



MÓDULO V

TÉCNICAS DIGITALES. SISTEMAS DE INSTRUMENTOS ELECTRÓNICOS



JAVIER GEJO GARCÍA

contenidos

1.-LÓGICA ELECTRÓNICA.....	10
1.1-SEÑALES ELÉCTRICAS DE INFORMACIÓN.....	10
1.2-SISTEMAS DE NUMERACIÓN.....	11
1.2.1-EXPRESIÓN DE NÚMEROS EN DIFERENTES SISTEMAS.....	12
1.3-REPRESENTACIÓN DE NÚMEROS.....	16
1.3.1-COMPLEMENTO A LA BASE, DE UN NÚMERO.....	16
1.3.2-REPRESENTACIÓN DE NÚMEROS BINARIOS CON SIGNO.....	17
1.4-OPERACIONES ARITMÉTICAS.....	22
1.4.1-SUMA.....	22
1.4.2-RESTA.....	23
1.4.3-SUMA Y RESTA BINARIA EN COMPLEMENTO A 2.....	23
1.4.4-MULTIPLICACIÓN BINARIA.....	25
1.4.5-DIVISIÓN BINARIA.....	25
1.5-CÓDIGOS.....	26
1.5.1-CÓDIGOS BCD.....	26
1.5.2-CÓDIGO GRAY.....	27
1.5.3-OTROS CÓDIGOS.....	28
1.5.4-CÓDIGOS ALFANUMÉRICOS.....	29
1.5.5-CÓDIGOS DETECTORES Y CORRECTORES DE ERROR.....	30
1.5.6-CÓDIGOS DE LÍNEA.....	34
1.6-FUNCIÓNES LÓGICAS.....	37
1.6.1-TABLA DE LA VERDAD DE UNA FUNCIÓN LÓGICA.....	37
1.6.2-FUNCIONES LÓGICAS BÁSICAS.....	37
1.6.3-ÁLGEBRA DE BOOLE.....	40
1.6.4-SÍMBOLOS ALTERNATIVOS.....	43
1.6.5-INTERPRETACIÓN DE SÍMBOLOS Y ESQUEMAS. NIVELES ACTIVOS.....	43
1.6.6-BLOQUEO Y PASO DE ENTRADAS.....	47
1.7-SIMPLIFICACIÓN DE FUNCIONES LÓGICAS.....	48
1.7.1-FORMA CANÓNICA DE UNA FUNCIÓN BOOLEANA.....	48
1.7.2-OBTENCIÓN DE TÉRMINOS DE LA TABLA DE LA VERDAD.....	50
1.7.3-INTERPRETACIÓN DE LOS DATOS DE LAS TABLAS DE LA VERDAD.....	51
1.7.4-SIMPLIFICACIÓN DE FUNCIONES.....	53
1.8-LÓGICA INTEGRADA.....	66
1.8.1-GLOSSARY.....	66
1.8.2-PARAMETERS.....	70
1.8.3-FAMILIAS LÓGICAS.....	71
1.8.4-TTL.....	73
1.8.5-CMOS.....	76
1.8.6-BICMOS TECHNOLOGY.....	77
1.8.7-ECL (CML).....	77
1.8.8-NIVELES DE INTEGRACIÓN.....	78
1.8.9-TRIGGER SCHMITT.....	79
1.8.10-MULTIVIBRADORES.....	79
2.-ESD (ELECTROSTATIC DISCHARGE).....	83
2.1-GENERACIÓN DE ESD.....	83

2.2-INFLUENCIA DEL MATERIAL EN LA CARGA ESTÁTICA.	84
2.3-EFECTOS PRODUCIDOS POR ESD.	85
2.3.1-DEFECTO LATENTE.	85
2.3.2-FALLO CATASTRÓFICO.	85
2.4-TIPOS DE DESCARGA.	86
2.5-CONTROL DE LA ESD.	87
2.5.1-PRINCIPIOS BÁSICOS.	87
2.5.2-ELEMENTOS DE PROTECCIÓN.	88
2.5.3-PROTECCIÓN DE IC-s.	90
2.6-ESD EN LA AERONAVE.	90
2.6.1-LRU-s. (Line Replaceable Units)	90
2.6.2-DESCARGADORES DE ESTÁTICA.	91
2.6.3-PUNTOS DE TIERRA.	92
3.-ELECTRÓNICA COMBINACIONAL.	94
3.1-CODIFICADOR (ENCODER)	95
3.2-DECODIFICADOR (DECODER)	98
3.2.1-DECODIFICADOR PARA IMPLEMENTAR UNA FUNCIÓN LÓGICA.	100
3.2.2-AMPLIACIÓN DE DECODIFICADORES.	101
3.3-CONVERTIDORES DE CÓDIGO. (TRANSCODIFICADORES)	102
3.4-MULTIPLEXERS.	113
3.4.1-MULTIPLEXORES EN PARALELO.	115
3.4.2-GENERACIÓN DE FUNCIONES LÓGICAS MEDIANTE MULTIPLEXORES	116
3.5-DEMULTIPLEXERS.	118
3.5.1-ACTUACIÓN DE UN DECODIFICADOR COMO DEMULTIPLEXOR:	119
3.6-COMPARADORES (COMPARATORS)	120
3.7-GENERADOR Y VERIFICADOR DE PARIDAD. (PARITY GENERATOR AND CHECKER)	123
3.7.1-EVEN PARITY:	123
3.7.2-ODD PARITY:	123
3.7.3-PARITY GENERATOR:	123
3.7.4-PARITY CHECKER	125
3.8-ARITMÉTICA. SUMADORES (ADDERS)	126
3.8.1-SUMADOR PARALELO	127
3.8.2-EJEMPLOS	129
4.-ELECTRÓNICA SECUENCIAL.	134
4.1-FLIP-FLOPS	134
4.1.1-FF ASÍNCRONOS RS.	135
4.1.2-FF ASÍNCRONOS JK.	138
4.1.3-CONMUTADOR SIN REBOTES	140
4.1.4-FF SÍNCRONOS.	140
4.1.5-ENTRADAS SÍNCRONAS Y ASÍNCRONAS.	144
4.1.6-TIMMING EN FLIP-FLOPS.	145
4.2-REGISTROS DE DESPLAZAMIENTO O ALMACENAMIENTO (SHIFT REGISTERS)	146
4.2.1-REGISTRO DE CARGA SERIE Y LECTURA SERIE.	146
4.2.2-REGISTRO DE CARGA PARALELO Y LECTURA PARALELO.	147
4.2.3-REGISTRO DE DESPLAZAMIENTO BIDIRECCIONAL, CARGA SERIE Y LECTURA SERIE O PARALELO.	147

4.3-CONTADORES (COUNTERS)	148
4.3.1-MÓDULO DE UN CONTADOR	148
4.3.2-CONTADORES ASÍNCRONOS O DE RIZO. MÓDULO BINARIO	148
4.3.3-CONTADORES ASÍNCRONOS O DE RIZO. MÓDULO NO BINARIO	149
4.3.4-CONTADORES SÍNCRONOS. MÓDULO BINARIO	152
4.3.5-CONTADORES SÍNCRONOS. MÓDULO NO BINARIO	153
4.3.6-CONTADORES PREINICIABLES	161
4.3.7-CONTADORES CÍCLICOS O CONTADORES DE REGISTRO DE DESPLAZAMIENTO	162
4.3.8-CONTADORES COMERCIALES. VARIACIÓN DE LA CUENTA	163
4.3.9-ACOTACIÓN DE LA CUENTA ENTRE DOS VALORES X E Y	164
4.3.10-DIVISORES DE FRECUENCIA	165
5.-PLD-S (PROGRAMMABLE LOGIC DEVICES)	167
5.1-IDEA BÁSICA:	167
5.2-SIMBOLOGÍA	168
5.3-ESTRUCTURAS LÓGICAS BÁSICAS	169
5.3.1-PROM (PROGRAMMABLE READ ONLY MEMORY)	169
5.3.2-PAL (PROGRAMMABLE ARRAY LOGIC)	169
5.3.3-PLA o FPLA. (PROGRAMMABLE LOGIC ARRAY o FIELD PROGRAMMABLE LOGIC ARRAY)	170
5.4-CONFIGURACIONES DE SALIDA DE LAS PAL	170
5.5-NOMENCLATURA	172
5.6-ACRÓNIMOS	172
5.7-PROGRAMACIÓN	173
5.8-EJEMPLO PAL COMERCIAL	174
6.-ENTORNO ANALÓGICO Y DIGITAL	176
6.1-DAC (DIGITAL TO ANALOG CONVERTER)	177
6.1.1-DAC CON RESISTENCIAS PONDERADAS	177
6.1.2-DAC CON RED EN ESCALERA R-2R	180
6.1.3-DAC CON ESCALERA INVERTIDA	181
6.1.4-DAC. ESPECIFICACIONES	182
6.1.5-EJEMPLOS COMERCIALES	184
6.1.6-CONEXIÓN EN UN SISTEMA MICROPROCESADOR	187
6.2-CIRCUITO DE MUESTREO Y RETENCIÓN (SAMPLE & HOLD-S/H)	188
6.2.1-CIRCUITO PRÁCTICO	189
6.2.2-TÉRMINOS S/H	190
6.2.3-CUANTIFICACIÓN	191
6.3-ADC (ANALOG TO DIGITAL CONVERTER)	192
6.3.1-MUESTREO – MANTENIMIENTO – CUANTIFICACIÓN – CODIFICACIÓN	192
6.3.2-MÉTODO PARALELO. FLASH ADC	193
6.3.3-HALF-FLASH	194
6.3.4-ADC DE RAMPA DIGITAL. CONTEO	195
6.3.5-CONTEO ASCENDENTE DESCENDENTE	196
6.3.6-ADC DE APROXIMACIONES SUCESIVAS	196
6.3.7-ADC DE SIMPLE RAMPA. (CONVERSIÓN TENSIÓN-TIEMPO)	198
6.3.8-ADC DE DOBLE RAMPA	200
6.3.9-ADC SERIE	201
6.3.10-ESPECIFICACIONES ADC	202

6.4-SISTEMA DE ADQUISICIÓN DE DATOS.....	203
7.-MEMORIAS	205
7.1-CARACTERÍSTICAS GENERALES (Memorias integradas):.....	206
7.1.1-VISIÓN GENERAL DE UNA MEMORIA INTEGRADA.	207
7.1.2-ORGANIZACIÓN DE UNA MEMORIA.	208
7.5-FLASH MEMORY.	209
7.6-APLICACIONES DE LAS ROM.	213
7.7-RAM (RANDOM ACCESS MEMORY).	214
7.7.1-SRAM. (STATIC RAM)	214
7.7.2-DRAM (DYNAMIC RAM).	217
7.7.3-SDRAM. (SYNCHRONOUS DRAM).	222
7.7.4-RDRAM.....	222
7.7.5-RAM CMOS	223
7.7.6-NVRAM (NO VOLATIL RAM)	223
7.7.7-RAM DUAL PORT	223
7.8-MEMORIAS SECUENCIALES INTEGRADAS.	224
7.9-CAM (CONTENT-ADDRESSABLE MEMORY)	227
7.10-MEMORIAS MAGNETICAS.	228
7.11-MEMORIAS ÓPTICAS.....	229
7.12-MEMORIAS MAGNETO-ÓPTICAS.....	230
7.13-EXPANSIÓN DE MEMORIA INTEGRADA.	231
7.13.1-EXPANSIÓN DE LA LONGITUD DE PALABRA.....	231
7.13.2-EXPANSIÓN DE LA CAPACIDAD.	231
7.13.3-EXPANSIÓN DE LA LONGITUD DE PALABRA Y LA CAPACIDAD.....	232
7.14-MAPA DE MEMORIA.	233
8.-SISTEMA MICROPROGRAMABLE. GENERALIDADES.	236
8.1-LÓGICA CABLEADA Y LÓGICA PROGRAMADA.....	236
8.2-SISTEMA MICROPROGRAMABLE.....	238
8.3-PROCEDIMIENTOS DE MANIPULACIÓN DE DISPOSITIVOS EXTERNOS.....	243
8.4-MEDIOS PARA PROVEER EL ACCESO A LA INFORMACIÓN EN EL I/O INTERFACE.	244
8.5-BUSES TÍPICOS. TIMING	245
8.6-ARQUITECTURAS BÁSICAS DE SISTEMAS μPROGRAMABLES.	248
8.7-PROGRAMACIÓN.....	249
8.8-DIRECTIVAS.....	253
8.9-CLASIFICACIÓN DE LAS INSTRUCCIONES:.....	254
8.10-PRINCIPIOS DE PROGRAMACIÓN	254
8.11-TAREAS DE ENTRADA / SALIDA. (I/O)	260

8.12-SUBROUTINAS.....	262
8.13-STACK (PILA)	264
8.14-PASO DE PARÁMETROS A UNA SUBROUTINA.....	266
8.15-LISTA ENLAZADA. (LINKED LIST).....	268
8.16-CADENA DE CARACTERES. (CHARACTER STRINGS)	270
8.17-MODOS DE DIRECCIONAMIENTO BÁSICOS.	272
8.18-INTERRUPCIONES.	275
8.19-EXCEPCIONES.	279
8.20-EJEMPLOS (6809 PROGRAMACIÓN).....	280
9.-μP DE 8 BITS. GENERALIDADES	285
9.1-6502.....	285
9.1.1-REGISTROS INTERNOS.....	285
9.1.2-MODOS DE DIRECCIONAMIENTO (6502).....	285
9.1.3-REGISTRO DE ESTADO (6502).....	286
9.2-8085.....	288
9.2.1-REGISTROS INTERNOS.....	288
9.2.2-MODOS DE DIRECCIONAMIENTO (8085).....	288
9.2.3-REGISTRO DE ESTADO (8085).....	289
9.3-6809.....	290
9.3.1-REGISTROS INTERNOS.....	290
9.3.2-MODOS DE DIRECCIONAMIENTO (6809).....	290
9.3.3-REGISTRO DE CÓDIGO DE CONDICIÓN. CCR (6809).	291
9.4-μP. 16 BITS. 68000.....	293
9.4.1-TERMINALES DEL 68000.....	294
9.4.2-68000. REGISTROS	302
9.4.3-ORGANIZACIÓN DE LA MEMORIA.	304
9.4.4-MODOS DE DIRECCIONAMIENTO.	306
9.4.5-SET DE INSTRUCCIONES. (NOTAS)	308
9.4.6-EXCEPCIONES (NOTAS).....	313
9.4.7-ASIGNACIÓN DE LOS VECTORES DE EXCEPCIÓN.....	318
9.4.8-PRIORIDAD DE LAS EXCEPCIONES:.....	319
9.5-SET DE INSTRUCCIONES	319
10.-INPUT/OUTPUT (I/O)	321
10.1-TRANSMISIÓN SERIE.	321
10.1.1-IDEA BÁSICA.	321
10.1.2-TRANSMISIÓN SÍNCRONA.....	322
10.1.3-TRANSMISIÓN ASÍNCRONA.	323
10.2-INTERFACE SERIE. UART (UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITER)	325
10.3-RS- 232-C (V.24, V.28)	329
10.4-RS- 449 [RS-422 (V.11), RS-423 (V.10)].....	333
10.5-TRANSMISIÓN PARALELO.	335

10.5.1-ENTRADAS POR LATCH.	335
10.5.2-I/O COMBINADAS.	335
10.5.3-HANDSHAKE. (PROTOCOLO).....	335
10.6-INTERFACE PARALELO. PIA (PERIPHERAL INTERFACE ADAPTER).....	339
10.7-INTERFACE PARALELO. VIA (VERSATILE INTERFACE ADAPTER)	346
10.8-DMA. (DIRECT MEMORY ACCESS).....	356
11.-LÍNEAS DE TRANSMISIÓN.....	364
11.1-CARACTERÍSTICAS.....	367
11.2-DATOS DE ALGUNAS LÍNEAS:	373
11.2.1-CABLE COAXIAL	373
11.2.2-PAR TRENZADO.....	374
11.2.3-RANGO DE APLICACIÓN DE LAS DIFERENTES LINEAS:.....	376
11.3-TRANSMISIÓN ÓPTICA.....	377
11.3.1-ESPECTRO DE FRECUENCIA ELECTROMAGNÉTICA.....	379
11.4-FIBRA ÓPTICA.....	379
11.4.1-COMPARATIVA CON LÍNEAS METÁLICAS.	379
11.4.2-COMPOSICIÓN.....	380
11.4.3-PROPAGACIÓN DE LA LUZ EN LA FIBRA.	381
11.4.4-CARACTERÍSTICAS	384
11.4.5-EMISORES.....	387
11.4.6-RECEPTORES.....	389
11.4.7-TIPOS DE CONECTORES. EJEMPLOS	391
12.-BUSES DE DATOS EN AERONAVES.	393
12.1-ARINC 429 (AERONAUTICAL RADIO INC. 429).....	393
12.1.1-CARACTERÍSTICAS GENERALES.....	393
12.1.2-PROTOCOLO.....	403
12.2-ESPECIFICACIÓN ARINC 629. (AERONAUTICAL RADIO INC. 629).....	419
12.2.1-CARACTERÍSTICAS GENERALES.....	419
12.2.2-OPERACIÓN	423
12.3-MIL-STD-1553.....	429
12.3.1-CARACTERÍSTICAS GENERALES.....	429
12.3.2-OPERACIÓN BÁSICA.....	434
12.4-MIL-STD-1773.....	439
13.- DISPLAYS.....	441
13.1-CRT (CATHODE RAY TUBE).....	441
13.1.1-CRT MONOCROMO.	441
13.1.2-CRT COLOR.....	444
13.1.3-MONITORES BASADOS EN CRT.....	445
13.1.4-CONTROLADORES DE PANTALLA.	451
13.2-LCD (LIQUID CRYSTAL DISPLAY).....	457
13.2.1-DISTRIBUCIÓN NEMÁTICA DE LOS CRISTALES.....	457
13.2.2-TIPOS.....	461
13.3-CAÑÓN PROYECTOR DE IMAGEN.....	463
13.3.1-PROYECTORES LCD.....	463
13.3.2-PROYECTORES DLP.....	463

13.4-LED (LIGHT EMITTING DIODE) DISPLAYS.....	464
13.4.1-DISPLAYS SEGMENTADOS.....	464
13.4.2-DISPLAYS DE MATRIZ DE PUNTOS.....	464
13.4.3-UTILIZACIÓN EN UN SISTEMA.....	464
14.-SOFTWARE.....	469
14.1-EVOLUCIÓN.....	469
14.2-SOFTWARE STANDARDS. D0-178B.....	469
14.2.1-PROCESOS.....	470
14.3-CAMBIOS EN EL SOFTWARE.....	471
14.4-REDUNDANCIA.....	473
14.5-CONTROL DEL SOFTWARE DE PROGRAMA.....	473
14.6-MÉTODOS DE CONTROL DEL SOFTWARE DE PROGRAMA.....	473
14.6.1-PARTICIONADO DE SOFTWARE.....	474
14.6.2-PINES PROGRAMABLES.....	474
15.-ENTORNO ELECTROMAGNÉTICO.....	477
15.1-ELECTROMAGNETIC COMPATIBILITY (EMC).....	477
15.1.1-ANTECEDENTES.....	477
15.1.2-ELECTROMAGNETIC INTERFERENCE (EMI).....	481
15.1.3-ELEMENTOS EN UN PROBLEMA DE EMC.....	482
15.1.4-EFECTOS PRODUCIDOS POR LA INCOMPATIBILIDAD ELECTROMAGNÉTICA.....	482
15.1.5-MEDIDAS DE PROTECCIÓN EN LA AERONAVE.....	484
15.2-TESTS DE EMC. ASPECTOS BÁSICOS.....	485
15.2.1-INSTRUMENTACIÓN.....	485
15.2.2-MEDICIONES.....	485
15.3-HIGH INTENSITY RADIATED FIELDS (HIRF).....	489
15.3.1-PROTECCIÓN EFECTIVA DE LOS SISTEMAS.....	490
15.3.2-PERTURBACIONES EN SISTEMAS AVIÓNICOS DEBIDOS A HIRF.....	496
15.4-RAYOS (LIGHTNING).....	497
15.4.1-INTRODUCCIÓN.....	497
15.4.2-EL FENÓMENO DEL RAYO EN LAS AERONAVES.....	500
16.-MANTENIMIENTO AUTOMÁTICO.....	505
16.1-AUTO PRUEBAS.....	505
16.1.1-CATEGORÍAS DE AUTO PRUEBA.....	505
16.1.2-MÉTODOS DE AUTOPRUEBA.....	508
16.1.3-TIPOS DE AUTOPRUEBA DE COMPUTADORES.....	508
16.1.4-FALLOS EN LA AUTOPRUEBA. ACCIONES.....	509
16.2-BUILT-IN TEST EQUIPMENT (BITE).....	510
16.2.1-CLASIFICACIÓN DE FALLOS.....	510
16.2.2-CATEGORÍAS BITE.....	510
16.2.3-TIPOS DE BITE.....	511
16.2.4-CONTROLES, INDICACIONES Y SALIDAS EN EL BITE.....	511
16.2.5-EJEMPLOS BITE.....	513
16.3-AUTOMATIC TEST EQUIPMENT (ATE).....	524
16.3.1-TIPOS DE ATE.....	524
16.3.2-AUTOMATIC TEST STATION (ATS).....	525
16.3.3-PROCESO DE MANTENIMIENTO.....	527

17.-AVIONICS SYSTEMS.....	530
17.1-COCKPIT LAYOUT.	530
17.2-EIS (ELECTRONIC INSTRUMENT SYSTEM).	532
17.2.1-DESCRIPCIÓN GENERAL.	532
17.2.2-ELECTRONIC CENTRALIZED AIRCRAFT MONITORING (ECAM).	537
17.2.3-ELECTRONIC FLIGHT INSTRUMENTS SYSTEM (EFIS).	541
17.3-CDS (COMMON DISPLAY SYSTEM).	545
17.3.1-DESCRIPCIÓN GENERAL.	545
17.4-SISTEMA DE NAVEGACIÓN.	550
17.4.1-DESCRIPCIÓN GENERAL.	550
17.5-ADIRS (AIR DATA INERTIAL REFERENCE SYSTEM).....	554
17.5.1-DESCRIPCIÓN GENERAL.	554
17.5.2-PRESENTACIÓN EN DISPLAYS.....	560
17.6-ACARS (ARINC COMMUNICATION ADDRESSING AND REPORTING SYSTEM).....	561
17.6.1-DESCRIPCIÓN GENERAL.	561
17.6.2-TEST BITE.....	568
17.7-GPS (GLOBAL POSITIONING SYSTEM).	573
17.7.1-DESCRIPCIÓN GENERAL.	573
17.7.2-INDICACIÓN Y CONTROL.	577
17.8-ATC / TCAS (AIR TRAFFIC CONTROL / TRAFFIC ALERT AND COLLISION AVOIDANCE SYSTEM).....	578
17.8.1-ATC. DESCRIPCIÓN GENERAL.	578
17.8.2-OPERACIÓN ATC.....	584
17.8.3-TCAS. DESCRIPCIÓN GENERAL.....	588
17.8.4-OPERACIÓN TCAS.	593
17.8.5-TCAS BITE.	598
17.9-ATIMS (AIR TRAFFIC AND INFORMATION MANAGEMENT SYSTEM)	602
17.9.1-INTRODUCCIÓN A LA INFORMACIÓN POR ENLACE DE DATOS.	602
17.9.2-ATIMS. DESCRIPCIÓN GENERAL.	603
17.9.3-TEST BITE.....	608
17.10-FMS (FLIGHT MANAGEMENT SYSTEM).	614
17.10.1-INTRODUCCIÓN. AFS	614
17.10.2-FMS. DESCRIPCIÓN GENERAL.....	619
17.11-FBW (FLY BY WIRE).....	624
17.11.1-ASPECTOS GENERALES.	624



Lógica Electrónica

1

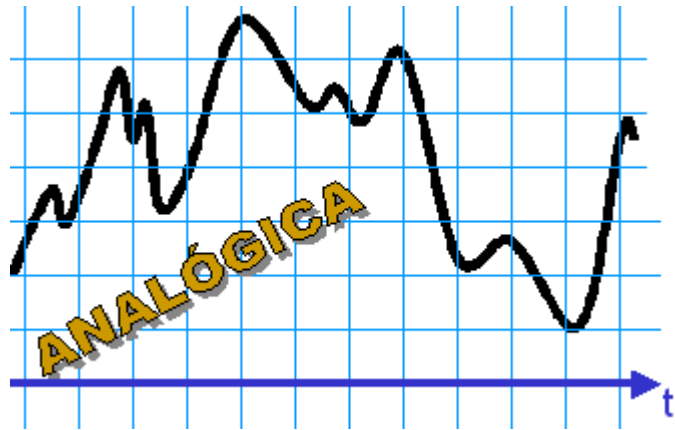
1.-LÓGICA ELECTRÓNICA.

La lógica electrónica es la ciencia de razonar a partir de los estados que pueden tener las variables de entrada a un sistema electrónico en orden a obtener una salida determinada.

1.1-SEÑALES ELÉCTRICAS DE INFORMACIÓN.

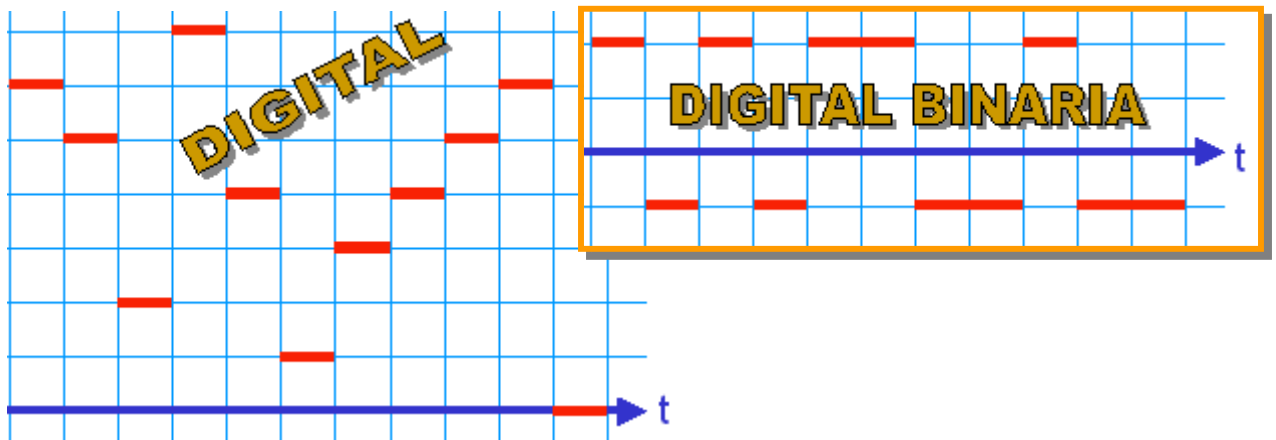
SEÑAL ANALÓGICA.

- En un instante dado, el valor instantáneo de la señal analógica puede ser cualquiera y único, de los infinitos valores comprendidos entre un valor máximo y un valor mínimo de acotación.



SEÑAL DIGITAL.

- La señal digital solamente puede tomar valores discretos, en un margen determinado de ellos. Cuando el número de valores que puede tomar se limita a dos (siempre los mismos) estamos ante una señal digital binaria.



SEÑAL DISCRETA.

- Esta señal se utiliza en el entorno aeronáutico. Puede tener dos valores, pero a diferencia de la señal digital, estos valores pueden ser diferentes, y están ya definidos. Otra diferencia fundamental con la señal digital binaria es que esta suele tratarse en grupos de, basta decir por el momento, grupos de estados (varios estados altos y/o bajos), para procesar la información que contiene, mientras que un discreto, que puede tener un valor u otro, por sí mismo es información tal como: puerta abierta o puerta cerrada, válvula activada o desactivada, motor activado o desactivado, etc.

1.2-SISTEMAS DE NUMERACIÓN.

- Los números pueden representarse en diversos sistemas de numeración que se diferencian en su base, la cual representa el número de símbolos distintos utilizados para la representación de magnitudes o cantidades en este sistema de numeración.

Ej. de sistemas de numeración más empleados:

SISTEMA	BASE (b)	SÍMBOLOS DEL SISTEMA
Binario	2	0, 1
Octal	8	0, 1, 2, 3 4, 5, 6, 7
Decimal	10	0, 1, 2, 3 4, 5, 6, 7, 8, 9
Hexadecimal	16	0, 1, 2, 3 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

DESCOMPOSICIÓN POLINÓMICA.

- Cualquier numero puede ser representado por su descomposición polinómica según la expresión:

$$N = a_n b^n + a_{n-1} b^{n-1} + a_{n-2} b^{n-2} + \dots + a_1 b + a_0 + a_{-1} b^{-1} + a_{-2} b^{-2} + \dots + a_{-p} b^{-p}$$

Donde:

b= base;

a_i = símbolo que pertenece al sistema y que cumple: $0 \leq a_i < b$

EJEMPLOS:

SISTEMA	EJEMPLO	REPRESENTACIÓN
Binario*	01101 _{/2}	$0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$
	10010,1011 _{/2}	$1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 + 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} + 1 \cdot 2^{-4}$
Octal	46170 _{/8}	$4 \cdot 8^4 + 6 \cdot 8^3 + 1 \cdot 8^2 + 7 \cdot 8^1 + 0 \cdot 8^0$
	7762,43 _{/8}	$7 \cdot 8^3 + 7 \cdot 8^2 + 6 \cdot 8^1 + 2 \cdot 8^0 + 4 \cdot 8^{-1} + 3 \cdot 8^{-2}$
Decimal	43819	$4 \cdot 10^4 + 3 \cdot 10^3 + 8 \cdot 10^2 + 1 \cdot 10^1 + 9 \cdot 10^0$
	32,69	$3 \cdot 10^1 + 2 \cdot 10^0 + 6 \cdot 10^{-1} + 9 \cdot 10^{-2}$
Hexadecimal	A132H	$A \cdot 16^3 + 1 \cdot 16^2 + 3 \cdot 16^1 + 2 \cdot 16^0$
	A3B, 2FH	$A \cdot 16^2 + 3 \cdot 16^1 + B \cdot 16^0 + 2 \cdot 16^{-1} + F \cdot 16^{-2}$

Los símbolos binarios se llaman **BIT**-s, (*B*inary dig*IT*). Un BIT puede tener el valor 0 o el valor 1.

En una cifra binaria el BIT situado a la izquierda se llama **MSB** (*Most Significant BIT*), el BIT situado a la derecha se llama **LSB** (*Least Significant BIT*).

1011100111011_{/2}
MSB LSB

En cifras de los otros sistemas, el número situado a la derecha se le llama dígito menos significativo, y el situado más a la izquierda, dígito más significativo.

1.2.1-EXPRESIÓN DE NÚMEROS EN DIFERENTES SISTEMAS

CAMBIO DE BASE b A BASE 10.

Enteros y fraccionarios.

- Se realiza la descomposición polinómica del número en base b y se efectúa la suma.

$$N = a_n b^n + a_{n-1} b^{n-1} + a_{n-2} b^{n-2} + \dots + a_1 b + a_0 + a_{-1} b^{-1} + a_{-2} b^{-2} + \dots + a_{-p} b^{-p}$$

Ej. :

H $\longrightarrow$ b10 (hexadecimal a decimal)

$$AF1H = A \cdot 16^2 + F \cdot 16^1 + 1 \cdot 16^0 = 10 \cdot 16^2 + 15 \cdot 16^1 + 1 = 2560 + 240 + 1 = 2801$$

$$A3,BH = A \cdot 16^1 + 3 \cdot 16^0 + B \cdot 16^{-1} = 10 \cdot 16^1 + 3 \cdot 16^0 + 11 \cdot 16^{-1} = 160 + 3 + 0.6875 = 163,6875$$

b8 $\longrightarrow$ b10 (octal a decimal)

$$7214_8 = 7 \cdot 8^3 + 2 \cdot 8^2 + 1 \cdot 8^1 + 4 \cdot 8^0 = 3584 + 128 + 8 + 4 = 3724$$

$$0.5605_8 = 0 \cdot 8^0 + 5 \cdot 8^{-1} + 6 \cdot 8^{-2} + 0 \cdot 8^{-3} + 5 \cdot 8^{-4} = 0 + 0.625 + 0.09375 + 0 + 1.22 \cdot 10^{-3} = 0.71997$$

b2 $\longrightarrow$ b10 (binario a decimal)

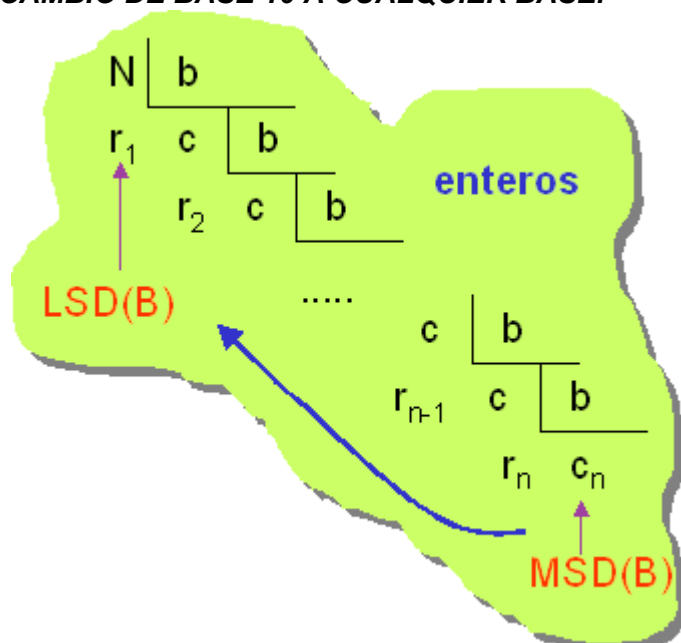
$$1101101_{/2} = 1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 64 + 32 + 0 + 8 + 4 + 0 + 1 = 109$$

$$10100000,110001_{/2} = 1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 0 \cdot 2^{-3} + 0 \cdot 2^{-4} + 0 \cdot 2^{-5} + 1 \cdot 2^{-6} = 128 + 0 + 32 + 0 + 0 + 0 + 0 + 0 + 0.5 + 0.25 + 0 + 0 + 0 + 15,062 \cdot 10^{-3} = 160.765$$

Convertir a decimal los siguientes números (*entre paréntesis, la respuesta*)

FFFFH (65535) 71_8 (57) $1111111_{/2}$ (255) A9,4CH (169,2968)

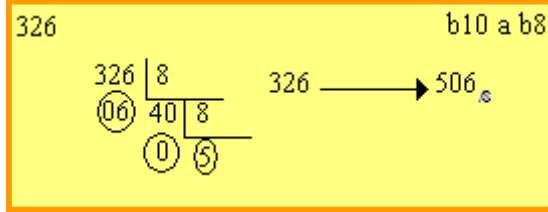
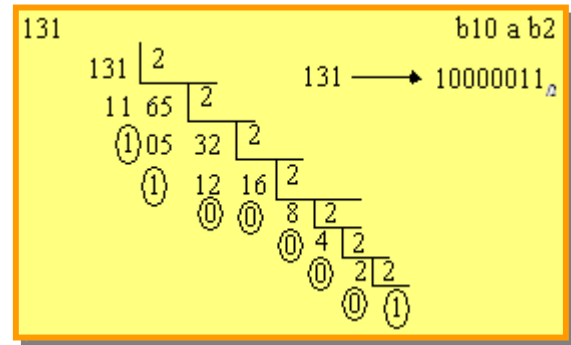
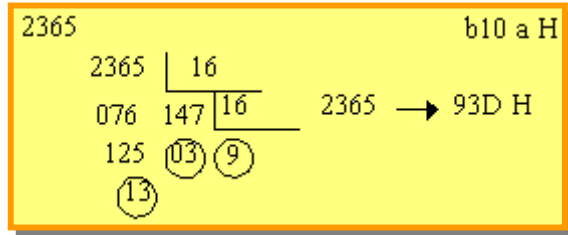
CAMBIO DE BASE 10 A CUALQUIER BASE.



Enteros.

- Se toma el número en base 10 y se divide por la base a la que se quiere transformar. El primer resto obtenido representa el dígito o BIT menos significativo del número en la nueva base. El cociente se divide por la base a la que se quiere transformar. El segundo resto es otro símbolo del nuevo número. Este proceso se realiza hasta que un cociente sea menor que la base; Este cociente representa el dígito o BIT más significativo del número en la nueva base.

Ej. :



Convertir: 1024 a hexadecimal (**400H**); 256 a binario (**100000000₂**); 1234 a octal (**2322₈**)

Fraccionarios.

- Se multiplica el número en base diez por la nueva base y se toma la parte entera del resultado como cifra más significativa del nuevo número; Con la parte fraccionaria se realiza la misma operación de multiplicar por la nueva base. De nuevo se toma la parte entera, que será la segunda cifra más significativa, y la parte fraccionaria se vuelve a multiplicar por la nueva base, repitiéndose este proceso hasta que el resultado sea nulo o hasta conseguir el número de decimales deseados.

Ej. :

b10 a H.

0.72	0.72 * 16 = 11.52	11 (B)
	0.52 * 16 = 8.32	8
	0.32 * 16 = 5.12	5
	0.12 * 16 = 1.92	1
	0.92 * 16 = 14.72	14 (E)
	0.72 * 16 = 11.52	11 (B)
	0.52 * 16 = ...	
	0.B851EBH	

b10 a b8.

0.72	0.72 * 8 = 5.76	5
	0.76 * 8 = 6.08	6
	0.08 * 8 = 0.64	0
	0.64 * 8 = 5.12	5
	0.12 * 8 = 0.96	0
	0.96 * 8 = 7.68	7
	0.68 * 8 = ...	
	0.560507₈	

b10 a b2.

0.72	$0.72 * 2 = 1.44$	1	
	$0.44 * 2 = 0.88$	0	
	$0.88 * 2 = 1.76$	1	
	$0.76 * 2 = 1.52$	1	
	$0.52 * 2 = 1.04$	1	
	$0.04 * 2 = 0.08$	0	
			0.101110₂

Parte entera y parte fraccionaria.

- Se realiza la conversión, por un lado de la parte entera y por otro lado de la parte fraccionaria según los métodos explicados.

160.765625 (= 160 + 0.765625)

160 $\longrightarrow$ **10100000₂**

0.765625	$0.765625 * 2 = 1.53125$	1	
	$0.53125 * 2 = 1.0625$	1	
	$0.0625 * 2 = 0.125$	0	
	$0.125 * 2 = 0.25$	0	
	$0.25 * 2 = 0.5$	0	
	$0.5 * 2 = 1$	1	
	$0 * 2 = 0$	0	
			110001₂

160.765625 $\longrightarrow$ **10100000.110001₂**

Convertir a binario 15,32 (**1111,01010**)

CAMBIO UN NÚMERO EN BASE b_x A OTRO EN BASE b_y ($b_y \neq 10$).

- En general este cambio se puede realizar pasando el número en base b_x a base 10 y después de base 10 a la base b_y , tal y como se ha indicado anteriormente.

Ej. :

b8 a H.

$$545_{/8} : 545_{/8} = 5 \cdot 8^2 + 4 \cdot 8^1 + 5 \cdot 8^0 = 320 + 32 + 5 = 357$$

$545_{/8} \longrightarrow 165H$

357	16
037	22
05	06
01	

b2 a H.

101101011₂:

$$101101011_{/2} = 1 \cdot 2^8 + 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^3 + 1 \cdot 2^1 + 1 \cdot 2^0 = 256 + 64 + 32 + 8 + 2 + 1 = 363$$

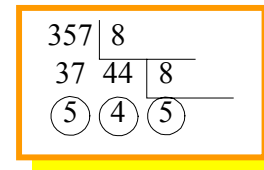
$101101011_{/2} \longrightarrow 16BH$

363	16
043	22
11	06
01	

H a b8.

165H: $165H = 1 \cdot 16^2 + 6 \cdot 16^1 + 5 \cdot 16^0 = 256 + 96 + 5 = 357$

165H $\longrightarrow$ 545₈

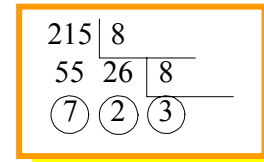


b2 a b8.

11010111₂:

$11010111_2 = 1 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^4 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 128 + 64 + 16 + 4 + 2 + 1 = 215$

11010111₂ $\longrightarrow$ 327₈



- Se pueden realizar ciertas conversiones utilizando un método de codificación. Según este método, los símbolos del sistema hexadecimal y octal pueden ser codificados directamente mediante bits.

Un BIT puede tener dos valores (0 o 1). Con n bits se pueden formar 2^n combinaciones diferentes. Como el sistema octal tiene 8 valores diferentes, para poder codificar esos valores se necesitarán el siguiente número de bits:

$$2^n = 8; \log 2^n = \log 8; n \log 2 = \log 8; n = (\log 8 / \log 2) = \underline{3 \text{ bits.}}$$

De igual forma, el sistema hexadecimal posee 16 valores diferentes por lo que para poder codificarlos necesitamos:

$$2^n = 16; \log 2^n = \log 16; n \log 2 = \log 16; n = (\lg 16 / \log 2) = \underline{4 \text{ bits.}}$$

- Para convertir un número de base octal o hexadecimal a binario, lo único que tenemos que hacer es codificar cada símbolo con la combinación de bits correspondiente según la tabla dada en la página siguiente.
- Para convertir un número binario al sistema octal o hexadecimal, lo que tenemos que hacer es tomar los bits en grupos de tres (octal) o de cuatro (hex.) comenzando el agrupamiento desde el LSB y hacia el MSB. Cada grupo de bits representa un símbolo en el sistema correspondiente (ver tabla en la página siguiente). Para la parte fraccionaria los agrupamientos se hacen desde la coma hacia la derecha.
- La equivalencia numérica entre el sistema binario y el decimal se obtiene directamente teniendo en cuenta la relación de peso de los bits.

El peso del LSB es 2^0 (1), los siguientes pesos según avanzamos hacia el MSB son 2^1 (2), 2^2 (4), 2^3 (8), 2^4 (16), 2^5 (32), 2^6 (64), 2^7 (128),...

Ej.:

732₈ 111(7) 011(3) 010(2) **111011010₂**

732H 0111(7) 0011(3) 0010(2) **011100110010₂**

101101011_2

 $\nearrow$ 101(5) 101(5) 011(3) **553₈**
 $\searrow$ 1(1) 0110(6) 1011(B) **16BH**

110110.11_2

 11(3) 0110(6) 11(C) **36.CH**

DECIMAL	OCTAL	HEXADECIMAL	BINARIO
0	0	0	0
1	1	1	1
2	2	2	10
3	3	3	11
4	4	4	100
5	5	5	101
6	6	6	110
7	7	7	111
8	10	8	1000
9	11	9	1001
10	12	A	1010
11	13	B	1011
12	14	C	1100
13	15	D	1101
14	16	E	1110
15	17	F	1111
16	20	10	10000
17	21	11	10001
18	22	12	10010
19	23	13	10011
20	24	14	10100
.....			

Equivalencia entre los primeros números en diferentes sistemas de numeración.

1.3-REPRESENTACIÓN DE NÚMEROS.

1.3.1-COMPLEMENTO A LA BASE, DE UN NÚMERO.

Sea **N** un número de **n** bits o dígitos expresado en base **b**. El complemento a la base menos uno, **C_{b-1}** de ese número viene dado por la expresión:

$$C_{b-1}(N) = C_b(N) - 1 = b^n - N - 1$$

donde **C_b(N) = bⁿ - N** es el complemento a la base.

Ej. :

Complemento a nueve de 63; $C_9(63) = 10^2 - 63 - 1 = 36$.

Complemento a uno de 10110_2 ; $C_1(10110) = [2^5 - 22 - 1 = 9] = 01001$

Complementar a 9 el número 8. (1)	Complementar a 10 el número 96 (4)
Complementar a 1 el número 10100_2 (01011)	Complementar a 2 el número 10110_2 (1010)

- El complemento del complemento de un número da como resultado ese mismo número.

$$C_b(N) = b^n - N = X$$

$$C_b(X) = b^n - X = b^n - (b^n - N) = b^n - b^n + N = N$$

- De la misma forma, el complemento a la base menos uno del complemento a la base menos uno de un número, da como resultado ese mismo número:

$$C_{b-1}(N) = b^n - N - 1 = Y$$

$$C_{b-1}(Y) = b^n - Y - 1 = b^n - (b^n - N - 1) - 1 = N$$

Observar que para complementar a 1 un número binario lo único que tenemos que hacer es cambiar los ceros por unos y viceversa. Para complementar a 2, se comienza por el LSB dejando los bits como están hasta que se encuentre el primer 1. A partir de aquí y hacia el MSB se cambian los unos por ceros y viceversa. También lo que se puede hacer es complementar a uno y al resultado sumarle 1.

1.3.2-REPRESENTACIÓN DE NÚMEROS BINARIOS CON SIGNO.

SISTEMA SIGNO-MAGNITUD.



MAGNITUD: Corresponde al equivalente binario.

BIT DE SIGNO: 0 valor positivo; 1 valor negativo.

Los valores máximos que se pueden representar para un número (n) dado de bits es:

$$\pm (2^{n-1} - 1).$$

La tabla muestra un ejemplo para 4 bits.

signo	Magnitud			Eq. dec
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	-0
1	0	0	1	-1
1	0	1	0	-2
1	0	1	1	-3
1	1	0	0	-4
1	1	0	1	-5
1	1	1	0	-6
1	1	1	1	-7

SISTEMA COMPLEMENTO A 1.



MAGNITUD:

Positivos: Corresponde con el equivalente binario.

Negativos: Corresponde con el complemento a 1 del equivalente binario.

BIT DE SIGNO: 0 valor positivo; 1 valor negativo.

Los valores máximos que se pueden representar para un número (n) dado de bits es $\pm (2^{n-1} - 1)$.

- La operación complemento a 1 de un número con signo convierte un número positivo en negativo y viceversa.

0011 (3) comp. a 1 $\longrightarrow$ 1100 (-3)

Ejemplo para 4 bits

signo	Magnitud			Eq. dec
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	-7
1	0	0	1	-6
1	0	1	0	-5
1	0	1	1	-4
1	1	0	0	-3
1	1	0	1	-2
1	1	1	0	-1
1	1	1	1	-0

SISTEMA COMPLEMENTO A 2.



MAGNITUD:

Positivos: Corresponde con el equivalente binario.

Negativos: Corresponde con el complemento a 2 del equivalente binario.

NOTA. (Excepción para los números negativos): Si todos los bits de la magnitud son 0, el equivalente decimal viene dado por la expresión -2^n donde n representa el n° de bits que forman la magnitud.

Ej. :

1000	-8
10000	-16
100000	-32

BIT DE SIGNO: 0 valor positivo; 1 valor negativo.

Los valores máximos que se pueden representar para un número (n) dado de bits son desde -2^{n-1} hasta $(2^{n-1} - 1)$.

signo	Magnitud			Eq. dec
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	-8
1	0	0	1	-7
1	0	1	0	-6
1	0	1	1	-5
1	1	0	0	-4
1	1	0	1	-3
1	1	1	0	-2
1	1	1	1	-1

Ej. : para 4 bits los valores van desde -8 hasta +7. Con 6 bits los valores van desde -32 hasta 31, etc.

- La operación complemento a 2 de un número con signo convierte un número positivo en negativo y viceversa.

Ej. :

011011(27)	comp. a 2	—————▶	100101(-27)
110110 (-10)	comp. a 2	—————▶	001010 (10)

REPRESENTACIÓN EN COMA FLOTANTE. ($M * b^E$)

La representación de un número binario con decimales, puede realizarse tomando, por ejemplo, para un número de 8 bits el convenio de que los cuatro bits de la izquierda representen la parte entera, y los cuatro restantes, representen la parte fraccionaria. En este caso una expresión como 11010111 representaría al número 1101,0111. El rango de valores que se podrían representar según este criterio sería para 8 bits desde 0 hasta 1111,1111. Esta forma de representación en la cual la coma tiene siempre un valor prefijado en el espacio se llama representación en COMA FIJA.

Otra forma de representar números binarios (o de cualquier otra base) es mediante el formato de COMA FLOTANTE. Esta representación tiene la forma:

$$M * b^E$$

E= exponente. (nº entero)

M= mantisa. (nº entero o fraccionario)

b = base del sistema de numeración

La mantisa y el exponente pueden ser positivos o negativos.

Según este sistema, la precisión de un número se puede controlar variando de forma independiente el número de dígitos que forman la mantisa (M). Por ejemplo la velocidad de la luz se puede expresar según el número de dígitos de la mantisa, con diferente precisión:

$$3 \times 10^8 \text{ ms}^{-1} \quad \longrightarrow \quad 2,997925 \times 10^8 \text{ ms}^{-1}$$

Por otro lado, un mismo número puede ser representado de diferentes maneras con este sistema:

Ej.

$$1/3 * 10^{12} \text{ puede representarse con 4 dígitos de mantisa como } \left\{ \begin{array}{l} 0.0003 \times 10^{15} \text{ (a)} \\ 0.0033 \times 10^{14} \text{ (b)} \\ 0.0333 \times 10^{13} \text{ (c)} \\ 0.3333 \times 10^{12} \text{ (d)} \\ 3.3333 \times 10^{10} \text{ (e)} \\ \dots\dots\dots \end{array} \right.$$

La normalización para la representación en este formato de números en base 10, establece que la parte entera de la mantisa debe ser cero, y la parte fraccionaria no debe comenzar por cero, de esta manera se obtienen más dígitos significativos. En el ejemplo anterior, la expresión normalizada es la (d).

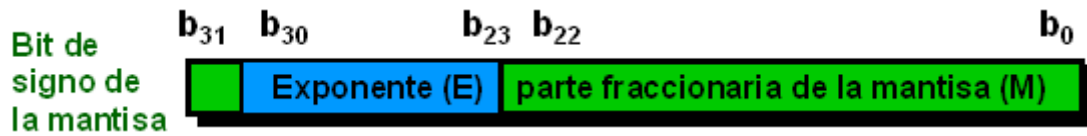
Para el caso de números binarios, también existen diversas formas de representación. Una de estas formas es la recomendada por el **IEEE** (*Institute of Electrical and Electronics Engineers*).

IEEE define dos estándares:

Formato de 32 bits (Precisión Simple)

Formato de 64 bits (Precisión Doble)

Formato en coma flotante (precisión simple):



- La mantisa se representa en la forma *Signo-Magnitud* (b_{31} para el signo). Se normaliza de tal manera que su MSB es siempre 1 y va a ser la parte entera. Como este valor es conocido, no se guarda. Se asume que el formato es de la forma 1.XXXXX y solo la parte fraccionaria XXXXX, es guardada.

Ej. : $0.0101101 \cdot 2^E \rightarrow 1.0110110 \cdot 2^{E-2}$ (se guardaría como valor de la mantisa 0110110)

- El rango de valores de la mantisa será 1.000.... hasta 1.111..... ($1.0 \leq |M| < 2.0$). Debido a esta normalización, la mantisa nunca puede ser 0; por lo tanto, nunca podría representarse un número de valor cero en representación coma flotante, siendo el mínimo valor $1 \cdot 2^{-126}$.

Para evitar esto, un valor de todos ceros en el exponente implica un valor de 0 del número en coma flotante. Por otra parte, un valor de todos unos en el exponente y todos ceros en la mantisa, representa un valor infinito.

- El exponente se representa en una variación del complemento a 2 conocida como exceso 127.

DEC	EXP	VALOR
0	00000000	RESERV
1	00000001	-126
...	...	...
127	01111111	+0
128	10000000	+1
129	10000001	+2
...	...	...
254	11111110	+127
255	11111111	RESERV

Para obtener el valor del exponente en base 10 lo que se debe hacer es restar la cantidad 127 (01111111) al valor del exponente una vez pasado a decimal.

Ej. : $2^{10010001}$. El exponente en formato exceso 127 es 10010001. En decimal, este valor representa 145, por lo tanto el exponente buscado será $145 - 127 = 18$.

Ej. Representar en formato de coma flotante en precisión simple el número decimal 178,125:

El número es positivo por lo tanto $b_{31} = 0$

$178.125 = 10110010,001_{1/2} = 1,0110010001 \cdot 2^7$. El exponente (7) será en la notación normalizada $127 + 7 = 134$ **10000110**.

El valor de la mantisa será: **011001000100000000000000**, por lo tanto, la representación será:

0 10000110 011001000100000000000000.

Ej. Cual será el valor decimal del número expresado en coma flotante de precisión simple siguiente:

10101000110111100100110000000000

$$\begin{array}{ccc}
 \underbrace{1}_{-} & \underbrace{01010001}_{81-127 = -46} & \underbrace{101111001001100000000000}_{1, 1011110010011}
 \end{array}$$

$$1, 1011110010011 * 2^{-46} = 11011110010011 * 2^{-59} = -2,467991089 * 10^{-14}$$

Precisión doble.

- El exponente se compone de 11 bits (2^{-1022} hasta 2^{1023}) y la mantisa se compone de 52 bits mas el BIT de signo.

1.4-OPERACIONES ARITMÉTICAS.

1.4.1-SUMA.

- Cuando realizamos una suma en sistema decimal (10), de forma manual lo que hacemos es una adición por columnas, anotamos un valor en la parte inferior de esa columna, el cual representa unas unidades “sobrantes” como resultado parcial, y anotamos un valor en la parte superior de la siguiente columna que representa lo que “nos llevamos”. Este último valor representa en realidad cuantas veces hemos llegado a la base (una vez en este caso).

$$\begin{array}{r} 1 \text{ arrastre} \\ 78 \\ + 19 \\ \hline 97 \end{array}$$

- La suma en cualquier otro sistema de numeración sigue el mismo principio que en el sistema decimal.

Binario.

Tabla de sumar:

		SUM	CAR
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

El Carry (“llevada”) nos indica cuantas veces se alcanza la base en el conteo. En la tabla de sumar para dos sumandos, podemos ver que solamente se alcanza la base (2) cuando sumamos $1 + 1$, por lo tanto el

Carry será 1 y no habrá ningún elemento sobrante (sum =0)

$$\begin{array}{r} \text{arrastre} \\ 1 \text{ } 11 \\ 00111001 \\ + 00100011 \\ \hline 01011100 \end{array}$$

$$\begin{array}{r} 1 \text{ } 1 \text{ arrastre} \\ 11111 \\ 1011 \\ + 1101 \\ + 1001 \\ + 0001 \\ + 1001 \\ \hline 101011 \end{array}$$

Octal

$$\begin{array}{r} 1 \text{ } 1 \text{ arrastre} \\ 47_{/8} \\ + 32_{/8} \\ \hline 101_{/8} \end{array}$$

Hexadecimal

$$\begin{array}{r} 1 \text{ arrastre} \\ E \text{ H} \\ + A \text{ H} \\ \hline 18 \text{ H} \end{array}$$

Hexadecimal

$$\begin{array}{r} 1 \text{ } 1 \text{ arrastre} \\ 3A.2B \text{ H} \\ + 21.F6 \text{ H} \\ + 5C.21 \text{ H} \\ \hline 5C.21 \text{ H} \end{array}$$

Realizar las siguientes sumas.

$$1010_{/2} + 101_{/2} \text{ (1111}_{/2}) \quad 100101_{/2} + 001001_{/2} \text{ (110011}_{/2}) \quad 1111_{/2} + 111_{/2} \text{ (10110}_{/2})$$

$$765_{/8} + 673_{/8} + 266_{/8} \text{ (2146}_{/8}) \quad 3AD9 \text{ H} + 45EF \text{ H} + ABCD \text{ H} \text{ (12C95 H)}$$

1.4.2-RESTA.

- Cuando realizamos una resta en forma manual, en la cual el minuendo es mayor que el sustraendo, el razonamiento que realizamos aplicado a la resta dada es el siguiente:
"de 1 a 10 **9** y me llevo 1, mas 2 son 3 al 9 van **6**. De 4 a 8 **4**. De 0 a 1 **1**..."

$$\begin{array}{r} 1890 \\ - 421 \\ \hline 1469 \end{array}$$

- Aplicando el mismo razonamiento, cuando el sustraendo es mayor que el minuendo (resultado negativo), tenemos unos resultados que en principio parecen cuanto menos, "galácticos", pero que analizados detenidamente, se puede observar que estos valores vienen dados en el complemento a la base (complemento a 10 en este caso). Si complementamos este resultado obtenemos el valor real en modo absoluto.

$$\begin{array}{r} 421 \\ - 1890 \\ \hline 8531 \end{array}$$

$$C_{10}(8531) = 10^4 - 8531 = 1469.$$

- La operación es análoga en otros sistemas de numeración.

Binario.

Tabla de restar:

		RES	CAR
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

El Carry se toma en este caso (0-1) solicitando un valor a la potencia superior.

Minuendo mayor que sustraendo

$$\begin{array}{r} 1010011 \\ - 11101 \\ \hline 1111 \text{ arrastre} \\ 0110110 \end{array}$$

Minuendo menor que sustraendo

$$\begin{array}{r} 0001 \\ - 1011 \\ \hline 1111 \text{ arrastre} \\ 0110 \end{array}$$

En este caso el resultado viene dado en complemento a 2. Si complementamos este número obtenemos el valor absoluto real.

$$C_2(0110) = 2^4 - 0110 = 16 - 10 = 10 = 1010$$

Realizar las siguientes restas.

$$1011_{/2} - 1111_{/2} (0100_{/2})$$

$$110_{/2} - 1001_{/2} (1101_{/2})$$

$$11_{/2} - 1100_{/2} (10111_{/2})$$

1.4.3-SUMA Y RESTA BINARIA EN COMPLEMENTO A 2

Una forma de realizar sumas y restas es mediante el formato de complemento 2. Esta forma se puede adaptar fácilmente a la forma de trabajo de un computador. Los resultados de las operaciones en este sistema, vienen determinados por la capacidad de trabajo en cuanto al número de bits con los que trabaja la máquina. La interpretación de los resultados la realiza el programa, siguiendo el estado de unos bits bandera (*flags*). Estos *flags* son el BIT de Acarreo (*Carry*), **C**; el BIT de Desbordamiento (*Overflow*), **V** y el BIT de Negativo (*Negative*), **N**.

- ❑ **C** se activará siempre que exista acarreo en el MSB de los operandos.
- ❑ **V** se activará cuando al trabajar con números con signo, el resultado no pueda ser contenido en los bits que sirven como magnitud. Si se trabaja sin signo, este BIT carece de sentido.

Suponiendo que trabajamos con una máquina cuya capacidad es de 8 bits, el rango de valores de trabajo será entre -128 hasta +127, cuando trabajamos con signo, y desde 0 hasta 255 si trabajamos sin signo. El flag V se activará cuando el valor a representar esté fuera del rango indicado para el trabajo con signo.

Esta bandera se activará por lo tanto en los casos en los que los números sean los dos positivos o los dos negativos y se cumpla lo anterior.

- ❑ **N** se activará cuando el valor del MSB sea 1.

Ej. : SUMA.

Dos números positivos.		
11111		
01000101	69	
00011011	27	
01100000	96	
C V N		
0 0 0		

Dos números negativos.		
1 1111111		
10111011	-69	
11100101	-27	
1 10100000	-96	
C V N		
1 0 1		

Un número positivo y otro negativo.		
NEG > POS:		
11 1		
10011010	-102	
00011011	27	
10110101	-75	
C V N		
0 0 1		
NEG < POS:		
111		
01100110	102	
10110001	-79	
1 00010111	23	
C V N		
1 0 0		

En los dos casos de abajo, se deben realizar las correcciones oportunas al resultado, interpretando los flags de V y N.

Dos números positivos.		
1 111		
01100110	102	
01001111	79	
10110101	181	
C V N		
0 1 1		

Dos números negativos.		
11		
10011010	-102	
10110001	-79	
1 01001011	-181	
C V N		
1 1 0		

RESTA. El procedimiento es análogo al realizado para la suma sin mas que tener en cuenta que la resta se reduce a sumar al minuendo el complemento a 2 del sustraendo.

1.4.4-MULTIPLICACIÓN BINARIA.

Tabla de multiplicar:

		MUL	CAR
0	0	0	0
0	1	0	0
1	0	0	0
1	1	1	0

$$\begin{array}{r}
 1101101 \\
 \times 10001 \\
 \hline
 1101101 \\
 0000000 \\
 0000000 \\
 0000000 \\
 1101101 \\
 \hline
 11100111101
 \end{array}$$

$$\begin{array}{r}
 1.01 \\
 \times 10.1 \\
 \hline
 101 \\
 101 \\
 \hline
 11.001
 \end{array}$$

$$\begin{array}{r}
 1101101 \\
 \times 10001 \\
 \hline
 1101101 \\
 1101101 \\
 \hline
 11100111101
 \end{array}$$

- En la práctica se copia el multiplicando tantas veces como unos tenga el multiplicador, teniendo en cuenta los desplazamientos a la izquierda debidos a los ceros del multiplicador.

Efectuar por pasos la siguiente multiplicación: $110110_2 \times 110011_2$ (101011000010)

1.4.5-DIVISIÓN BINARIA.

$$\begin{array}{r}
 \overline{110110111} \overline{) 1101} \\
 \underline{1101} \downarrow \downarrow \downarrow \downarrow \\
 000010111 \\
 \underline{1101} \\
 01010
 \end{array}$$

- Consiste en ir restando el divisor del dividendo por partes. Cada vez que se pueda restar (dividendo > divisor) se pone un uno en el cociente. Si no se puede restar (dividendo < divisor) se pone cero en el cociente y se toma la cifra siguiente del dividendo hasta conseguir que dividendo > divisor ...

Realizar por partes la siguiente división: $1001010_2 / 101_2$ ($1110 - 100$)

1.5-CÓDIGOS.

- ❑ Codificar se puede entender como establecer una correspondencia biunívoca entre los elementos de un conjunto A y los de otro B. A cada combinación de los elementos de A le corresponde una determinada combinación de los del B, según unas determinadas reglas prefijadas de antemano.

Código binario natural. Corresponde con el sistema de numeración binario.

Códigos binarios continuos. Son aquellos en los cuales las combinaciones correspondientes a números decimales consecutivos, son adyacentes, es decir, de una combinación a la anterior o siguiente solamente varía un único BIT.

Códigos binarios cíclicos. Son aquellos códigos continuos en los cuales la última combinación es adyacente con la primera. (código GRAY)

1.5.1-CÓDIGOS BCD.

- ❑ Estos códigos codifican un número decimal de tal manera que cada una de las cifras que componen el número es codificada por separado, en binario.
- ❑ Como tiene que codificar 10 símbolos (0 al 9), se necesitan 4 bits, los cuales dan lugar a 16 combinaciones (2^4) de las cuales para codificar en BCD solamente se utilizan 10.

Los códigos BCD se dividen en ponderados y no ponderados.

***PONDERADOS:** A cada dígito binario (BIT) se le asigna un peso o valor en función de la posición que ocupa dentro de la combinación. El número decimal representado se obtiene sumando los pesos asignados a las posiciones que tienen estado "1". (BCD-NATURAL, BCD-AIKEN, 5421, 7421).

Relación de pesos:

BCD-N:	BCD- AIKEN.	Las 5 primeras combinaciones del código Aiken, tienen el MSB con valor 0 y las 5 últimas el MSB es 1.
$2^3 \quad 2^2 \quad 2^1 \quad 2^0$ b3 b2 b1 b0	$2^1 \quad 2^2 \quad 2^1 \quad 2^0$ b3 b2 b1 b0	

5421	7421
$5 \quad 4 \quad 2 \quad 1$ b3 b2 b1 b0	$7 \quad 4 \quad 2 \quad 1$ b3 b2 b1 b0

dec	5421				7421			
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	0
3	0	0	1	1	0	0	1	1
4	0	1	0	0	0	1	0	0
5	1	0	0	0	0	1	0	1
6	1	0	0	1	0	1	1	0
7	1	0	1	0	1	0	0	0
8	1	0	1	1	1	0	0	1
9	1	1	0	0	1	0	1	0

*NO PONDERADOS: No existe relación de pesos. (BCD-X3).

BCD EXCESO 3. (BCD-X3).

La combinación binaria asignada a una cifra decimal D coincide con la asignada a D+3 en el código BCD-N.

(VER TABLA COMPARATIVA DE CÓDIGOS MAS ADELANTE)

1.5.2-CÓDIGO GRAY. (Código Binario Reflejado).

Es un código continuo cíclico. Al cambiar solamente un BIT de una combinación a la siguiente o anterior, se reducen las posibilidades de combinaciones transitorias no deseadas entre una y otra*.

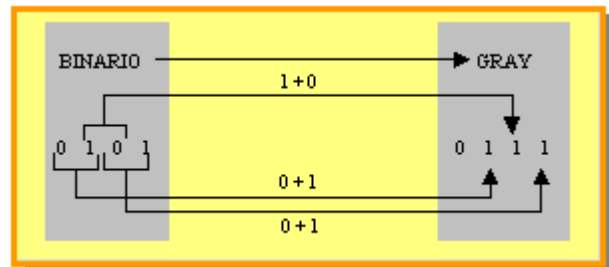
La línea gruesa entre el 7 y el 8 hace de eje de simetría horizontal. La parte superior, exceptuando el MSB, queda "reflejada" en la parte de abajo del eje.

0	0	0	0	0
1	0	0	0	1
2	0	0	1	1
3	0	0	1	0
4	0	1	1	0
5	0	1	1	1
6	0	1	0	1
7	0	1	0	0
8	1	1	0	0
9	1	1	0	1
10	1	1	1	1
11	1	1	1	0
12	1	0	1	0
13	1	0	1	1
14	1	0	0	1
15	1	0	0	0

Conversión Binario-Gray.

El MSB es el mismo en binario y en Gray. Para obtener los demás bits operamos como sigue:

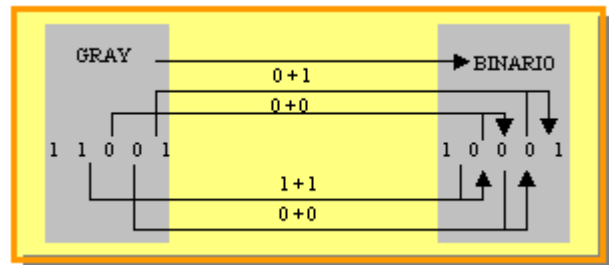
De izquierda a derecha, se van sumando, en binario, cada par de bits contiguos. El resultado de cada suma (sin tener en cuenta el carry), es el nuevo BIT en código Gray.



Conversión Gray-Binario.

El MSB es el mismo para ambos números. Para la obtención de los demás bits, el proceso es el siguiente:

De izquierda a derecha, a cada BIT del código binario generado, se suma el BIT de código Gray de la siguiente posición contigua (el carry no se tiene en cuenta).



Convertir de binario a Gray el número 10011_{10} [11010].
Convertir de Gray a binario el número 1011 [1101₁₀].

* El código Gray es utilizado en decodificadores ópticos. Con este código se evitan transitorios de corriente al cambiar de combinación consecutiva, al ser mínima la variación entre una y otra.

Tabla comparativa de algunos códigos y sistemas de numeración.

DEC.	OCTAL	HEX	BIN	BCD		
				BCD-N	BCD-AK	BCD-X3
0	0	0	0000	0000	0000	0011
1	1	1	0001	0001	0001	0100
2	2	2	0010	0010	0010	0101
3	3	3	0011	0011	0011	0110
4	4	4	0100	0100	0100	0111
5	5	5	0101	0101	1011	1000
6	6	6	0110	0110	1100	1001
7	7	7	0111	0111	1101	1010
8	10	8	1000	1000	1110	1011
9	11	9	1001	1001	1111	1100
10	12	A	01010	0001-0000	0001-0000	0100-0011
11	13	B	01011	0001-0001	0001-0001	0100-0100
12	14	C	01100	0001-0010	0001-0010	0100-0101
13	15	D	01101	0001-0011	0001-0011	0100-0110
14	16	E	01110	0001-0100	0001-0100	0100-0111
15	17	F	01111	0001-0101	0001-1011	0100-1000
16	20	10	10000	0001-0110	0001-1100	0100-1001
17	21	11	10001	0001-0111	0001-1101	0100-1010
18	22	12	10010	0001-1000	0001-1110	0100-1011

1.5.3-OTROS CÓDIGOS**Códigos de 5 bits:**

	2 ENTRE 5					51111					JOHNSON				
0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	1	0	0	0	0	1	0	0	0	0	1
2	0	0	1	1	0	0	0	0	1	1	0	0	0	1	1
3	0	1	0	0	1	0	0	1	1	1	0	0	1	1	1
4	0	1	0	1	0	0	1	1	1	1	0	1	1	1	1
5	0	1	1	0	0	1	0	0	0	0	1	1	1	1	1
6	1	0	0	0	1	1	1	0	0	0	1	1	1	1	0
7	1	0	0	1	0	1	1	1	0	0	1	1	1	0	0
8	1	0	1	0	0	1	1	1	1	0	1	1	0	0	0
9	1	1	0	0	0	1	1	1	1	1	1	0	0	0	0

Código biquinario

	5	0	4	3	2	1	0
0	0	1	0	0	0	0	1
1	0	1	0	0	0	1	0
2	0	1	0	0	1	0	0
3	0	1	0	1	0	0	0
4	0	1	1	0	0	0	0
5	1	0	0	0	0	0	1
6	1	0	0	0	0	1	0
7	1	0	0	0	1	0	0
8	1	0	0	1	0	0	0
9	1	0	1	0	0	0	0

1.5.4-CÓDIGOS ALFANUMÉRICOS.

- Estos códigos representan un carácter (letra, número, símbolo) o instrucciones para transferencia de información.
- El código alfanumérico más utilizado es el **ASCII** (*American Standard Code for Information Interchange*), que utiliza 7 bits para codificar cada carácter o instrucción. (tabla)

b3 b2 b1 b0	b6 b5 b4							
	000	001	010	011	100	101	110	111
0000	NUL	DLE	SPACE	0	@	P	'	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	`	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DEL

NUL	NUL/IDLE	SI	SHIFT IN
SOH	START OF HEADER	DLE	DATA LINK ESCAPE
STX	START OF TEXT	DC1-DC4	DEVICE CONTROL
ETX	END OF TEXT	NAK	NEGATIVE ACKNOWLEDGEMENT
EOT	END OF TRANSMISION	SYN	SYNCHRONOUS IDLE
ENQ	ENQUIRY	ETB	END OF TRANSMITTED BLOCK
ACK	ACKNOWLEDGEMENT	CAN	CANCEL (ERROR IN DATA)
BEL	AUDIBLE SIGNAL	EM	END OF MEDIUM
BS	BACK SPACE	SUB	SPECIAL SEQUENCE
HT	HORIZONTAL TAB	ESC	ESCAPE
LF	LINE FEED	FS	FILE SEPARATOR
VT	VERTICAL TAB	GS	GROUP SEPARATOR
FF	FORM FEED	RS	RECORD SAPARATOR
CR	CARRIAGE RETURN	US	UNIT SAPARATOR
SO	SHIFT OUT	DEL	DELETE/IDLE

Código EBCDIC. (*Extended Binary Coded Decimal Interchange Code*)

- Similar al ASCII pero compuesto por 8 bits. Introducido por IBM®, permite ampliar la cantidad de símbolos utilizados por el ASCII en los ordenadores.

1.5.5-CÓDIGOS DETECTORES Y CORRECTORES DE ERROR.

Se utilizan para garantizar la fidelidad y exactitud en la transmisión de información digital.

DISTANCIA: La distancia entre dos combinaciones binarias es la cantidad de bits que deben ser modificados en una para obtener la otra. La distancia mínima de un código es la menor de las distancias entre dos combinaciones cualesquiera pertenecientes a ese código.

- ❑ La condición necesaria para que un código detecte errores, es no utilizar todas las combinaciones posibles de ese código, como combinaciones portadoras de información; Si se hace esto, resulta que un error en una de ellas, origina otra combinación que también es válida dentro de ese código con lo cual, el error no se detecta.
- ❑ Según lo anterior, para que un código pueda detectar errores, su distancia mínima debe ser mayor que la unidad.

*DETECTORES:

PARIDAD.

- ❑ El código detector más utilizado es el **código de paridad**. Este código resulta de añadir un BIT extra al código que se quiere transferir; Este BIT se llama BIT de paridad y puede ser 0 o 1 según el método de paridad utilizado (par o impar):

Paridad par:

El número total de unos en la combinación del código (incluido el de paridad), debe ser par.

Paridad impar:

El número total de unos en la combinación del código (incluido el de paridad), debe ser impar.

BP	X-3			
1	0	0	1	1
0	0	1	0	0
1	0	1	0	1
1	0	1	1	0
0	0	1	1	1
0	1	0	0	0
1	1	0	0	1
1	1	0	1	0
0	1	0	1	1
1	1	1	0	0

Ej.

Código de paridad impar construido a partir del código BCD X-3

La distancia mínima en este código es de 2, por lo que se puede detectar un error cuando en una combinación varía sólo un BIT, pues la nueva combinación no pertenece al código.

El BIT de paridad puede colocarse también al final de la combinación. Por lo general se pone delante.

CYCLIC REDUNDANCY CHECK (CRC)

- ❑ El método CRC para la detección de errores es ampliamente utilizado en la comunicación digital. Este método utiliza un código polinómico aplicado a una secuencia de bits.
- ❑ La secuencia de bits se representa como $x^{k-1} + x^{k-2} + \dots + x + 1$, donde k es la longitud en bits de la secuencia binaria a representar, y donde el coeficiente de cada término tiene el valor 1 o 0 dependiendo del valor del correspondiente bit en dicha secuencia.

Por ejemplo la secuencia s = **11000110** se representa por $s(x) = x^7 + x^6 + x^2 + x$.

- La idea básica para la detección de errores consiste en codificar en el emisor la secuencia de bits a ser transmitida, **p**. Para realizar dicha codificación se hace uso de un polinomio generador **g(x)** de grado **r**, previamente establecido, y por lo tanto, conocido por transmisor y receptor. Mediante este polinomio generador se obtiene un código binario CRC (**c**), que se añade al final de la palabra binaria a transmitir. Todo el conjunto forma el mensaje codificado CRC que se transmite, (**m**).

La forma de generar el CRC es la siguiente:

- 1º. Se añade a la palabra origen a transmitir (**p**), después de su LSB, tantos ceros como indique el grado del polinomio generador **g(x)**. Es decir, si la palabra a transmitir (**p**) tiene de longitud **n** bits, y el polinomio generador es de grado **r**, al añadir los **r** ceros a **p**, obtenemos una nueva palabra de longitud **n+r**, que podemos representar en forma polinómica como $p(x) x^r$. La secuencia final a transmitir en representación polinómica **m(x)**, se obtiene mediante la operación:

$$m(x) = p(x) x^r + c.$$

- 2º. Se divide en formato módulo 2, $p(x) x^r$, entre el polinomio generador **g(x)**. De aquí obtenemos:

$$p(x) x^r / g(x) = Q + R / g(x)$$

(donde Q representa el cociente y R el resto de la división).

- 3º. El resto R de la división es el valor **c** a sumar a $p(x) x^r$ para obtener la palabra **m(x)** a transmitir.

En el emisor se cumple siempre que el resto de la división de **m(x)** entre **g(x)** es nulo.

En el receptor se realiza la operación de dividir **m(x)** entre **g(x)**. Si el resto de esta división es distinta de cero, indica que en la transmisión ha habido algún error.

Ejemplo: Se quiere transmitir la cadena 1010001101 codificada con CRC. El polinomio generador a utilizar es:

$$g(x) = x^5 + x^4 + x^2 + 1.$$

Tenemos **p** = 1 0 1 0 0 0 1 1 0 1; **g** = 1 1 0 1 0 1; **r** = 5.

Se añaden 5 ceros al final de la palabra **p**:

$$1\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0 = p'.$$

Se realiza la división de la secuencia **p'** entre **g** = 110101 en módulo 2. No se tienen en cuenta los acarreo.

Se suma el resto de la división a la secuencia **p'**. La cadena resultante es la cadena codificada CRC a transmitir.

$$m = 1\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0$$

1	0	1	0	0	0	1	1	0	1	0	0	0	0	0
1	1	0	1	0	1									
0	1	1	1	0	1	1								
	1	1	0	1	0	1								
	0	0	1	1	1	0	1	0						
			1	1	0	1	0	1						
			0	0	1	1	1	1	0					
					1	1	0	1	0	1				
					0	0	1	0	1	1	0	0		
							1	1	0	1	0	1		
							0	1	1	0	0	1	0	
								1	1	0	1	0	1	
								0	0	0	1	1	1	0

GENERADORES POLINÓMICOS MAS UTILIZADOS

Código CRC. Polinomio generador

CRC 12 $x^{12} + x^{11} + x^3 + x^2 + x + 1$

CRC 16 $x^{16} + x^{15} + x^2 + 1$

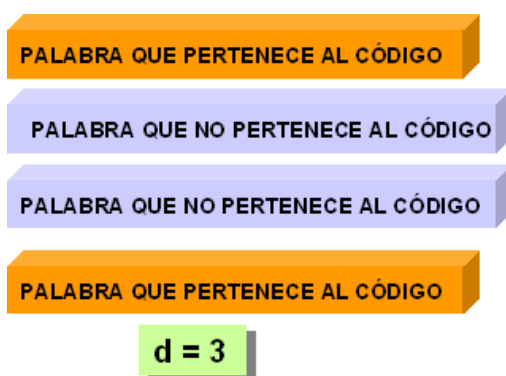
CRC 32 $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$

CHECKSUM.

- Este método de detección se utiliza con pequeños mensajes. Tiene mejores características de detección que el método de paridad, pero peores que el método CRC. Se forma sumando todas las palabras que deben ser incluidas en el chequeo, sin tener en cuenta cualquier acarreo, y añadiendo el resultado de esta suma como palabra al final del mensaje. El receptor realiza la misma operación para comparar el checksum recibido con el calculado por él mismo.

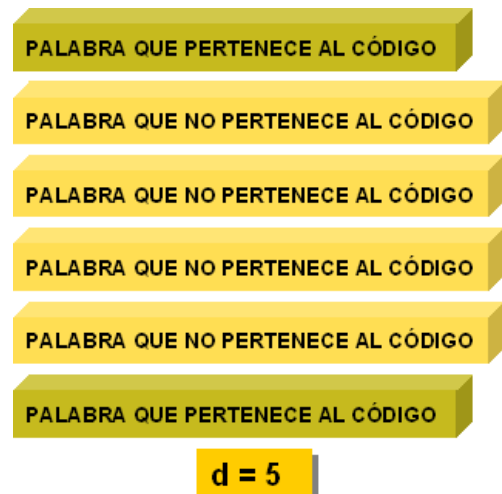
*CORRECTORES:

- Indican la existencia de errores y proporcionan información sobre las cifras binarias erróneas, permitiendo la corrección de las mismas.
- Un código con distancia mínima d , detecta errores de $(d-1)$ bits y corrige errores de $[(d-1)/2]$ bits.



Si tomamos un código con distancia mínima 3, cualquier combinación con un sólo BIT erróneo será adyacente a una y solo una combinación perteneciente al código por lo que será corregido el error al conocer de que combinación se trata. Si varían dos bits, se detectará el error pues la combinación resultante no pertenece al código, pero en este caso no se puede saber la combinación de la que partió el error con lo que no se puede corregir.

Si la distancia mínima es de 5, un error en un BIT dará una combinación que no pertenece al código pero que es adyacente a la combinación de la cual partió el error. Si el error es de dos bits, también se puede corregir pues la combinación resultante estará más próxima a la palabra de la que partió el error que a ninguna otra. Si el error es de tres bits, este se detecta pues la nueva combinación no pertenece al código, pero en este caso no se puede corregir debido a la ambigüedad en la "proximidad" a palabras pertenecientes al código. Para errores de cuatro bits sucede lo mismo.



CÓDIGO HAMMING.

	b7	b6	b5	b4	b3	b2	b1
0	0	0	0	0	0	0	0
1	0	0	0	0	1	1	1
2	0	0	1	1	0	0	1
3	0	0	1	1	1	1	0
4	0	1	0	1	0	1	0
5	0	1	0	1	1	0	1
6	0	1	1	0	0	1	1
7	0	1	1	0	1	0	0
8	1	0	0	1	0	1	1
9	1	0	0	1	1	0	0

Es de los códigos correctores más utilizados. Se parte del código BCD-N, formado por los bits b7-b6-b5-b3.

Se define un número binario de tres bits, $B_2B_1B_0$, donde cada BIT cumple las siguientes relaciones:

$$B_0 = b1 \text{ X-OR } b3 \text{ X-OR } b5 \text{ X-OR } b7$$

$$B_1 = b2 \text{ X-OR } b3 \text{ X-OR } b6 \text{ X-OR } b7$$

$$B_2 = b4 \text{ X-OR } b5 \text{ X-OR } b6 \text{ X-OR } b7$$

La obtención de los restantes bits del código (b1, b2 y b4) para obtener las 10 combinaciones de 7 bits que lo forman, se hace partiendo de los bits b7-b6-b5-b3 que forman el código BCD-N y teniendo en cuenta el criterio de que $B_2=B_1=B_0=0$.

Cuando se transmiten palabras de este código, y no existe error en la transmisión, resulta que $B_2B_1B_0 = 000$. Cuando hay error, el equivalente decimal de $B_2B_1B_0$ indica el BIT erróneo.

Ej.:

Supongamos que queremos transmitir el número 7 (0110100) y en el receptor tenemos la combinación 0100100,

$$B_0 = 0 \text{ X-OR } 1 \text{ X-OR } 0 \text{ X-OR } 0 = 1$$

$$B_1 = 0 \text{ X-OR } 1 \text{ X-OR } 1 \text{ X-OR } 0 = 0$$

$$B_2 = 0 \text{ X-OR } 0 \text{ X-OR } 1 \text{ X-OR } 0 = 1$$

$$B_2B_1B_0 = 101 = 5; \text{ b5 erróneo .}$$

Hay que observar que la distancia mínima en este código es 3, por lo tanto, puede detectar errores cuando éstos se producen como máximo en dos bits, dado que surgirán combinaciones que no pertenecen al código. Por otro lado, puede corregir errores cuando estos se producen en un solo BIT.

1.5.6-CÓDIGOS DE LÍNEA.

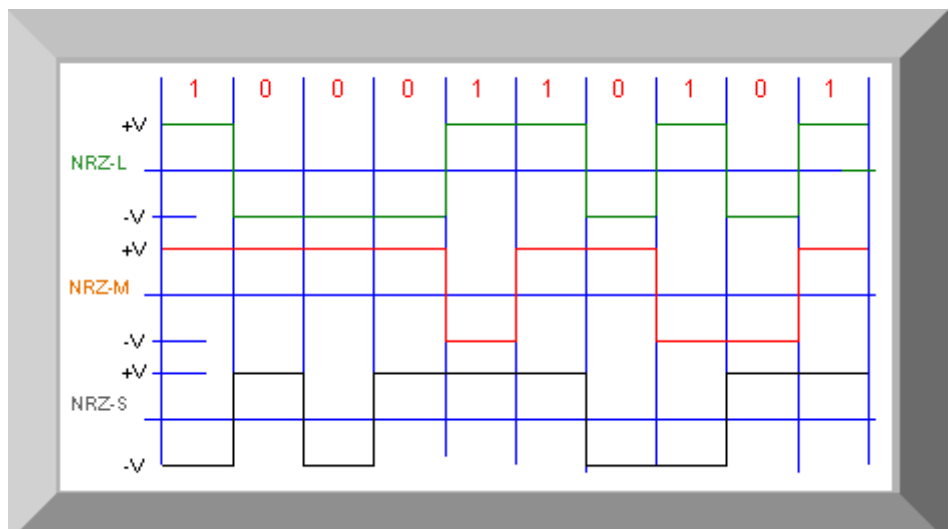
- ❑ Cuando se quiere transmitir los bits generados por una fuente, se deben emplear señales que se adapten al canal transmisor a utilizar, para conseguir que la información transmitida llegue al receptor con la máxima fidelidad (mínima distorsión) posible. La correspondencia que se establece entre los símbolos a transmitir y las señales que se transmiten se llama *codificación de línea*.
- ❑ En el diseño de un código de línea se buscan las siguientes características:
 - a.- *Eliminación de componente continua en la señal transmitida.*
Esto permite el acoplamiento en AC. Algunos sistemas que utilizan el acoplamiento por transformador, tienen poca sensibilidad a las bajas frecuencias.
 - b.- *Facilidad para la extracción del reloj.*
El receptor necesita tener una copia del reloj emisor para poder decodificar la señal recibida.
 - c.- *Detección de errores.*
Algunas implementaciones tienen la propiedad de detectar errores sin necesidad de incrementar el número de bits en el código primario.
 - d.- *Eficiencia espectral.*
Se trata de aprovechar mejor el ancho de banda para poder enviar más datos con la misma ocupación de un ancho determinado.
 - e.- *Inmunidad frente al ruido.*
Algunos códigos son más susceptibles de ser afectados por el ruido que otros. Los códigos bipolares se comportan mejor frente al ruido que los unipolares.
- ❑ Los distintos códigos de línea que se emplean se clasifican en: **NRZ** (*Non-Return-to-Zero*), **RZ** (*Return-to-Zero*), **Ø** (*Phase-Encoded*).

NRZ:

-NRZ-L (*Level*). Un uno binario se representa por un nivel de tensión alto, en tanto que un cero se representa por un nivel de tensión bajo. Hay un cambio de nivel siempre que los datos cambien de uno a cero o de cero a uno.

-NRZ-M (*Mark*).
El uno binario (marca) provoca un cambio de nivel. El cero no provoca cambio.
(Codificación diferencial).

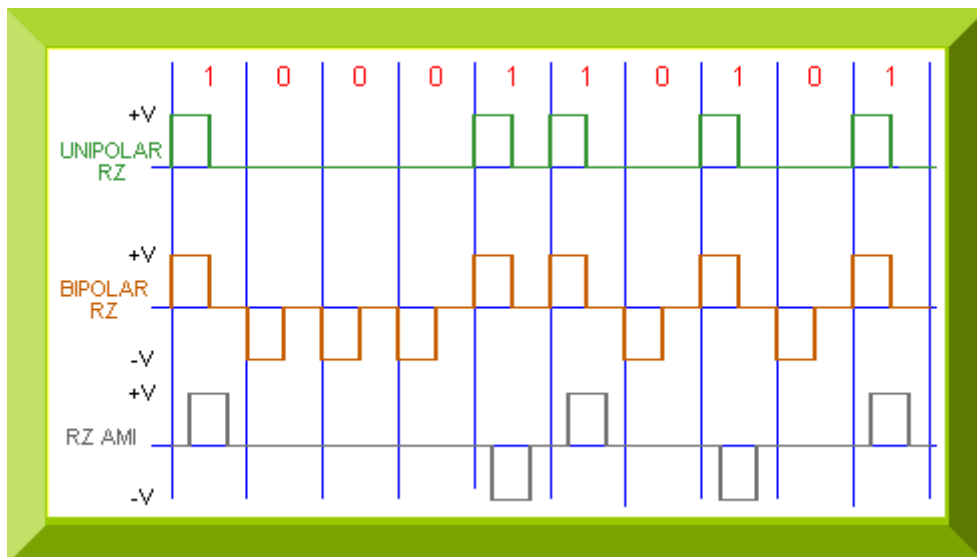
-NRZ-S (*Space*). El uno binario no provoca cambio de nivel en tanto que el cero si lo provoca.



RZ:

-Unipolar RZ. Un uno se representa mediante un impulso de duración igual a la de medio BIT. Un cero se representa por la ausencia de impulso.

-Bipolar RZ. Un uno se representa mediante un impulso positivo al comienzo del periodo de BIT, con una anchura de medio BIT; Un cero mediante un impulso negativo también al comienzo del periodo de BIT y de medio BIT de anchura.



-RZ-AMI. (*Alternate Mark Inversion*). Los unos se representan mediante impulsos alternados de igual amplitud, de medio BIT de duración y centrados en el semiperiodo BIT. Los ceros se representan por la ausencia de impulsos.

Ø:

(VER FIGURA EN LA SIGUIENTE PÁGINA)

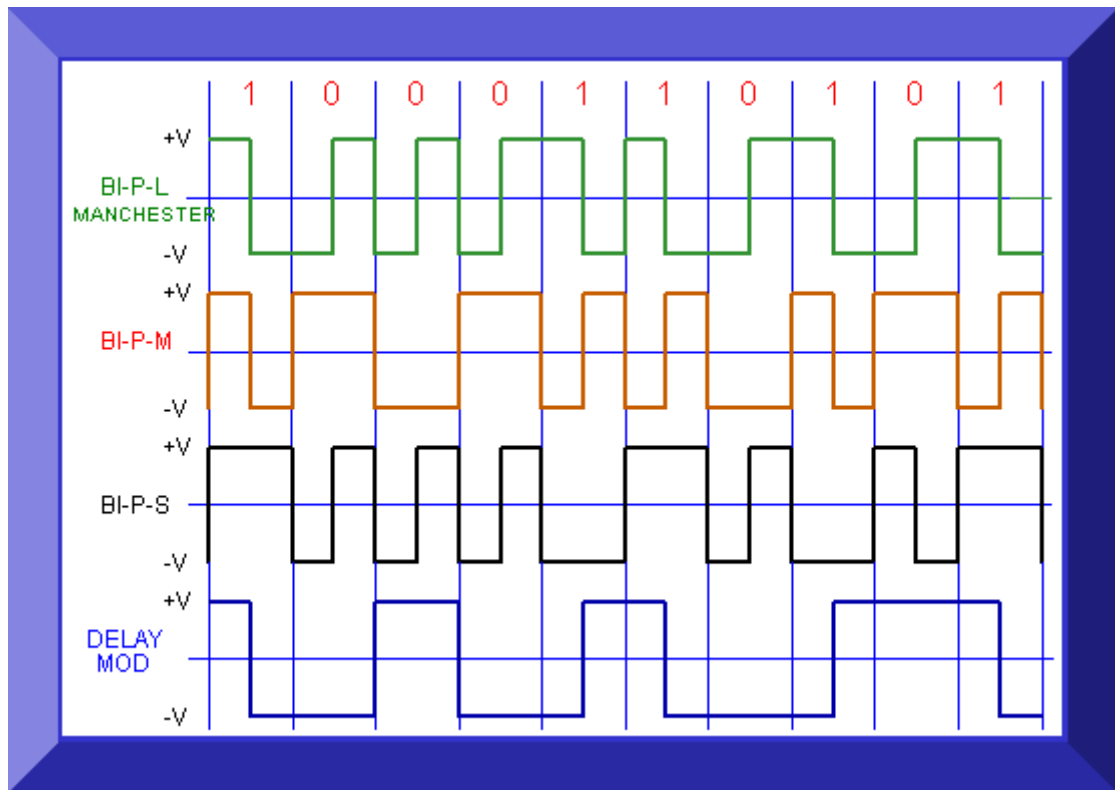
-Bi-Ø-L (*Bi-Phase-Level*). *Código Manchester*. Un uno binario se representa por un pulso, de medio BIT de ancho, durante la primera mitad del periodo de BIT. Un cero se representa por un pulso también de medio BIT de ancho, durante la segunda mitad del periodo de BIT.

También se puede observar que un uno provoca una transición descendente de la señal en la mitad del periodo de BIT, mientras que un cero lógico provoca una transición ascendente, también en la mitad del periodo de BIT.

-Bi-Ø-M (*Bi-Phase-Mark*). Existe una transición siempre al comienzo del intervalo de BIT. Un uno se representa por una segunda transición medio intervalo de BIT más tarde, y un cero no provoca transición.

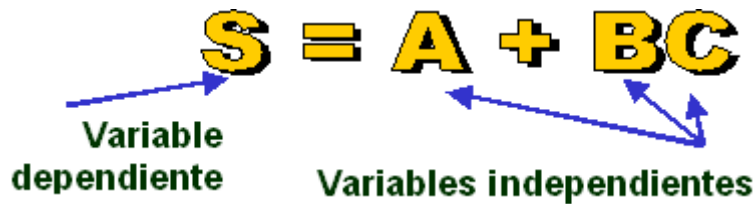
-Bi-Ø-S (*Bi-Phase-Space*). También existe una transición al comienzo del intervalo de BIT. Un cero se representa por una segunda transición medio intervalo de BIT más tarde, y un uno no provoca transición.

-Modulación por retardo (*código Miller*). Un uno se representa por una transición en el punto medio del intervalo de BIT. Un cero no produce transición a menos que esté seguido por otro cero, en cuyo caso se produce una transición al final del intervalo de BIT del primer cero.



1.6-FUNCIONES LÓGICAS.

- Se puede definir la función lógica como toda variable binaria cuyo valor depende de una expresión lógica formada por otras variables binarias relacionadas mediante sumas y/o productos y/o inversiones lógicas.



- En el álgebra booleana, solamente existen tres operaciones básicas:

- Suma lógica o función **OR**.
- Producto lógico o función **AND**
- Inversión lógica o función **NOT**

1.6.1-TABLA DE LA VERDAD DE UNA FUNCIÓN LÓGICA

- Es una representación gráfica de una función lógica. La tabla se forma con tantas columnas como variables (dependiente e independientes) contenga la función y con tantas filas como combinaciones posibles (2^n) se originen con el número (n) de variables independientes dadas.

B	A	S
0	0	?
0	1	?
1	0	?
1	1	?

S = f(A,B)

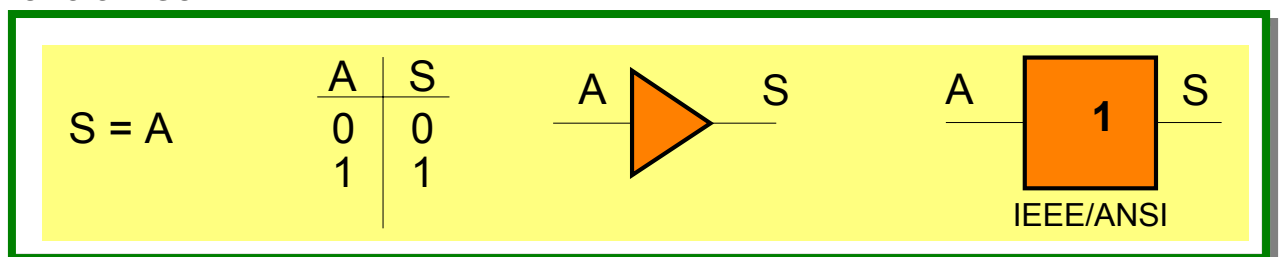
C	B	A	S
0	0	0	?
0	0	1	?
0	1	0	?
0	1	1	?
1	0	0	?
1	0	1	?
1	1	0	?
1	1	1	?

S = f(A,B,C)

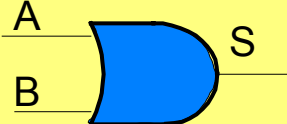
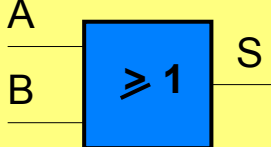
1.6.2-FUNCIONES LÓGICAS BÁSICAS.

En las figuras que aparecen a continuación, se representa para cada función, su expresión, tabla de la verdad, y el símbolo del circuito electrónico (PUERTAS LÓGICAS) que realiza dicha función. (se representa el símbolo tradicional y símbolo más moderno IEEE/ANSI)

FUNCIÓN IGUALDAD.

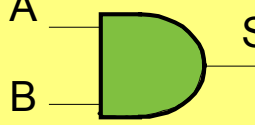
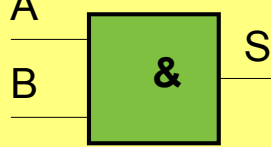


FUNCIÓN OR (SUMA LÓGICA).

$S = A + B$	A	B	S		
	0	0	0		
	0	1	1		
	1	0	1		
	1	1	1		

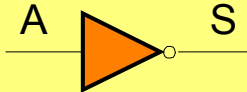
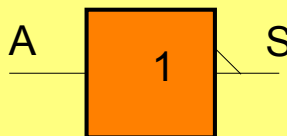
IEEE/ANSI

FUNCIÓN AND (PRODUCTO LÓGICO).

$S = A * B$	A	B	S		
	0	0	0		
	0	1	0		
	1	0	0		
	1	1	1		

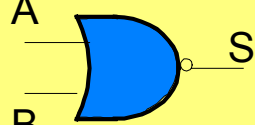
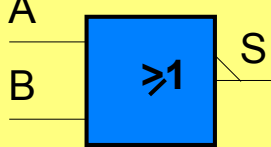
IEEE/ANSI

FUNCIÓN NOT (INVERSIÓN, NEGACIÓN, COMPLEMENTO LÓGICO).

$S = \overline{A}$	A	S		
	0	1		
	1	0		

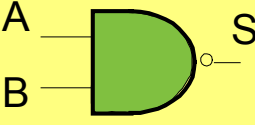
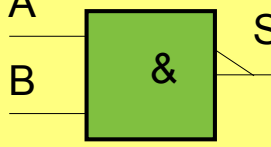
IEEE/ANSI

FUNCIÓN NOR (NOT-OR).

$S = \overline{A + B}$	A	B	S		
	0	0	1		
	0	1	0		
	1	0	0		
	1	1	0		

IEEE/ANSI

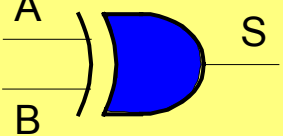
FUNCIÓN NAND (NOT-AND)

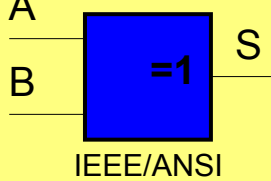
$S = \overline{A * B}$	A	B	S		
	0	0	1		
	0	1	1		
	1	0	1		
	1	1	0		

IEEE/ANSI

FUNCIÓN OR-EXCLUSIVA (X-OR)

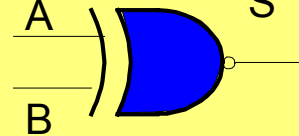
A	B	S
0	0	0
0	1	1
1	0	1
1	1	0

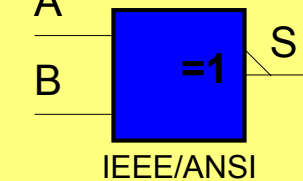




FUNCIÓN NOR-EXCLUSIVA (X-NOR)

A	B	S
0	0	1
0	1	0
1	0	0
1	1	1





NOTAS:

-Las tablas de la verdad están obtenidas empleando lógica positiva, esto es, un 1 representa un valor de tensión alto, mientras que un cero representa un nivel de tensión nulo. En lógica negativa, es justo lo contrario.

-Observar por ejemplo, que la tabla de la verdad de la función OR en lógica negativa, es la misma que la tabla de la verdad de la función AND si empleamos lógica positiva. Normalmente, se emplea la lógica positiva, y así aparece en los manuales de componentes digitales.

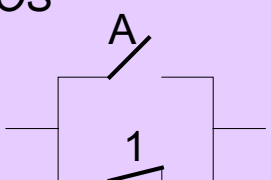
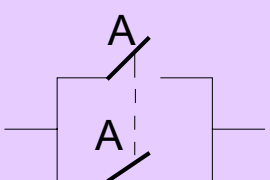
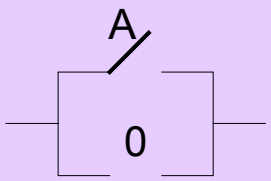
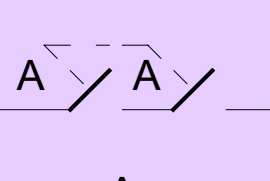
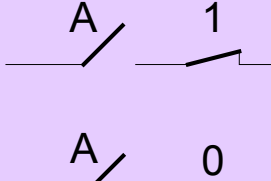
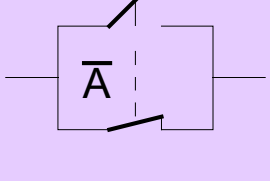
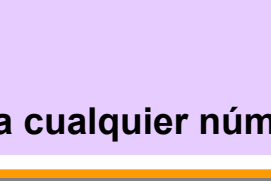
-Para indicar una negación de una variable o expresión, se utiliza indistintamente una línea encima de la variable o expresión, una almohadilla al final de la variable o expresión, o una comilla al final de la variable o expresión:

$\bar{A}$, $\overline{A + B}$, $A\#$, $(A + B)\#$, A' , $(A + B)'$

1.6.3-ÁLGEBRA DE BOOLE.

- Tiene como función definir una serie de símbolos para representar objetos o fenómenos que encadenados convenientemente dan lugar a expresiones matemáticas más o menos complejas.
- El álgebra de Boole opera sobre un conjunto que solamente tiene dos valores, el 0 y el 1. A los elementos de este conjunto se les llama variables binarias. Estas variables, relacionadas mediante operaciones lógicas (no cuánticas) cumplen:

POSTULADOS

$A + 1 = 1$		$A + A = A$	
$A + 0 = A$		$A * A = A$	
$A * 1 = A$		$A + \bar{A} = 1$	
$A * 0 = 0$		$A * \bar{A} = 0$	

$A \# \# = A$ para cualquier número par de inversiones.

PROPIEDADES:

De la suma y productos lógicos.

CONMUTATIVA.

$$A + B = B + A \quad A * B = B * A$$

ASOCIATIVA.

$$A + B + C = A + (B + C)$$

$$A * B * C = A * (B * C)$$

DISTRIBUTIVA.

$$A * (B + C) = A * B + A * C$$

$$A + (B * C) = (A + B) * (A + C)$$

TEOREMAS:

Ley de absorción.

$$A + AB = A$$

$$A * (A + B) = A$$

$$A + \bar{A}B = A + B$$

$$(A + \bar{B}) * B = AB$$

Leyes de De Morgan

$$\overline{A + B} = \bar{A} * \bar{B}$$

$$\overline{A * B} = \bar{A} + \bar{B}$$

- La demostración de los teoremas y de las propiedades puede realizarse mediante los postulados, mediante teoría de conjuntos o mediante tablas de la verdad.

Demostración de propiedades mediante la tabla de la verdad:

A	B	C	A+B	AB	B+C	BC	A+C	AC	ABC	A+B+C	A+BC	A(BC)	A+(B+C)	A(B+C)	AB+AC	(A+B)(A+C)
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	1	0	0	1	0	0	1	0	0	0
0	1	0	1	0	1	0	0	0	0	1	0	0	1	0	0	0
0	1	1	1	0	1	1	1	0	0	1	1	0	1	0	0	1
1	0	0	1	0	0	0	1	0	0	1	1	0	1	0	0	1
1	0	1	1	0	1	0	1	1	0	1	1	0	1	1	1	1
1	1	0	1	1	1	0	1	0	0	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Demostración de teoremas

$$A + AB = A: \quad A + AB = A(1 + B) = A \cdot 1 = A$$

$$A(A + B) = A: \quad A(A+B) = A + AB = A(1+B) = A \cdot 1 = A.$$

$$A + \bar{A}B = A+B: \quad A + \bar{A}B = (A+A)(\bar{A}+B) = 1(A+B) = A+B$$

$$B(A+\bar{B}) = AB: \quad BA + B\bar{B} = BA + 0 = BA = AB$$

Demostración de las Leyes de De Morgan.

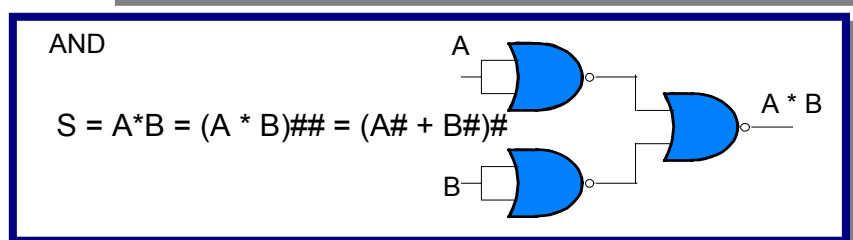
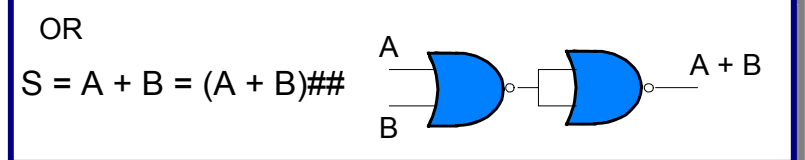
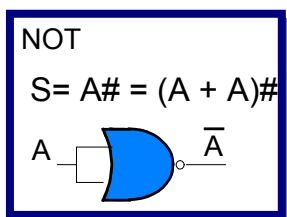
$$\overline{A+B} = \bar{A} \cdot \bar{B}$$

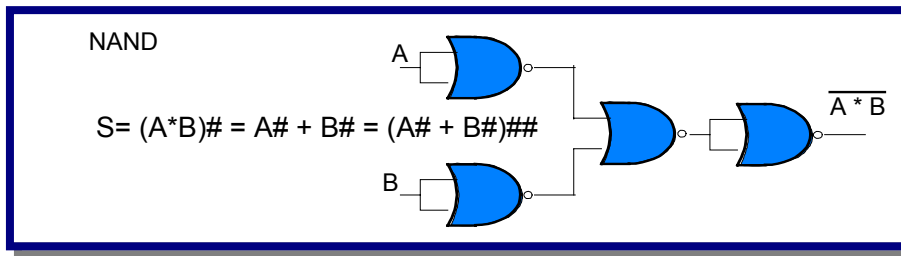
$$\overline{A \cdot B} = \bar{A} + \bar{B}$$

A	B	A+B	(A+B)#	AB	(AB)#	A#	B#	A#B#	A#+B#
0	0	0	1	0	1	1	1	1	1
0	1	1	0	0	1	1	0	0	1
1	0	1	0	0	1	0	1	0	1
1	1	1	0	1	0	0	0	0	0

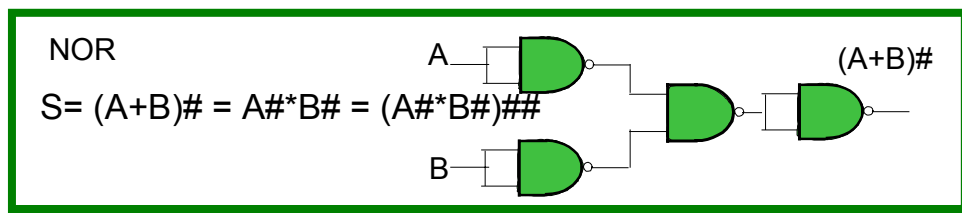
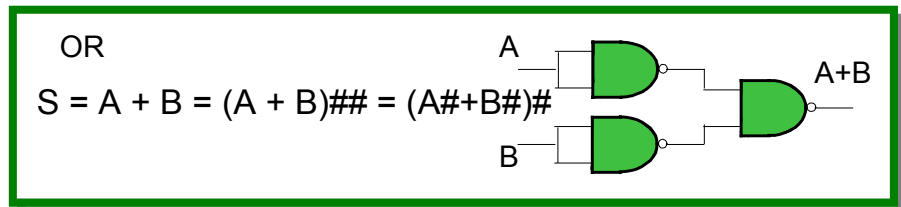
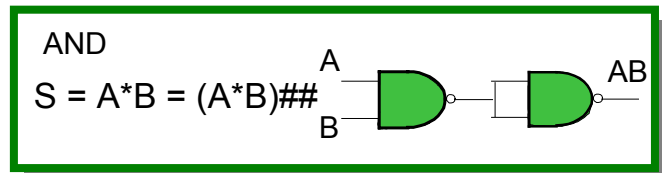
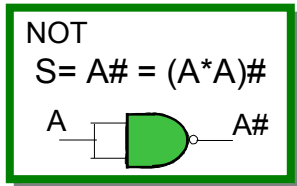
IMPLEMENTACIÓN DE FUNCIONES OR, AND, NOT Y NAND MEDIANTE PUERTAS NOR.

- Cualquier función puede ser representada mediante las funciones OR, AND y NOT, y estas pueden ser representadas o implementadas mediante funciones NOR y NAND, por lo que cualquier función puede realizarse mediante estas dos últimas funciones.





IMPLEMENTACIÓN DE FUNCIONES OR, AND, NOT Y NOR MEDIANTE PUERTAS NAND.



- La implementación de funciones con un solo tipo de puertas permite ahorrar elementos en el circuito.

Ej. Sea la función $S = AB + CD$. Para implementarla, necesitamos dos puertas AND y una puerta OR (dos IC diferentes).

Si la función la pasamos a puertas NAND, tenemos $S = [(AB \#)(CD \#)] \#$, tres puertas NAND, con lo que se podría realizar el montaje con un sólo IC.

1.6.4-SÍMBOLOS ALTERNATIVOS.

NOT				
AND				
OR				
NAND				
NOR				
	Tradicionales	Alternativos	IEEE/ANSI	Alternativos IEEE/ANSI

- Cada función básica puede ser representada mediante los símbolos tradicionales o IEEE/ANSI, y también, en ambos casos, mediante los símbolos alternativos.

Los símbolos alternativos se obtienen al representar la función que resulta de aplicar las leyes de De Morgan a las funciones básicas:

AND

$$AB = (A\# + B\#)\#$$

OR

$$A+B = (A\# B\#)\#$$

NAND

$$(AB)\# = A\# + B\#$$

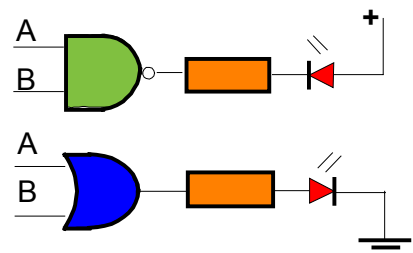
NOR

$$(A+B)\# = A\# B\#$$

1.6.5-INTERPRETACIÓN DE SÍMBOLOS Y ESQUEMAS. NIVELES ACTIVOS.

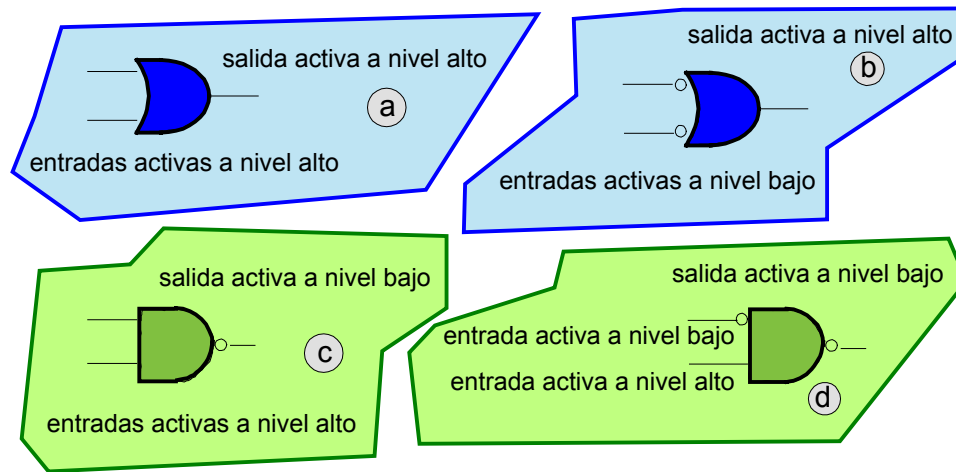
- La presencia de un círculo en un símbolo lógico indica que ese terminal donde está el círculo, es activo a nivel bajo. Si el terminal no tiene círculo, es activo a nivel alto.

Un efecto determinado se puede conseguir tanto con un nivel alto como con un nivel bajo. Por ejemplo, que un diodo LED se encienda, se puede conseguir conectando el diodo a una salida cuyo nivel activo sea el alto o el bajo. El diodo superior de la figura, se iluminará cuando la salida de la puerta NAND sea activa, esto es, cuando tenga un cero lógico.



El diodo inferior se iluminará cuando la salida de la puerta OR sea activa, esto es, cuando tenga un uno lógico.

La interpretación que se hace de los símbolos va en función de los niveles activos.



a. **La salida será activa cuando una u (OR) otra entrada sea activa.** En este caso eso significa que la salida tendrá un nivel alto, cuando cualquiera de las dos entradas tenga un nivel alto.

b. **La salida será activa cuando una u (OR) otra entrada sea activa.** En este caso eso significa que la salida tendrá un nivel alto cuando cualquiera de las dos entradas tenga un nivel bajo.

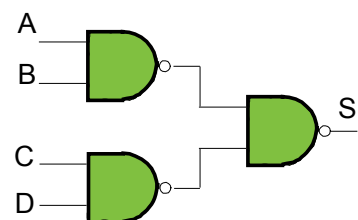
c. **La salida será activa cuando una y (AND) otra entrada sean activas.** En este caso eso significa que la salida tendrá un nivel bajo cuando las dos entradas tengan un nivel alto.

d. **La salida será activa cuando una y (AND) otra entrada sean activas.** En este caso eso significa que la salida tendrá un nivel bajo cuando una entrada (la superior) tenga un nivel bajo y (AND) la otra (la inferior) tenga un nivel alto.

- La utilización de los símbolos alternativos junto con los tradicionales facilita la traza del comportamiento de un circuito lógico.

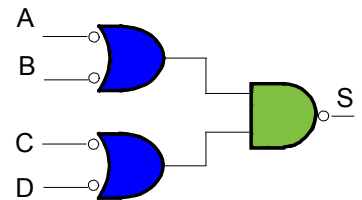
Ejemplo. Queremos hacer la traza del siguiente circuito, es decir, saber qué niveles han de estar en las entradas para que la salida sea activa.

Comenzamos la traza desde la salida a la entrada. Para que la salida sea activa (nivel bajo en este caso), las dos entradas de la puerta NAND de la salida, deben tener un nivel alto. Esto quiere decir que ninguna de las salidas de las puertas NAND que dan paso a las entradas, deben ser activas (0), por lo tanto, las entradas A **O** B **O** ambas deben ser inactivas (0), Y B **O** C **O** ambas deben ser inactivas (0).

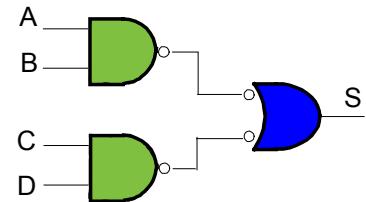


Si hacemos la traza del circuito usando los símbolos alternativos junto con los tradicionales, es decir, convirtiendo algunos tradicionales, según convenga, a sus equivalentes alternativos, de tal manera que en las conexiones intermedias del circuito haya coincidencia de niveles activos entre salidas de unas puertas y entradas de las siguientes (salidas con círculos a entradas con círculos y salidas sin círculos a entradas sin círculos), solamente tendremos que tener en cuenta las entradas del circuito para realizar la traza y no tendremos que preocuparnos de los niveles intermedios.

Para que la salida sea activa (0), $A \text{ O } B \text{ O } C \text{ O } D$ ambas deben ser 0 Y ambas también deben ser 0.



Si lo que queremos es saber que deben cumplir las entradas para que la salida sea alta (1) cambiamos la puerta de salida por su equivalente, buscando como siempre, la coincidencia de niveles activos intermedios en el circuito.



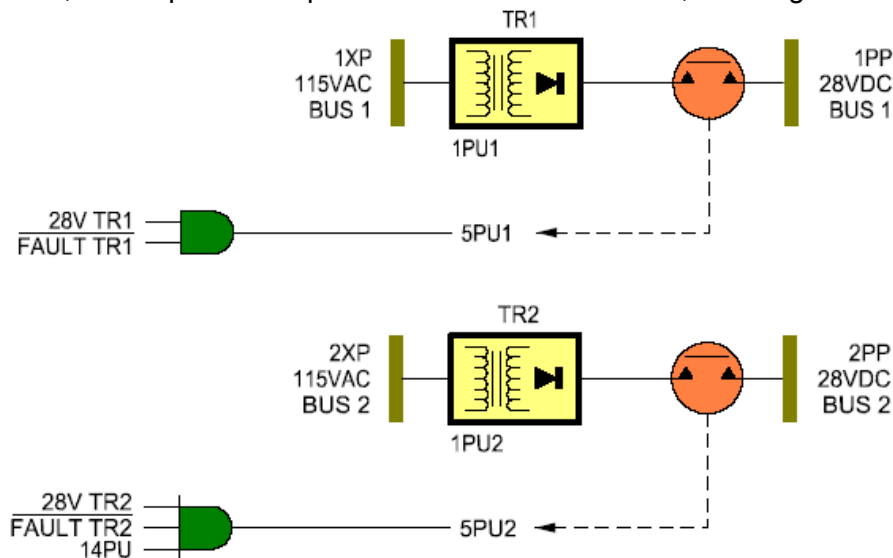
La salida será alta cuando $A \text{ Y } B$ sean altas $\text{O } C \text{ Y } D$ sean altas.

Observar que en los tres casos, se trata del mismo circuito, es decir, la misma función, simplemente lo adaptamos para facilitar la interpretación.

- En los esquemas eléctricos, donde se quiere representar la condición o condiciones que deben cumplir ciertas variables para que se realice una cierta acción, se hace uso de la simbología lógica que acabamos de ver. En unos casos, utilizando símbolos tradicionales junto con los alternativos, en otros casos indicando la condición de existencia o no de una variable a través de la negación correspondiente ($\#$, $'$, $\neg$), etc.

EJEMPLO.

El esquema siguiente muestra la lógica de control para poder alimentar el bus de 28VDC a través del TR asociado. Las condiciones que deben cumplirse para que se cierre el contactor correspondiente se muestran mediante los símbolos lógicos que ya conocemos. La condición en la que deben estar las variables (existencia o no, activación o no, etc.), también se representa mediante las barras de inversión o negación colocadas en la parte superior de las mismas. Según esto, la interpretación que debe hacerse del circuito, es la siguiente:

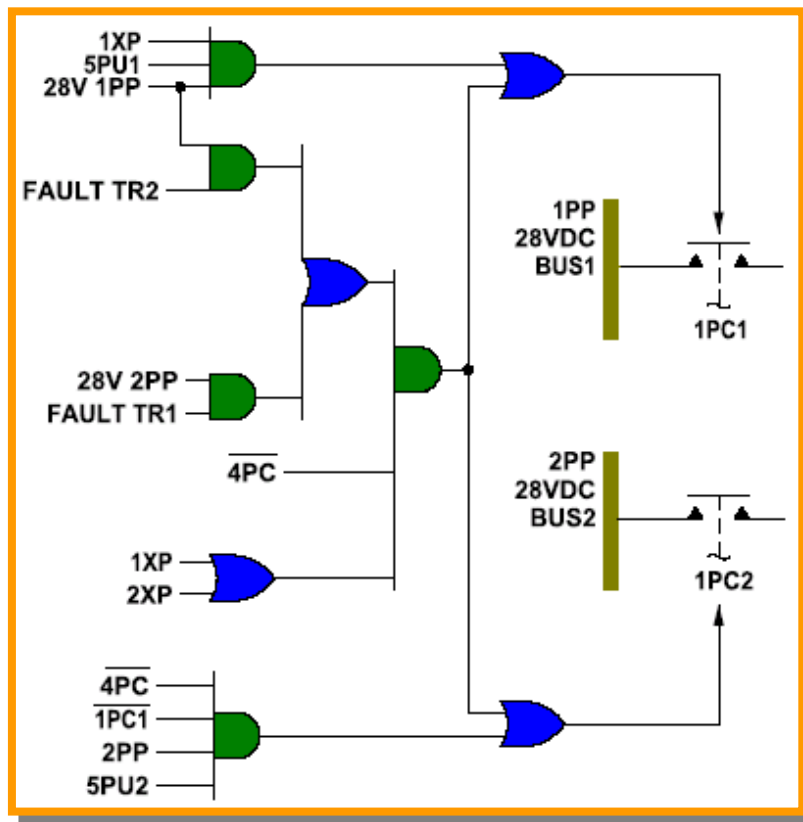


Para que se active el contactor **5PU1**, se debe cumplir que haya 28V del TR1 Y que **NO** haya una condición de fallo en el mismo TR1. Si no se cumple cualquiera de las dos condiciones, el contactor no se cierra.

Para que se active el contactor **5PU2**, se debe cumplir que haya 28V del TR2 Y que **NO** haya una condición de fallo en el mismo TR2 Y que exista 14PU (que esté energizado, activado, etc., según el elemento del que se trate). Si no se cumple cualquiera de las tres condiciones, el contactor no se cierra.

El siguiente circuito muestra también la lógica de control para que se activen dos contactores (1PC1 y 1PC2). La interpretación es la siguiente:

El contactor **1PC1** se va a activar cuando se cumpla:



por un lado,

[que exista (1XP Y 5PU1 Y 28V 1PP), estas tres condiciones simultáneamente,

O

por otro lado,

[(que exista 28V 1PP Y FALLO en TR2) O (que exista 28V 2PP Y FALLO en TR1)

Y

que **NO** exista 4PC

Y

que exista 1XP O 2XP

El contactor **1PC2** se va a activar cuando se cumpla:

por un lado,

[que exista (2PP Y 5PU2) Y **NO** existan (4PC Y 1PC1), estas cuatro condiciones simultáneamente]

O

por otro lado,

[(que exista 28V 1PP Y FALLO en TR2) O (que exista 28V 2PP Y FALLO en TR1)

Y

que **NO** exista 4PC

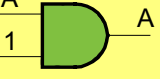
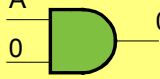
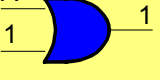
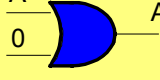
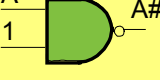
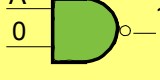
Y

que exista 1XP O 2XP

]

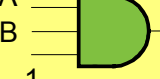
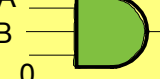
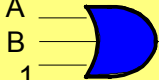
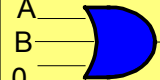
1.6.6-BLOQUEO Y PASO DE ENTRADAS.

- Podemos utilizar las puertas como elementos que bloquean o dejan pasar la información que tienen en una de sus entradas, o para dejar pasar la información pero invertida. Por ejemplo, cuando una puerta AND de dos entradas tiene un 0 en una de ellas, está bloqueando la otra entrada. Independientemente del valor que esta última entrada tenga, la salida siempre será 0.

AND	A	B	S		
	0	0	0		
	0	1	0		
	1	0	0		
	1	1	1		
OR	A	B	S		
	0	0	0		
	0	1	1		
	1	0	1		
	1	1	1		
NAND	A	B	S		
	0	0	1		
	0	1	1		
	1	0	1		
	1	1	0		
NOR	A	B	S		
	0	0	1		
	0	1	0		
	1	0	0		
	1	1	0		

Si una de las entradas tiene el valor 1, está dando paso a la salida a la información presente en la otra entrada.

En las tablas adjuntas, se dan diferentes casos de las salidas que se pueden dar según el valor de una de las entradas de las puertas de dos y tres entradas.

AND	A	B	C	S		
	0	0	0	0		
	0	0	1	0		
	0	1	0	0		
	0	1	1	0		
	1	0	0	0		
	1	0	1	0		
	1	1	0	0		
	1	1	1	1		
	1	1	1	1		
OR	A	B	C	S		
	0	0	0	0		
	0	0	1	1		
	0	1	0	1		
	0	1	1	1		
	1	0	0	1		
	1	0	1	1		
	1	1	0	1		
	1	1	1	1		
	1	1	1	1		
NAND	A	B	C	S		
	0	0	0	1		
	0	0	1	1		
	0	1	0	1		
	0	1	1	1		
	1	0	0	1		
	1	0	1	1		
	1	1	0	1		
	1	1	1	0		
	1	1	1	0		
NOR	A	B	C	S		
	0	0	0	1		
	0	0	1	0		
	0	1	0	0		
	0	1	1	0		
	1	0	0	0		
	1	0	1	0		
	1	1	0	0		
	1	1	1	0		
	1	1	1	0		

1.7-SIMPLIFICACIÓN DE FUNCIONES LÓGICAS.

1.7.1-FORMA CANÓNICA DE UNA FUNCIÓN BOOLEANA.

- Es todo producto de sumas o sumas de productos en los cuales aparecen todas las variables, bien en forma directa o bien en forma complementada, en cada uno de los términos que constituyen la expresión. Un término canónico contiene, por lo tanto, todas las variables.

$$S = \bar{A}\bar{B}C + \bar{A}B\bar{C} + ABC$$

Forma canónica

$$S = AB\bar{C} + AB + \bar{A}BC$$

Forma no canónica

MINTERMS.

- Son los términos canónicos cuya relación interna es la operación producto lógico.

Cuando una función viene dada como suma de productos, se dice que la función está dada en forma de MINTERMS.

$$S = A\bar{B}\bar{C}D + \bar{A}BC\bar{D} + \bar{A}BCD + A\bar{B}CD$$

MAXTERMS.

- Son los términos canónicos cuya relación interna es la operación suma lógica.

Cuando una función viene dada como producto de sumas, se dice que la función está dada en forma de MAXTERMS.

$$S = (A+\bar{B}+\bar{C}+\bar{D})(\bar{A}+\bar{B}+\bar{C}+D)(A+\bar{B}+C+D)$$

REPRESENTACIÓN SIMPLIFICADA DE MAXTERMS Y MINTERMS.

MINTERMS.

$$S = \sum_n (X_1, X_2, \dots, X_P)$$

MAXTERMS.

$$S = \prod_n (X_1, X_2, \dots, X_P)$$

Donde: **n** indica el número de variables.

X₁, X₂, ..., X_P representan los números decimales asociados al minterm o maxterm.

Para la representación del número decimal hay que tener presente las siguientes consideraciones:

MINTERMS: Una variable directa tiene el valor 1; Una variable negada tiene el valor 0.

MAXTERMS: Una variable directa tiene el valor 0; Una variable negada tiene el valor 1.

Las variables tienen asociado un peso dentro del término. Este peso de la variable se mantiene independientemente del lugar donde se coloque aquella dentro del término. Una variable representará siempre el MSB y otra el LSB.

EJEMPLOS:

$$S = \overline{A}\overline{B}\overline{C}\overline{D} + \overline{A}B\overline{C}\overline{D} + A\overline{B}\overline{C}\overline{D} + A\overline{B}C\overline{D} \quad S = \sum_4 (10, 6, 14, 11)$$

$$S = (A+\overline{B}+\overline{C}+\overline{D})(\overline{A}+B+\overline{C}+D)(A+\overline{B}+C+D) \quad S = \prod_4 (7, 10, 4)$$

Para ambos casos se ha considerado la variable A como la de más peso (MSB).

PASO DE UNA FUNCIÓN DE FORMA NO CANÓNICA A CANÓNICA.

- El método para la transformación de funciones consiste en la aplicación de los siguientes postulados: $A + A\# = 1$ y $A * A\# = 0$. Y también de la propiedad distributiva respecto del producto y la suma [$A+BC = (A+B)(A+C)$ y $A*(B+C) = AB + AC$].

Ej.

$$S = \overline{B}C + A\overline{C}$$

En el primer término falta la variable A: $\overline{B}C (A+\overline{A}) = \overline{A}\overline{B}C + A\overline{B}C$

En el segundo término falta la variable B: $A\overline{C} (B+\overline{B}) = AB\overline{C} + A\overline{B}\overline{C}$

Por lo tanto la función en forma canónica y minterms será:

$$S = \overline{A}\overline{B}C + \overline{A}B\overline{C} + AB\overline{C} + A\overline{B}\overline{C} = \sum_3 (5, 1, 6, 4)$$

Si lo que quiero es pasar la función a canónica en forma de maxterms:

$$S = \overline{B}C + A\overline{C} = (\overline{B}C + A)(\overline{B}C + \overline{C}) = (A+\overline{B})(A+C)(\overline{B}+\overline{C})(C+\overline{C}).$$

En el primer término falta C: $A+\overline{B}+C\overline{C} = (A+\overline{B}+C)(A+\overline{B}+\overline{C})$

En el segundo término falta B: $A+C+B\overline{B} = (A+B+C)(A+\overline{B}+C)$

En el tercer término falta A: $\overline{B}+\overline{C}+A\overline{A} = (A+\overline{B}+\overline{C})(\overline{A}+\overline{B}+\overline{C}).$

El cuarto término vale 1.

Por lo tanto la función en forma canónica y maxterms será:

$$S = (A+\overline{B}+C)(A+\overline{B}+\overline{C})(A+B+C)(A+\overline{B}+C) (A+\overline{B}+\overline{C})(\overline{A}+\overline{B}+\overline{C}) = \prod_3 (2, 3, 0, 2, 3, 7) = \prod_3 (2, 3, 0, 7)$$

Pasar a forma canónica, dada en minterms y maxterms, la siguiente función:

$$S = A(B+C). \quad (\text{minterms } \sum_3 (7,6,5), \text{ maxterms } \prod_3 (4, 3, 2, 1, 0))$$

PASO DE MINTERMS A MAXTERMS Y VICEVERSA MEDIANTE LA FORMA SIMPLIFICADA DE REPRESENTACIÓN.

- Lo único que tenemos que observar es de cuantas variables (n) se compone la función. Este número de variables origina un número de combinaciones que es 2^n , cuyo equivalente decimal va desde 0 hasta $2^n - 1$.

Una vez que tenemos la representación simplificada en uno u otro modo y teniendo en cuenta el punto anterior, lo que hacemos es observar que números decimales faltan en el modo dado, pues estos números, son los que representan la función en el otro modo. No olvidar el símbolo que representa una y otra forma.

Hay que tener siempre presente que el valor de las variables es diferente según consideremos minterms (variable directa vale 1 y variable negada vale 0) o maxterms (variable directa vale 0 y variable negada vale 1).

Por ejemplo, para cuatro variables, el número de combinaciones es de 16. Los números equivalentes en decimal de estas combinaciones van desde 0 hasta el 15. Entonces, si tenemos la función dada en forma de maxterms, lo que tenemos que hacer es mirar cuales son las combinaciones que faltan, y éstas serán las que forman los minterms:

$$S = \prod_4 (5, 8, 9, 15, 7, 0) \longrightarrow S = \sum_4 (1, 2, 3, 4, 6, 10, 11, 12, 13, 14)$$

$$S = \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}CD + \bar{A}B\bar{C}\bar{D} + \bar{A}B\bar{C}D + \bar{A}BC\bar{D} + \bar{A}BCD + A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D + AB\bar{C}\bar{D} + ABC\bar{D} + A\bar{B}C\bar{D} + A\bar{B}CD + AB\bar{C}D + ABCD$$

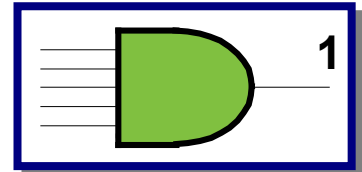
$$S = (A+B+C+\bar{D})(A+B+\bar{C}+D)(A+B+C+\bar{D})(A+\bar{B}+C+D)(A+\bar{B}+\bar{C}+D)(\bar{A}+B+\bar{C}+D)(\bar{A}+B+C+\bar{D})(\bar{A}+\bar{B}+C+D)(\bar{A}+\bar{B}+C+\bar{D})(\bar{A}+\bar{B}+\bar{C}+D)(\bar{A}+\bar{B}+\bar{C}+\bar{D})(\bar{A}+B+C+D)(\bar{A}+B+C+\bar{D})(\bar{A}+B+\bar{C}+D)(\bar{A}+B+\bar{C}+\bar{D})$$

1.7.2-OBTENCIÓN DE TÉRMINOS DE LA TABLA DE LA VERDAD.

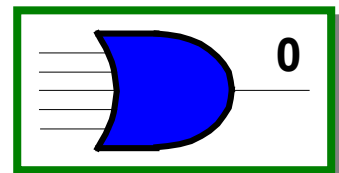
D	C	B	A	S
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

variables del maxterm que son 1, aparecen negadas en la expresión de la función.

- La función en forma de minterms, se obtiene tomando los términos que originan en la salida un nivel alto (1). Teniendo en cuenta que la relación interna de los minterms es el producto lógico y que cada minterm hace 1 en la salida, las variables del minterm que son 0, aparecen negadas en la expresión de la función.



- La función en forma de maxterms, se obtiene tomando los términos que originan en la salida un nivel bajo (0). Teniendo en cuenta que la relación interna de los maxterms es la suma lógica y que cada maxterm hace 0 en la salida, las variables del maxterm que son 1, aparecen negadas en la expresión de la función.



MINTERMS:

$$S = \bar{D}\bar{C}\bar{B}\bar{A} + \bar{D}\bar{C}\bar{B}A + \bar{D}\bar{C}B\bar{A} + \bar{D}\bar{C}BA + \bar{D}C\bar{B}\bar{A} + \bar{D}C\bar{B}A + \bar{D}CB\bar{A} + \bar{D}CBA$$

MAXTERMS:

$$S = (D+C+B+A)(D+C+B+\bar{A})(D+C+\bar{B}+\bar{A})(D+\bar{C}+\bar{B}+A)(D+\bar{C}+\bar{B}+\bar{A})(\bar{D}+C+\bar{B}+A)(\bar{D}+C+\bar{B}+\bar{A})(\bar{D}+C+B+A)(\bar{D}+C+B+\bar{A})(\bar{D}+\bar{C}+B+A)(\bar{D}+\bar{C}+B+\bar{A})(\bar{D}+\bar{C}+\bar{B}+A)(\bar{D}+\bar{C}+\bar{B}+\bar{A})(\bar{D}+\bar{C}+\bar{B}+A)(\bar{D}+\bar{C}+\bar{B}+\bar{A})$$

REPRESENTACIÓN SIMPLIFICADA:

$$S = \sum_4 (2, 4, 5, 8, 9, 13, 15).$$

$$S = \prod_4 (0, 1, 3, 6, 7, 10, 11, 12, 14)$$

Dec	C	B	A	S
0	0	0	0	1
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

$$S = \sum_3 (0, 1, 3, 7)$$

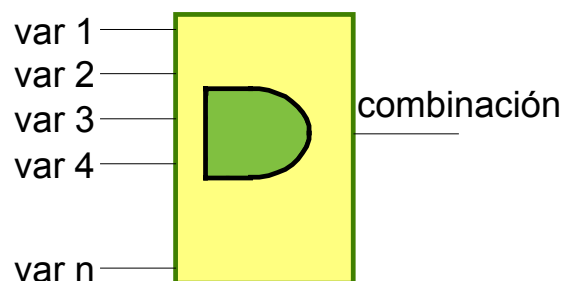
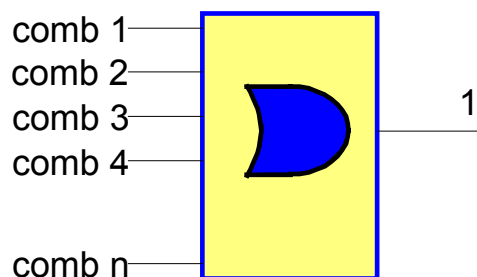
$$S = \prod_3 (2, 4, 5, 6)$$

1.7.3-INTERPRETACIÓN DE LOS DATOS DE LAS TABLAS DE LA VERDAD.

Minterms.

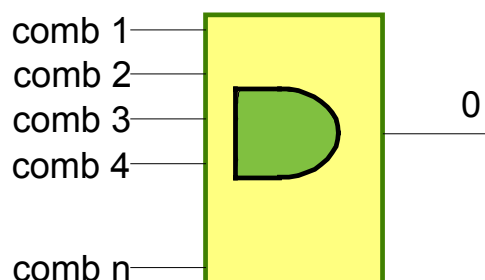
- Los minterms, son aquellos términos de la tabla de la verdad que hacen que la salida sea un uno lógico. Esto es, la salida tiene que ser 1 cuando cualquiera de las combinaciones de la tabla dé como resultado 1, por lo tanto, la relación entre las combinaciones debe ser una suma lógica, OR (cuando cualquiera de las combinaciones sea 1, la salida será 1).

Por otro lado, todas y cada una de las variables que componen la combinación debe cumplirse para obtener el 1 (AND).

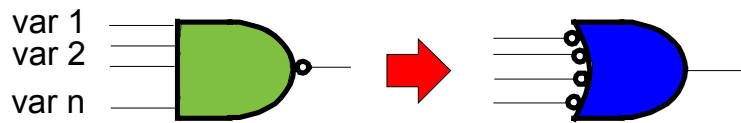


Maxterms.

- Los maxterms, son aquellos términos de la tabla de la verdad que hacen que la salida sea un cero lógico. Esto es, la salida tiene que ser 0 cuando cualquiera de las combinaciones de la tabla dé como resultado 0, por lo tanto, indica de forma implícita que la relación entre combinaciones debe ser un producto lógico, AND (cuando cualquiera de las combinaciones sea 0, la salida será 0).



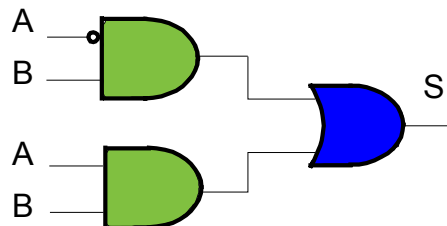
Por otro lado, todas y cada una de las variables que componen la combinación debe cumplirse para obtener el 0 (NAND). Haciendo uso del álgebra de Boole, la función NAND vemos que es la función OR con las entradas invertidas, por esto, el valor de las variables es justo el complementario respecto a los minterms, cuando pasamos a decimal el valor de los maxterms.



Ej.

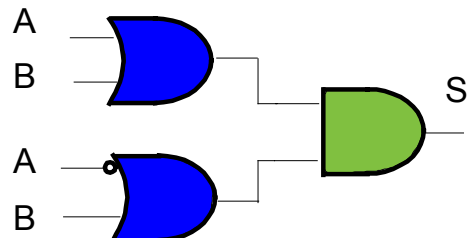
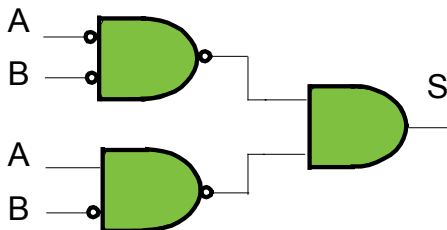
A	B	S
0	0	0
0	1	1
1	0	0
1	1	1

Minterms.



$$S = \bar{A}B + A\bar{B}$$

Maxterms



$$S = \overline{\bar{A}B} \cdot \overline{A\bar{B}} =$$

$$(A+B)(\bar{A}+\bar{B})$$

1.7.4-SIMPLIFICACIÓN DE FUNCIONES.

- ❑ Partiendo de una función lógica, tenemos que hallar otra, equivalente a la primera, pero con menor número de términos y menor número de variables por término que la función de la que se parte.
- ❑ Para la simplificación podemos hacer uso del método algebraico, basado en la aplicación de propiedades, teoremas y postulados del álgebra de **Boole**. También se puede realizar la simplificación utilizando el método gráfico de **Karnaugh** y el método numérico de **Quine McCluskey**.

MÉTODO ALGEBRAICO.

- ❑ Se utiliza en funciones sencillas, de pocas variables. Su aplicación en funciones complejas se ve dificultado debido a la imposibilidad de distinguir las circunstancias específicas favorables a la aplicación de una determinada propiedad o teorema del álgebra.

$$S = A + B (\overline{\overline{A+C}}) = A + B (\overline{\overline{A} * \overline{C}}) = A + B (\overline{A * C}) = A + B (AC) = A (1 + BC) = A$$

MÉTODO GRÁFICO DE KARNAUGH.

REPRESENTACIÓN.

- ❑ Este método es muy útil hasta cuatro variables. Para mas de cuatro variables se vuelve complicado. El formato básico consiste en una tabla con tantas casillas como combinaciones se puedan dar según el número de variables independientes implicadas en la función; es decir, para n variables, el número de casillas o celdas será 2^n y la distribución gráfica será:

Dos variables
Cuatro casillas

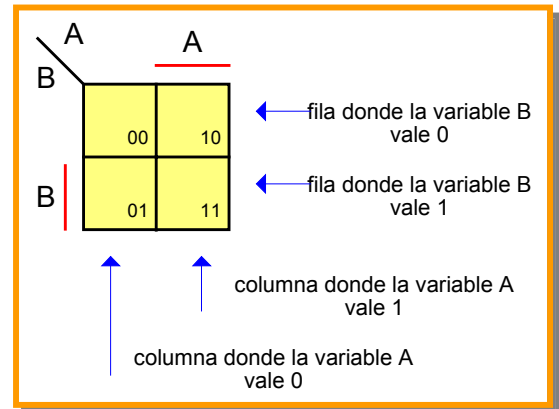
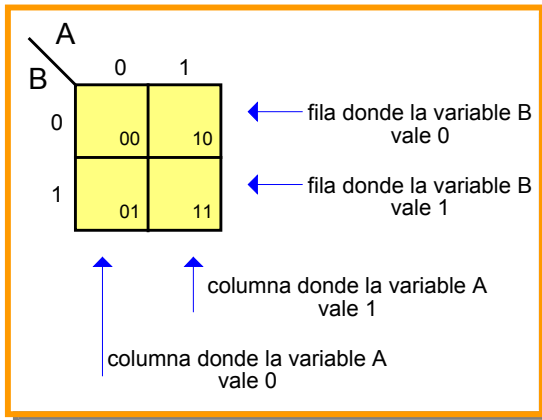
Tres variables
Ocho casillas

Cuatro variables
Dieciseis casillas

Cada fila y columna representan unos valores determinados de las variables de la función. Cada una de las celdas representa por lo tanto, una de las 2^n posibles combinaciones que se pueden dar con todas las variables.

Al situar las variables sobre la tabla, hay que tener en cuenta que entre dos casillas cualesquiera, comparadas en sentido horizontal o vertical, debe existir adyacencia algebraica, es decir, entre la combinación que representa una celda y la combinación que representa la otra celda, solamente puede haber una diferencia en el valor de una variable.

Representación para dos variables.

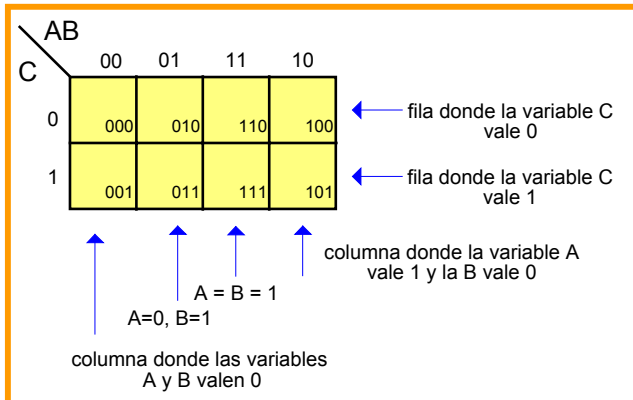


Una forma de identificar cada casilla con cada combinación, es señalar cada fila y cada columna con un valor binario. La intersección de fila y columna, dará como resultado el valor de la casilla correspondiente (figura superior izquierda).

También se puede hacer uso de las letras de las variables y colocar estas letras encima de las columnas donde su valor sea un uno lógico, y al lado de las filas donde también, su valor sea un uno lógico (figura de la derecha). Está claro que en la fila y columna donde no hay letra de variable, el valor de esa fila y columna es de cero lógico.

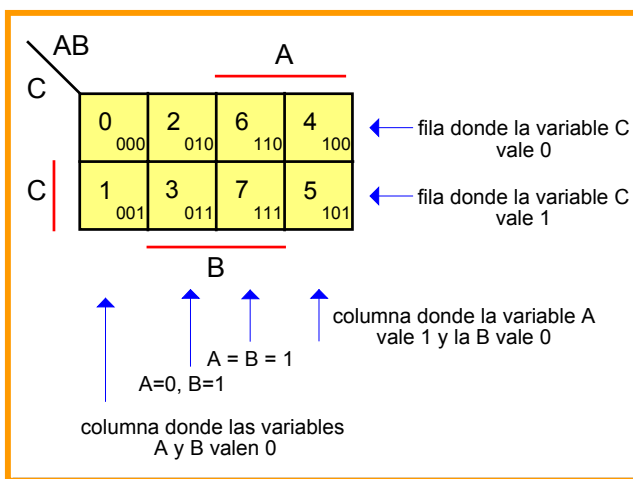
En los casos mostrados, a las columnas se les ha asignado el valor de la variable A y a las filas el valor de la variable B.

Representación para tres variables.



Para tres variables, se sigue el mismo procedimiento, teniendo cuidado de que los valores que se dan a las filas y columnas tienen que ser de tal forma que el valor final de las casillas, tienen que cumplir con la adyacencia algebraica.

En las figuras de la izquierda, a las filas se les ha asignado el valor de la variable C, y a las columnas los valores de las variables A y B.



Es importante tener en cuenta el peso que cada variable tiene en la expresión que vamos a simplificar, pues según ese peso, el valor final de cada casilla será diferente. En las figuras, la variable con más peso es la A y la de menos peso la C (ABC)

También podemos asignar el valor decimal equivalente a cada casilla, para facilitar su identificación.

Representación para cuatro variables.

AB		A			
		00	01	11	10
CD	00	0000	4	1100	8
	01	1	5	13	9
	11	0011	0111	15	11
	10	2	6	14	1010

B

A=B=0 A=0, B=1 A=B=1 A=1, B=0

C=D=0
C=0, D=1
C=D=1
C=1, D=0

El procedimiento, también es el mismo que en los casos anteriores.

Las asignación de casillas está fijada según los pesos ABCD.

Como en los casos anteriores, cada celda puede ser identificada de forma decimal, con lo que se facilita el trabajo a la hora de colocar los términos (minterms o maxterms) en la tabla.

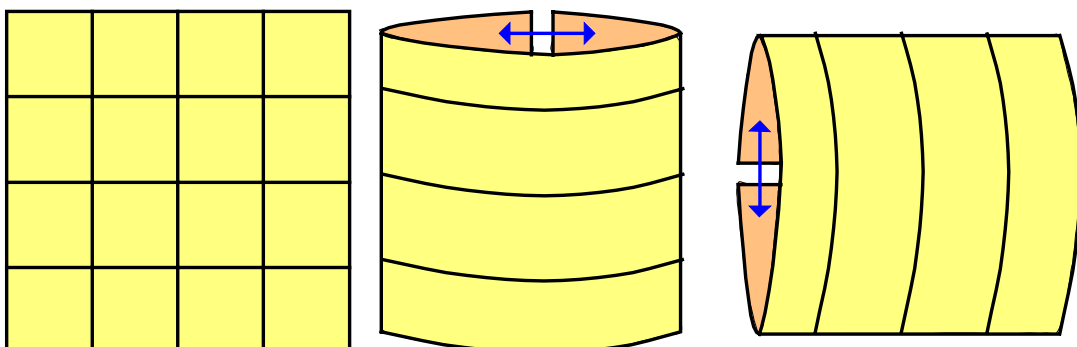
SIMPLIFICACIÓN.

Minterms.

- Se pone un 1 en la casilla de la tabla de Karnaugh, que representa a la combinación origen del minterm en la tabla de la verdad. En la tabla de Karnaugh, habrá tantos unos como minterms tengamos en la tabla de la verdad. A la hora de asociar la combinación que origina el minterm, con la casilla de la tabla de Karnaugh, las variables negadas valen 0 y las no negadas valen 1.

Una vez pasados todos los minterms a la tabla de Karnaugh, se hacen grupos que abarquen el mayor número de unos posible, teniendo en cuenta las siguientes consideraciones:

- El número de unos dentro de un grupo debe ser un valor de 2^n , con $n = 0, 1, 2, 3, \dots$
- Los unos que se toman dentro de un bucle han de estar en casillas adyacentes. Hay que observar que las casillas superiores de la tabla de Karnaugh son adyacentes con las inferiores de la misma columna y viceversa. Con las casillas de la izquierda y la derecha, ocurre lo mismo.
- Para lograr la máxima simplificación cada bucle debe tener el mayor número de unos posible y el número de bucles debe ser el mínimo, teniendo en cuenta que no debe quedar ningún uno que no pertenezca a algún bucle. Un uno puede pertenecer a más de un bucle.



- Para hacer la simplificación de términos, hay que observar la relación entre los bucles y las filas y columnas (variables). En la función final resultante, habrá tantos términos como bucles; un bucle representa un término. Si un bucle tiene los dos valores posibles de una variable, esta variable queda eliminada en el término. En los demás casos el bucle tendrá un valor de 0 (variable correspondiente negada), o 1 (variable sin negar).

Según lo anterior se puede observar que un bucle que contenga un sólo uno, no elimina ninguna variable para el término correspondiente. Un bucle con dos unos, elimina una variable en el término correspondiente; si el bucle tiene cuatro unos, elimina dos variables, con ocho unos, elimina 3 variables ...

Ej. Simplificar:

1º. OBTENGO LA FUNCIÓN LÓGICA DE LA TABLA DE LA VERDAD.

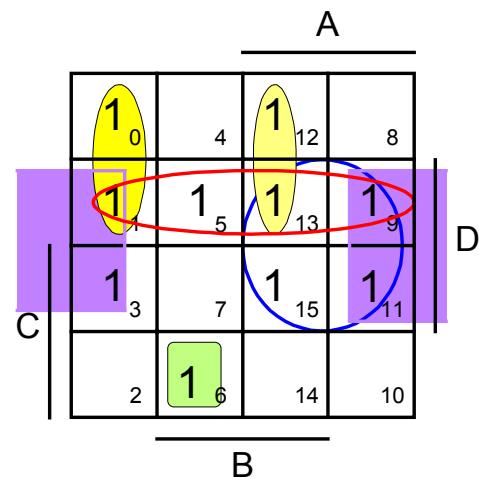
dec	A	B	C	D	S
0	0	0	0	0	1
1	0	0	0	1	1
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	1
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	1

$$S = \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}CD + \bar{A}B\bar{C}\bar{D} + \bar{A}B\bar{C}D + \bar{A}BC\bar{D} + \bar{A}BCD + A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D + AB\bar{C}\bar{D} + ABC\bar{D} + ABCD$$

De forma simplificada:

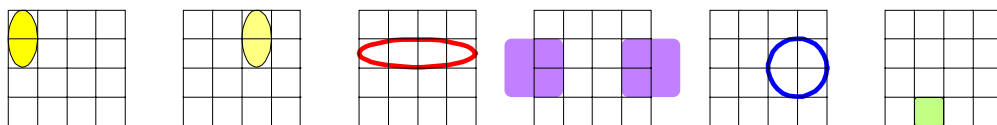
$$S = \sum_4 (0, 1, 3, 5, 6, 9, 11, 12, 13, 15)$$

2º SITUO LOS TÉRMINOS EN LA TABLA DE KARNAUGH.



Una vez situados los términos, los agrupo en bucles tal y como se ha indicado mas arriba. No debe quedar ningún término sin agrupar.

3º ESTUDIO LA RELACIÓN DE LOS BUCLES (GRUPOS) CON LAS VARIABLES Y OBTENGO LA FUNCIÓN SIMPLIFICADA.



$$S = \bar{A}\bar{B}\bar{C} + A\bar{B}\bar{C} + \bar{C}D + \bar{B}D + AD + \bar{A}BC\bar{D}$$

Maxterms.

- ❑ Se pone un 0 en la casilla de la tabla de Karnaugh que representa la combinación origen del maxterm en la tabla de la verdad. En la tabla de Karnaugh, habrá tantos ceros como maxterms tengamos en la tabla de la verdad. A la hora de asociar la combinación que origina el maxterm, con la casilla de la tabla de Karnaugh, en este caso las variables negadas valen 1 y las no negadas valen 0.
- ❑ El proceso que se sigue es el mismo que para la simplificación por minterms.

Ej. Simplificar:

dec	A	B	C	D	S
0	0	0	0	0	1
1	0	0	0	1	1
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	1
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	1

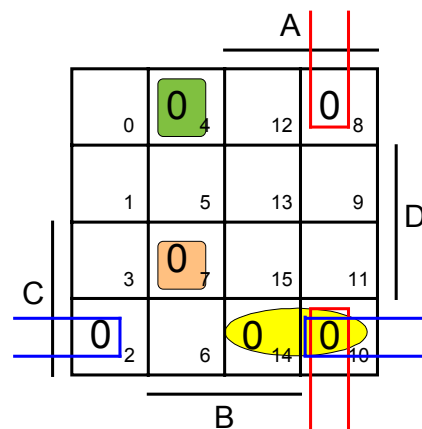
1º. OBTENGO LA FUNCIÓN LÓGICA DE LA TABLA DE LA VERDAD.

$$S = (A+B+\bar{C}+D)(A+\bar{B}+C+D)(A+\bar{B}+\bar{C}+\bar{D})(\bar{A}+B+C+D)(\bar{A}+B+\bar{C}+D)$$

De forma simplificada:

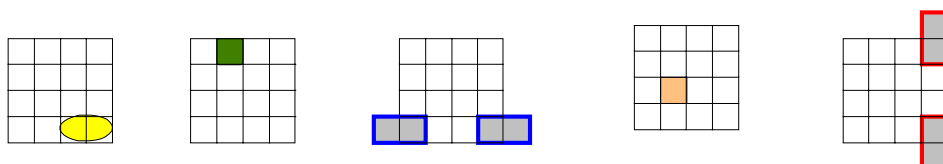
$$S = \Pi_4 (2, 4, 7, 8, 10, 14)$$

2º SITUO LOS TÉRMINOS EN LA TABLA DE KARNAUGH.



Agrupo los términos.

3º ESTUDIO LA RELACIÓN DE LOS BUCLES CON LAS VARIABLES Y OBTENGO LA FUNCIÓN SIMPLIFICADA.



$$S = \bar{A} + \bar{C} + D * A + \bar{B} + C + D * B + \bar{C} + D * A + \bar{B} + \bar{C} + \bar{D} * \bar{A} + B + D$$

Ej. Obtener los maxterms y minterms de la tabla de la verdad y simplificar las dos funciones.

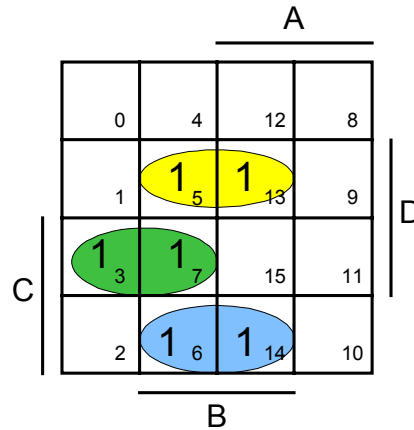
dec	A	B	C	D	S
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	0
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	0

Minterms.

$$S = \bar{A}\bar{B}CD + \bar{A}B\bar{C}D + \bar{A}BC\bar{D} + \bar{A}BCD + A\bar{B}\bar{C}D + ABC\bar{D}$$

también,

$$S = \sum_4 (3, 5, 6, 7, 13, 14)$$



$$S = \bar{A}CD + BCD + B\bar{C}D$$

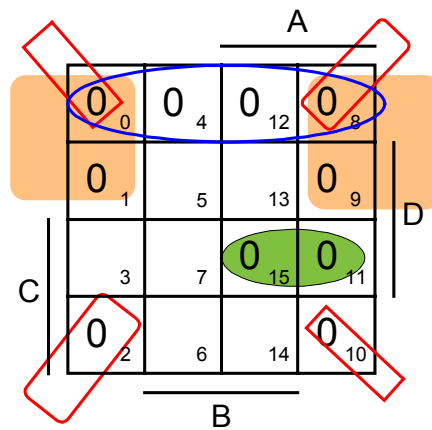
Maxterms.

$$S = (A+B+C+D)(A+B+C+\bar{D})(A+B+\bar{C}+D)(A+\bar{B}+C+D)(\bar{A}+B+C+D)(\bar{A}+B+C+\bar{D})(\bar{A}+B+\bar{C}+D)(\bar{A}+B+\bar{C}+\bar{D})(\bar{A}+\bar{B}+C+D)(\bar{A}+\bar{B}+C+\bar{D})(\bar{A}+\bar{B}+\bar{C}+D)(\bar{A}+\bar{B}+\bar{C}+\bar{D})$$

también,

$$S = \prod_4 (0, 1, 2, 4, 8, 9, 10, 11, 12, 15)$$

$$S = (B+C)(B+D)(C+D)(\bar{A}+\bar{C}+\bar{D})$$



Simplificar y expresar en forma de producto de sumas y suma de productos.

$$S = \sum_4 (1, 4, 5, 9, 10, 12, 13, 14) \quad [S = (B+C+D)(\bar{C}+\bar{D})(A+\bar{C}); \quad S = \bar{B}\bar{C} + \bar{C}D + ACD]$$

$$S = \prod_4 (0, 1, 2, 3, 7, 8, 9, 10, 12, 14) \quad [S = \bar{A}\bar{B}D + \bar{B}\bar{C}D + ACD; \quad S = (\bar{A}+D)(B+C)(A+\bar{C}+\bar{D})(A+B)]$$

FUNCIÓN EN FORMA NO CANÓNICA.

- Cuando la función no está especificada en forma canónica, se pasan los términos a las celdas de la tabla de Karnaugh teniendo en cuenta que cada término no canónico pertenece a mas de una celda; la variable o variables no especificadas en el término, pueden tener el valor 0 o el valor 1, originando varias combinaciones posibles para ese término.

Ej. $S = AB + AC$. Esta función, de cara a la colocación de los términos en la tabla puede ser:

$$S = AB(C + \bar{C}) + AC(B + \bar{B})$$

Es decir,

$$S = ABC + AB\bar{C} + ABC + A\bar{B}C = ABC + AB\bar{C} + A\bar{B}C$$

AB		A	
C		1	0
1	1	1	0
0	0	1	0

Para pasar los términos directamente (sin poner la función en forma canónica) a la tabla de Karnaugh, nos valemos de los valores que aparecen en la función, los cuales están claramente especificados. Para el ejemplo dado, el término AB indica que $A=B=1$, y C puede ser 0 o 1, por lo que en la tabla aparecen los unos de las casillas 110 (6) y 111 (7) donde se cumple $A=B=1$ y $C=0$ (6) y $C=1$ (7).

Para el término AC tenemos los unos de las casillas 101 (5) y 111 (7), dado que B puede ser 0 o 1.

Ej. $S = \bar{B}C + C\bar{D} + \bar{A}\bar{B}\bar{C}D + \bar{A}\bar{C}D$

AB		A	
CD		1	0
1	1	1	0
0	1	1	0

AB		A	
CD		1	0
1	1	1	0
0	1	1	0

$$S = \bar{A}\bar{C}D + \bar{B}C + C\bar{D}$$

Ej. $S = \bar{A}\bar{B}\bar{C}\bar{D} + B\bar{C}D + \bar{A}\bar{C} + A$

AB		A	
CD		1	0
1	1	1	0
0	1	1	0

AB		A	
CD		1	0
1	1	1	0
0	1	1	0

$$S = A + \bar{C}$$

FUNCIÓN INCOMPLETAMENTE ESPECIFICADA (REDUNDANCIAS O CONDICIONES “NO IMPORTA”).

- Una especificación incompleta, se presenta en la práctica de dos maneras. Unas veces, ocurre que simplemente no interesa el valor asumido por la función para ciertas combinaciones de las variables. Otras veces, se sabe que nunca podrán darse esas combinaciones de las variables. Esas combinaciones que nunca interesan o nunca podrán darse, se llaman redundancias o condiciones no importa.

Estas combinaciones se pasan a la tabla de Karnaugh mediante una X y a la hora de simplificar, las tomamos como ceros o como unos, según nos interese para simplificar lo máximo posible.

En la forma simplificada, estas combinaciones aparecen en un paréntesis con una “d” delante, como suma a la forma simplificada conocida.

Ej:

$$S = \sum_4 (1, 2, 5, 6, 9) + d(10, 11, 13, 14, 15)$$

$$S = \bar{C}D + C\bar{D} = C \text{ X-OR } D$$

AB		A	
CD			
	0	4	X ₁₂
	1	5	X ₁₃
	3	7	X ₁₅
	2	6	X ₁₄
			X ₁₁
			X ₁₀

CONVERSIÓN ENTRE MINTERMS Y MAXTERMS EN LA SIMPLIFICACIÓN CON REDUNDANCIAS.

- Lo único que se debe tener en cuenta es que las redundancias pueden ser ceros o unos, es decir, tratarlos como maxterms o minterms según nos convenga. (Si estamos simplificando los minterms, las redundancias serán unos, si estamos simplificando los maxterms, las redundancias serán ceros).

Ej. $S = \prod_4 (0, 3, 4, 7, 10, 13, 14, 15) + d(1, 2, 5, 6)$

AB		A	
CD			
	0	4	12
	X ₁	X ₅	13
	3	7	15
	X ₂	X ₆	14
			10

$$S = A(\bar{C}+D)(\bar{B}+\bar{D})$$

AB		A	
CD			
	0	4	12
	X ₁	X ₅	13
	3	7	15
	X ₂	X ₆	14
			10

$$S = A\bar{C}\bar{D} + A\bar{B}D$$

Ej. Simplificar dando el resultado en forma de producto de sumas y sumas de productos.

$$S = \sum_4 (0, 2, 4, 5, 6, 7, 14) + d(1, 9, 10, 11, 12) \quad [S = (\bar{A}+C)(\bar{A}+\bar{D})(B+\bar{D}); S = \bar{A}\bar{C} + \bar{A}B + C\bar{D}]$$

TABLA DE KARNAUGH PARA 5 VARIABLES.

- En este caso tendremos 32 combinaciones posibles con las 5 variables. Esto implica que necesitamos una tabla que tenga 32 celdas; para su realización, partimos de dos tablas de 16 celdas como las de 4 variables. Para una de las tablas, la variable de mas peso valdrá 0 (para toda la tabla) y para la otra tabla, esta variable tendrá el valor 1.

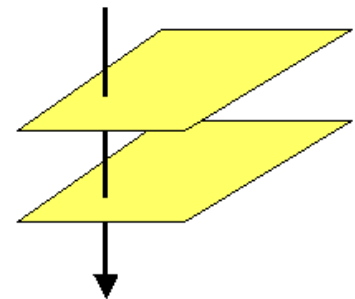
BC		B			
DE	D	E			
		C			
		0	4	12	8
		1	5	13	9
		3	7	15	11
		2	6	14	10

A=0

BC		B			
DE	D	E			
		C			
		16	20	28	24
		17	21	29	25
		19	23	31	27
		18	22	30	26

A=1

Las adyacencias en las tablas son iguales que para cuatro variables, pero aquí, además, hay otras: si pusiéramos una tabla encima de la otra, todas las celdas que coinciden en la misma vertical, son adyacentes. Son adyacentes la casilla 0 y la 16, la 15 y la 31, la 18 y la 2...etc.



Ej. Simplificar

$$S = \sum_5 (0, 2, 4, 5, 12, 13, 14, 16, 18, 19, 20, 21, 28, 29, 31)$$

BC		B			
DE	D	E			
		C			
		1	4	1	8
		1	5	1	9
		3	7	15	11
		1	6	1	10

A=0

BC		B			
DE	D	E			
		C			
		1	20	1	24
		1	21	1	25
		1	23	1	27
		1	22	30	26

A=1

$$S = \overline{C}\overline{D} + \overline{B}\overline{C}\overline{E} + \overline{A}\overline{B}\overline{C}\overline{E} + \overline{A}\overline{B}\overline{C}D + \overline{A}BCE$$

Simplificar mediante Karnaugh:

$$S = \overline{A}BCD + \overline{A}B + \overline{A}B\overline{D}\overline{E} + \overline{A}BD \quad [S = \overline{A}B]$$

MÉTODO DE QUINE McCLUSKEY.

- Cuando el número de variables es superior a 4, una buena alternativa es el empleo de este método de simplificación (aunque también es válido para menos de 4 variables).

El proceso de simplificación se explica siguiendo la resolución de un ejercicio práctico:

Para 4 variables:

Tomando los minterms:

Dec	A	B	C	D	S
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	0
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	0

$$S = \bar{A}\bar{B}CD + \bar{A}B\bar{C}D + \bar{A}BC\bar{D} + \bar{A}BCD + A\bar{B}\bar{C}D + ABC\bar{D}$$

$$S = \sum_4 (3, 5, 6, 7, 13, 14)$$

1º Se realiza una tabla donde se agrupan los términos por el número de unos que contengan. Este número de unos determina el **índice** de cada grupo.

Las columnas de la tabla serán las que correspondan al término expresado en binario, el equivalente decimal y el índice de cada grupo:

DEC.	BINARIO	ÍNDICE
3	0011	2
5	0101	
6	0110	
7	0111	3
13	1101	
14	1110	

2º Se realizan después una serie de tablas llamadas **tablas de reducción**. La primera tabla será la tabla de reducción de primer orden, la segunda será la tabla de reducción de segundo orden y así sucesivamente.

Para la realización de estas tablas, se va comparando cada término de un índice con cada uno de los términos de los índices siguientes. Cuando en la comparación, dos términos difieran solamente en un BIT, estos dos se agruparan como un sólo término, sustituyendo el BIT diferente, por un guión, y anotando el valor decimal de los términos que han dado lugar al nuevo elemento.

Por ejemplo si comparamos el 0110 (6) con el 0111 (7) resulta el término 011- y anotamos (6,7). En esta nueva tabla también estarán agrupados los términos según su nuevo índice.

Tabla de reducción de primer orden

Si salen términos que se repiten, se anulan y se deja solamente uno de ellos.

La comparación debe continuar hasta que sólo quede un índice o con varios índices, no salga ninguna reducción más.

DEC.	BINARIO	ÍNDICE
(3,7)	0-11	2
(5,7)	01-1	
(5,13)	-101	
(6,7)	011-	
(6,14)	-110	

3º Se realiza una última tabla donde aparecen por un lado las variables según los pesos y por otro lado el número decimal correspondiente a los términos antes de la simplificación. Se colocan debajo de las variables los términos obtenidos de la última tabla de reducción. Para cada fila correspondiente a cada término se coloca una X en la columna correspondiente al valor decimal que haya estado implicado en la reducción. La colocación de esta X es de suma importancia para la mayor simplificación. La función estará simplificada cuando aparezca al

menos una X en cada una de las columnas pertenecientes a los términos decimales sin simplificar.

ABCD	3	5	6	7	13	14
0-11	X			X		
01-1		X		X		
-101		X			X	
011-			X	X		
-110			X			X

Para que la simplificación sea máxima, hay que tratar de tomar el mínimo número de filas (términos simplificados) y que a la vez intervengan todos los valores decimales.

En las columnas donde solamente hay una X, está claro que el término de esa fila a la que pertenece la X debe ser tomado. En los demás casos hay que estudiar la selección. Puede darse el caso de varias soluciones, todas válidas, según la elección.

En algunos casos puede que con la última tabla de reducción no se cubran todas las columnas con las X. En este caso hay que recurrir a términos de la anterior tabla o tablas de reducción, de donde se obtendrán otros términos (menos simplificados).

ABCD	3	5	6	7	13	14
0-11	X			X		
01-1						
-101		X			X	
011-						
-110			X			X

4º Una vez seleccionados los términos que forman parte de la función simplificada, las variables de los términos con guión, son eliminadas. En nuestro caso el resultado será:

$$S = \bar{A}CD + B\bar{C}D + BCD\bar{D}$$

Tomando los maxterms:

$$S = (A+B+C+D)(A+B+C+\bar{D})(A+B+\bar{C}+D)(A+\bar{B}+C+D)(\bar{A}+B+C+D)(\bar{A}+B+C+\bar{D})(\bar{A}+B+\bar{C}+D)(\bar{A}+B+\bar{C}+\bar{D})(\bar{A}+\bar{B}+C+D)(\bar{A}+\bar{B}+\bar{C}+\bar{D})$$

$$S = \prod_4 (0, 1, 2, 4, 8, 9, 10, 11, 12, 15)$$

DEC.	BINARIO	ÍNDICE
0,1	000-	0
0,2	00-0	
0,4	0-00	
0,8	-000	
1,9	-001	1
2,10	-010	
4,12	-100	
8,9	100-	
8,10	10-0	
8,12	1-00	2
9,11	10-1	
10,11	101-	3
11,15	1-11	

DEC.	BINARIO	ÍNDICE
0	0000	0
1	0001	1
2	0010	
4	0100	
8	1000	
9	1001	2
10	1010	
12	1100	
11	1011	3
15	1111	4

DEC.	BINARIO	ÍNDICE
(0,1)(8,9)	-00-	0
(0,2)(8,10)	-0-0	
(0,4)(8,12)	--00	
(0,8)(1,9)	-00-	
(0,8)(2,10)	-0-0	
(0,8)(4,12)	--00	1
(8,9)(10,11)	10--	
(8,10)(9,11)	10--	

ABCD	0	1	2	4	8	9	10	11	12	15
- 0 0 -	X	X			X	X				
- 0 - 0	X		X		X		X			
- - 0 0	X			X	X				X	
1 0 - -					X	X	X	X		
1 - 1 1								X		X

Lo único que hay que tener en cuenta ahora es que las variables directas valen 0 y las negadas 1. La solución es:

$$S = (B+C)(B+D)(C+D)(\bar{A}+\bar{C}+\bar{D})$$

Para 5 variables:

$$S = \sum_5 (0, 2, 4, 5, 12, 13, 14, 16, 18, 19, 20, 21, 28, 29, 31)$$

(00000, 00010, 00100, 00101, 01100, 01101, 01110, 10000, 10010, 10011, 10100, 10101, 11100, 11101, 11111)

DEC	BIN	INDICE
0	00000	0
2	00010	1
4	00100	
16	10000	
5	00101	2
12	01100	
18	10010	
20	10100	
13	01101	3
14	01110	
19	10011	
21	10101	
28	11100	4
29	11101	
31	11111	

DEC	BIN	IND
(0,2)	000-0	0
(0,4)	00-00	
(0,16)	-0000	
(2,18)	-0010	1
(4,5)	0010-	
(4,12)	0-100	
(4,20)	-0100	
(16,18)	100-0	
(16,20)	10-00	
(5,13)	0-101	2
(5,21)	-0101	
(12,13)	0110-	
(12,14)	011-0	
(12,28)	-1100	
(18,19)	1001-	
(20,21)	1010-	
(20,28)	1-100	3
(13,29)	-1101	
(21,29)	1-101	
(28,29)	1110-	4
(29,31)	111-1	

Tabla de reducción de primer orden.

DEC	BIN	IND
(0,2)(16,18)	-00-0	0
(0,4)(16,20)	-0-00	
(0,16)(2,18)	-00-0	
(0,16)(4,20)	-0-00	
(4,5)(12,13)	0-10-	1
(4,5)(20,21)	-010-	
(4,12)(5,13)	0-10-	
(4,12)(20,28)	--100	
(4,20)(5,21)	-010-	
(4,20)(12,28)	--100	
(5,13)(21,29)	--101	2
(5,21)(13,29)	--101	
(12,13)(28,29)	-110-	
(12,28)(13,29)	-110-	
(20,21)(28,29)	1-10-	
(20,28)(21,29)	1-10-	

Tabla de reducción de segundo orden.

DEC	BIN	IND
(4,5)(12,13) (20,21)(28,29)	--10-	1
(4,5)(20,21) (12,13)(28,29)	--10-	
(4,20)(12,28) (5,13)(21,29)	--10-	

Tabla de reducción de tercer orden.

ABCDE	0	2	4	5	12	13	14	16	18	19	20	21	28	29	31	
- - 1 0 -			X	X	X	X					X	X	X	X		*
- 0 0 - 0	X	X						X	X							**
1 0 0 1 -									X	X						***
0 1 1 - 0					X		X									***
1 1 1 - 1														X	X	***

* término tomado de la tabla de reducción de tercer orden

** término tomado de la tabla de reducción de segundo orden

*** términos tomados de la tabla de reducción de tercer orden

La solución es por lo tanto:

$$S = C \bar{D} + \bar{B} \bar{C} \bar{E} + A \bar{B} \bar{C} D + \bar{A} B C \bar{E} + A B C E.$$

Para trabajar con funciones incompletamente especificadas mediante el método de McCluskey, se opera de la misma forma que lo visto hasta ahora. Lo único que cambia es que en la tabla de simplificación, no aparecen los valores decimales de las condiciones no importa.

$$Ej. S = \sum_4 (1, 2, 5, 6, 9) + d(10, 11, 13, 14, 15)$$

DEC	BIN	INDICE
1	0001	1
2	0010	
5	0101	
6	0110	
9	1001	2
10	1010	
11	1011	
13	1101	3
14	1110	
15	1111	4

DEC	BIN	IND
(1,5)	0-01	
(1,9)	-001	1
(2,6)	0-10	
(2,10)	-010	
(5,13)	-101	
(6,14)	-110	
(9,11)	10-1	2
(9,13)	1-01	
(10,11)	101-	
(10,14)	1-10	
(11,15)	1-11	
(13,15)	11-1	3
(14,15)	111-	

DEC	BIN	IND
(1,5)(9,13)	--01	
(1,9)(5,13)	--01	1
(2,6)(10,14)	--10	
(2,10)(6,14)	--10	
(9,11)(13,15)	1--1	
(9,13)(11,15)	1--1	2
(10,11)(14,15)	1-1-	
(10,14)(11,15)	1-1-	

ABCD	1	2	5	6	9
- - 0 1	X		X		X
- - 1 0		X		X	

$$S = \bar{C}D + C\bar{D} = C \text{ X-OR } D$$

Simplificar por McCluskey:

$$S = \sum_5 (2, 3, 7, 10, 12, 15, 27) + d(5, 18, 19, 21, 23) [S = \bar{A}\bar{B}\bar{C}\bar{D}\bar{E} + \bar{A}\bar{C}DE + \bar{A}\bar{C}DE + \bar{A}\bar{C}DE + \bar{B}DE]$$

1.8-LÓGICA INTEGRADA.

A continuación se dan una serie de definiciones y parámetros tal y como las podemos encontrar en cualquier manual u hoja de datos relacionada con las familias lógicas.

1.8.1-GLOSSARY

These symbols, terms, and definitions are in accordance with those currently agreed upon by the **JEDEC** (*Join Electronics Device Engineering Council*) of the *Electronic Industries Association (EIA)* for use in the USA and by the *International Electrotechnical Commission (IEC)* for international use.

(Current out of a terminal, is given as a negative value)

- **Ci.** Input capacitance.
The internal capacitance at an input of the device.
- **Cio.** Input/output capacitance.
Input-to-output internal capacitance; transcapacitance.
- **Co.** Output capacitance.
The internal capacitance at an output of the device.
- **Cpd.** Power dissipation capacitance
Used to determine the no-load dynamic power dissipation per logic function.
 $PD = Cpd \cdot V_{CC}^2 \cdot f + I_{CC} \cdot V_{CC}$.
- **fmax,** Maximum clock frequency.
The highest rate at which the clock input of a bistable circuit can be driven through its required sequence while maintaining stable transitions of logic level at the output with input conditions established that should cause changes of output logic level in accordance with the specification I_{CC} .
- **Icc.** Supply current.
The current into the VCC supply terminal of an integrated circuit.
- **ΔI_{CC} .** Supply current change.
The increase in supply current for each input that is at one of the specified TTL voltage levels rather than 0 V or V_{CC} I_{CEX} .
- **I_{CEX} .** Output high leakage current.
The maximum leakage current into the collector of the pulldown output transistor when the output is high and the output forcing condition $V_O = 5.5$ V.
- **$I_{I(hold)}$.** Input hold current.
Input current that holds the input at the previous state when the driving device goes to a high-impedance state.
- **I_{IH} .** High-level input current.
The current into an input when a high-level voltage is applied to that input.
- **I_{IL} .** Low-level input current.
The current into an input when a low-level voltage is applied to that input.
- **I_{off} .** Input/output power-off leakage current.
The current into a circuit mode when the device or a portion of the device affecting that circuit mode is in the off state.

- **I_{OH}** . High-level output current.

The current into an output with input conditions applied that, according to the product specification, establish a high level at the output.

- **I_{OL}** . Low-level output current.

The current into an output with input conditions applied that, according to the product specification, establish a low level at the output.

- **I_{OZ} $I_{OZPU/PD}$** . Off-state (high-impedance-state) output current (of a 3-state output).

The current flowing into* an output having 3-state capability with input conditions established that, according to the product specification, establish the high-impedance state at the output.

- **R_{JA}** . Junction-to-ambient thermal resistance.

The thermal resistance from the semiconductor junction(s) to the ambient.

- **R_{JC}** . Junction-to-case thermal resistance.

The thermal resistance from the semiconductor junction(s) to a stated location on the case.

- **t_a** . Access time.

The time interval between the application of a specified input pulse and the availability of valid signals at an output.

- **t_c** . Clock cycle time.

Clock cycle time is $1/f_{max}$.

- **t_{dis}** . Disable time (of a 3-state or open-collector output).

The propagation time between the specified reference points on the input and output voltage waveforms with the output changing from either of the defined active levels (high or low) to a high-impedance (off) state

NOTE: For 3-state outputs, $t_{dis} = t_{PHZ}$ or t_{PLZ} . Open-collector outputs change only if they are low at the time of disabling, so $t_{dis} = t_{PLH}$

- **t_{en}** . Enable time (of a 3-state or open-collector output).

The propagation time between the specified reference points on the input and output voltage waveforms with the output changing from a high-impedance (off) state to either of the defined active levels (high or low).

NOTE: In the case of memories, this is the access time from an enable input (e.g., OE). For 3-state outputs, $t_{en} = t_{PZH}$ or t_{PZL} . Open-collector outputs change only if they are responding to data that would cause the output to go low, so $t_{en} = t_{PHL}$

- **t_h** . Hold time.

The time interval during which a signal is retained at a specified input terminal after an active transition occurs at another specified input terminal.

NOTES: 1. The hold time is the actual time interval between two signal events and is determined by the system in which the digital circuit operates. A minimum value is specified that is the shortest interval for which correct operation of the digital circuit is to be expected.

2. The hold time may have a negative value in which case the minimum limit defines the longest interval (between the release of the signal and the active transition) for which correct operation of the digital circuit is to be expected.

- **t_{pd}** . Propagation delay time.

The time between the specified reference points on the input and output voltage waveforms with the output changing from one defined level (high or low) to the other defined level ($t_{pd} = t_{PHL}$ or t_{PLH}).

- **t_{PHL}** , Propagation delay time, high-to-low level output.

The time between the specified reference points on the input and output voltage waveforms with the output changing from the defined high level to the defined low level.

- **t_{PHZ}** , Disable time (of a 3-state output) from high level.

The time interval between the specified reference points on the input and the output voltage waveforms with the 3-state output changing from the defined high level to the high-impedance (off) state.

- **t_{PLH}** , Propagation delay time, low-to-high level output.

The time between the specified reference points on the input and output voltage waveforms with the output changing from the defined low level to the defined high level.

- **t_{PLZ}** , Disable time (of a 3-state output) from low level.

The time interval between the specified reference points on the input and the output voltage waveforms with the 3-state output changing from the defined low level to the high-impedance (off) state.

- **t_{PZH}** , Enable time (of a 3-state output) to high level.

The time interval between the specified reference points on the input and output voltage waveforms with the 3-state output changing from the high-impedance (off) state to the defined high level.

- **t_{PZL}** , Enable time (of a 3-state output) to low level.

The time interval between the specified reference points on the input and output voltage waveforms with the 3-state output changing from the high-impedance (off) state to the defined low level.

- **$tsk(o)$** , Output Skew.

The difference between any two propagation delay times when a single switching input or multiple inputs switching simultaneously cause multiple outputs to switch, as observed across all switching output. This parameter is used to describe the fanout capability of a clock driver and is of concern when making decisions on clock buffering and distribution networks.

- **tsu** , Setup time

The time interval between the application of a signal at a specified input terminal and a subsequent active transition at another specified input terminal.

NOTES: 1. The setup time is the actual time interval between two signal events and is determined by the system in which the digital circuit operates. A minimum value is specified that is the shortest interval for which correct operation of the digital circuit is guaranteed.

2. The setup time may have a negative value, in which case the minimum limit defines the longest interval (between the active transition and the application of the other signal) for which correct operation of the digital circuit is guaranteed.

- **t_w** , Pulse duration (width).

The time interval between specified reference points on the leading and trailing edges of the pulse waveform.

- **V_{IH}** , High-level input voltage.

An input voltage within the more positive (less negative) of the two ranges of values used to represent the binary variables.

NOTE: A minimum is specified that is the least-positive value of high-level input voltage for which operation of the logic element within specification limits is to be expected.

- V_{IL} , Low-level input voltage.

An input voltage within the less positive (more negative) of the two ranges of values used to represent the binary variables.

NOTE: A maximum is specified that is the most-positive value of low-level input voltage for which operation of the logic element within specification limits is to be expected.

- V_{OH} , High-level output voltage.

The voltage at an output terminal with input conditions applied that, according to product specification, establish a high level at the output.

- V_{OL} , Low-level output voltage.

The voltage at an output terminal with input conditions applied that, according to product specification, establish a low level at the output.

- V_{IT+} , Positive-going input threshold level.

The voltage level at a transition-operated input that causes operation of the logic element according to specification as the input voltage rises from a level below the negative-going threshold voltage, V_{IT-} .

- V_{IT-} , Negative-going input threshold level.

The voltage level at a transition-operated input that causes operation of the logic element according to specification as the input voltage falls from a level above the positive-going threshold voltage, V_{IT+} .

- V_{OHV} , High-level output voltage change during simultaneous switching.

The minimum (valley) voltage induced on a quiescent high-level output during switching of other outputs.

- V_{OLP} , Low-level output voltage change during simultaneous switching.

The maximum (peak) voltage induced on a quiescent low-level output during switching of other outputs.

1.8.2-PARAMETERS.

□ **Signal Amplitude.**

The signal amplitude is specified as the difference between the two stable levels for high (V_H), and low (V_L); overshoot and undershoot of the signal is not taken into account. The difference $V_H - V_L$ is taken as 100% of the amplitude.

t_r , Rise Time.

Time taken to rise from 10% to 90% of the full amplitude.

t_f , Fall Time.

Time taken to fall from 90% to 10% of the amplitude.

Fan Out.

The maximum number of parallel loads that a single logic circuit can drive safely.

Slew Rate: $SR = dv/dt = (V_{OH} - V_{OL}) * 0.8 / t_r$ (o t_f)

Temperature Margin:

<u>Serie.</u>	<u>Margin</u>	
74...	0	70° C
54...	-55	125°C
MOS BASED	-40	85°C

Power Consumption:

$$P_D = V_{CC} * I_{CC} \quad I_{CC} = (I_{CCH} + I_{CCL})/2$$

TTL, CMOS y LV levels.

DC Noise Margin (High & Low)

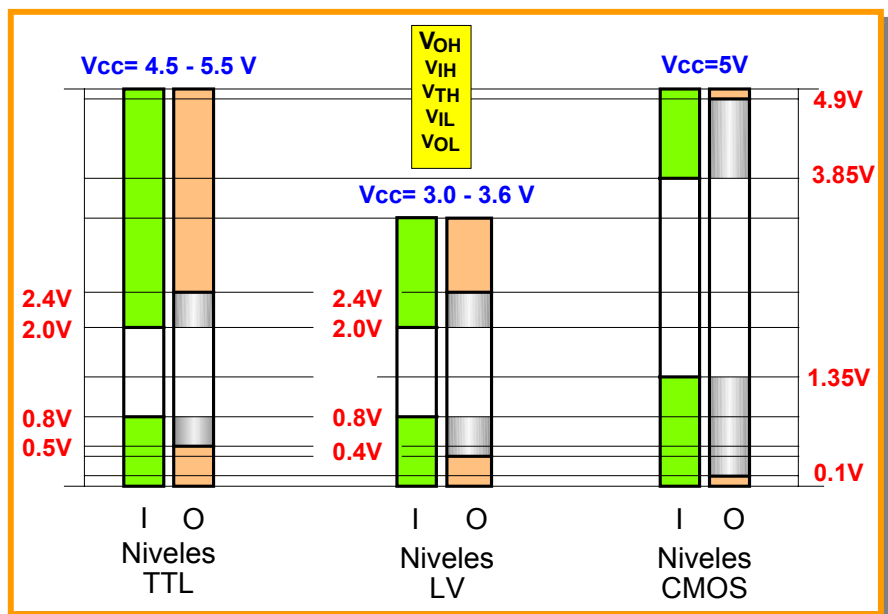
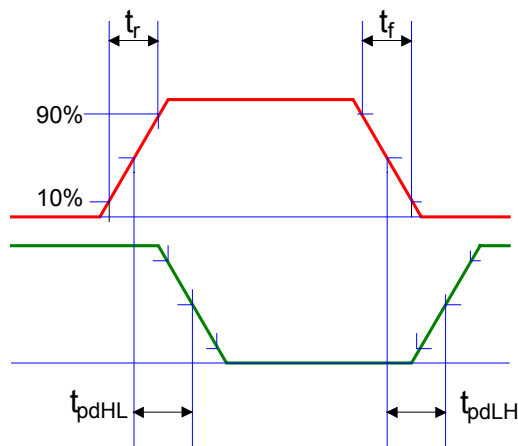
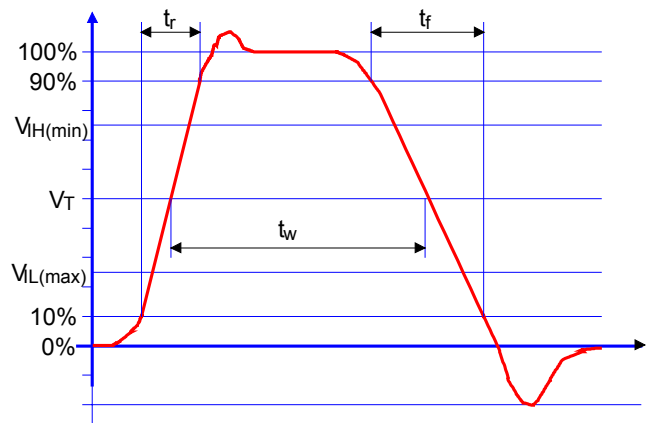
$$V_{NH} = V_{OH(MIN)} - V_{IH(MIN)}$$

$$V_{NL} = V_{IL(MX)} - V_{OL(MX)}$$

The margin of voltage (noise) that a circuit can tolerate without any change in its function.

Speed-Power product:

$$T_{pd} * P_D$$



It is a comparative term between series or families.

1.8.3-FAMILIAS LÓGICAS.

NOMENCLATURA.

Serie	Significado	Input level- output level
74/54.....	TTL Transistor-Transistor Logic.	TTL-TTL
ABT	Advanced BICMOS Technology.	TTL-TTL
AC	Advanced CMOS Logic.	CMOS-CMOS
ACT	Advanced CMOS Logic with TTL compatible inputs.	TTL-CMOS
AHC	Advanced High-Speed CMOS Logic.	CMOS-CMOS
AHCT	Advanced High Speed CMOS Logic with TTL compatible inputs.	TTL-TTL
ALB	Advanced Low-Voltage BICMOS.	LVTTL-LVTTL
ALS	Advanced Low-Power Schottky Logic.	TTL-TTL
ALVC	Advanced Low-Voltage CMOS Technology.	LVTTL-LVTTL
AS	Advanced Schottky Logic.	TTL-TTL
BCT	BICMOS Technology.	TTL-TTL
F	Fast Logic.	TTL-TTL
HC	High Speed CMOS Logic.	CMOS-CMOS
HCT	High Speed CMOS Logic with TTL compatible inputs.	TTL-CMOS
LS	Low Power Schottky Logic.	TTL-TTL
LV	Low Voltage CMOS Technology.	LVTTL-LVTTL
LVC	Low Voltage CMOS Technology.	TTL/CMOS-LVTTL
LVT	Low Voltage (BICMOS) Technology.	TTL/CMOS-LVTTL
S	Schottky Logic.	TTL-TTL
4000.....	CMOS Technology.	CMOS-CMOS

IMPLANTACIÓN DE LAS FAMILIAS LÓGICAS.

Declibe	Madurez	Crecimiento	Introducción
TTL ¹ Standard.	AS ¹	BCT ³	LVT ³
S ¹	F ¹	ACT ²	LV, LVC ²
LS ¹	ALS ¹	ABT ³	ALVC ²
4000	HC ²		AHC ²
AS ¹			ALVT ³

¹ Tecnología Bipolar.² Tecnología CMOS³ Tecnología Bi-CMOS

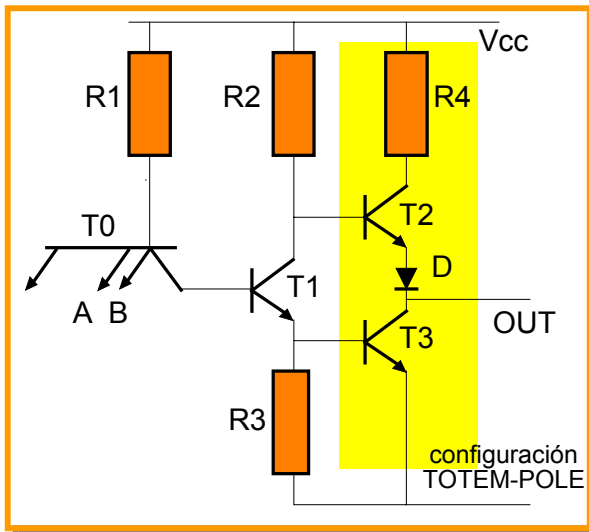
Dentro de cada apartado, la serie a la que se hace referencia, es mas reciente en el tiempo según descendemos.

FAMILIAS LÓGICAS. CIFRAS COMPARATIVAS.													
TIPO	V _{CC}	V _{IH}	V _{IL}	V _{OH}	V _{OL}	I _{OH}	I _{OL}	I _{IH}	I _{IL}	t _{PLH}	t _{PHL}	I _{CCH}	I _{CCL}
	v	V	V	v	v	mA	mA	mA	mA	nS	nS	mA	mA
---	5,25		0,8		0,4	-0,4	16	0,04	-1,6	22	15	8	22
	4,75	2		2,4									
LS	5,25		0,8		0,5	-0,4	8	0,02	-0,4	15	15	1,6	4,4
	4,75	2		2,7									
S	5,25		0,8		0,5	-1	20	0,05	-2	4,5	5	16	36
	4,75	2		2,7									
AS	5,5		0,8		0,5	-2	20	0,02	-0,5	4,5	4	3,2	17,4
	4,5	2		VCC-2									
ALS	5,5		0,8		0,5	-0,4	8	0,02	-0,1	11	8	0,85	3
	4,5	2		VCC-2									
F	5,5		0,8		0,5	-1	20	0,02	-0,6	6	5,3	2,8	10,2
	4,5	2		2,7									
AHCT	5,5		0,8		0,44	-8	8	0,001	0,001	9	9	0,02	0,02
	4,5	2		4									
HC	6		1,2		0,3	-4,5	4,5	0,001	0,001	23	23	0,02	0,02
	2	3,15		VCC-0,1									
AC	6		1,6		0,44	-24	2,4	0,001	0,001	8,5	7	0,02	0,02
	2	3,15		VCC-0,1									
AHC	5,5		1,5		0,44	-8	8	0,001	0,001	8,5	8,5	0,02	0,02
	2	3,7		3,8									
ACT	5,5		0,8		0,44	-24	24	0,001	0,001	9,5	8	0,02	0,02
	4,5	2		4									
HCT	5,5		0,8		0,3	-4	4	0,001	0,001	23	23	0,02	0,02
	4,5	2		3,8									
LV	5,5		0,8		0,4	-6	6	0,001	0,001	18	18	20	20
	2,7	2		2,4									
LVC	3,6		0,8		0,55	-24	24	0,005	0,005	5,1	5,1	10	10
	2	2		2,2									

FAMILIAS LÓGICAS. CIFRAS COMPARATIVAS						
TIPO	t _p	I _{CC}	V _{CC}	P	P * t _p	dt/dv
	nS	mA	v	mW	PJ	nS/V
-----	18,5	15	5	75	1387,5	100
LS	15	3	5	15	225	50
S	4,75	26	5	130	617,5	50
AS	4,25	10,3	5	51,5	218,875	8
ALS	9,5	1,925	5	9,625	91,4375	15
F	5,65	6,5	5	32,5	183,625	8
AHCT	9	0,02	5	0,1	0,9	
HC	23	0,02	4	0,08	1,84	110
AC	7,75	0,02	4	0,08	0,62	10
AHC	8,5	0,02	3,75	0,075	0,6375	
ACT	8,75	0,02	5	0,1	0,875	10
HCT	23	0,02	5	0,1	2,3	125
LV	18	20	4,1	82	1476	100
LVC	5,1	10	2,8	28	142,8	0,5

1.8.4-TTL.

NAND BÁSICA.



Valores típicos del circuito son $R1=4K\Omega$; $R2=1K\Omega$; $R3=1K\Omega$; $R4=130\Omega$; $V_{cc}=+5V$.

A	B	T0	T1	T2	T3	OUT
0	0	C	NC	C	NC	H
0	1	C	NC	C	NC	H
1	0	C	NC	C	NC	H
1	1	NC	C	NC	C	L

C=conduce; NC= no conduce

Notas:

1.- Cuando las dos entradas valen 1 (T1 conduce), tenemos:

$V_{ET1} = V_{R3} = V_{BE T3} \approx 0.7V$ y $V_{CE T1} = V_{CE T3} \approx 0.2V$ (SAT).

$V_{CT1} = V_{ET1} + V_{CE T1} = 0.7 + 0.2 = 0.9V$; y $V_{CT3} = 0.2V$. Si no estuviera el diodo D, la $V_{ET2} = V_{CT3} = 0.2V$ y la tensión $V_{BT2} = V_{CT1} = 0.9V$, por lo que $V_{BE T2}$ sería: $V_{BE T2} = V_{BT2} - V_{ET2} = 0.9 - 0.2 = 0.7V$, con lo cual, el transistor T2 estaría conduciendo. Con el diodo D ($V_D \approx 0.7V$), la $V_{BE T2} \approx 0V$ manteniendo T2 al corte.

2.- Observar que cuando la salida (OUT) tiene el estado lógico "L", absorbe corriente de la entrada que está manejando.

3.- Con la salida TOTEM POLE se logra que cuando T3 está en saturación la corriente que circula por él sea muy baja al estar T2 al corte.

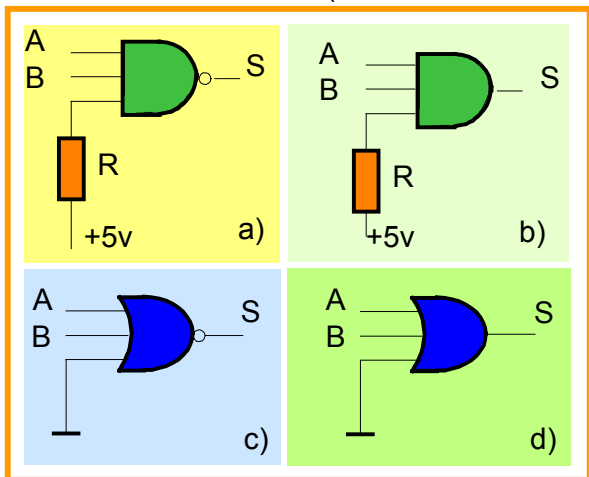
4.- Para el estado lógico de salida "H" T2 actúa como un seguidor de tensión, con su Z_o baja, lo que implica una constante de tiempo (τ) baja. Esto tiene gran importancia en cargas capacitivas.

5.- El convenio tomado para definir el sentido de las corrientes, es considerar como positivas las que entran al circuito y negativas las corrientes salientes.

ENTRADAS FLOTANTES.

Las entradas flotantes (terminales de entrada al aire) actúan como un nivel lógico H, aún así,

no es conveniente dejar estas entradas al aire, aunque se necesite siempre un nivel H en las mismas, pues pueden inducirse en ellas ciertos niveles que hagan modificar la función del circuito.



Cuando alguna entrada no vaya a ser utilizada conviene conectarla a masa o positivo según el requerimiento del circuito. En la figura se dan unos ejemplos.

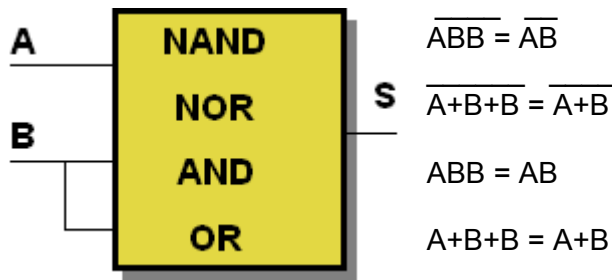
a) $S = \overline{AB1} = \overline{AB}$

c) $S = \overline{A+B+0} = \overline{A+B}$

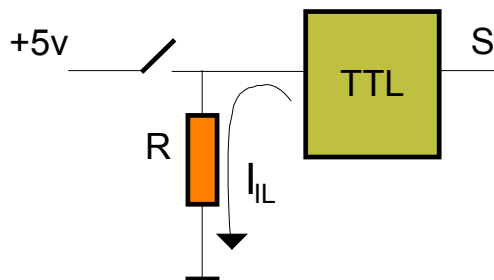
b) $S = AB1 = AB$

d) $S = A+B+0 = A+B$

También puede utilizarse la configuración de la figura en el caso de puertas OR, AND, NOR y NAND, aunque es preferible la anterior solución.



ENTRADA TTL A NIVEL BAJO DE FORMA MOMENTANEA.

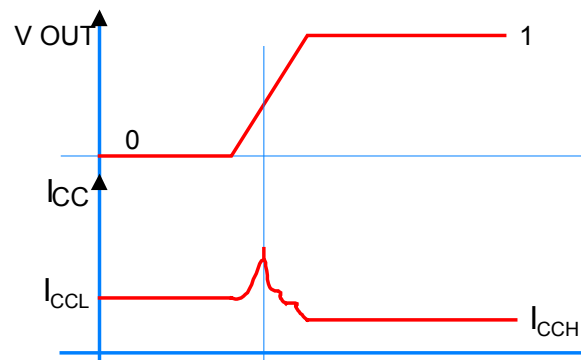
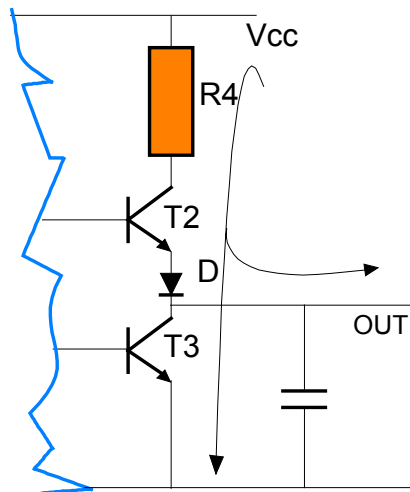


- Para evitar que esta entrada pase a nivel lógico H cuando lo que se busca es un nivel L (interruptor abierto), la resistencia debe tomarse de tal manera que cumpla:

$$R \cdot I_{IL} < V_{IL\ MAX}, \text{ es decir } R < V_{IL\ MAX} / I_{IL}$$

TRANSITORIOS DE CORRIENTE.

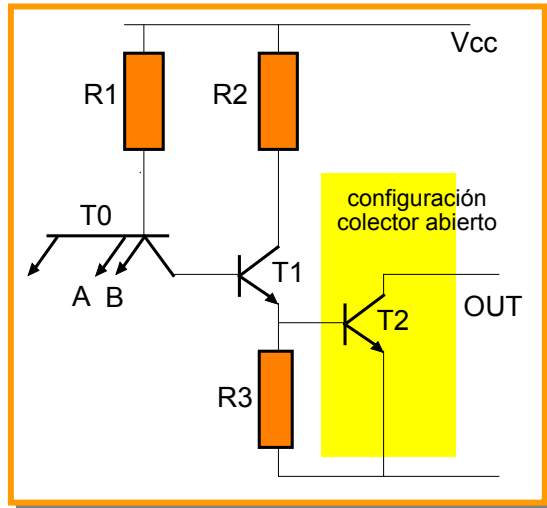
- Siempre que una salida TTL tipo TOTEM POLE pasa de nivel L a nivel H, se genera un consumo transitorio de corriente de gran amplitud de la fuente de alimentación. Esto se debe a que T3 tarda más en pasar de la saturación al corte que T2 del corte a la saturación, por lo que durante un tiempo están conduciendo los dos transistores.



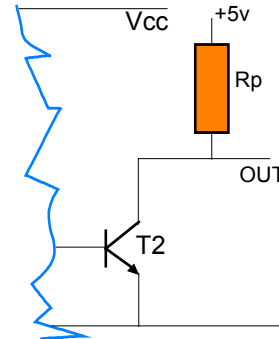
El efecto acumulativo de un número importante de circuitos conmutando al mismo tiempo hace que el impulso de corriente deba ser tenido en cuenta, pues ocasiona unos impulsos de tensión en la línea debidos a la inductancia distribuida de la misma. ($v = L (di/dt)$).

Para evitar estos impulsos de tensión, se colocan condensadores de desacoplo de la FA cerca de cada IC, conectados entre la Vcc y la masa.

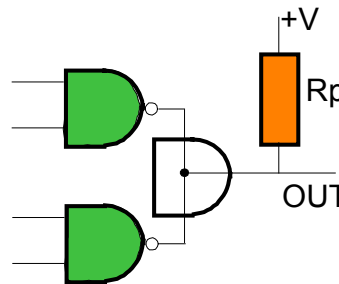
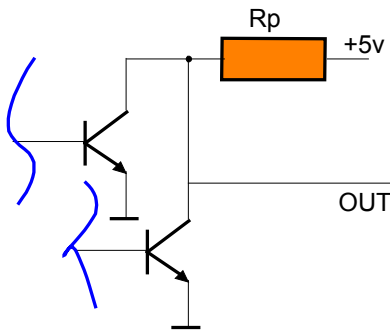
TTL DE COLECTOR ABIERTO.



- Para que el circuito cumpla su función, necesita una resistencia R_p (**pull-up**) externa, conectada al colector de T2 y a la alimentación.



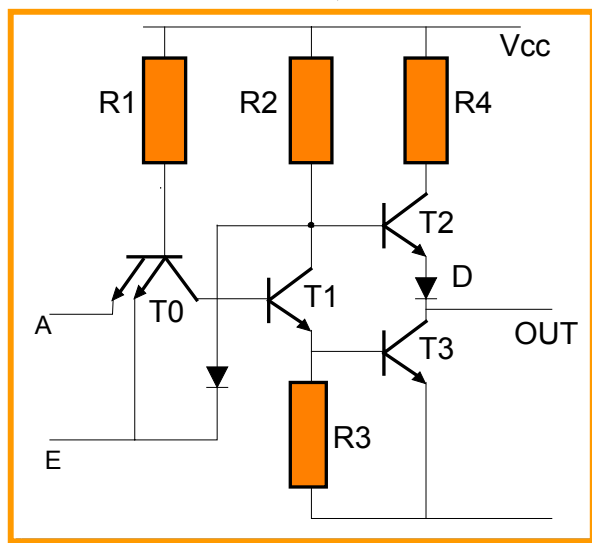
Este tipo de configuración permite la llamada AND alamburada:



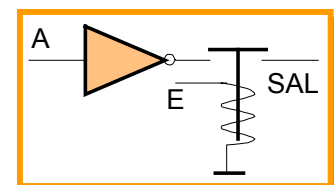
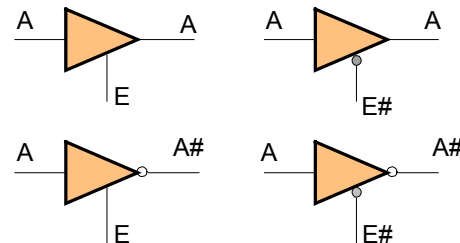
El valor de la R_p debe ser elegido de tal manera que cuando una salida de la puerta alamburada pase a nivel L, en tanto que las otras sean H, la corriente obtenida a través de la salida en el estado lógico L, no exceda su límite I_{OL} (≈ 16 mA).

Cuanto mayor es la R_p más aseguramos los transistores de salida de las puertas pero a costa de la velocidad de trabajo. ($\tau = RC$). De todas maneras, esta versión es más lenta que la de salida TOTEM POLE.

TTL TRIESTADO.



E	A	T0	T1	T2	T3	OUT
0	0	C	NC	NC	NC	Z
0	1	C	NC	NC	NC	Z
1	0	C	NC	C	NC	H
1	1	NC	C	NC	C	L



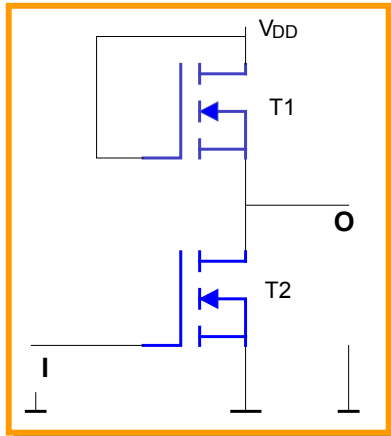
Observar que el estado de alta impedancia (Z), idealmente se puede comparar a un terminal al aire.

1.8.5-CMOS.

- La técnica CMOS integra en el mismo IC transistores PMOS (MOSFET de canal P) y NMOS (MOSFET de canal N).

Las características más importantes son la elevada densidad de integración, y el consumo reducido de energía. La velocidad es sensiblemente inferior a la que se obtiene con TTL, aunque cada vez las características de ambos son más parecidas.

INVERSOR NMOS.



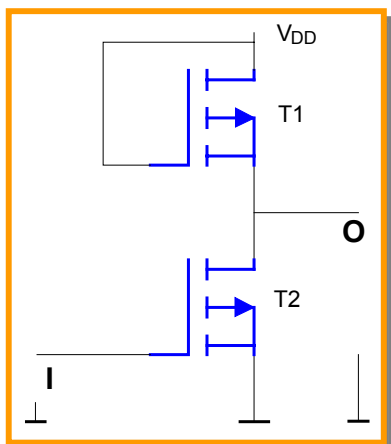
I	T1	T2	O
0	ON ($R_{DS} = 100K$)	OFF ($R_{DS} = 10^{10} \Omega$)	H $\approx 5V$
1	ON ($R_{DS} = 100K$)	ON ($R_{DS} = 1K$)	L $\approx 0V$

Los transistores se diseñan de tal manera que las R_{DS} -s dadas en la tabla, se cumplan.

El transistor T1 actúa como una resistencia (*MOSFET DE CARGA*).

La $V_{DD} = +5V$. Los NMOS, poseen una densidad de integración casi doble que los PMOS, además son más rápidos que estos.

INVERSOR PMOS.



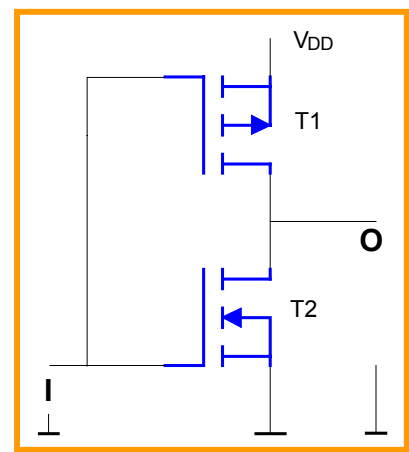
- El comportamiento del circuito es igual que el del inversor NMOS, pero en este caso, debemos cambiar todas las polaridades.

Al aplicar una tensión negativa a la puerta del T2, este transistor se pondrá en ON, obteniendo en la salida un nivel de tensión prácticamente de 0 voltios. Cuando no apliquemos tensión a dicha puerta, el T2 queda en OFF, teniendo en la salida (O), una tensión de -5V.

INVERSOR CMOS.

$V_{DD} = +5V.$

I	T1	T2	O
0V (L)	$V_{GS} = -5V$. ON ($R_{DS} = 1K$)	$V_{GS} = 0V$. OFF ($R_{DS} = 10^{10} \Omega$)	H $\approx 5V$
5V (H)	$V_{GS} = -0V$. OFF ($R_{DS} = 10^{10} \Omega$)	$V_{GS} = 5V$. ON ($R_{DS} = 1K$)	L $\approx 0V$



- La CMOS posee más complejidad y menos densidad de integración que PMOS, mayor velocidad que NMOS, y menor disipación de potencia que PMOS o NMOS.

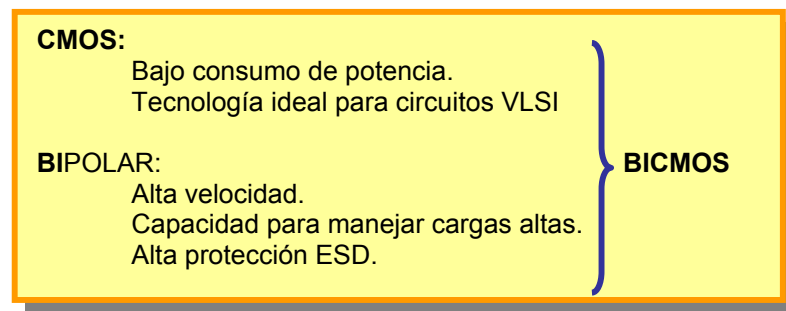
El factor de carga, suele estar limitado por las capacidades de las entradas de otros circuitos.

Existe un aumento de la potencia disipada con la frecuencia de trabajo. Esto se debe a que al pasar del estado bajo al alto, tiene que suministrarse una corriente (con oscilación momentánea), a las capacidades parásitas, originándose impulsos de corriente; a mas ciclos/s, mas impulsos de I , lo que se traduce en más disipación de potencia.

La CMOS, tiene menor resistencia de salida para cada estado con relación a la NMOS; esto quiere decir que la constante de tiempo del circuito de salida es menor, con lo que la velocidad de conmutación es mayor.

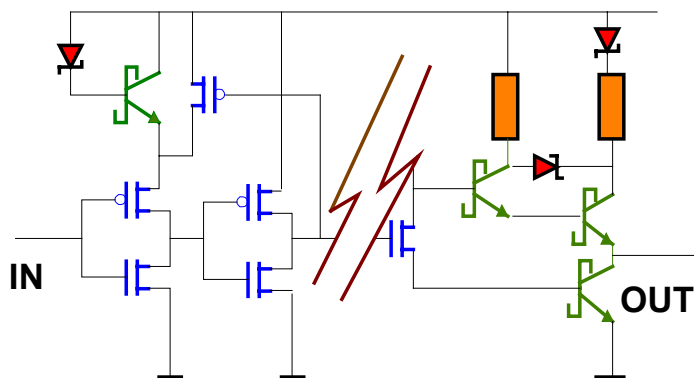
Existen CMOS de drenaje abierto (el equivalente a colector abierto en TTL) y CMOS Triestado.

1.8.6-BICMOS TECHNOLOGY.



- ❑ Esta técnica combina elementos TTL con CMOS dentro de un mismo IC. El circuito se basa en un núcleo realizado con CMOS y con una salida implementada mediante un transistor bipolar NPN. Aunque el transistor bipolar tiende a tener un alto consumo estático de corriente, su menor consumo dinámico le hace adaptarse mejor para altas frecuencias que un bipolar puro o un CMOS puro, pues el consumo dinámico constituye la mayor parte del consumo total.
- ❑ Aprovecha las ventajas de la familia TTL y las de CMOS; trabaja a velocidades equivalentes a las de la familia TTL avanzada pero con una reducción de cerca del 90% de potencia con relación a esta.

En la figura se muestra las etapas de entrada y salida simplificadas de transceiver ABT (Advanced BICMOS Technology).



Las entradas se diseñan para ofrecer niveles compatibles TTL, con $V_{IH(min)}$ de 2V y $V_{IL(max)}$ de 0,8V.

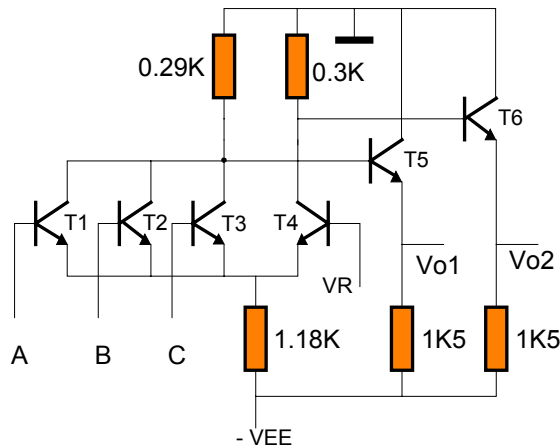
Al estar las entradas implementadas en circuitos CMOS, la impedancia de entrada es muy grande, con lo cual cargan poco al bus donde se conectan.

1.8.7-ECL (CML)

Emitter Coupled Logic o Current Mode Logic. Esta técnica, bipolar también, evita que los transistores trabajen a saturación (al contrario que la TTL donde los transistores conmutan entre corte y saturación), de esta manera se consiguen mayores velocidades al reducirse el tiempo de almacenamiento asociado al paso de la saturación al corte.

- El circuito básico consta esencialmente de una configuración amplificador diferencial, con una corriente fija a través de la resistencia de emisor.

Los transistores están polarizados a una corriente siempre menor que la corriente de saturación, y el principio de funcionamiento está basado en la conmutación de la corriente (CML), entre un transistor y otro.



Puerta ECL:

$$V_R = -1.175V; V_{EE} = -5.2V$$

A	B	C	T1	T2	T3	T4	T5	T6	V1	V2
0	0	0	NC	NC	NC	C	C	NC	1	0
0	0	1	NC	NC	C	NC	NC	C	0	1
0	1	0	NC	C	NC	NC	NC	C	0	1
0	1	1	NC	C	C	NC	NC	C	0	1
1	0	0	C	NC	NC	NC	NC	C	0	1
1	0	1	C	NC	C	NC	NC	C	0	1
1	1	0	C	C	NC	NC	NC	C	0	1
1	1	1	C	C	C	NC	NC	C	0	1

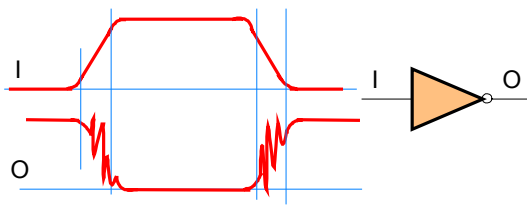
- Las tensiones correspondientes a los niveles lógicos son para el uno lógico ≈ 0 voltios y para el cero lógico ≈ -2 voltios.
- Dependiendo de donde se tome la salida, esta puerta funciona como OR (V2) o como NOR (V1).
- Los transistores T5 y T6 son dos etapas seguidor de emisor, con lo que se logra una muy baja impedancia de salida, de esta manera se disminuye la constante de tiempo RC. Estas etapas, también adaptan los niveles de salida a los niveles ECL correctos.

1.8.8-NIVELES DE INTEGRACIÓN.

- Esta clasificación de los IC-s se realiza en función de la complejidad de la circuiteria interna, medida de diversos modos según fabricantes, en número de puertas lógicas, circuitos, transistores equivalentes etc.

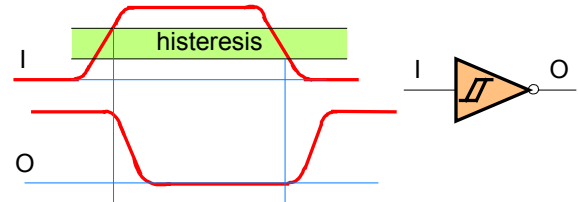
Nivel de integración	Número de puertas lógicas	Aplicación
SSI (Small Scale Integration)	<12	AND, OR, NAND, NOR, X-OR, X-NOR, INV
MSI (Medium Scale Integration)	12 a 99	ENCODERS, DECOD, MUX, DMUX, REG, CONT
LSI (Large Scale Integration)	100 a 9999	MEMORY
VLSI (Very Large Scale Integration)	10000 a 99999	ASIC, μP
ULSI (Ultra Large Scale Integration)	>100000	ASIC, μP

1.8.9-TRIGGER SCHMITT



- En un circuito lógico puede ocurrir que, cuando los tiempos de transición son relativamente largos, la respuesta presente oscilaciones ante el intervalo de incertidumbre del estado lógico presente como excitación.

- En los circuitos lógicos con entrada tipo Schmitt, las transiciones solamente se realizan en puntos claramente definidos de tensión correspondientes a los niveles lógicos del cero y uno, evitándose las oscilaciones.



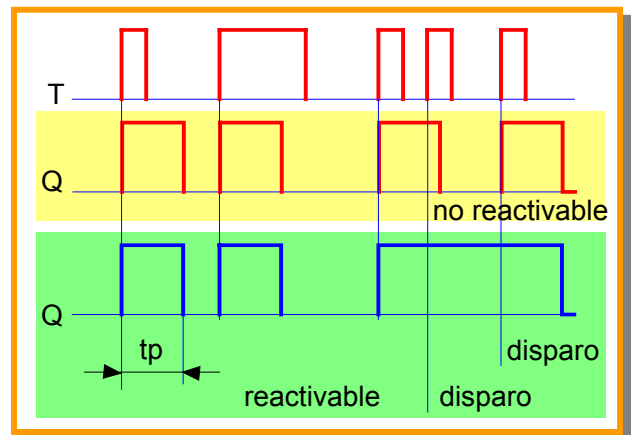
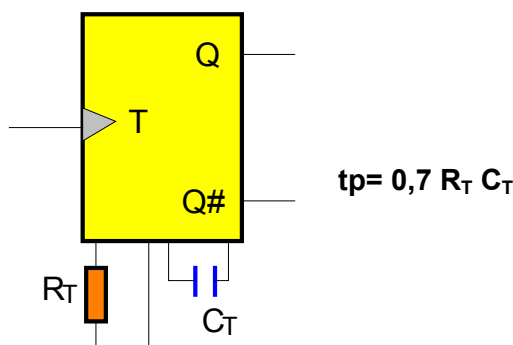
1.8.10-MULTIVIBRADORES.

MULTIVIBRADOR MONOESTABLE.

- Estos circuitos poseen un estado estable o de reposo. Mediante una señal de disparo o activación del monoestable, se los lleva a un estado inestable en el cual permanecen un tiempo t_p .

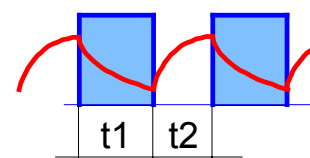
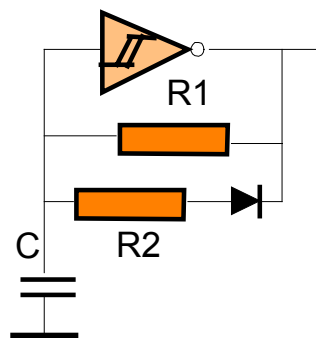
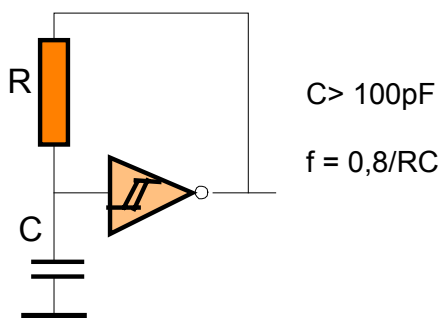
MONOESTABLE NO REACTIVABLE. Este monoestable debe regresar de su estado de disparado o activado, a su estado estable antes de que pueda ser activado de nuevo.

MONOESTABLE REACTIVABLE. Este monoestable puede ser disparado o activado, mientras está en su estado inestable.



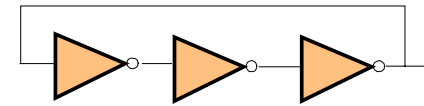
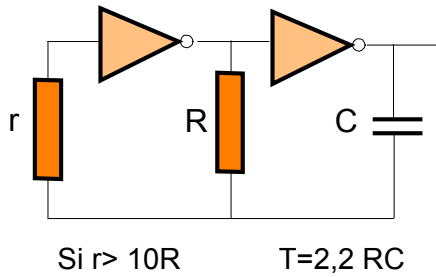
MULTIVIBRADOR ASTABLE

- Estos circuitos no poseen ningún estado estable, es decir, están oscilando continuamente.

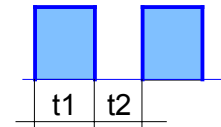


$$t_1 = 0,4 R_1 C$$

$$t_2 = 0,4 (R_1 // R_2) C$$



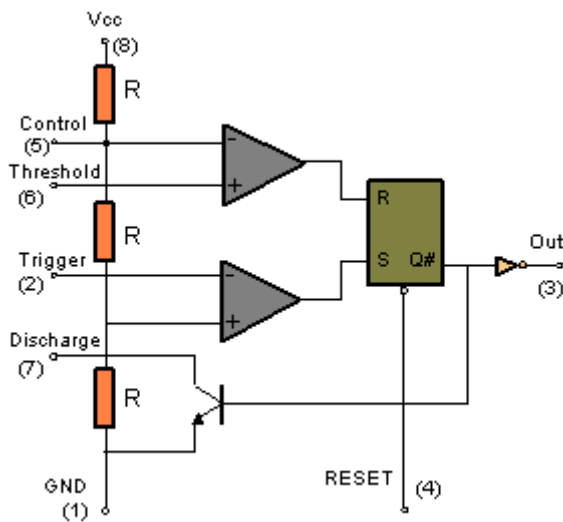
$t_D = t_{pd}$ de un inversor;



$t_1 \approx t_2 = 3 t_D$

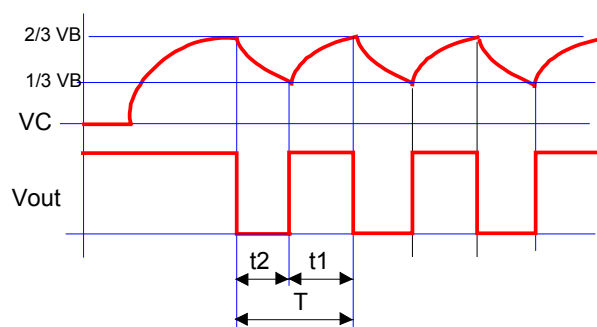
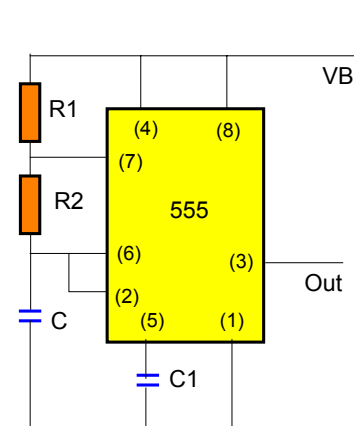
TIMER 555

CONFIGURACIÓN INTERNA



V(2)	V(6)	R	S	Q#	V(3)
$< 1/3 V_{cc}$	$< 2/3 V_{cc}$	0	1	0	$\approx V_{CC}$
$> 1/3 V_{cc}$	$< 2/3 V_{cc}$	0	0	Q#	Q
$< 1/3 V_{ee}$	$> 2/3 V_{ee}$				
$> 1/3 V_{cc}$	$> 2/3 V_{cc}$	1	0	0	≈ 0

CONFIGURACIÓN COMO ASTABLE

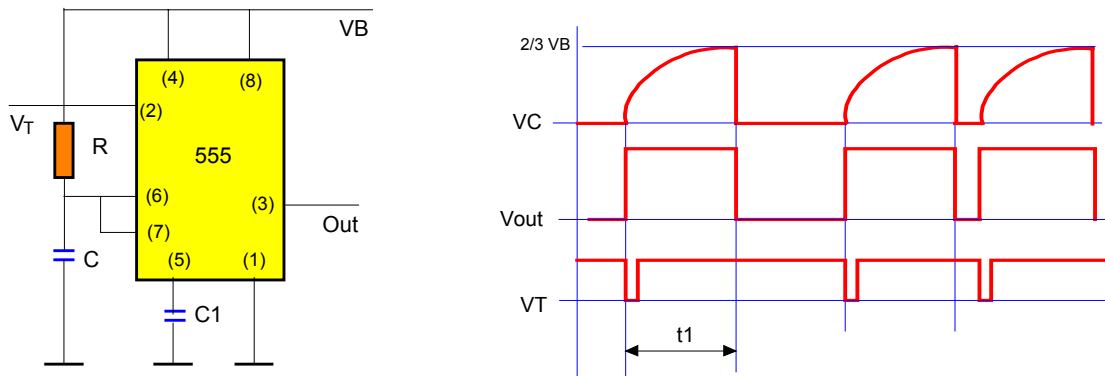


$$t_1 = -(R_1 + R_2) C \ln 0,5$$

$$t_2 = -R_2 C \ln 0,5$$

$$T = -(R_1 + 2R_2) C \ln 0,5$$

CONFIGURACIÓN COMO MONESTABLE



Cuando la V_T es menor de $1/3$ de la tensión de alimentación del circuito, el monoestable se dispara. El tiempo en el cual permanece en este estado inestable, viene dado por la fórmula

$$t_1 = -RC \ln |1/3|$$



Electrostatic Sensitive Devices

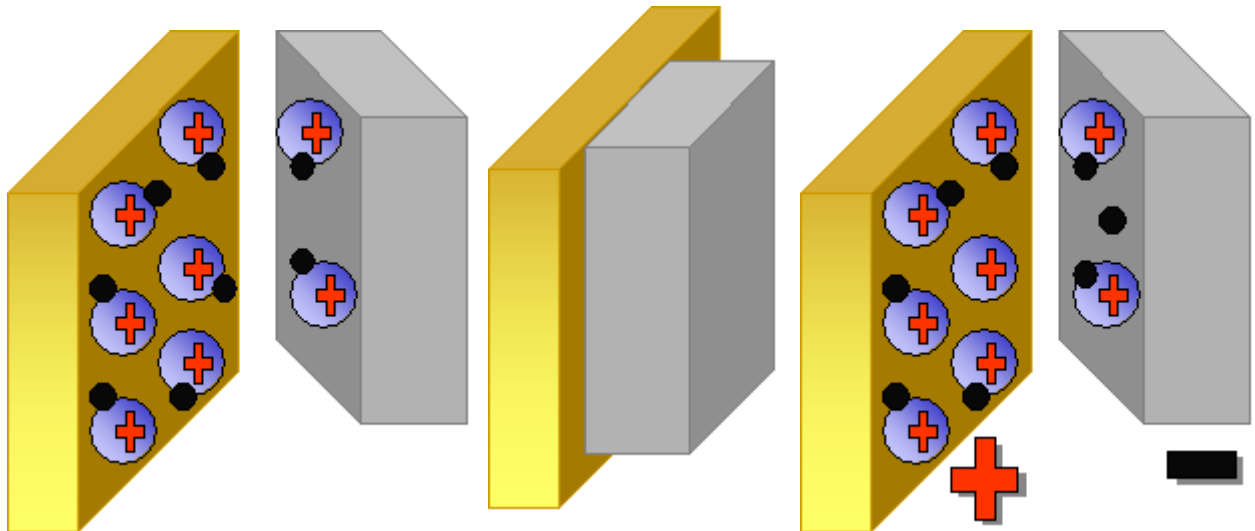
2

2.-ESD (ELECTROSTATIC DISCHARGE)

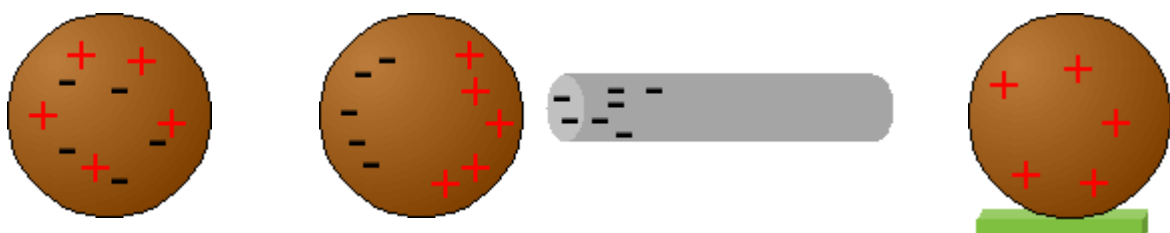
En la vida cotidiana, la electricidad estática no es más que “algo con lo cual una persona se carga”. Sabemos de su existencia cuando al acercarnos a un objeto u otra persona, sentimos esa chispa que más que dolorosa es “sorprendente”, sobre todo por el desconocimiento de cómo y porqué, nuestro cuerpo puede llegar a generarla. Pero la realidad es que la ESD se ha convertido en los últimos años, debido al desarrollo de componentes electrónicos de alta tecnología y sobre todo a partir de la incorporación de la tecnología MOS (Metal Oxide Semiconductor), en uno de los problemas más serios con los que un equipo electrónico fabricado con esos componentes, tiene que enfrentarse y a la vez, un reto para el personal que debe manipular estos equipos.

2.1-GENERACIÓN DE ESD.

- ❑ La electricidad estática se produce por un desequilibrio de electrones, bien por defecto o bien por exceso de los mismos, en la superficie de un material. Este desequilibrio produce un campo eléctrico que influye sobre otros cuerpos situados en su radio de acción.
- ❑ La forma más común de generar este desequilibrio, es decir, electricidad estática es cuando las superficies de dos cuerpos diferentes se ponen en contacto por frotamiento más o menos pronunciado, y después se separan.



Otras formas son por contacto con un objeto cargado y por inducción. En la figura que sigue, una esfera metálica es puesta cerca de una barra cargada negativamente. Las cargas de la esfera se reorganizan para compensar la proximidad de las cargas negativas de la barra. Estas cargas que aparecen definidas en diferentes partes de la esfera, han sido creadas por inducción. No existe transferencia de cargas de la barra a la esfera ni viceversa. Si la esfera se pone en contacto con otro elemento, perderá las cargas negativas y quedará cargada positivamente.



Esto pone de manifiesto, que un componente electrónico puede ser cargado de estática por inducción, por un elemento cotidiano (bolígrafo de plástico, vaso de plástico, etc.) que esté cargado de estática y que esté cercano a dicho componente. Si este componente es puesto en contacto con tierra (por ejemplo a través de un operario con muñequera), el componente se quedará cargado de estática. De aquí la gran importancia que tiene el hecho de acondicionar un puesto de trabajo teniendo en cuenta todas las posibles causas que pueden originar una ESD.

- Una persona por el simple hecho de caminar, genera electricidad estática ya que su zapato toca el suelo y se separa de éste. Por ejemplo una persona caminando por una alfombra en un ambiente de humedad relativa por debajo del 25%, puede crear una carga electrostática equivalente a 35.000 V. Si la humedad relativa está comprendida entre el 65 y 90%, el potencial que se crea baja hasta 1.500V.

Esta forma de generar electricidad estática por contacto y separación de superficies es conocida como *triboelectricidad*.

Las cargas negativas se transfieren desde una superficie a la otra. La superficie que gana electrones, quedará por lo tanto cargada negativamente, mientras que la que pierde esas cargas, quedará cargada positivamente. El sentido de esta transferencia y la cantidad de carga que se crea (transfiere) va a depender de la naturaleza de los materiales, del área de la superficie puesta en contacto, de la velocidad con la cual los dos cuerpos se separan, de la humedad relativa del espacio donde se produce el contacto y separación, etc. La forma de saber el sentido de transferencia la tenemos consultando la tabla de la serie triboeléctrica.

En esta tabla se muestra una serie de elementos ordenados según la facilidad que tienen para ceder electrones cuando son puestos en contacto. Según se asciende en la tabla el elemento correspondiente tiene más facilidad para ceder electrones (quedar cargado positivamente) que el que le precede por debajo. Existe una relación directa entre el voltaje electrostático generado y la separación que existe entre los materiales en la tabla triboeléctrica. Lógicamente, éste no es el único factor que influye en el voltaje, también hay que tener en cuenta la forma de contacto, la velocidad de separación entre ellos, el grado de humedad, etc.

- En el momento en el que la carga se crea en un material, ésta constituye una carga electrostática. Cuando esta carga se transfiere desde el material, constituye lo que se llama una descarga electrostática (*Electrostatic Discharge*, o **ESD**). Este fenómeno ocurre cuando la diferencia de potencial entre las dos superficies es lo suficiente grande como para perforar el dieléctrico del medio que las separa.

SERIE TRIBOELECTRICA

AIRE SECO
CABELLO HUMANO
PELO DE CONEJO
VIDRIO
MICA
NYLON
LANA
PIEL
PLOMO
SEDA
ALUMINIO
PAPEL
ALGODÓN
ACERO
MADERA
LACRE
GOMA DURA
NIQUEL, COBRE
LATÓN, PLATA
ORO, PLATINO
POLIÉSTER
ACETATO
POLIURETANO
POLIETILENO
POLIPROPILENO
PVC
SILICIO
TEFLÓN

2.2-INFLUENCIA DEL MATERIAL EN LA CARGA ESTÁTICA.

AISLANTES.

- En la superficie de un aislante puede crearse una importante cantidad de carga. Debido a la dificultad para que los electrones se desplacen por el material, la superficie de éste puede estar a la vez cargado positivamente en una zona y negativamente en otra, es decir, la carga electrostática está en puntos localizados del material.

CONDUCTORES.

- ❑ La carga electrostática puede ser creada en los conductores de forma triboeléctrica (al igual que en los aislantes). Cuando un material conductor se carga, esta carga se distribuye uniformemente por toda la superficie del material. Si un conductor cargado se pone en contacto con otro conductor, los electrones serán transferidos entre ambos de manera fácil y rápida. Si el conductor permanece aislado de otros cuerpos, la carga estática permanece en el conductor.

MATERIALES DISIPADORES DE ESTÁTICA.

- ❑ Este tipo de materiales tiene una resistencia eléctrica comprendida entre los materiales aislantes y los materiales conductores. El flujo de electrones se controla mediante la resistencia de la superficie o volumen del material.

La carga electrostática se puede generar también como en los anteriores tipos de materiales, de forma triboeléctrica. En estos casos, este material al igual que un conductor permitirá una transferencia fácil de la carga hacia el suelo u otro elemento conductor, pero de una manera mas controlada, a menor velocidad que un conductor de las mismas dimensiones. En cualquier caso esta velocidad será siempre muy superior a la de un aislante.

2.3-EFECTOS PRODUCIDOS POR ESD.

- ❑ El daño que se produce en un componente electrónico que es manipulado en unas condiciones en las que no se cumplen las normas básicas de control de ESD, y es afectado por una descarga de este tipo, puede ser clasificado en dos categorías: *defecto latente* y *fallo catastrófico*.

2.3.1-DEFECTO LATENTE.

- ❑ Un dispositivo con un defecto latente, es aquel que ha sido deteriorado por una descarga electrostática, pero no ha sido destruido. Este tipo de defecto ocurre cuando el impulso de la descarga electrostática no es suficientemente energético como para destruir dicho dispositivo, aunque si degradarlo en mayor o menor medida.
- ❑ Generalmente estos dispositivos después de sufrir estas descargas, todavía están dentro de los límites de funcionamiento que marcan las hojas de datos correspondientes, por lo tanto, es extremadamente difícil identificar un defecto de este tipo.
- ❑ Un dispositivo puede estar sometido a numerosas descargas que crean defectos latentes, cada una de las cuales va degradándolo más, antes de que finalmente, el defecto sea catastrófico. Si el defecto latente se produce en el proceso de fabricación o manipulación hasta la aplicación, esa aplicación habrá nacido ya defectuosa y posiblemente causará un fallo prematuro en el funcionamiento de la misma.
- ❑ Se calcula que alrededor del 90% de los daños producidos por la ESD pertenecen a este tipo de efecto.

2.3.2-FALLO CATASTRÓFICO.

- ❑ En este caso, el dispositivo sometido a una descarga electrostática queda dañado permanentemente debido a la fusión de metalizaciones, rotura de uniones o destrucción de óxidos.

- ❑ La detección de este tipo de fallo es sencilla y se realiza mediante el test correspondiente del dispositivo. Aproximadamente el 10% de los daños por ESD pertenecen a este grupo.

El daño que se puede producir a un dispositivo por una ESD, depende de la capacidad del dispositivo para disipar la energía de la descarga o soportar los niveles de tensión implicados en la misma. Este factor se conoce como *sensibilidad a la ESD*.

Por ejemplo, existen algunos componentes electrónicos del tipo MOS que no llevan protección a la entrada. Estos dispositivos son de alta sensibilidad y pueden ser dañados por voltajes estáticos inferiores a los 10 V.

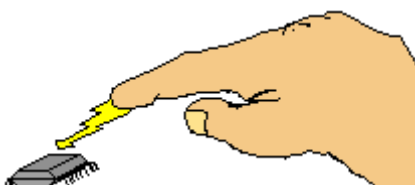
En la tabla de la derecha, se muestra la sensibilidad a la ESD de varios tipos de componentes electrónicos.

COMPONENTE	VOLTIOS
VMOS	30 – 1.200
MOSFET, EPROM	100 - 300
FET	150 – 7.000
O. A.	190 – 2.500
DIODOS SCHOTTKY	300 - 2500
RESISTENCIA DE PELÍCULA	300 – 3.000
TRANSISTOR BIPOLAR	380 – 7.000
SCR	680 – 1.000
TTL SCHOTTKY	1.000 – 2.500

2.4-TIPOS DE DESCARGA.

- ❑ Una ESD puede ocurrir de una de las tres maneras siguientes: *descarga directa desde un cuerpo cargado, hacia un dispositivo sensible a descargas (ESDS), descarga desde un dispositivo cargado o por campo electrostático inducido.*

DESCARGA HACIA UN DISPOSITIVO.



- ❑ Este tipo de descarga ocurre cuando un cuerpo cargado se descarga hacia un dispositivo ESDS (Electrostatic Discharge Sensitive).

La causa más común de daños por este tipo de ESD es la descarga directa del cuerpo humano hacia un elemento ESDS. El cuerpo humano está generando continuamente electricidad estática como resultado de la actividad cotidiana. Esta electricidad se descarga de forma regular la mayor parte del tiempo sin que nosotros lo percibamos, hacia elementos de menos potencial, es decir, no solo existe descarga cuando nosotros la percibimos. En este sentido, los humanos sentimos la descarga cuando el potencial que adquiere la carga acumulada está por encima de los 3.000 voltios, llegamos incluso a oírla por encima de los 5.000 voltios y la vemos en forma de chispa por encima de los 8.000 voltios.

Otra descarga similar es la que se produce desde un objeto conductor, como puede ser una herramienta metálica, hacia un elemento ESDS.

DESCARGA DESDE UN DISPOSITIVO.

- ❑ La transferencia de carga desde un dispositivo ESDS se considera también una ESD. Por ejemplo, un elemento semiconductor que por un motivo determinado se desliza por una superficie, se cargará de electricidad estática. Si después este dispositivo entra en contacto con una superficie metálica, se producirá una descarga rápida desde aquel, a la superficie. Esta descarga puede ser tan destructiva (o más, según el tipo de dispositivo) como la que se produce desde el cuerpo humano hacia un dispositivo.

DESCARGA POR CAMPO INDUCIDO.

- ❑ Siempre que un cuerpo queda cargado de electricidad estática, se forma un campo electrostático asociado a dicha carga. Si un dispositivo ESDS se sitúa en dicho campo, en dicho dispositivo se puede inducir una carga electrostática, con lo cual, si entra en contacto con otro cuerpo, se puede producir una descarga. Por otro lado, el campo electrostático puede ser de tal magnitud que puede inducir un voltaje a través de un determinado dieléctrico tal que lo perfora.

2.5-CONTROL DE LA ESD.**2.5.1-PRINCIPIOS BÁSICOS.****DISEÑO.**

- ❑ Se trata de diseñar los dispositivos para que sean inmunes en la medida de lo posible al fenómeno ESD. Para ello en los diseños electrónicos se deben usar menos elementos sensibles a la ESD, implementar protecciones en las entradas de los dispositivos, etc.

REDUCCIÓN DE LA GENERACIÓN DE ESTÁTICA.

- ❑ Para ello hay que comenzar en la medida de lo posible por reducir en el entorno de trabajo los procesos o materiales que originan la generación de ESD, como el contacto y separación de diferentes materiales y plásticos comunes. Los procesos o materiales que no se puedan eliminar, los mantendremos al mismo potencial.

DISIPACIÓN Y NEUTRALIZACIÓN DE ESTÁTICA.

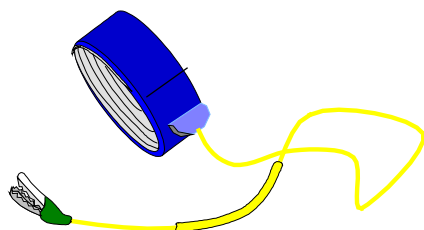
- ❑ Este principio trata de limitar los efectos perniciosos una vez que se produce una ESD. Una buena puesta a tierra y el uso de materiales disipadores o conductores de estática son los pilares fundamentales de este principio. Por ejemplo, un operario que está cargado de estática, puede eliminar esa carga haciendo uso de la *pulsera antiestática*, o caminando por encima de una alfombrilla de disipación de estática. En estos casos la carga se desplaza a tierra en lugar de a un elemento ESDS. Cuando un dispositivo está cargado, para evitar el deterioro del mismo, se realiza la descarga a través de un elemento disipador de estática, con lo cual se controla el flujo de dicha descarga.
- ❑ Para el caso de ciertos elementos como los plásticos comunes y algunos otros aislantes, la puesta a tierra no elimina la carga electrostática ya que no existe un camino conductor para ello. En estos casos lo que se utiliza es un ionizador. El ionizador se encarga de generar iones positivos y negativos que son atraídos por la superficie del plástico o aislante cargado, de tal manera que neutraliza la carga.

PROTECCIÓN DE LOS DISPOSITIVOS.

- ❑ En este caso se trata de evitar las descargas que se producen al alcanzar elementos susceptibles. Para ello estos elementos deben estar provistos de una buena toma de tierra o en contacto con un elemento disipador. La manipulación de todos los componentes sensibles a la descarga de estática debe realizarse dentro de un área acondicionada de seguridad. Por otro lado, estos componentes se deben transportar en contenedores o embalajes con el blindaje apropiado, que los proteja de la acumulación y de la descarga de electrostática.

2.5.2-ELEMENTOS DE PROTECCIÓN.

PULSERA ANTIESTÁTICA.



- El medio primario para controlar la carga estática en las personas es mediante la pulsera antiestática. Esta pulsera se conecta a un potencial de tierra, mediante el cordón de tierra, manteniendo por lo tanto, a la persona que la porta a dicho potencial.

La mayoría de estas pulseras lleva una resistencia limitadora de corriente en la unión entre el cordón de tierra y la propia pulsera. El valor de esta resistencia suele ser de **1MΩ, ¼ W**.

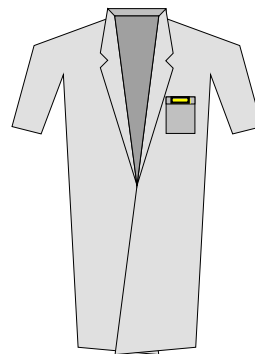
SUELOS Y CALZADO ANTIESTÁTICO.

- Otro método para controlar la carga electrostática en las personas es mediante la utilización de materiales especiales en el suelo, junto con calzado capaz de controlar la ESD. La combinación de estos elementos proporciona un camino para la disipación de la carga electrostática, por lo tanto, reducen la acumulación de carga estática en las personas.



ROPA ANTIESTÁTICA.

- La ropa que se viste es muy importante en entornos como habitaciones limpias y ambientes muy secos. La carga electrostática que se genera en la ropa puede descargarse sobre un dispositivo ESDS o crear campos electrostáticos que pueden inducir cargas en el cuerpo humano. El hecho de que una persona sea puesta a potencial de tierra no garantiza que la ropa pueda eliminar su carga a tierra a través del cuerpo. La única manera de garantizar que la ropa queda descargada es mediante vestimenta especial (batas, monos,...) de control de estática.



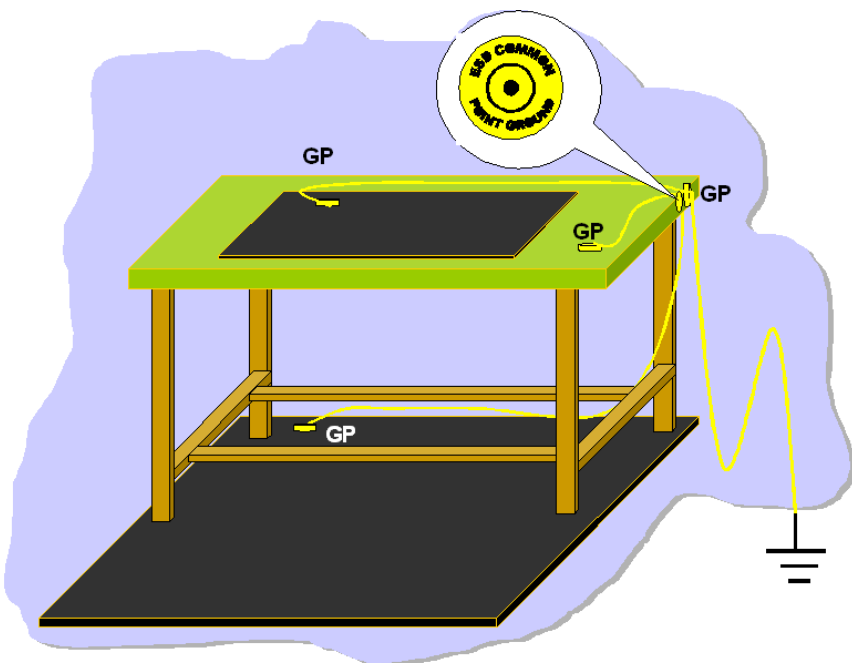
Esta vestimenta en unos casos puede tener una conexión de puesta a tierra (batas conductivas) con las limitaciones por un lado, de movilidad que ello supone y por otro lado, de seguridad para la persona al moverse en entornos donde existen voltajes de alimentación de más de 50 V.

En otros casos, la alternativa está en el uso de batas de algodón con broches de conexión a la muñequera antiestática y con partes de la misma en conexión directa con el cuerpo, como son los puños elásticos de las mangas. La fibra natural de algodón tiene desde el punto de vista de la ESD, propiedades beneficiosas ya que absorbe humedad del aire y se mantiene con un cierto grado de conductividad que facilita la evacuación de las cargas generadas por la ropa y también facilita la disipación de las mismas al aire.

ESTACIÓN DE TRABAJO CON PROTECCIÓN ESD.

- Muchos elementos ESDS son ensamblados, manipulados y reparados en una estación fija de trabajo. Una estación de trabajo protegida contra ESD es el espacio de trabajo que está diseñado, construido y equipado con materiales y equipos destinados a limitar el daño a los elementos sensibles a la ESD. Puede ser un lugar donde un operario realiza ciertos trabajos con equipos electrónicos dentro de un laboratorio, o el espacio destinado a albergar los equipos aviónicos en una aeronave, o un lugar dentro de una **EPA (ESD Protected Area)**.

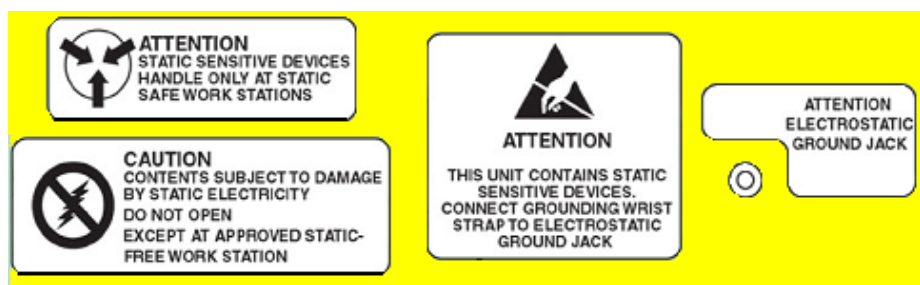
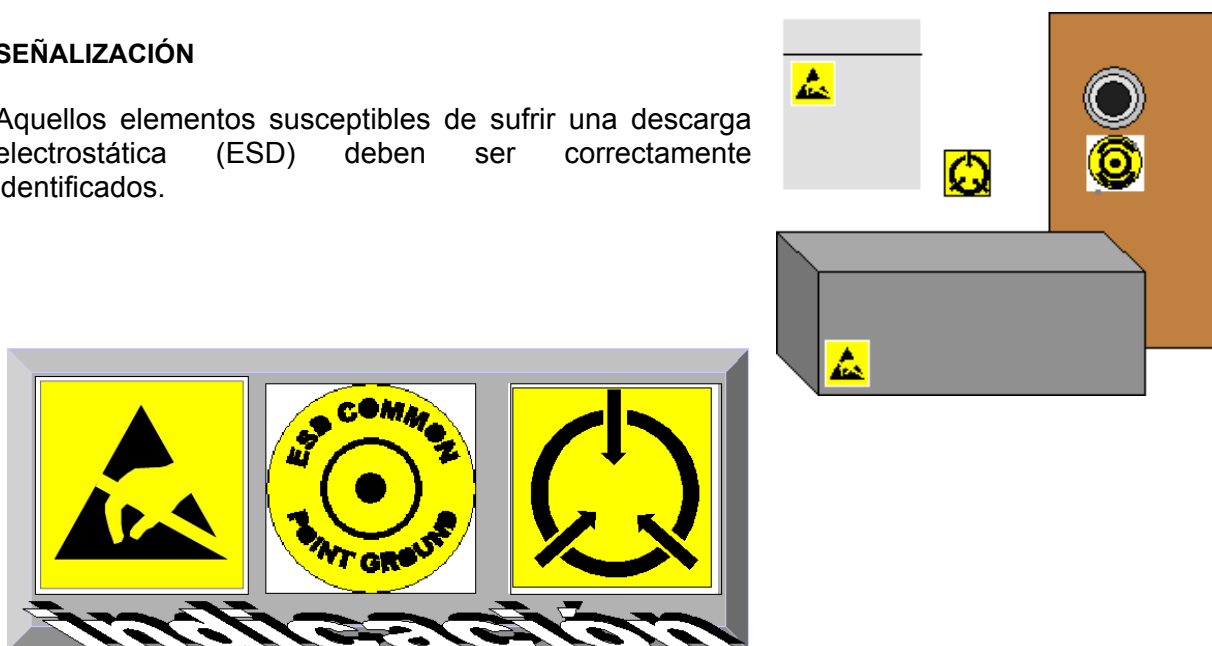
- Estas estaciones de trabajo tienen un material disipativo de estática, tanto en el suelo por donde se moverán las personas alrededor del puesto, como en la superficie donde se manipularán los ASDS. Están también provistas de medios para poder conectar todas las superficies de trabajo, el material que se debe manipular, y los equipos a un *punto común de tierra* o *Ground Point (Gp)*. Este punto común deberá estar conectado a la tierra del edificio. También se pueden conectar elementos para poner a potencial de tierra a los operarios que trabajan en la estación (generalmente este elemento es la pulsera antiestática).



El punto común de tierra debe estar perfectamente identificado en la estación de trabajo. Para ello se coloca el símbolo de punto común de tierra en el lugar que corresponde al potencial de tierra.

SEÑALIZACIÓN

Aquellos elementos susceptibles de sufrir una descarga electrostática (ESD) deben ser correctamente identificados.



IONIZACIÓN.

- Como se ha indicado mas arriba, la forma de ejercer un control sobre la ESD en elementos conductores, materiales disipadores de estática y personal, es la conexión directa de esos elementos a tierra (**Gp**). Para los materiales aislantes como son la mayoría de los plásticos comunes, la puesta a tierra no es efectiva ya que no puede producirse la necesaria transferencia de cargas hacia aquella. Para estos casos, el método empleado es el de ionización. Mediante un ionizador balanceado (este tipo produce cantidades similares de los dos tipos de iones, al contrario del ionizador doméstico, que solo los produce negativos) las moléculas de los gases del aire en el entorno de trabajo, son cargadas con diferente polaridad. La carga estática en la superficie de los materiales aislantes en dicho entorno es neutralizada por estas cargas.



2.5.3-PROTECCIÓN DE IC-s.

- Teniendo en cuenta todo lo expuesto más arriba, se puede decir que la mejor manera de proteger los IC-s (Integrated Circuits) es mantener a éstos, al mismo potencial que los demás elementos del entorno donde se encuentran. En base a lo anterior, existen cinco reglas cuyo cumplimiento prácticamente elimina el riesgo de daño a un IC por ESD:

1.- Los IC-s y todo lo que está próximo y relacionado a ellos, se debe mantener a potencial de tierra.

2.- La superficie de trabajo donde los componentes son situados para su manipulación, debe ser de material disipador de estática y estar conectado a tierra.

3.- Todos los materiales aislantes deben ser eliminados del área de trabajo o si esto no es posible, deben ser neutralizados con un ionizador. La ropa generadora de estática debe ser cubierta con un traje anti-ESD

4.- Cuando los IC-s van a ser movidos de un lugar a otro, deben situarse en recipientes disipadores de estática con las superficies interiores en contacto con los componentes.

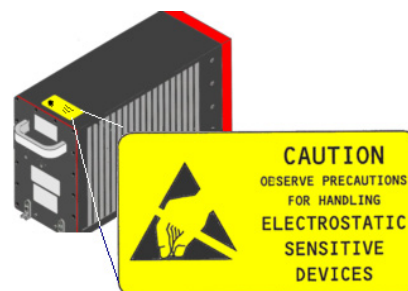
5.- El operario que manipule los IC-s, debe estar puesto a tierra mediante la pulsera antiestática, y en un entorno donde exista la alfombra disipadora de estática.

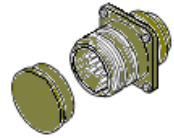
Se debe tener en cuenta también el grado de humedad en el entorno de trabajo. Es recomendable un grado de humedad entre el 40% y 60%. Por debajo de este rango se incrementa el riesgo de generación estática en materiales aislantes.

2.6-ESD EN LA AERONAVE.

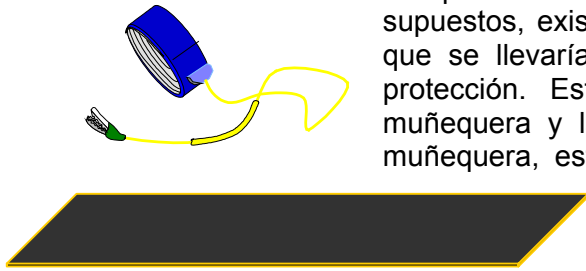
2.6.1-LRU-s. (Line Replaceable Units)

- La LRU, cuando se manipula en la aeronave, no presenta realmente, ningún problema de cara a la ESD, siempre que el técnico manipule como un todo, la caja que forma la LRU, dado que esta caja si es metálica, hace de Caja de Faraday con lo que presta un blindaje a la circuitería interna. Sin embargo, hay que tener presente en todo momento, que los conectores de la LRU puede que no estén protegidos contra la ESD, por lo tanto, cuando se desmonta una LRU de un avión, hay que emplear tapones conductivos antiestáticos para proteger los conectores, procurando no tocar los pines de los mismos durante la operación.





Por otro lado, si la manipulación afecta a una tarjeta electrónica, deben tomarse las precauciones necesarias para evitar la ESD. Para estos supuestos, existen conjuntos móviles para protección contra ESD que se llevarían al lugar de la aeronave donde requiera dicha protección. Estos conjuntos suelen constar de una esterilla, muñequera y las conexiones pertinentes de Ground Point para muñequera, esterilla y masa de los equipos. Además, la tarjeta electrónica, en caso de tener que ser transportada, se debe introducir en bolsas o contenedores antiestáticos.

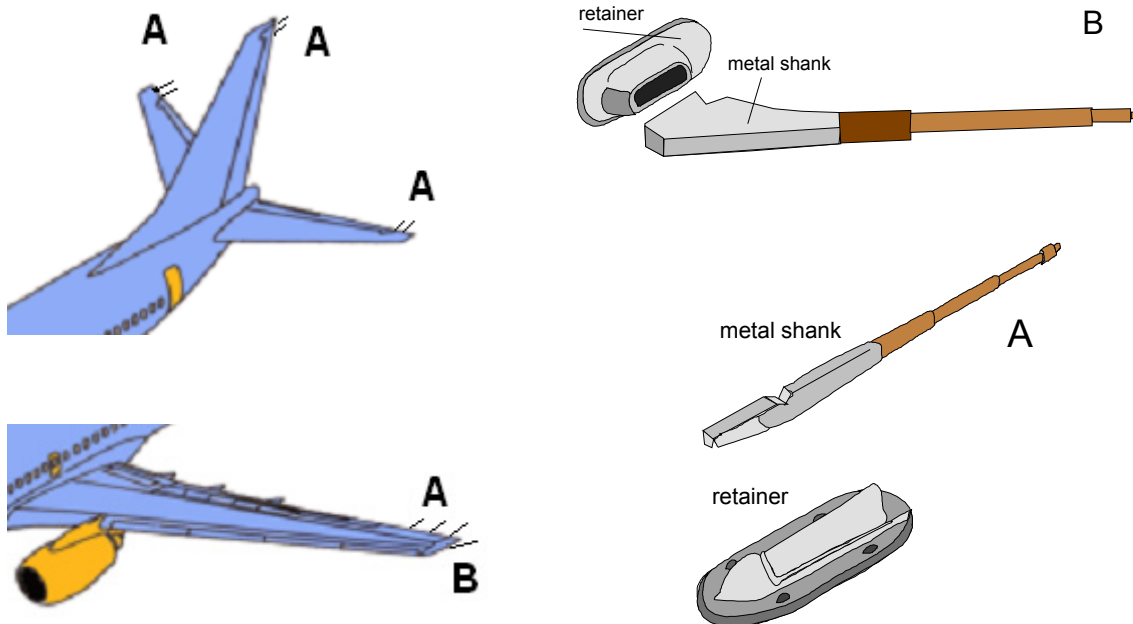


2.6.2-DESCARGADORES DE ESTÁTICA.

- Durante el vuelo, una aeronave rozando continuamente con el aire, se carga de electricidad estática. Esta carga eléctrica es el resultado entre otros motivos, del choque con partículas de precipitación y polvo en la atmósfera. Si la descarga de esta electricidad no es controlada de algún modo, puede causar interferencias en los sistemas de comunicación y navegación. Para evitar este problema, se hace uso de los *descargadores de estática*, cuya misión es realizar la descarga a la atmósfera, a través de un camino de resistencia continua y de una manera controlada, de las cargas acumuladas en la superficie de la aeronave y de esta manera, evitar la descarga brusca debido a la acumulación de cargas (y consecuentemente incremento de potencial) y por lo tanto, evitar las interferencias.

DISTRIBUCIÓN.

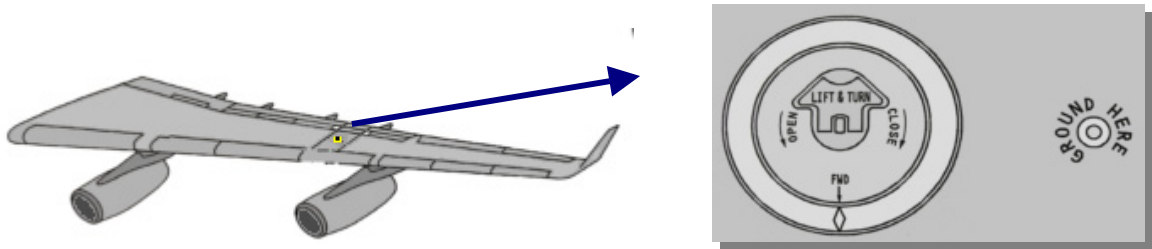
Los descargadores de estática se instalan en dos tipos de emplazamientos: bordes de salida (marcados con la letra A en la figura) y en las puntas (marcados con la letra B) de las alas y estabilizador horizontal para el caso de la figura representada. La vara de los descargadores del borde de salida son de longitud mayor que los de las puntas. En ambos casos la vara está hecha de nylon moldeado.



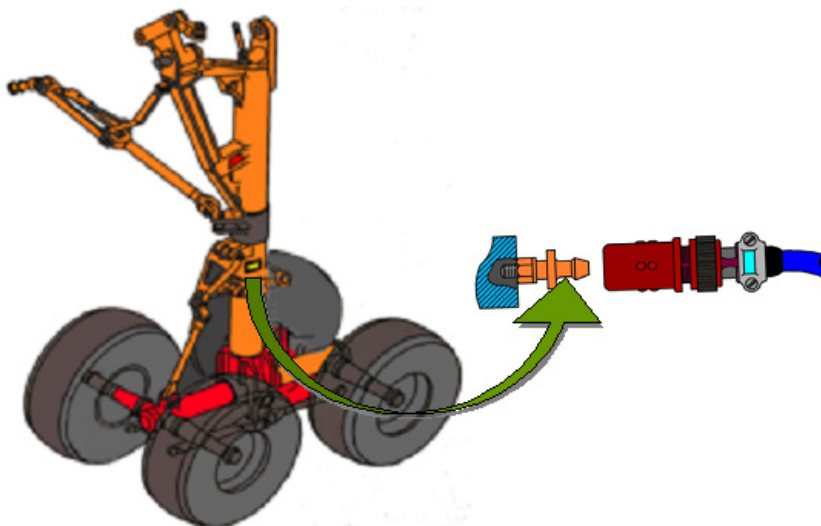
- Los descargadores de estática, mantienen la estructura de la aeronave a, un voltaje por debajo del nivel de umbral de la *descarga de corona*. La descarga de corona es una descarga eléctrica de color azulado-púrpura brillante, que se produce cuando el voltaje en un conductor excede cierto valor crítico y se debe a la ionización de los gases que rodean a dicho conductor. Los descargadores de estática, son el camino de resistencia continua por donde se produce la descarga de estática hacia la atmósfera, ya que éstos tienen el nivel de umbral de corona, por debajo de cualquier otro punto de la estructura de la aeronave. De esta manera, la carga estática que se acumula en las superficies, no llega al nivel crítico que puede provocar una descarga tal que afecte a los sistemas NAV y COM. Como dato de referencia, la resistencia entre la punta del descargador y su base, puede estar comprendida entre 6 y 200 MΩ.

2.6.3-PUNTOS DE TIERRA.

- En la aeronave existen ciertos puntos en el fuselaje (tren de aterrizaje, superficie del ala,...) llamados puntos de tierra. Los técnicos o técnicas y su equipamiento, deben conectarse a los puntos de tierra de la aeronave antes de realizar trabajos de mantenimiento sobre cualquier elemento o sistema sensibles a la descarga electrostática en la misma. En procesos como el llenado de los tanques de combustible de la aeronave, los procedimientos indican la conexión directa mediante un cable entre el camión cisterna y uno de los puntos de tierra de la aeronave. De este modo se evita una posible chispa por descarga electrostática en una atmósfera potencialmente inflamable.



PUNTO DE TIERRA EN EL ALA



PUNTO DE TIERRA EN EL MLG (Main Landing Gear)

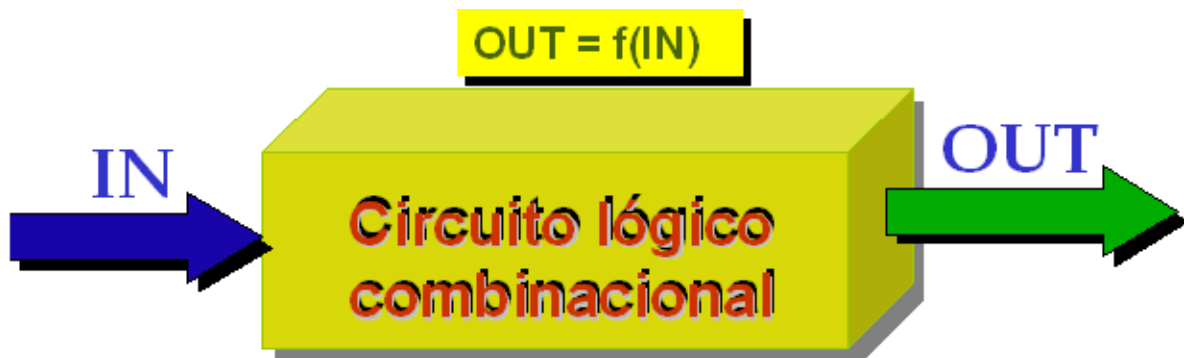


Circuitos Combinacionales

3

3.-ELECTRÓNICA COMBINACIONAL.

- Los circuitos combinacionales son aquellos en los cuales se cumple que los valores de la salida dependen únicamente de los distintos estados que puedan tener las variables de entrada.



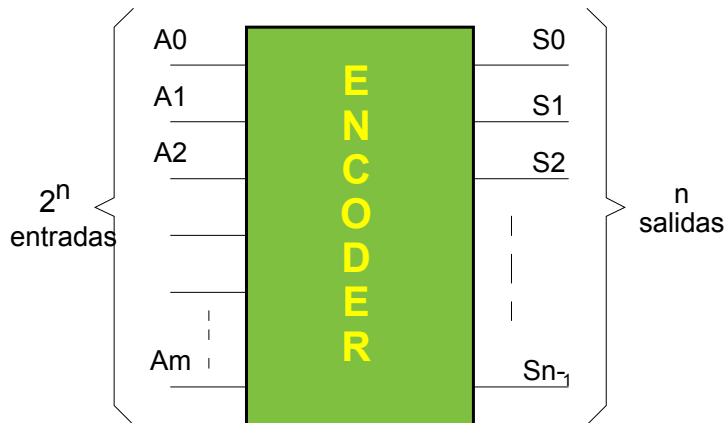
DISEÑO DE CIRCUITOS COMBINACIONALES.

- Los pasos a seguir para diseñar un circuito combinacional son los siguientes:
 1. Traducir el problema en cuestión a valores en la *tabla de la verdad*.
 2. Extraer de la tabla en forma canónica la función que representa el problema. Esta función la extraemos como más nos interese, bien por Minterms o bien por Maxterms.
 3. Simplificar la función.
 4. Construcción del circuito con los dispositivos más adecuados.
- Los circuitos combinacionales pueden clasificarse según su aplicación en dos grandes grupos:

APLICACIÓN ESPECIFICA: El diseño del circuito se realiza en base a unos criterios particulares y específicos para satisfacer unos determinados requerimientos en una aplicación concreta.

APLICACIÓN GENERAL: Estos circuitos combinacionales, generalmente son base o forman parte de otros circuitos electrónicos más complejos. Debido a esto, se encuentran integrados a nivel MSI o LSI por lo que no necesitan ser implementados a partir de puertas lógicas. Ejemplo de este tipo de circuitos de aplicación general son los *codificadores*, *decodificadores*, *multiplexores*, *demultiplexores*, etc.

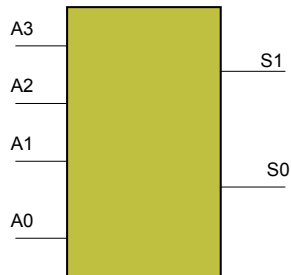
3.1-CODIFICADOR (ENCODER)



- La salida produce un código de n-bits que va a depender de la entrada que esté activa.
- Solamente una de las entradas puede ser activa en un momento dado. El código que se genera en la salida, es el equivalente en binario del valor decimal de la entrada activa.
- Un caso particular de codificador es el *Priority Encoder* (Codificador de prioridad). En este codificador,

si en un momento determinado existe más de una entrada activa, el código de salida va a corresponder a la entrada cuyo valor decimal sea el mayor.

4 LINE TO 2 LINE ENCODER



Partiendo de la premisa de que solamente una de las entradas puede ser activa en un momento dado, realizamos la tabla de la verdad:

A_3	A_2	A_1	A_0	S_1	S_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

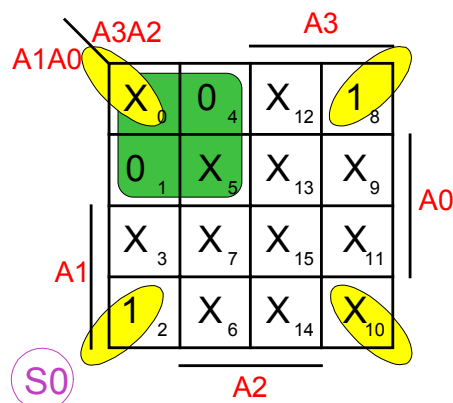
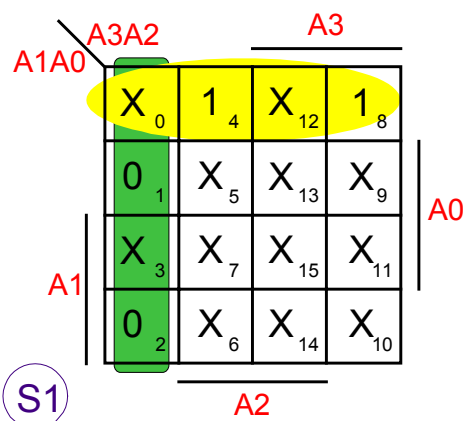
También se puede representar:

IN	S_1	S_0
A_0	0	0
A_1	0	1
A_2	1	0
A_3	1	1

$$S_1 = \Sigma_4 (4, 8) + d(0, 3, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15)$$

$$S_0 = \Sigma_4 (2, 8) + d(0, 3, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15)$$

Tomando Minterms tenemos:



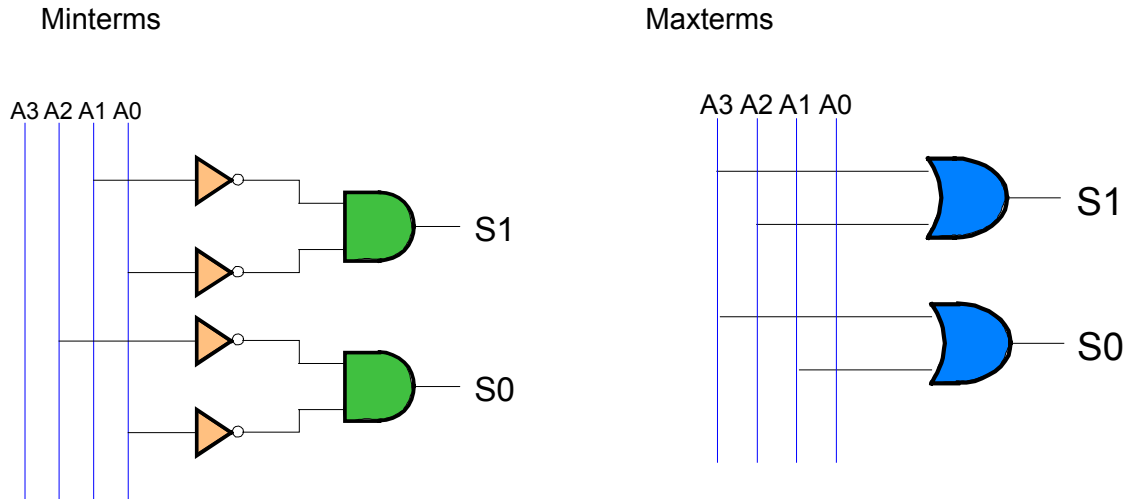
$$S_1 = \bar{A}_1 \bar{A}_0$$

$$S_0 = \bar{A}_2 \bar{A}_0$$

Si tomamos los Maxterms: $S_1 = A_3 + A_2$; $S_0 = A_3 + A_1$

El análisis también se puede realizar teniendo en cuenta que en un momento dado solamente una de las entradas es activa. Si observamos la tabla, S_1 va a ser 1 cuando A_3 sea 1 o A_2 sea 1; es decir, $S_1 = A_3 + A_2$. De la misma forma, S_0 va a ser 1 cuando A_1 sea 1 o A_3 sea 1; es decir $S_0 = A_3 + A_1$. Estos son los mismos resultados que se obtienen sacando los maxterms de la tabla de Karnaugh.

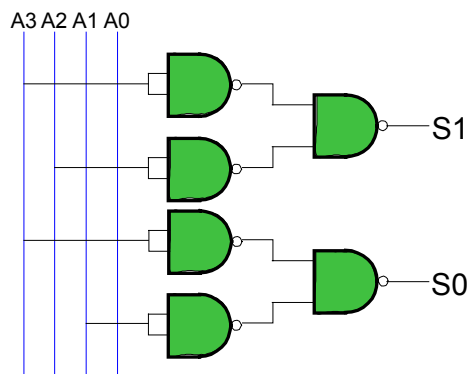
Implementación:



Si queremos implementar con puertas NAND tenemos:

$$S_1 = A_3 + A_2; \quad S_0 = A_3 + A_1$$

$$S_1 = \overline{\overline{A_3} \overline{A_2}}; \quad S_0 = \overline{\overline{A_3} \overline{A_1}}$$

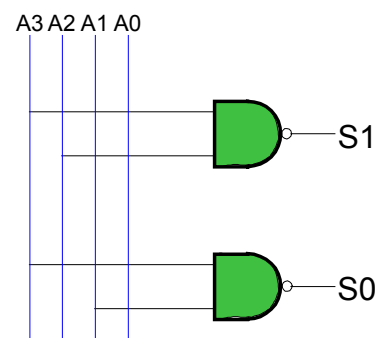


También podemos considerar las entradas activas a nivel bajo:

A_3	A_2	A_1	A_0	S_1	S_0
1	1	1	0	0	0
1	1	0	1	0	1
1	0	1	1	1	0
0	1	1	1	1	1

$$S_1 = \overline{A_2} + \overline{A_3} = \overline{A_2 A_3}$$

$$S_0 = \overline{A_1} + \overline{A_3} = \overline{A_1 A_3}$$

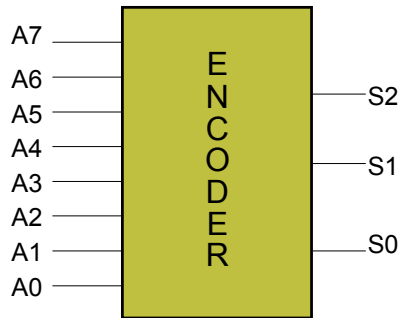


O representada con niveles:

A_3	A_2	A_1	A_0	S_1	S_0
H	H	H	L	L	L
H	H	L	H	L	H
H	L	H	H	H	L
L	H	H	H	H	H

H: HIGH L: LOW

8 LINE TO 3 LINE ENCODER

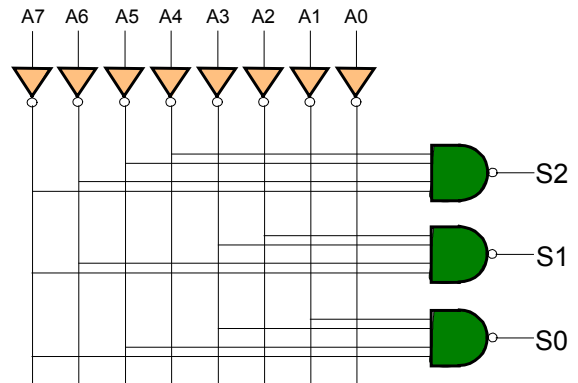


IN	S ₂	S ₁	S ₀
A ₀	0	0	0
A ₁	0	0	1
A ₂	0	1	0
A ₃	0	1	1
A ₄	1	0	0
A ₅	1	0	1
A ₆	1	1	0
A ₇	1	1	1

$$S_2 = A_4 + A_5 + A_6 + A_7 = \overline{\overline{A_4} \overline{A_5} \overline{A_6} \overline{A_7}}$$

$$S_1 = A_2 + A_3 + A_6 + A_7 = \overline{\overline{A_2} \overline{A_3} \overline{A_6} \overline{A_7}}$$

$$S_0 = A_1 + A_3 + A_5 + A_7 = \overline{\overline{A_1} \overline{A_3} \overline{A_5} \overline{A_7}}$$



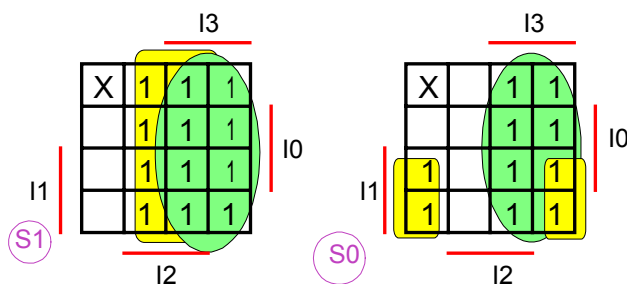
4 LINE TO 2 LINE PRIORITY ENCODER

- Cuando hay mas de una entrada activa, la salida corresponde a la entrada con el mayor valor decimal.

I ₃	I ₂	I ₁	I ₀	S ₁	S ₀
0	0	0	1	0	0
0	0	1	X	0	1
0	1	X	X	1	0
1	X	X	X	1	1

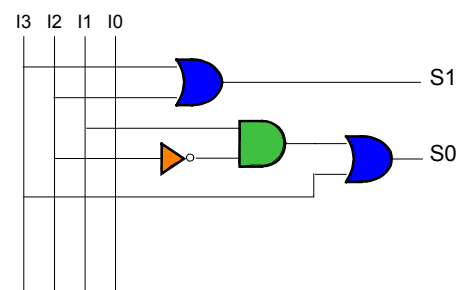
$$S_1 = \overline{I_3} I_2 + I_3 = (\overline{I_3} + I_3)(I_2 + I_3) = I_2 + I_3$$

$$S_0 = \overline{I_3} \overline{I_2} I_1 + I_3 = \dots = \overline{I_2} I_1 + I_3$$



$$S_1 = I_2 + I_3$$

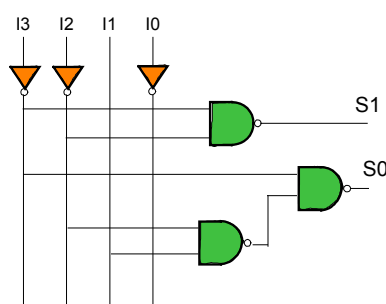
$$S_0 = \overline{I_2} I_1 + I_3$$



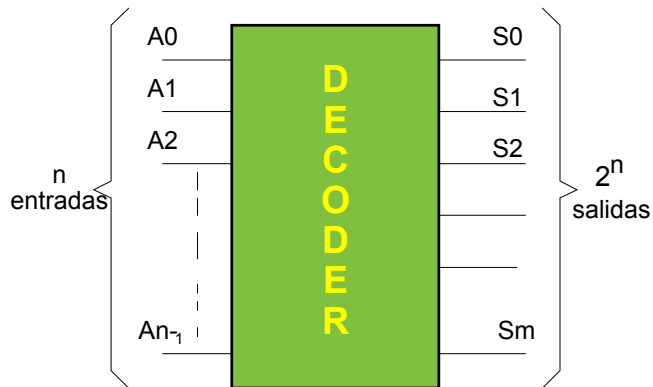
Con puertas NAND:

$$S_1 = \overline{\overline{I_2} \overline{I_3}}$$

$$S_0 = \overline{\overline{\overline{I_2} I_1} \overline{I_3}}$$



3.2-DECODIFICADOR (DECODER)



- Para cada código de entrada, solamente una de las salidas es activa.
- La salida que está activa, es la que corresponde al equivalente decimal del código binario presente en la entrada.

2 LINE TO 4 LINE DECODER

Nivel activo de salida: H

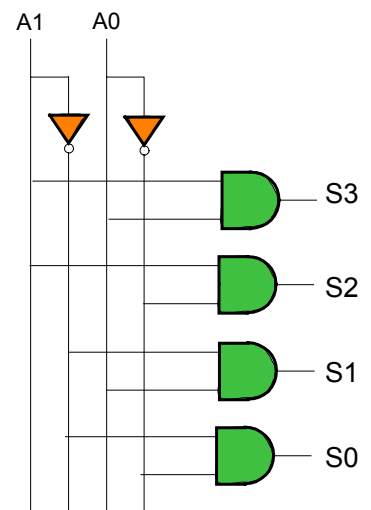
A_1	A_0	S_3	S_2	S_1	S_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

$$S_0 = \bar{A}_1 \bar{A}_0$$

$$S_1 = \bar{A}_1 A_0$$

$$S_2 = A_1 \bar{A}_0$$

$$S_3 = A_1 A_0$$



Nivel activo de salida: L

A_1	A_0	S_3	S_2	S_1	S_0
0	0	1	1	1	0
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	1	1	1

$$S_0 = A_1 + A_0 = \overline{\bar{A}_1 \bar{A}_0}$$

$$S_1 = A_1 + \bar{A}_0 = \overline{\bar{A}_1 A_0}$$

$$S_2 = \bar{A}_1 + A_0 = \overline{A_1 \bar{A}_0}$$

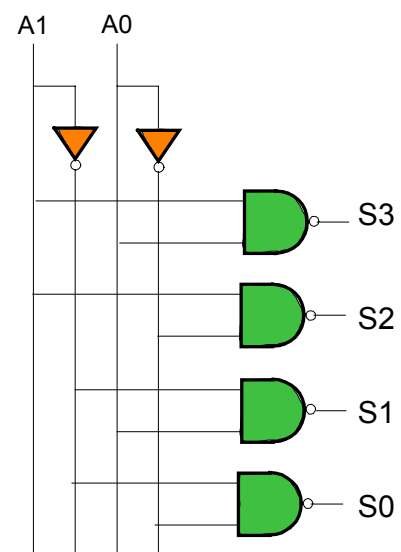
$$S_3 = \bar{A}_1 + \bar{A}_0 = \overline{A_1 A_0}$$

Los términos también se pueden obtener como sigue:

$$\bar{S}_0 = \bar{A}_1 \bar{A}_0 \implies S_0 = \overline{\bar{A}_1 \bar{A}_0}$$

$$\bar{S}_1 = \bar{A}_1 A_0 \implies S_1 = \overline{\bar{A}_1 A_0}$$

...etc.



3 LINE TO 8 LINE DECODER (BINARIO-OCTAL)

A ₂	A ₁	A ₀	S ₇	S ₆	S ₅	S ₄	S ₃	S ₂	S ₁	S ₀
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

$$S_0 = \bar{A}_2 \bar{A}_1 \bar{A}_0$$

$$S_1 = \bar{A}_2 \bar{A}_1 A_0$$

$$S_2 = \bar{A}_2 A_1 \bar{A}_0$$

$$S_3 = \bar{A}_2 A_1 A_0$$

$$S_4 = A_2 \bar{A}_1 \bar{A}_0$$

$$S_5 = A_2 \bar{A}_1 A_0$$

$$S_6 = A_2 A_1 \bar{A}_0$$

$$S_7 = A_2 A_1 A_0$$

Si tomamos como nivel activo el nivel bajo tendremos:

$$S_0 = A_2 + A_1 + A_0 = \overline{\bar{A}_2 \bar{A}_1 \bar{A}_0}$$

$$S_1 = A_2 + A_1 + \bar{A}_0 = \overline{\bar{A}_2 \bar{A}_1 A_0}$$

.....
.....

$$S_7 = \bar{A}_2 + \bar{A}_1 + \bar{A}_0 = \overline{A_2 A_1 A_0}$$

o

$$\bar{S}_0 = \bar{A}_2 \bar{A}_1 \bar{A}_0$$



$$S_0 = \overline{\bar{A}_2 \bar{A}_1 \bar{A}_0}$$

o

$$\bar{S}_1 = \bar{A}_2 \bar{A}_1 A_0$$



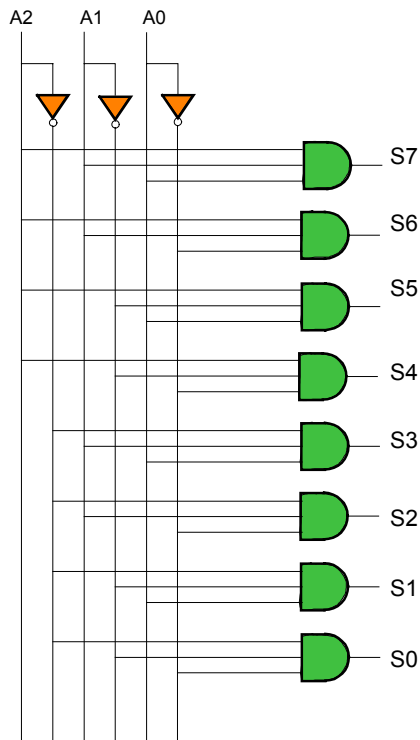
$$S_1 = \overline{\bar{A}_2 \bar{A}_1 A_0}$$

o

$$\bar{S}_7 = A_2 A_1 A_0$$



$$S_7 = \overline{A_2 A_1 A_0}$$



De la misma forma realizaríamos el análisis de un decodificador binario-hexadecimal (4 LINE TO 16 LINE DECODER)

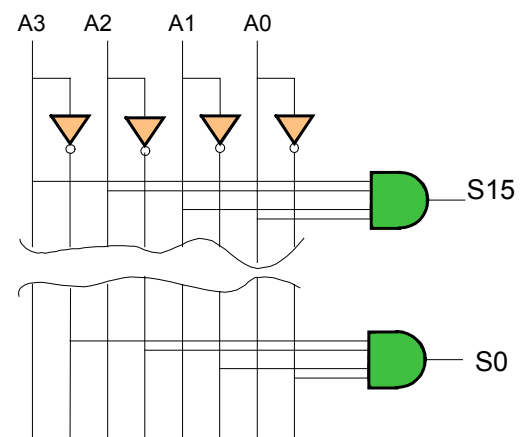
$$S_0 = \bar{A}_3 \bar{A}_2 \bar{A}_1 \bar{A}_0$$

$$S_1 = \bar{A}_3 \bar{A}_2 \bar{A}_1 A_0$$

$$S_2 = \bar{A}_3 \bar{A}_2 A_1 \bar{A}_0$$

.....

$$S_{15} = A_3 A_2 A_1 A_0$$

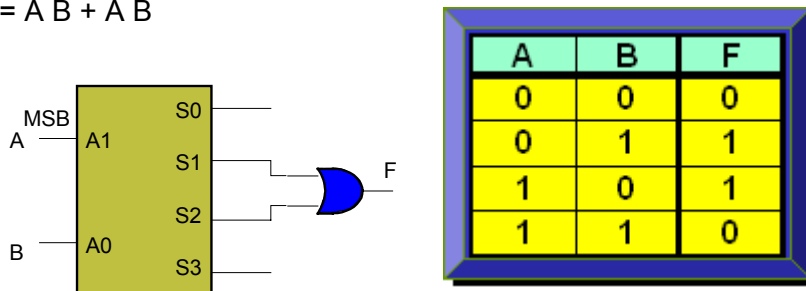


3.2.1-DECODIFICADOR PARA IMPLEMENTAR UNA FUNCIÓN LÓGICA

Decoder con salidas activas a nivel alto.

Implementemos la función $F = A \text{ X-OR } B$.

$$F = \bar{A} B + A \bar{B}$$



originando un estado alto en la salida de la puerta OR.

Se puede observar que la función F tendrá un valor alto cuando A valga 0 y B valga 1. Observando el circuito basado en decodificador, cuando se cumpla esta condición, se activará la salida $S1$,

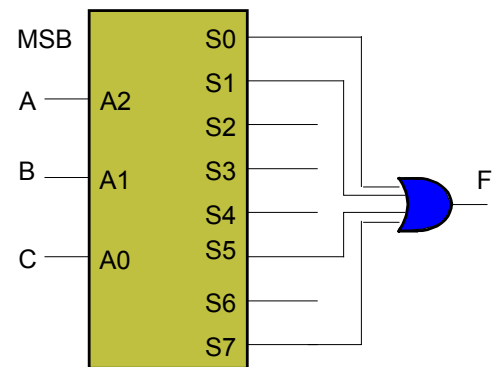
También F tendrá un estado alto cuando A valga 1 y B valga 0. En este caso la salida que se activa en el decodificador será la $S2$, dando también un estado alto a la salida de la puerta OR. En las demás combinaciones de las variables A y B , la salida debe ser 0, condición que se cumple con el circuito dado.

Otra función: $F = A B C + A \bar{B} C + \bar{A} \bar{B} \bar{C}$

Ponemos la función en su forma canónica:

$$F = A B C + A \bar{B} C + \bar{A} \bar{B} \bar{C} + \bar{A} \bar{B} C.$$

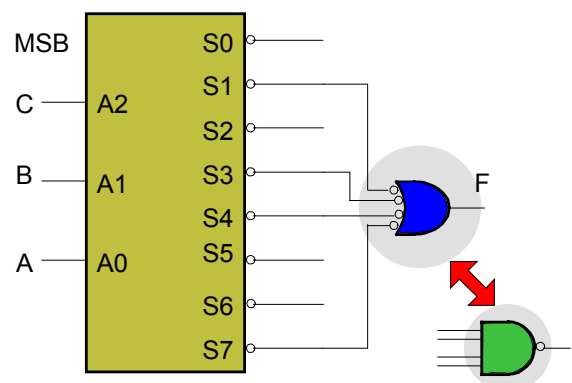
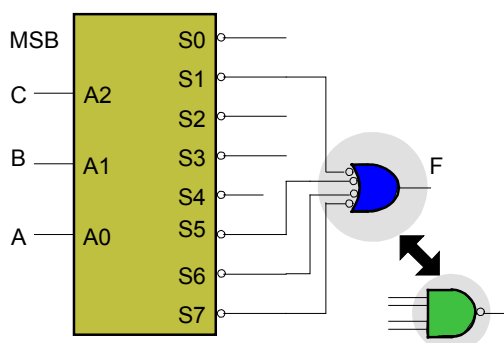
Estas combinaciones representan los valores decimales 7, 5, 0, 1 respectivamente, tomando como MSB la variable A . Aplicando A , a la entrada de más peso del decodificador y C a la de menos peso, las salidas que se activarán, serán la 0, 1, 7 y 5.



Decoder con salidas activas a nivel bajo.

El razonamiento es el mismo que para el caso anterior.

$$F = \underbrace{\bar{C} \bar{B} A}_{1} + \underbrace{\bar{C} B A}_{3} + \underbrace{C \bar{B} \bar{A}}_{4} + \underbrace{C B A}_{7}$$



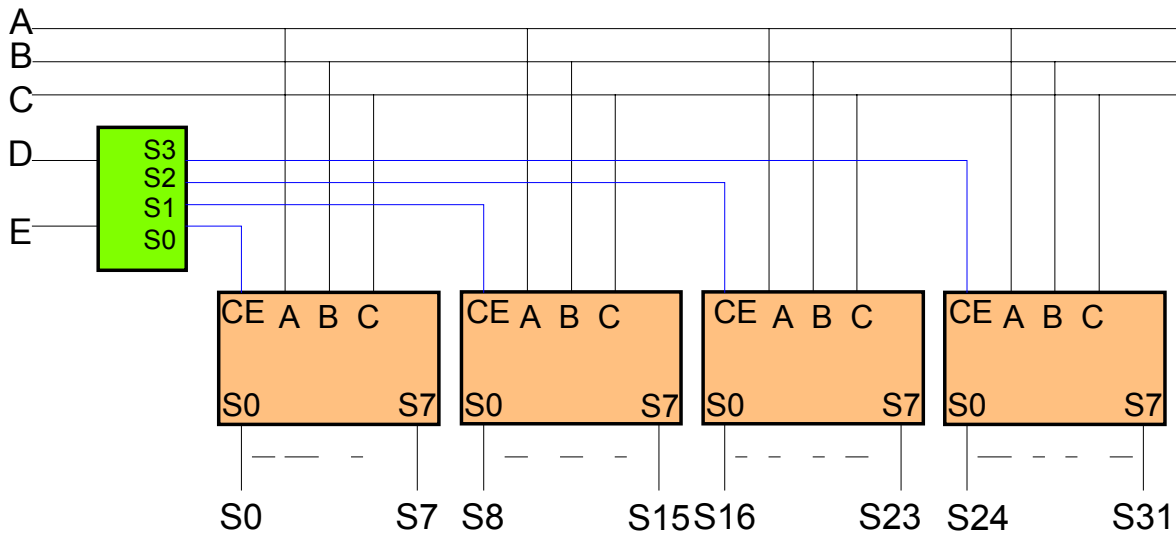
$$F = C B A + C B \bar{A} + \bar{C} B A;$$

$$F = \underbrace{C B A}_{7} + \underbrace{C B \bar{A}}_{7} + \underbrace{\bar{C} B A}_{6} + \underbrace{\bar{C} B \bar{A}}_{5} + \underbrace{\bar{C} \bar{B} A}_{1}$$

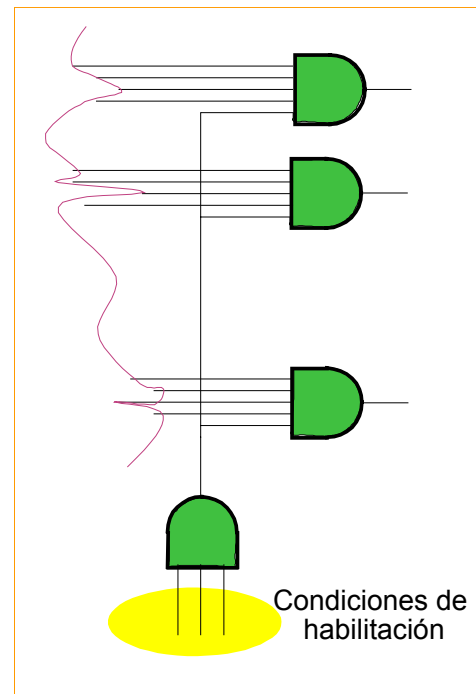
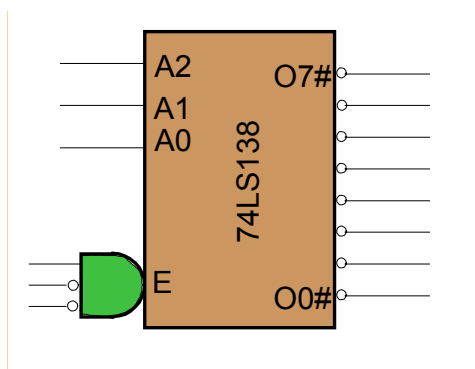
3.2.2-AMPLIACIÓN DE DECODIFICADORES.

- Se puede diseñar decodificadores de grandes capacidades (gran número de entradas-salidas) a partir de decodificadores de capacidades menores. Para ello se hace uso de un terminal adicional que poseen los decodificadores denominado CE (*Chip Enable*), o CS (*Chip Select*), o simplemente E (*Enable*) cuya función es, básicamente, hacer operativo (cuando este terminal está activo) o inoperativo, el decodificador.

Ej. Construcción de un decodificador de 5 a 32 líneas a partir de decodificadores de 3 a 8 líneas y 2 a 4 líneas.



- La entrada de habilitación del circuito no se limita en general a que se active o no una entrada. En muchos casos el que se active o no el circuito está condicionado por más de una entrada, las cuales deben cumplir ciertas condiciones para habilitar el circuito. Por otro lado, estos terminales, pueden actuar sobre las entradas al circuito, sobre las salidas, sobre la alimentación del circuito, etc.

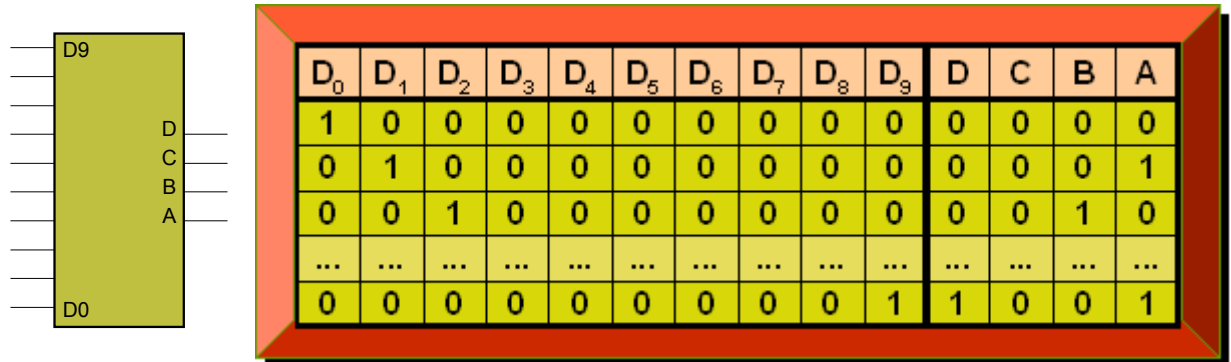


El circuito será operativo cuando se cumpla que las tres entradas de habilitación (E) sean activas, es decir, las dos inferiores estén a nivel bajo y la superior a nivel alto.

3.3-CONVERTIDORES DE CÓDIGO. (TRANSCODIFICADORES)

- Realizan la conversión de un código determinado en otro código diferente.

TRANSCODIFICADOR DECIMAL-BCD-N (10-LINE TO 4-LINE BCD ENCODER)



Teniendo en cuenta que sólo puede ser activa una entrada en un momento dado, podemos representar la tabla como sigue:

IN	D	C	B	A
D ₀	0	0	0	0
D ₁	0	0	0	1
D ₂	0	0	1	0
D ₃	0	0	1	1
D ₄	0	1	0	0
D ₅	0	1	0	1
D ₆	0	1	1	0
D ₇	0	1	1	1
D ₈	1	0	0	0
D ₉	1	0	0	1

En este caso la mejor forma de obtener las funciones de la tabla de la verdad es observando directamente que condiciones deben cumplir las entradas para que se activen las correspondientes salidas.

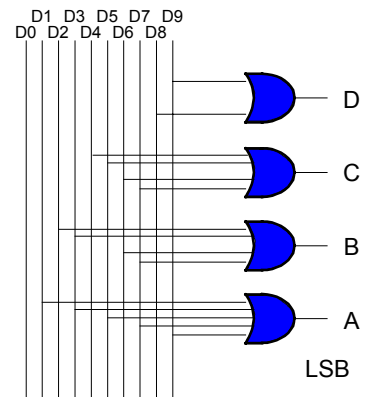
D será activa cuando sea D₈ o D₉....

$$D = D_8 + D_9$$

$$C = D_4 + D_5 + D_6 + D_7$$

$$B = D_2 + D_3 + D_6 + D_7$$

$$A = D_1 + D_3 + D_5 + D_7 + D_9$$



Hay que observar que si se activa más de una entrada al mismo tiempo, el código no sería el buscado.

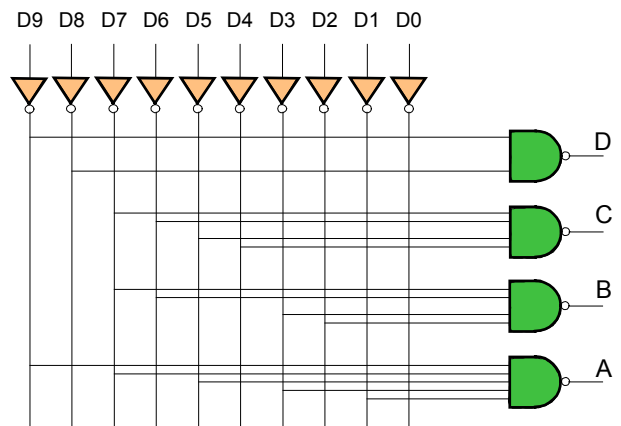
El circuito implementado con puertas NAND será

$$D = D_8 + D_9 = \overline{\overline{D_8} \overline{D_9}}$$

$$C = D_4 + D_5 + D_6 + D_7 = \overline{\overline{D_4} \overline{D_5} \overline{D_6} \overline{D_7}}$$

$$B = D_2 + D_3 + D_6 + D_7 = \overline{\overline{D_2} \overline{D_3} \overline{D_6} \overline{D_7}}$$

$$A = D_1 + D_3 + D_5 + D_7 + D_9 = \overline{\overline{D_1} \overline{D_3} \overline{D_5} \overline{D_7} \overline{D_9}}$$



TRANSCODIFICADOR BCD(N)-DECIMAL; BCD TO DECIMAL DECODER

D	C	B	A	OUT
0	0	0	0	S ₀
0	0	0	1	S ₁
0	0	1	0	S ₂
0	0	1	1	S ₃
0	1	0	0	S ₄
0	1	0	1	S ₅
0	1	1	0	S ₆
0	1	1	1	S ₇
1	0	0	0	S ₈
1	0	0	1	S ₉

$$S_0 = \bar{D} \bar{C} \bar{B} \bar{A}$$

$$S_1 = \bar{D} \bar{C} \bar{B} A$$

$$S_2 = \bar{D} \bar{C} B \bar{A}$$

$$S_3 = \bar{D} \bar{C} B A$$

$$S_4 = \bar{D} C \bar{B} \bar{A}$$

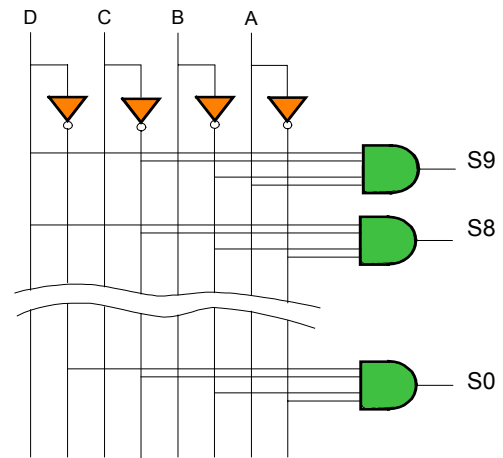
$$S_5 = \bar{D} C \bar{B} A$$

$$S_6 = \bar{D} C B \bar{A}$$

$$S_7 = \bar{D} C B A$$

$$S_8 = D \bar{C} \bar{B} \bar{A}$$

$$S_9 = D \bar{C} \bar{B} A$$



Según el diseño anterior, si se da una combinación en la entrada, superior a una combinación BCD, todas las salidas permanecen en 0. Podemos reducir el circuito, haciendo uso de las condiciones no importa (las que no se pueden dar en la entrada del decodificador) y que son aquellas que no son BCD, es decir: 1010(10), 1011 (11), 1100 (12), 1101 (13), 1110 (14), 1111 (15) que corresponden a los términos:

$$D \bar{C} B \bar{A}, D \bar{C} B A, D C \bar{B} \bar{A}, D C \bar{B} A, D C B \bar{A}, D C B A.$$

D	C	B	A	OUT
0	0	0	0	S ₀
0	0	0	1	S ₁
0	0	1	0	S ₂
0	0	1	1	S ₃
0	1	0	0	S ₄
0	1	0	1	S ₅
0	1	1	0	S ₆
0	1	1	1	S ₇
1	0	0	0	S ₈
1	0	0	1	S ₉
1	0	1	0	X
1	0	1	1	X
1	1	0	0	X
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

$$S_0 = \Sigma_4 (0)$$

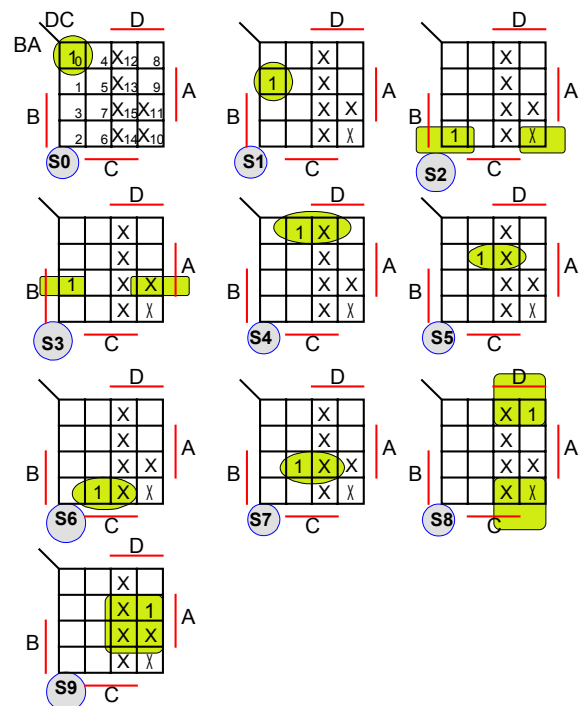
$$S_1 = \Sigma_4 (1)$$

$$S_2 = \Sigma_4 (2)$$

$$S_3 = \Sigma_4 (3)$$

.....

$$S_9 = \Sigma_4 (9)$$



$$S_0 = \bar{D} \bar{C} \bar{B} \bar{A}; S_1 = \bar{D} \bar{C} \bar{B} A; S_2 = \bar{C} B \bar{A}; S_3 = \bar{C} B A;$$

$$S_4 = C \bar{B} \bar{A}; S_5 = C \bar{B} A; S_6 = C B \bar{A}; S_7 = C B A; S_8 = D \bar{A}; S_9 = D A$$

TRANSCODIFICADOR BINARIO-GRAY

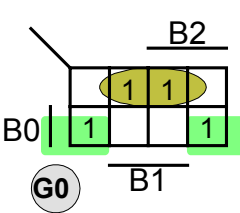
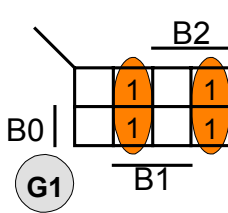
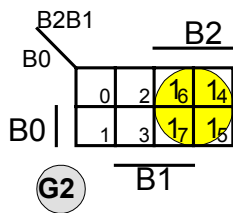
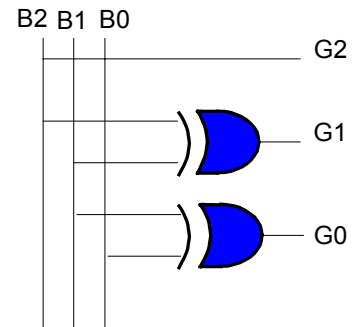
Tres variables:

B ₂	B ₁	B ₀	G ₂	G ₁	G ₀
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	1
1	0	1	1	1	0
1	1	0	1	0	0
1	1	1	1	0	1

$$G_2 = \Sigma_3 (4, 5, 6, 7)$$

$$G_1 = \Sigma_3 (2, 3, 4, 5)$$

$$G_0 = \Sigma_3 (1, 2, 5, 6)$$



$$G_2 = B_2$$

$$G_1 = B_2 \bar{B}_1 + \bar{B}_2 B_1 = B_2 \text{ X-OR } B_1$$

$$G_0 = \bar{B}_0 B_1 + B_0 \bar{B}_1 = B_0 \text{ X-OR } B_1$$

Cuatro variables.

B ₃	B ₂	B ₁	B ₀	G ₃	G ₂	G ₁	G ₀
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

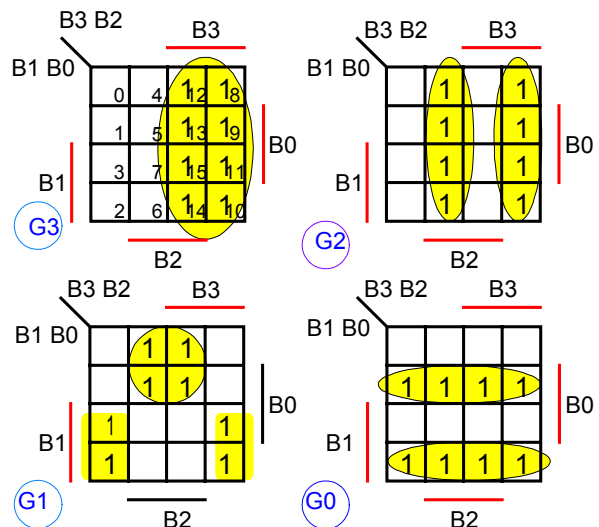
Tomando los minterms:

$$G_3 = \Sigma_4 (8, 9, 10, 11, 12, 13, 14, 15)$$

$$G_2 = \Sigma_4 (4, 5, 6, 7, 8, 9, 10, 11)$$

$$G_1 = \Sigma_4 (2, 3, 4, 5, 10, 11, 12, 13)$$

$$G_0 = \Sigma_4 (1, 2, 5, 6, 9, 10, 13, 14)$$

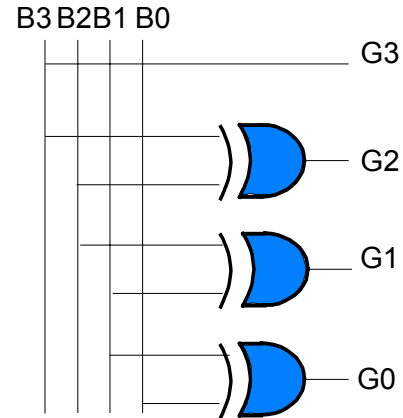


$$G_3 = B_3$$

$$G_2 = B_3 \bar{B}_2 + \bar{B}_3 B_2 = B_3 \text{ X-OR } B_2$$

$$G_1 = B_2 \bar{B}_1 + \bar{B}_2 B_1 = B_2 \text{ X-OR } B_1$$

$$G_0 = \bar{B}_0 B_1 + B_0 \bar{B}_1 = B_0 \text{ X-OR } B_1$$



Tenemos:

Para tres variables

$$G_2 = B_2; G_1 = B_2 \text{ X-OR } B_1; G_0 = B_0 \text{ X-OR } B_1$$

Para cuatro variables

$$G_3 = B_3; G_2 = B_3 \text{ X-OR } B_2; G_1 = B_2 \text{ X-OR } B_1; G_0 = B_1 \text{ X-OR } B_0$$

Entonces:

Para n variables

$$G_n = B_n; G_{n-1} = B_n \text{ X-OR } B_{n-1}; G_{n-2} = B_{n-1} \text{ X-OR } B_{n-2}; \dots G_0 = B_1 \text{ X-OR } B_0$$

Si tomamos los maxterms:

$$G_3 = \Sigma_4 (0, 1, 2, 3, 4, 5, 6, 7)$$

$$G_2 = \Sigma_4 (0, 1, 2, 3, 12, 13, 14, 15)$$

$$G_1 = \Sigma_4 (0, 1, 6, 7, 8, 9, 14, 15)$$

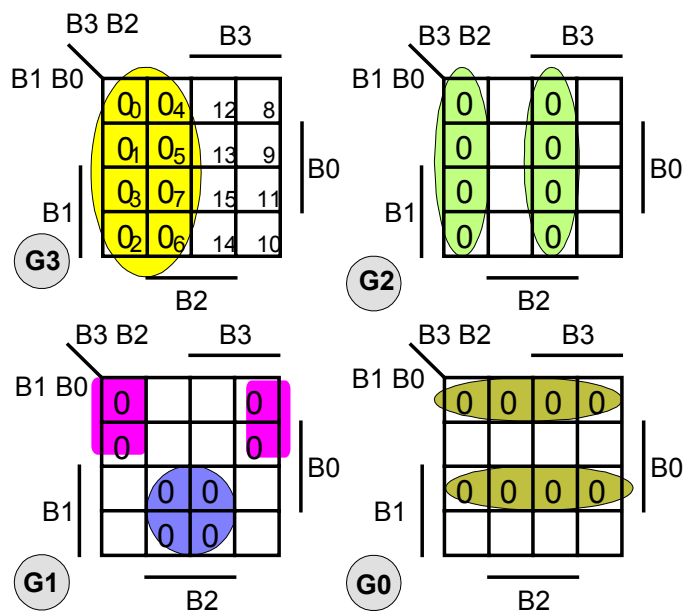
$$G_0 = \Sigma_4 (0, 3, 4, 7, 8, 11, 12, 15)$$

$$G_3 = B_3$$

$$G_2 = (\bar{B}_3 + \bar{B}_2)(B_3 + B_2)$$

$$G_1 = (\bar{B}_2 + \bar{B}_1)(B_2 + B_1)$$

$$G_0 = (\bar{B}_0 + \bar{B}_1)(B_0 + B_1)$$



TRANSCODIFICADOR GRAY-BINARIO

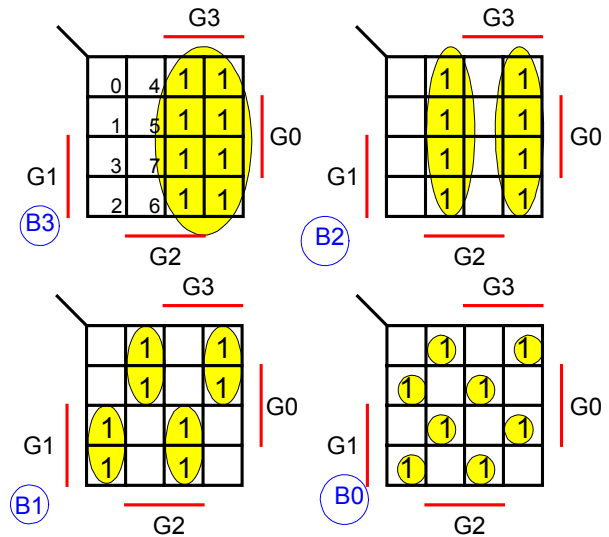
G ₃	G ₂	G ₁	G ₀	B ₃	B ₂	B ₁	B ₀
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	1	0	0	1	0
0	0	1	0	0	0	1	1
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	1
0	1	0	1	0	1	1	0
0	1	0	0	0	1	1	1
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	1
1	1	1	1	1	0	1	0
1	1	1	0	1	0	1	1
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	0	0	1	1	1	1	0
1	0	0	0	1	1	1	1

$$B_3 = \Sigma_4 (12, 13, 15, 14, 10, 11, 9, 8)$$

$$B_2 = \Sigma_4 (6, 7, 5, 4, 10, 11, 9, 8)$$

$$B_1 = \Sigma_4 (3, 2, 5, 4, 15, 14, 9, 8)$$

$$B_0 = \Sigma_4 (1, 2, 7, 4, 13, 14, 11, 8)$$

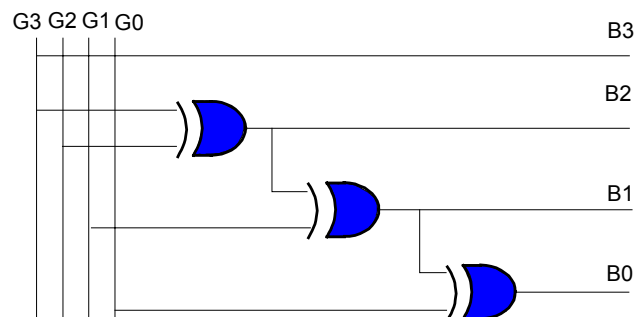


$$B_3 = G_3$$

$$B_2 = G_3 \bar{G}_2 + \bar{G}_3 G_2 = G_3 \text{ X-OR } G_2$$

$$B_1 = \bar{G}_3 \bar{G}_2 G_1 + \bar{G}_3 G_2 \bar{G}_1 + G_3 G_2 G_1 + G_3 \bar{G}_2 \bar{G}_1 = (\bar{G}_3 \bar{G}_2 + G_3 G_2)^1 G_1 + (\bar{G}_3 \bar{G}_2 + G_3 G_2) \bar{G}_1 = \overline{(G_3 \text{ X-OR } G_2)} G_1 + (G_3 \text{ X-OR } G_2) \bar{G}_1 = G_3 \text{ X-OR } G_2 \text{ X-OR } G_1$$

$$B_0 = \bar{G}_3 \bar{G}_2 \bar{G}_1 G_0 + \bar{G}_3 \bar{G}_2 G_1 \bar{G}_0 + \bar{G}_3 G_2 G_1 G_0 + \bar{G}_3 G_2 \bar{G}_1 \bar{G}_0 = G_3 \text{ X-OR } G_2 \text{ X-OR } G_1 \text{ X-OR } G_0$$



$$\begin{aligned} \bar{G}_3 \bar{G}_2 + G_3 G_2 &= \overline{\overline{\bar{G}_3 \bar{G}_2 + G_3 G_2}} = \overline{\bar{G}_3 \bar{G}_2 G_3 G_2} = \overline{(G_3 + G_2) (\bar{G}_3 + \bar{G}_2)} = \overline{G_3 \bar{G}_3 + G_3 \bar{G}_2 + \bar{G}_3 G_2 + G_2 \bar{G}_2} = \\ &= \overline{G_3 \bar{G}_2 + \bar{G}_3 G_2} = \overline{G_3 \text{ X-OR } G_2} = G_3 \text{ X-NOR } G_2 \end{aligned}$$

Para 8 bits tenemos:

$$B_7 = G_7$$

$$B_6 = G_7 \text{ X-OR } G_6$$

$$B_5 = G_7 \text{ X-OR } G_6 \text{ X-OR } G_5$$

$$B_4 = G_7 \text{ X-OR } G_6 \text{ X-OR } G_5 \text{ X-OR } G_4$$

$$B_3 = G_7 \text{ X-OR } G_6 \text{ X-OR } G_5 \text{ X-OR } G_4 \text{ X-OR } G_3$$

$$B_2 = G_7 \text{ X-OR } G_6 \text{ X-OR } G_5 \text{ X-OR } G_4 \text{ X-OR } G_3 \text{ X-OR } G_2$$

$$B_1 = G_7 \text{ X-OR } G_6 \text{ X-OR } G_5 \text{ X-OR } G_4 \text{ X-OR } G_3 \text{ X-OR } G_2 \text{ X-OR } G_1$$

$$B_0 = G_7 \text{ X-OR } G_6 \text{ X-OR } G_5 \text{ X-OR } G_4 \text{ X-OR } G_3 \text{ X-OR } G_2 \text{ X-OR } G_1 \text{ X-OR } G_0$$

Para n bits

$$B_n = G_n$$

$$B_{n-1} = G_n \text{ X-OR } G_{n-1}$$

.....

$$B_0 = G_n \text{ X-OR } G_{n-1} \text{ X-OR } G_{n-2} \cdots G_2 \text{ X-OR } G_1 \text{ X-OR } G_0$$

TRANSCODIFICADOR BCD(N)-BCD X-3

D	C	B	A	D ₁	C ₁	B ₁	A ₁
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0
X	X	X	X				

$$D_1 = D + C A + C B$$

$$C_1 = A \bar{C} + B \bar{C} + \bar{A} \bar{B} C$$

$$B_1 = \bar{A} \bar{B} + AB = A \text{ X-NOR } B$$

$$A_1 = \bar{A}$$

Implementado con puertas NAND:

$$D_1 = \overline{\overline{D} \overline{C} A \overline{C} B}$$

$$C_1 = \overline{\overline{A} \overline{C} \overline{B} C \overline{A} \bar{B} C}$$

$$B_1 = \overline{\overline{A} \bar{B} \overline{AB}}$$

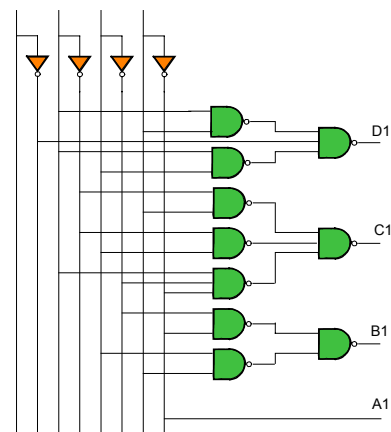
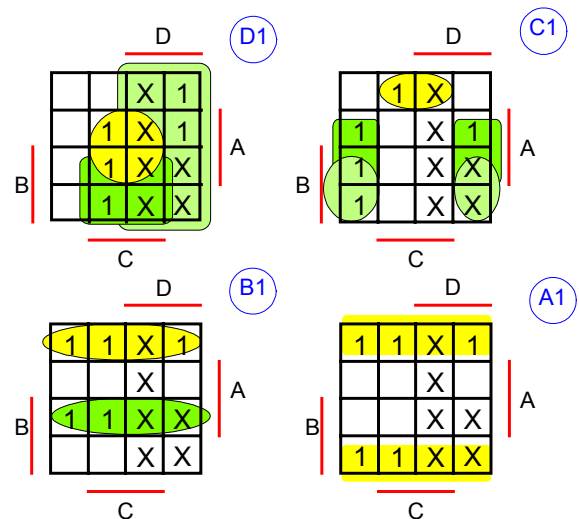
$$A_1 = \bar{A}$$

$$D_1 = \Sigma_4 (5, 6, 7, 8, 9) + d(10, 11, 12, 13, 14, 15)$$

$$C_1 = \Sigma_4 (1, 2, 3, 4, 9) + d(10, 11, 12, 13, 14, 15)$$

$$B_1 = \Sigma_4 (0, 3, 4, 7, 8) + d(10, 11, 12, 13, 14, 15)$$

$$A_1 = \Sigma_4 (0, 2, 4, 6, 8) + d(10, 11, 12, 13, 14, 15)$$



TRANSCODIFICADOR BCD X-3-BCD(N)

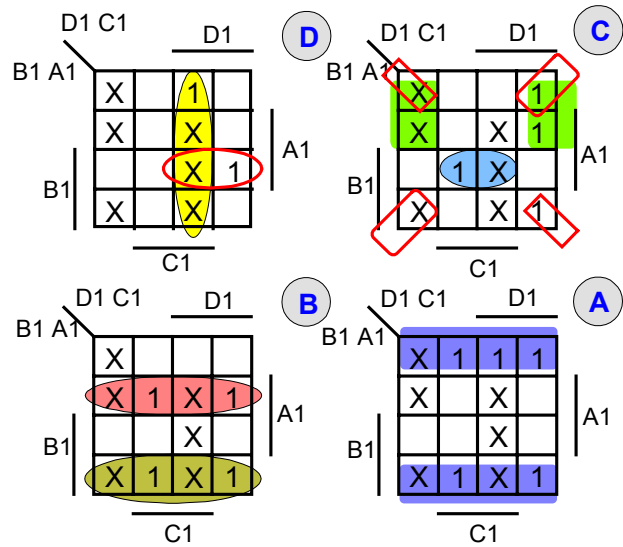
D	C	B	A	D ₁	C ₁	B ₁	A ₁
0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	0	0	1	1
0	1	1	1	0	1	0	0
1	0	0	0	0	1	0	1
1	0	0	1	0	1	1	0
1	0	1	0	0	1	1	1
1	0	1	1	1	0	0	0
1	1	0	0	1	0	0	1
X	X	X	X				

$$D = \Sigma_4 (11, 12) + d(0, 1, 2, 13, 14, 15)$$

$$C = \Sigma_4 (7, 8, 9, 10) + d(0, 1, 2, 13, 14, 15)$$

$$B = \Sigma_4 (5, 6, 9, 10) + d(0, 1, 2, 13, 14, 15)$$

$$A = \Sigma_4 (4, 6, 8, 10, 12) + d(0, 1, 2, 13, 14, 15)$$

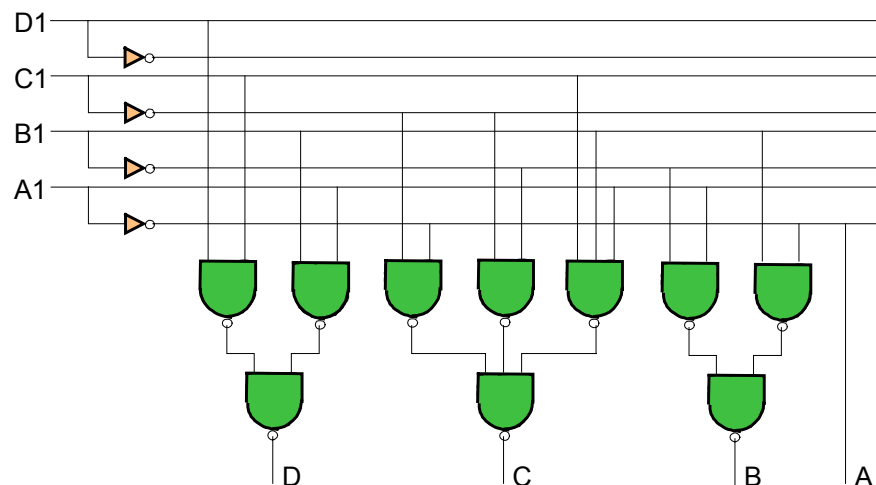


$$A = \bar{A}_1$$

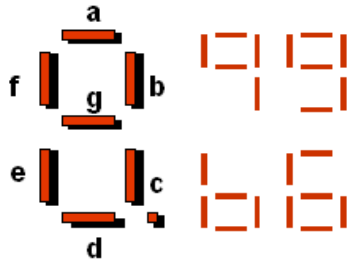
$$B = A_1 \bar{B}_1 + \bar{A}_1 B_1 = \bar{A}_1 \bar{B}_1 + \bar{A}_1 B_1$$

$$C = \bar{A}_1 \bar{C}_1 + \bar{C}_1 \bar{B}_1 + A_1 C_1 B_1 = \bar{A}_1 \bar{C}_1 + \bar{C}_1 \bar{B}_1 + A_1 C_1 B_1$$

$$D = D_1 C_1 + A_1 B_1 = \bar{D}_1 \bar{C}_1 + \bar{A}_1 \bar{B}_1$$

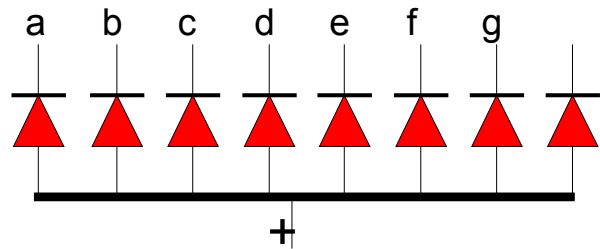
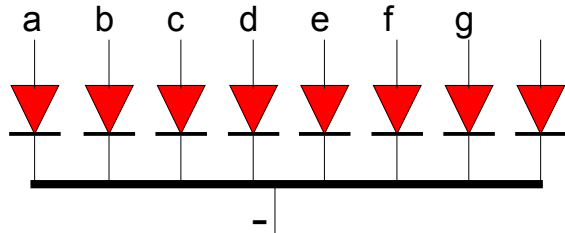


BCD TO SEVEN SEGMENTS DECODERS



□ Los dígitos se forman por la iluminación o no de siete segmentos LED. Todos los segmentos tienen un punto común que puede ser el ánodo, o el cátodo.

□ Los de ánodo común para que se iluminen, necesitan que los extremos libres, cátodos en este caso, sean excitados con un nivel bajo (decodificador con las siete salidas activas a nivel bajo).

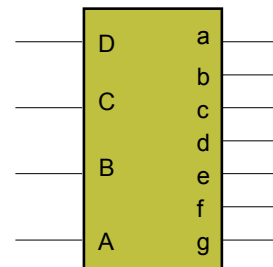


□ Los de cátodo común (ánodos con extremos libres), necesitan ser excitados con un decodificador cuyas siete salidas sean activas a nivel alto.

□ La configuración de los números puede ser diferente de unos displays a otros. Observar por ejemplo el número 6 y el 9 en la figura superior.

D	C	B	A	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1
X	X	X	X							

Diseño de un decodificador donde se considera que solo se pueden dar entradas BCD-N.



$$a = \sum_4 (0, 2, 3, 5, 6, 7, 8, 9) + d(10, 11, 12, 13, 14, 15)$$

$$b = \sum_4 (0, 1, 2, 3, 4, 7, 8, 9) + d(10, 11, 12, 13, 14, 15)$$

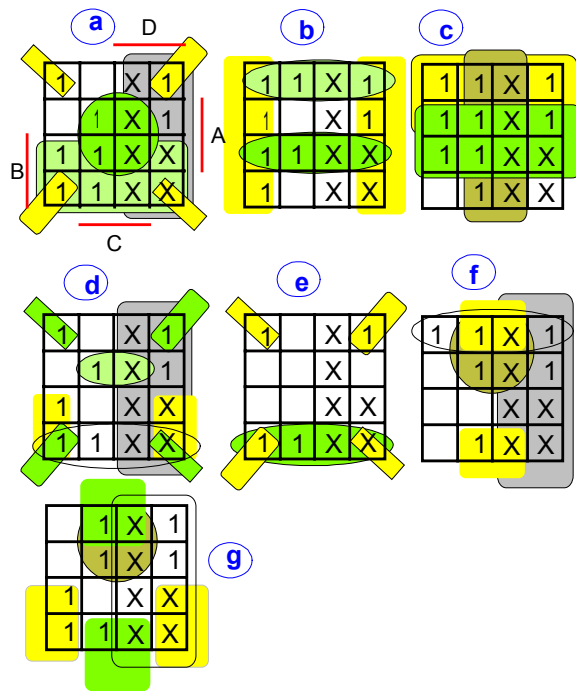
$$c = \sum_4 (0, 1, 3, 4, 5, 6, 7, 8, 9) + d(10, 11, 12, 13, 14, 15)$$

$$d = \sum_4 (0, 2, 3, 5, 6, 8, 9) + d(10, 11, 12, 13, 14, 15)$$

$$e = \sum_4 (0, 2, 6, 8) + d(10, 11, 12, 13, 14, 15)$$

$$f = \sum_4 (0, 4, 5, 6, 8, 9) + d(10, 11, 12, 13, 14, 15)$$

$$g = \sum_4 (2, 3, 4, 5, 6, 8, 9) + d(10, 11, 12, 13, 14, 15)$$



$$a = D + C A + B + \overline{C} \overline{A} = \overline{D} \overline{C} \overline{A} \overline{B} \overline{C} \overline{A}$$

$$b = \overline{C} + \overline{B} \overline{A} + B A = C \overline{B} \overline{A} \overline{B} \overline{A}$$

$$c = \overline{B} + A + C = \overline{B} \overline{A} \overline{C}$$

$$d = \overline{C} \overline{A} + D + \overline{C} B + B \overline{A} + C \overline{B} A$$

$$= \overline{C} \overline{A} \overline{D} \overline{C} B \overline{B} \overline{A} C \overline{B} A$$

$$e = \overline{C} \overline{A} + B \overline{A} = \overline{C} \overline{A} \overline{B} \overline{A}$$

$$f = D + \overline{B} \overline{A} + C \overline{B} + C \overline{A} = \overline{D} \overline{B} \overline{A} \overline{C} \overline{B} \overline{C} \overline{A}$$

$$g = D + C \overline{B} + C \overline{A} + \overline{C} B = \overline{D} + \overline{C} \overline{B} + \overline{C} \overline{A} + \overline{C} B$$

Si en vez de tomar los minterms, tomamos los maxterms:

D	C	B	A	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1
X	X	X	X							

$$a = \prod_4 (1, 4) + d(10, 11, 12, 13, 14, 15)$$

$$b = \prod_4 (5, 6) + d(10, 11, 12, 13, 14, 15)$$

$$c = \prod_4 (2) + d(10, 11, 12, 13, 14, 15)$$

$$d = \prod_4 (1, 4, 7) + d(10, 11, 12, 13, 14, 15)$$

$$e = \prod_4 (1, 3, 4, 5, 7, 9) + d(10, 11, 12, 13, 14, 15)$$

$$f = \prod_4 (1, 2, 3, 7) + d(10, 11, 12, 13, 14, 15)$$

$$g = \prod_4 (0, 1, 7) + d(10, 11, 12, 13, 14, 15)$$

$$a = (D + C + B + A)(\overline{C} + \overline{B} + \overline{A}) =$$

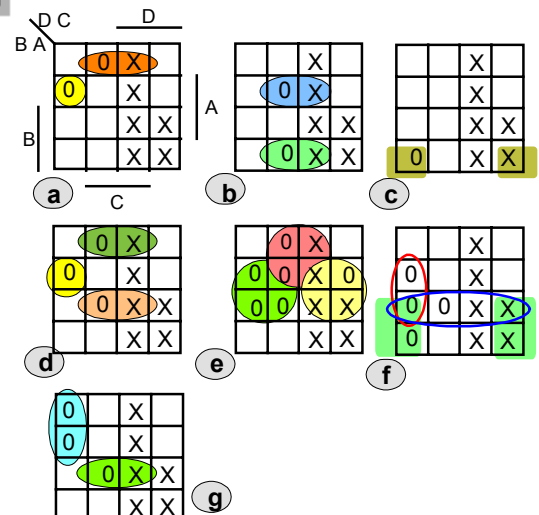
$$= \overline{D} \overline{C} \overline{B} \overline{A} \overline{C} \overline{B} \overline{A}$$

$$b = (\overline{C} + \overline{B} + A)(\overline{C} + B + \overline{A}) = \overline{C} \overline{B} \overline{A} \overline{C} \overline{B} \overline{A}$$

$$c = C + \overline{B} + A = \overline{C} \overline{B} \overline{A}$$

$$d = (\overline{C} + \overline{B} + \overline{A})(\overline{C} + B + A)(D + C + B + \overline{A}) =$$

$$= \overline{C} \overline{B} \overline{A} \overline{C} \overline{B} \overline{A} \overline{D} \overline{C} \overline{B} \overline{A}$$

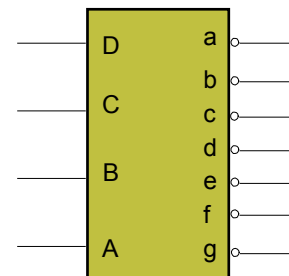


$$e = (\bar{C} + B)(\bar{A}) = \bar{C} \bar{B} \bar{A}$$

$$f = (D + C + \bar{A})(C + \bar{B})(\bar{B} + \bar{A}) = \bar{D} \bar{C} \bar{A} \bar{C} \bar{B} \bar{B} \bar{A}$$

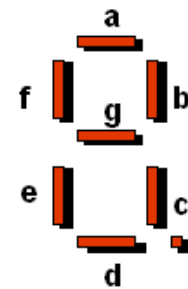
$$g = (D + C + B)(\bar{C} + \bar{B} + \bar{A}) = \bar{D} \bar{C} \bar{B} \bar{C} \bar{B} \bar{A}$$

- Los decodificadores vistos, sirven para excitar los displays de 7 segmentos del tipo cátodo común. En estos, cada ánodo debe tener un nivel alto para excitar el segmento (LED) correspondiente.
- Para excitar displays de ánodo común, el decodificador debe tener sus salidas activas a nivel bajo.



Ej. Diseñar un decodificador BCD-7 segmentos que indique, mediante una “E”, que el valor aplicado al mismo no pertenece al código BCD-N. Implementar con puertas NOR.

D	C	B	A	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1
1	0	1	0	1	0	0	1	1	1	1
1	0	1	1	1	0	0	1	1	1	1
1	1	0	0	1	0	0	1	1	1	1
1	1	0	1	1	0	0	1	1	1	1
1	1	1	0	1	0	0	1	1	1	1
1	1	1	1	1	0	0	1	1	1	1



$$\begin{aligned}
 a &= \prod_4 (1, 4) \\
 b &= \prod_4 (5, 6, 10, 11, 12, 13, 14, 15) \\
 c &= \prod_4 (2, 10, 11, 12, 13, 14, 15) \\
 d &= \prod_4 (1, 4, 7) \\
 e &= \prod_4 (1, 3, 4, 5, 7, 9) \\
 f &= \prod_4 (1, 2, 3, 7) \\
 g &= \prod_4 (0, 1, 7)
 \end{aligned}$$

$$a = (D + \bar{C} + B + A)(D + C + B + \bar{A})$$

$$b = (\bar{D} + \bar{B})(\bar{D} + \bar{C})(\bar{C} + \bar{B} + A)(\bar{C} + B + \bar{A})$$

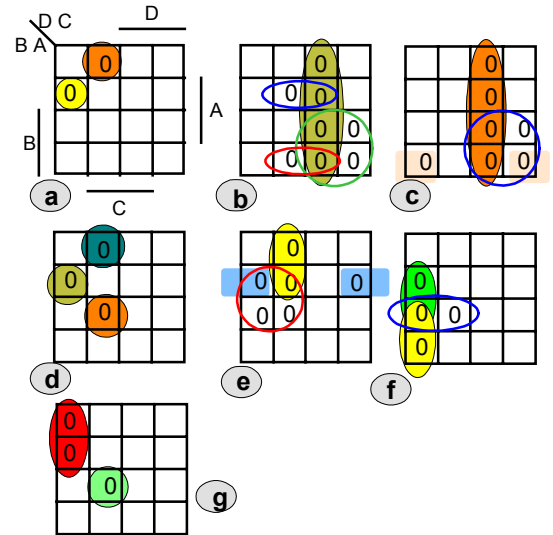
$$c = (\bar{D} + \bar{B})(C + \bar{B} + A)(\bar{D} + \bar{C})$$

$$d = (D + \bar{C} + \bar{B} + \bar{A})(D + \bar{C} + B + A)(D + C + B + \bar{A})$$

$$e = (D + \bar{A})(D + \bar{C} + B)(C + B + \bar{A})$$

$$f = (D + \bar{B} + \bar{A})(D + C + \bar{B})(D + C + \bar{A})$$

$$g = (D + \bar{C} + \bar{B} + \bar{A})(D + C + B)$$



Pasando las funciones a NOR.

$$a = \overline{D + \bar{C} + B + A + D + C + B + \bar{A}}$$

$$b = \overline{\bar{D} + \bar{B} + \bar{D} + \bar{C} + \bar{C} + \bar{B} + A + \bar{C} + B + \bar{A}}$$

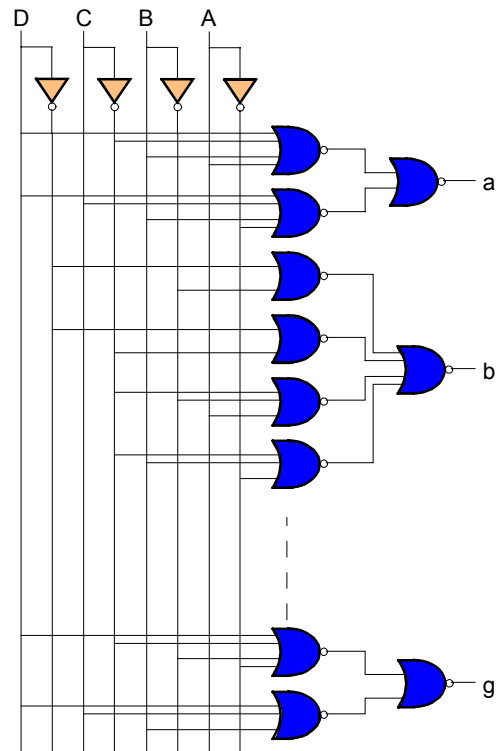
$$c = \overline{\bar{D} + \bar{B} + C + \bar{B} + A + \bar{D} + \bar{C}}$$

$$d = \overline{D + \bar{C} + \bar{B} + \bar{A} + D + \bar{C} + B + A + D + C + B + \bar{A}}$$

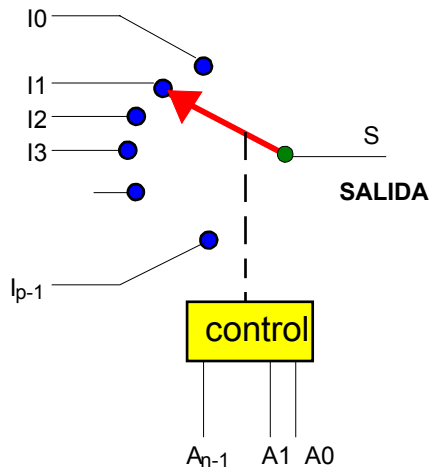
$$e = \overline{D + \bar{A} + D + \bar{C} + B + C + B + \bar{A}}$$

$$f = \overline{D + \bar{B} + \bar{A} + D + C + \bar{B} + D + C + \bar{A}}$$

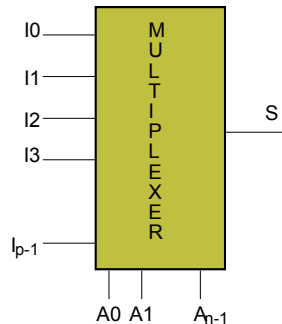
$$g = \overline{D + \bar{C} + \bar{B} + \bar{A} + D + C + B}$$



3.4-MULTIPLEXERS.



Un multiplexor puede ser representado según la figura dada. Esta representa un conmutador con varias entradas y una salida. En un momento dado una de las entradas está conectada a la salida mediante un contacto móvil. La entrada que queda conectada a la salida se selecciona mediante una lógica de control.



- En un multiplexor MSI, la entrada cuya información va a “pasar” a la salida, viene seleccionada por el código binario presente en las entradas de control. Este código, una vez transformado en decimal nos dará el número de entrada seleccionada.
- La relación entre el número de entradas de datos y el de entradas de dirección, de selección o de control viene dada por:

Entradas de datos = 2^n , donde n es el número de entradas de control.

MULTIPLEXOR DE 2 A 1 LÍNEA

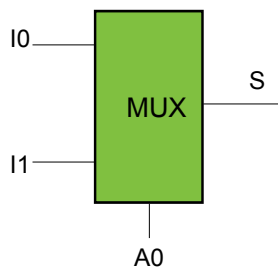
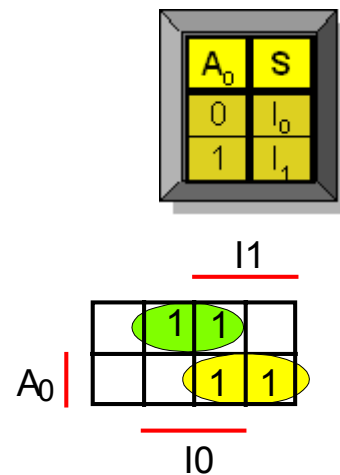


Tabla de la verdad:

I_1	I_0	A_0	S
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

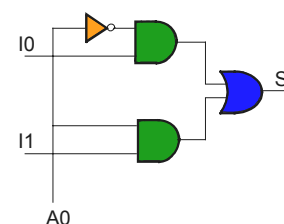
O también:



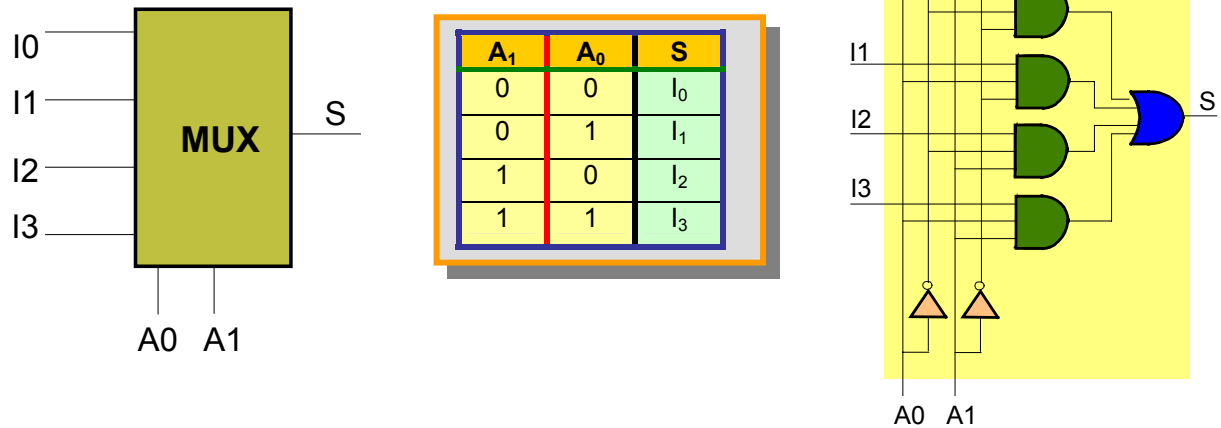
$S = \bar{I}_1 I_0 \bar{A}_0 + I_1 \bar{I}_0 A_0 + I_1 I_0 \bar{A}_0 + I_1 I_0 A_0$ Simplificando esta función tenemos:

$$S = I_0 \bar{A}_0 + I_1 A_0$$

La salida tendrá el valor de la entrada I_0 cuando la entrada de control A_0 , valga 0 y tendrá el valor de la entrada I_1 cuando A_0 valga 1. A esta conclusión se puede llegar partiendo de la tabla simplificada.

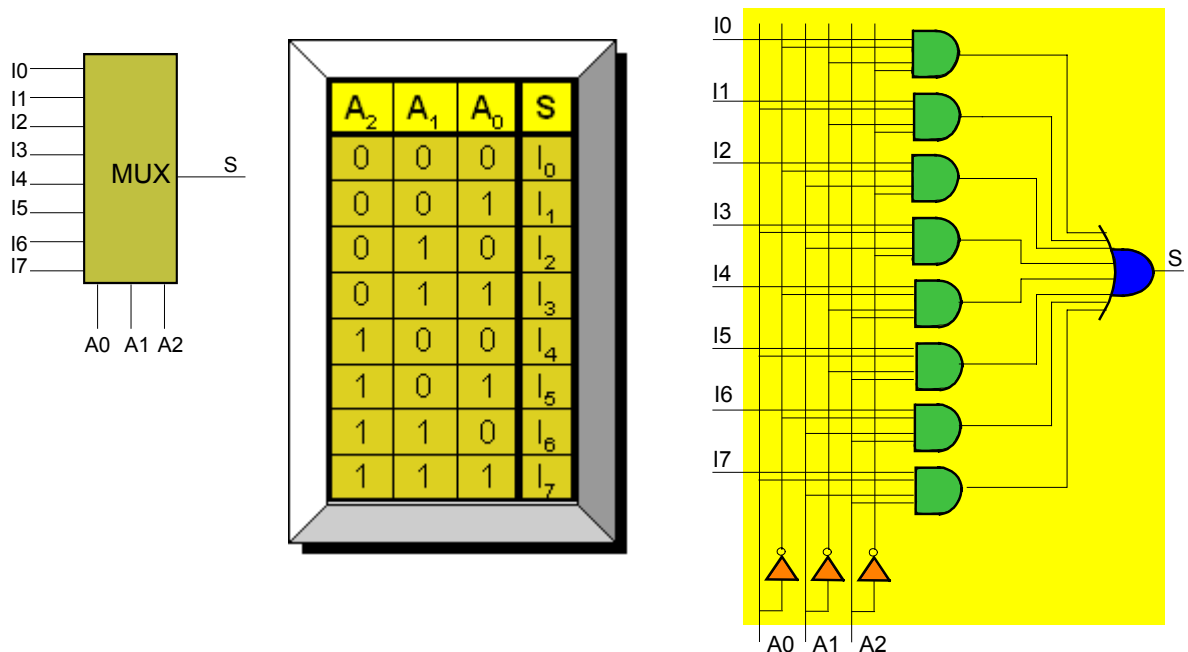


MULTIPLEXOR DE 4 A 1 LÍNEA



$$S = \bar{A}_1 \bar{A}_0 I_0 + \bar{A}_1 A_0 I_1 + A_1 \bar{A}_0 I_2 + A_1 A_0 I_3$$

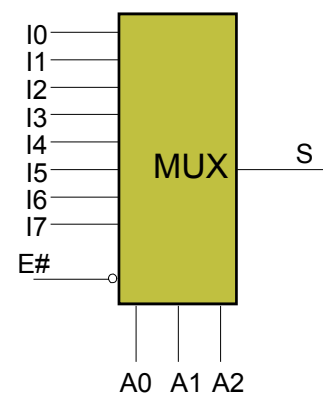
MULTIPLEXOR DE 8 A 1 LÍNEA



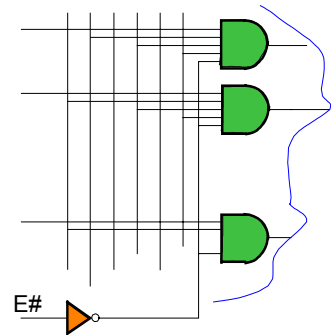
$$S = \bar{A}_2 \bar{A}_1 \bar{A}_0 I_0 + \bar{A}_2 \bar{A}_1 A_0 I_1 + \bar{A}_2 A_1 \bar{A}_0 I_2 + \bar{A}_2 A_1 A_0 I_3 + A_2 \bar{A}_1 \bar{A}_0 I_4 + A_2 \bar{A}_1 A_0 I_5 + A_2 A_1 \bar{A}_0 I_6 + A_2 A_1 A_0 I_7$$

- ❑ Muchos circuitos MSI llevan un terminal adicional llamado terminal de habilitación (*enable*, *E*), también denominado habilitación de chip (*chip enable*, *CE*), o de selección de chip (*chip select*, *CS*). La misión de este terminal es hacer que el circuito cumpla su función o no la cumpla dependiendo de sí el terminal está activo o no lo está.

El modo de operación de este o estos terminales (porque puede haber varios), para inhabilitar o habilitar el circuito es variado:



- En unos casos esta entrada de habilitación, lo que hace es aislar las entradas del circuito, de la circuitería interna, con lo cual no es posible una respuesta del mismo puesto que es insensible a las variaciones de entrada. Otras veces el aislamiento lo hace en la salida o partes intermedias (figura). El método mas utilizado para la inhabilitación es aislar parte del circuito poniéndolo en un estado de alta impedancia.



- Este tipo de entradas son también utilizadas para llevar a los chips a un estado de reposo (Stand-By), no sólo en la realización de su función, cuando no se precisa de su operación, sino también en cuanto al consumo de potencia, el cual queda reducido de forma automática en esos intervalos.

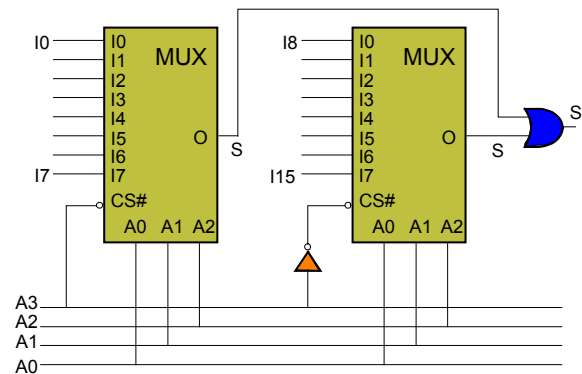
3.4.1-MULTIPLEXORES EN PARALELO.

CONSTRUCCIÓN DE UN MUX DE 16 A 1 LINEA PARTIENDO DE DOS MUX DE 8 A 1 LÍNEA.

Para realizar este multiplexor, se hace uso de la entrada de habilitación.

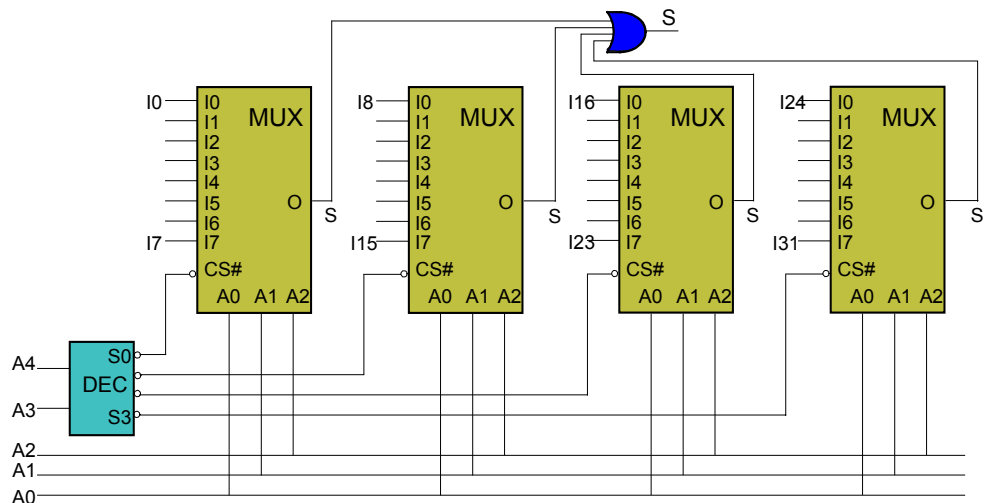
Las líneas de selección de entrada A_2 , A_1 , y A_0 van a seleccionar la misma entrada en ambos MUX., pero la línea A_3 va a discriminar entre uno y otro circuito, haciendo que solamente una de las 16 entradas “pase” a la salida a través de la puerta OR.

La entrada de selección de chip (CS) para ambos multiplexores, es activa a nivel bajo, lo cual quiere decir que cuando en esa entrada haya un cero, el circuito realizara su función, y cuando haya un 1 el circuito estará inactivo o no operativo. Cuando A_3 valga 0, estará activando el circuito con entradas desde I_0 hasta I_7 , la entrada exacta dependerá del estado de las otras tres entradas de selección (A_2 , A_1 , A_0). Cuando A_3 valga 1, el circuito activado será el que corresponde a las entradas desde I_8 hasta I_{15} . También ahora, la entrada exacta dependerá del estado de las otras tres entradas de selección (A_2 , A_1 , A_0).



CONSTRUCCIÓN DE UN MUX DE 32 A 1 LINEA PARTIENDO DE CUATRO MUX DE 8 A 1 LÍNEA.

En este caso, se hace uso de un decodificador para activar uno u otro multiplexor. El principio de funcionamiento es el mismo que el explicado para el caso anterior.



3.4.2-GENERACIÓN DE FUNCIONES LÓGICAS MEDIANTE MULTIPLEXORES

- ❑ Para generar una función determinada, se puede hacer uso de las entradas de selección de un multiplexor, tomando cada una de ellas como variables de la función.
- ❑ Para generar la función, primero hay que ponerla en forma de minterms, con lo cual, cada vez que se cumple una combinación determinada de las variables (entradas de selección en el multiplexor) el resultado de ese término ha de ser uno lógico. Como con esa combinación, vamos a seleccionar una entrada de datos determinada en el multiplexor, lo único que tenemos que hacer es poner un uno lógico en esa entrada. Los términos que no figuran en la función, los pondremos como términos que generan un cero lógico, con lo que en la entrada correspondiente a la combinación de selección que la selecciona, pondremos un cero lógico.
- ❑ Se distinguen dos casos: a) El número de entradas de selección y de variables es el mismo y b) hay una variable más que el número de entradas de selección.

a) El número de variables y el de entradas de selección es el mismo.

Ex:

*Generar la función: $A \text{ X-OR } B$, con un multiplexor de 4 a 1 línea.

La función para el multiplexor demandado es:

$$S = \bar{A}_1 \bar{A}_0 I_0 + \bar{A}_1 A_0 I_1 + A_1 \bar{A}_0 I_2 + A_1 A_0 I_3$$

Si conectamos la variable A, a la entrada de selección A_0 y la variable B, a la entrada de selección A_1 tenemos la función puesta de la forma:

$$S = \bar{B} \bar{A} I_0 + \bar{B} A I_1 + B \bar{A} I_2 + B A I_3. \text{ Como la función que queremos es:}$$

$S = \bar{B} A + B \bar{A}$, es decir $\bar{B} A 1 + B \bar{A} 1$, la función anterior (dando valores a las entradas de datos), la podemos poner como:

$$S = \bar{B} \bar{A} 0 + \bar{B} A 1 + B \bar{A} 1 + B A 0 = \bar{B} A + B \bar{A} = A \text{ X-OR } B$$

*Generar la función:

$$S = A \bar{B} \bar{C} + \bar{A} B \bar{C} + A B C$$

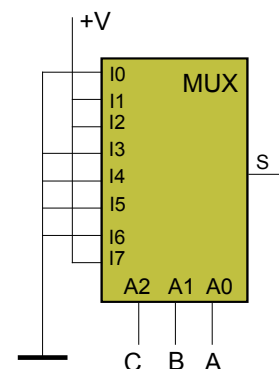
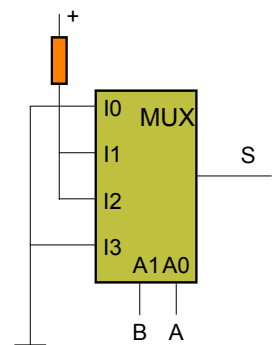
$$S = \bar{A}_2 \bar{A}_1 \bar{A}_0 I_0 + \bar{A}_2 \bar{A}_1 A_0 I_1 + \bar{A}_2 A_1 \bar{A}_0 I_2 + \bar{A}_2 A_1 A_0 I_3 + A_2 \bar{A}_1 \bar{A}_0 I_4 + A_2 \bar{A}_1 A_0 I_5 + A_2 A_1 \bar{A}_0 I_6 + A_2 A_1 A_0 I_7$$

Si conectamos A_2 con C, A_1 con B y A_0 con A, tenemos:

$$S = \bar{C} \bar{B} \bar{A} I_0 + \bar{C} \bar{B} A I_1 + \bar{C} B \bar{A} I_2 + \bar{C} B A I_3 + C \bar{B} \bar{A} I_4 + C \bar{B} A I_5 + C B \bar{A} I_6 + C B A I_7$$

Las entradas de datos que deben ser un uno lógico, es decir, que nos originan los minterms son

$$\underbrace{A \bar{B} \bar{C}}_1 + \underbrace{\bar{A} B \bar{C}}_2 + \underbrace{A B C}_7; A B C I_1 + A B C I_2 + A B C I_7;$$



b) Hay una variable más que entradas de selección.

Generar la función siguiente mediante un multiplexor de 4 a 1 línea.

$$S = \bar{A} B C + A \bar{B} \bar{C} + A B C + A B \bar{C}$$

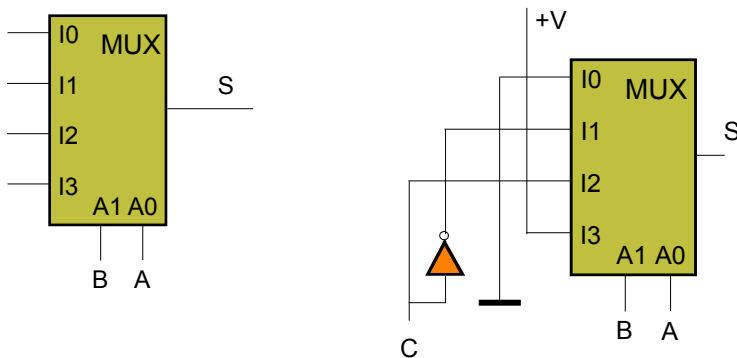
La función que define a este multiplexor es $S = \bar{A}_1 \bar{A}_0 I_0 + \bar{A}_1 A_0 I_1 + A_1 \bar{A}_0 I_2 + A_1 A_0 I_3$. Conectando la variable B a la entrada de selección A_1 y la variable A, a la entrada de selección A_0 , tenemos que la entrada que va a ser excitada por cada uno de los términos de la función a generar es:

$\bar{A} B C$ excita la entrada I_2 , $A \bar{B} \bar{C}$ excita la entrada I_1 , $A B C$ excita la entrada I_3 y $A B \bar{C}$ la entrada I_0 .

Observar que para llegar a esta conclusión solamente debemos tener en cuenta las variables que están conectadas a las entradas de selección del multiplexor. Por otro lado, tenemos dos combinaciones que excitan la entrada I_3 pero para cada uno de los casos, la variable C toma valores complementarios. Esto quiere decir que estos minterms son independientes de C

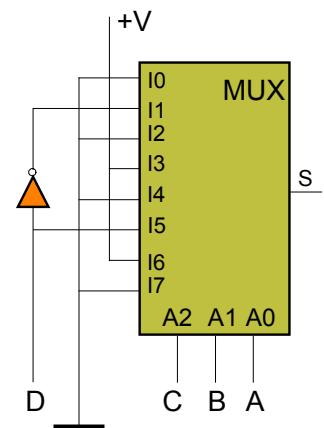
$$A B C + A B \bar{C} = A B (C + \bar{C}) = A B$$

El circuito del que se parte y el circuito solución son los siguientes.

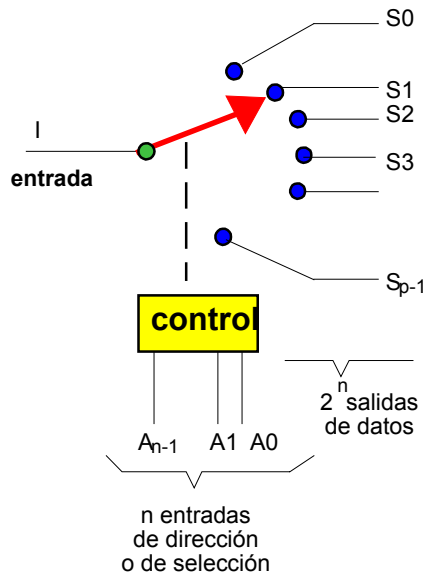


Generar mediante un multiplexor de 8 a 1 línea la función:

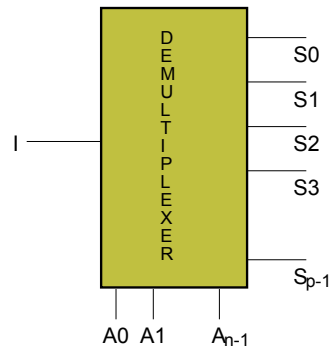
$$S = \underbrace{D C \bar{B} A}_{5} + \underbrace{\bar{D} C B \bar{A}}_{6} + \underbrace{D C B \bar{A}}_{6} + \underbrace{D \bar{C} B A}_{3} + \underbrace{\bar{D} \bar{C} B A}_{3} + \underbrace{\bar{D} \bar{C} \bar{B} A}_{1}$$



3.5-DEMULPLEXERS.



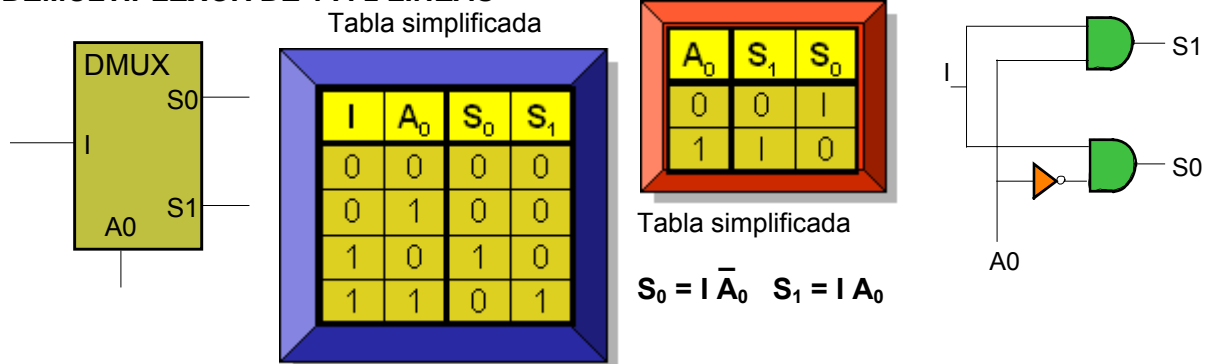
Un demultiplexor puede ser representado según la figura dada. Esta representa un conmutador con una entrada y varias salidas. En un momento dado, la entrada está conectada a una de las salidas mediante un contacto móvil. La salida que queda conectada a la entrada, se selecciona mediante una lógica de control.



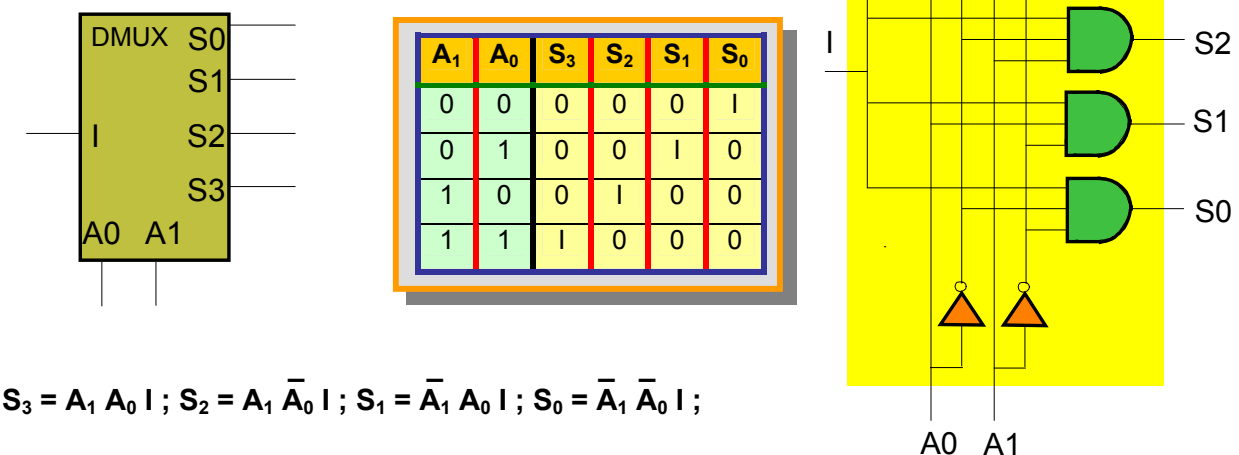
- En un multiplexor MSI, la salida que va a "recibir" internamente la información que hay en la entrada, viene seleccionada por el código binario presente en las entradas de control. Este código, una vez transformado en decimal nos dará el número de salida seleccionada.
- La relación entre el número de salidas de datos y el de entradas de dirección, de selección o de control viene dada por:

Salidas de datos = 2^n , donde n es el número de entradas de control.

DEMULPLEXOR DE 1 A 2 LÍNEAS



DEMULPLEXOR DE 1 A 4 LÍNEAS



DEMÚLTIPLEXOR DE 1 A 8 LÍNEAS

A ₂	A ₁	A ₀	S ₇	S ₆	S ₅	S ₄	S ₃	S ₂	S ₁	S ₀
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

A ₂	A ₁	A ₀	S
0	0	0	S ₀
0	0	1	S ₁
0	1	0	S ₂
0	1	1	S ₃
1	0	0	S ₄
1	0	1	S ₅
1	1	0	S ₆
1	1	1	S ₇

$$S_0 = \bar{A}_2 \bar{A}_1 \bar{A}_0 I$$

$$S_1 = \bar{A}_2 \bar{A}_1 A_0 I$$

$$S_2 = \bar{A}_2 A_1 \bar{A}_0 I$$

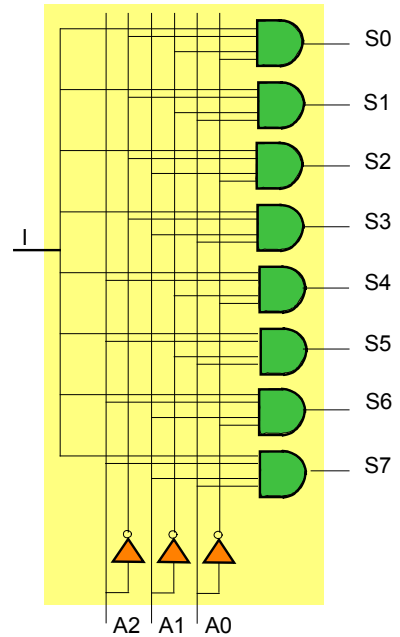
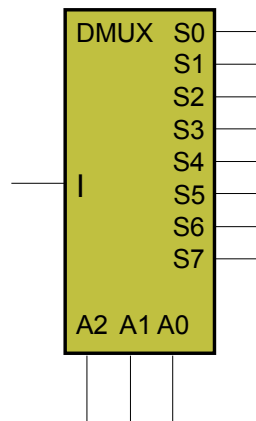
$$S_3 = \bar{A}_2 A_1 A_0 I$$

$$S_4 = A_2 \bar{A}_1 \bar{A}_0 I$$

$$S_5 = A_2 \bar{A}_1 A_0 I$$

$$S_6 = A_2 A_1 \bar{A}_0 I$$

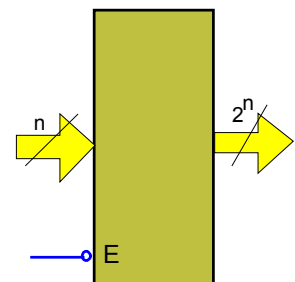
$$S_7 = A_2 A_1 A_0 I$$



3.5.1-ACTUACIÓN DE UN DECODIFICADOR COMO DEMÚLTIPLEXOR:

Generalmente, los demultiplexores los encontramos como circuito MSI, bajo la denominación *DECODERS-DEMÚLTIPLEXERS*, pues realmente la función de decodificador y demultiplexor se puede realizar con el mismo circuito siempre que éste tenga un terminal de habilitación.

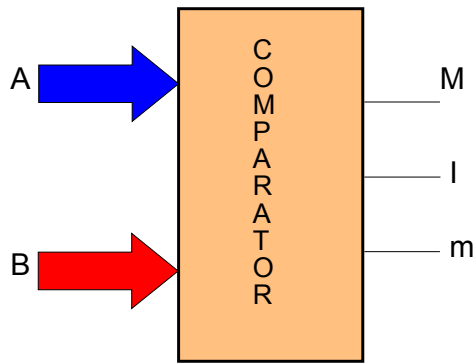
En este caso, las entradas de código en el decodificador, se convierten en entradas de selección en el demultiplexor, y el terminal de habilitación del IC, se convierte en la entrada de datos.



Cuando en las entradas de código (n), esté presente un valor, éste activará una de las 2ⁿ salidas siempre y cuando la entrada de habilitación sea activa. El hecho de que tengamos, o no, salida, va a depender por lo tanto del nivel del terminal de habilitación. Si las salidas son activas a nivel alto, y el terminal de habilitación también es activo a nivel alto, aparecerá un uno lógico, en la salida seleccionada por la información presente en las entradas de código, cuando haya un uno lógico en el terminal de habilitación, y aparecerá un cero lógico cuando haya un cero lógico en el terminal de habilitación. Según el tipo de circuito utilizado, se pueden dar los casos dados en la tabla.

Terminal de habilitación	Salidas del circuito	Correcciones en el circuito
Activo a L	Activas a L	Ninguna
Activo a L	Activas a H	Inversor en la entrada de habilitación
Activo a H	Activas a L	Inversor en la entrada de habilitación
Activo a H	Activas a H	Ninguna

3.6-COMPARADORES (COMPARATORS)

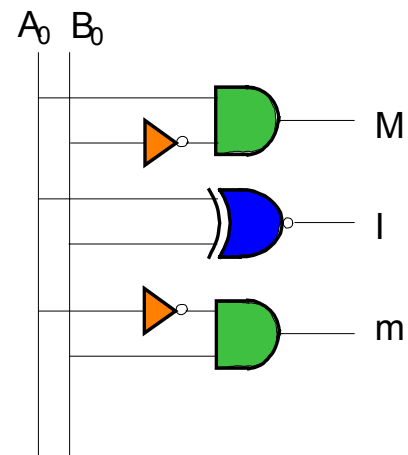


- Estos circuitos se van a encargar de comparar dos palabras de n bits (palabra A y palabra B). Según la magnitud relativa de A respecto a B se activará una salida u otra; **M** se activará cuando la palabra A sea mayor que la palabra B. Cuando ambas sean iguales se activará la salida **I**, y cuando B sea mayor que A, se activará la salida **m**. Es decir:

M → **A > B**
I → **A = B**
m → **A < B**

COMPARADOR DE DOS PALABRAS DE UN BIT.

A ₀	B ₀	M	I	m
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0



$$M = A_0 \bar{B}_0$$

$$I = \bar{A}_0 \bar{B}_0 + A_0 B_0 = A_0 \text{ X-OR } B_0 = A_0 \text{ X-NOR } B_0$$

$$m = \bar{A}_0 B_0$$

COMPARADOR DE DOS PALABRAS DE DOS BITS.

A ₁	A ₀	B ₁	B ₀	M	I	m
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

Tabla completa.

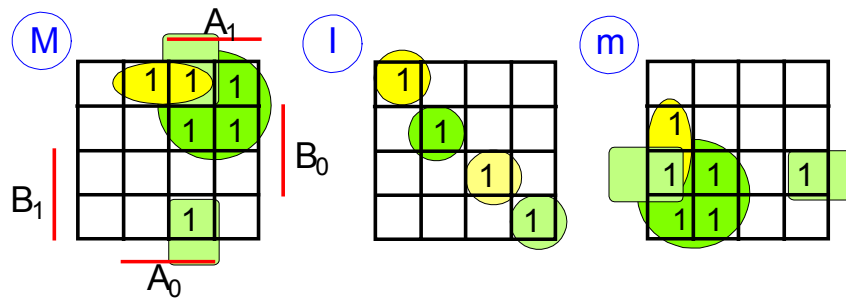
A	B	S
0	0	I
0	1	m
0	2	m
0	3	m
1	0	M
1	1	I
1	2	m
1	3	m
2	0	M
2	1	M
2	2	I
2	3	m
3	0	M
3	1	M
3	2	M
3	3	I

Tabla simplificada

$$M = \sum_4 (4, 8, 9, 12, 13, 14)$$

$$I = \sum_4 (0, 5, 10, 15)$$

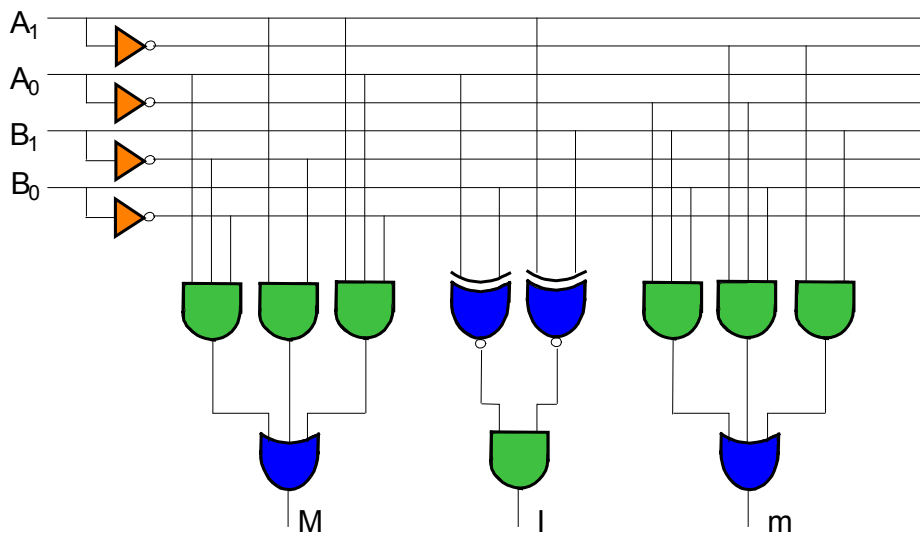
$$m = \sum_4 (1, 2, 3, 6, 7, 11)$$



$$M = A_0 \bar{B}_1 \bar{B}_0 + A_1 \bar{B}_1 + A_1 A_0 \bar{B}_0$$

$$m = \bar{A}_0 B_1 B_0 + \bar{A}_1 B_1 + \bar{A}_1 \bar{A}_0 B_0$$

$$I = \bar{A}_1 \bar{A}_0 \bar{B}_1 \bar{B}_0 + \bar{A}_1 A_0 \bar{B}_1 B_0 + A_1 \bar{A}_0 B_1 \bar{B}_0 + A_1 A_0 B_1 B_0 = \bar{A}_1 \bar{B}_1 (\bar{A}_0 \bar{B}_0 + A_0 B_0) + A_1 B_1 (\bar{A}_0 \bar{B}_0 + A_0 B_0) \\ = \bar{A}_1 \bar{B}_1 \overline{A_0 \oplus B_0} + A_1 B_1 \overline{A_0 \oplus B_0} = \overline{A_0 \oplus B_0} (\bar{A}_1 \bar{B}_1 + A_1 B_1) = \overline{A_0 \oplus B_0} \overline{A_1 \oplus B_1}$$



- La comprobación de la magnitud relativa entre una palabra y otra, puede realizarse también atendiendo a otro criterio:

Se comparan los dos MSB de cada palabra. Si los dos bits son diferentes, queda determinado ya, cual de las dos palabras es mayor y menor. Si por el contrario, los dos bits son iguales, debemos seguir comparando los dos siguientes bits de cada palabra.

Ej: Comparar dos palabras A y B, de tres bits.

A(A₂, A₁, A₀); B(B₂, B₁, B₀). Empezamos comparando los dos MSB de cada palabra, es decir, A₂ y B₂.

A ₂	B ₂	M	I	M
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

$$M_2 = A_2 \bar{B}_2$$

$$m_2 = \bar{A}_2 B_2$$

$$I_2 = \bar{A}_2 \bar{B}_2 + A_2 B_2 = \overline{A_2 \oplus B_2}$$

- Si se cumple que el resultado de la primera comparación (comparación de los MSB) es I₂ = 1, entonces Y solo entonces, se debe seguir la comparación con los siguientes bits:

Si $A_2 = B_2$

A_1	B_1	M	I	M
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

$$M_1 = A_1 \bar{B}_1$$

$$m_1 = \bar{A}_1 B_1$$

$$I_1 = \bar{A}_1 \bar{B}_1 + A_1 B_1 = \overline{A_1 \text{ X-OR } B_1}$$

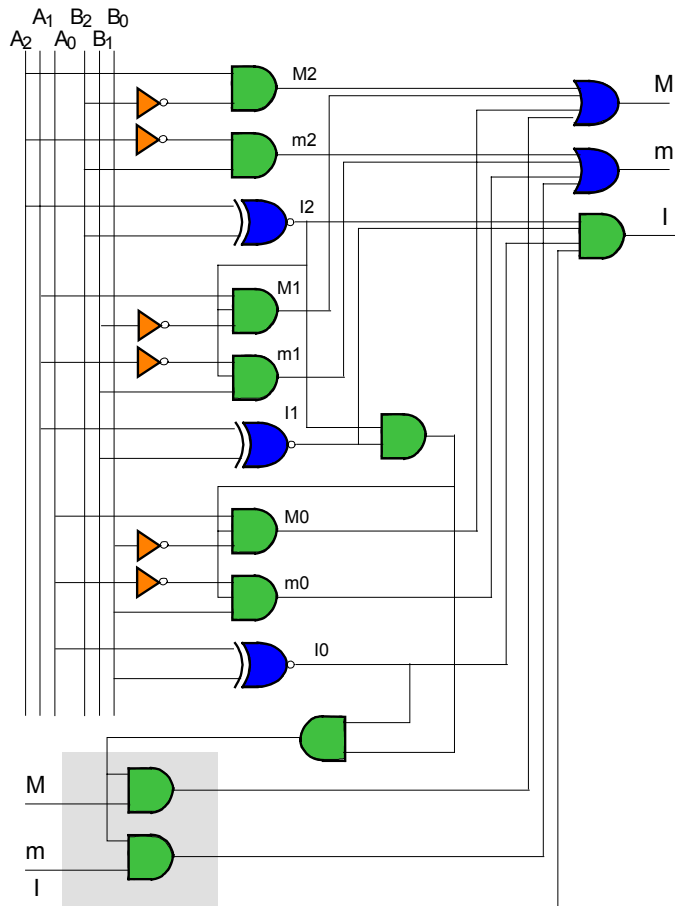
Si $I_1 = 1$ es decir, $A_1 = B_1$ se sigue comparando.

A_0	B_0	M	I	M
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

$$M_0 = A_0 \bar{B}_0$$

$$m_0 = \bar{A}_0 B_0$$

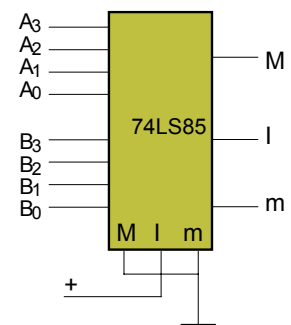
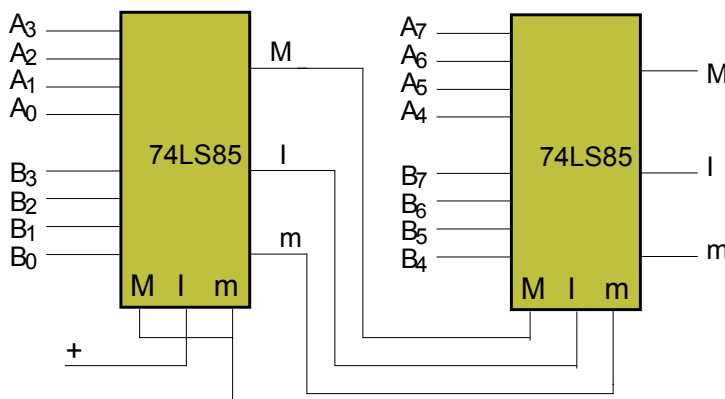
$$I_0 = \bar{A}_0 \bar{B}_0 + A_0 B_0 = \overline{A_0 \text{ X-OR } B_0}$$



Si llegados a esta comparación, $I = 1$, quiere decir que los dos números binarios son iguales. Los circuitos comerciales implementan unas entradas adicionales que sirven para conectar comparadores para el trabajo en paralelo, comparando así, palabras de más bits a base de la interconexión de dos o más de estos circuitos.

COMPARADOR INTEGRADO DE DOS PALABRAS DE CUATRO BITS.

COMPARADOR DE OCHO BITS CON EL 74XX85



Aunque la comparación se hace simultáneamente en los dos circuitos, el resultado de la comparación de los cuatro bits de menos peso, no tiene influencia sobre la salida total del comparador (M, I y m del circuito de la izquierda), a menos que el resultado de la comparación de los cuatro bits más significativos haya sido el de una igualdad entre esos cuatro bits.

3.7-GENERADOR Y VERIFICADOR DE PARIDAD. (PARITY GENERATOR AND CHECKER)

- El BIT de paridad es un BIT que se añade a un código determinado para poder detectar un error en la transmisión o recepción de las diferentes palabras que componen ese código. Este nuevo BIT que se añade al código “matriz”, pasa a formar parte de las nuevas palabras que componen el nuevo código resultante.

Existen dos tipos de códigos de paridad. *CÓDIGO DE PARIDAD PAR (Even Parity)* y *CÓDIGO DE PARIDAD IMPAR (Odd Parity)*

3.7.1-EVEN PARITY:

- El número de unos en cada una de las palabras del código debe ser un número par. Entonces, partiendo de un código, se añadirá un 1 o un 0 a cada una de las combinaciones para que en el nuevo código, el número de unos en cada palabra sea par.

El BIT de paridad puede ir antes del MSB o después del LSB del código “matriz”. Generalmente se coloca antes.

D	C	B	A	P- bit	New code				
0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	1
0	0	1	0	1	1	0	0	1	0
0	0	1	1	0	0	0	0	1	1
0	1	0	0	1	1	0	1	0	0
0	1	0	1	0	0	0	1	0	1
0	1	1	0	0	0	0	1	1	0
0	1	1	1	1	1	0	1	1	1
1	0	0	0	1	1	1	0	0	0
1	0	0	1	0	0	1	0	0	1

Even parity code, partiendo del código BCD-N.

3.7.2-ODD PARITY:

- El número de unos en cada una de las palabras del código debe ser un número impar. Entonces, partiendo de un código, se añadirá un 1 o un 0 a cada una de las combinaciones para que en el nuevo código, el número de unos en cada palabra sea impar.

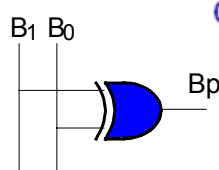
D	C	B	A	P- bit	New code				
0	0	0	0	1	1	0	0	0	0
0	0	0	1	0	0	0	0	0	1
0	0	1	0	0	0	0	0	1	0
0	0	1	1	1	1	0	0	1	1
0	1	0	0	0	0	0	1	0	0
0	1	0	1	1	1	0	1	0	1
0	1	1	0	1	1	0	1	1	0
0	1	1	1	0	0	0	1	1	1
1	0	0	0	0	0	1	0	0	0
1	0	0	1	1	1	1	0	0	1

Odd parity code, partiendo del código BCD-N.

3.7.3-PARITY GENERATOR:

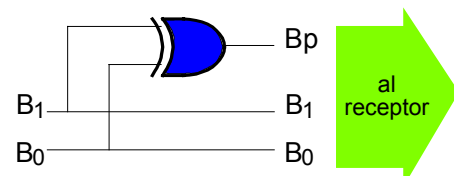
B ₁	B ₀	B _p
0	0	0
0	1	1
1	0	1
1	1	0

Even parity:



Dos bits:

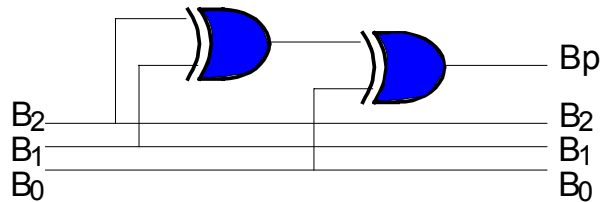
$$B_p = \bar{B}_1 B_0 + B_1 \bar{B}_0 = B_1 \text{ X-OR } B_0$$



Tres bits

B ₂	B ₁	B ₀	B _p
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

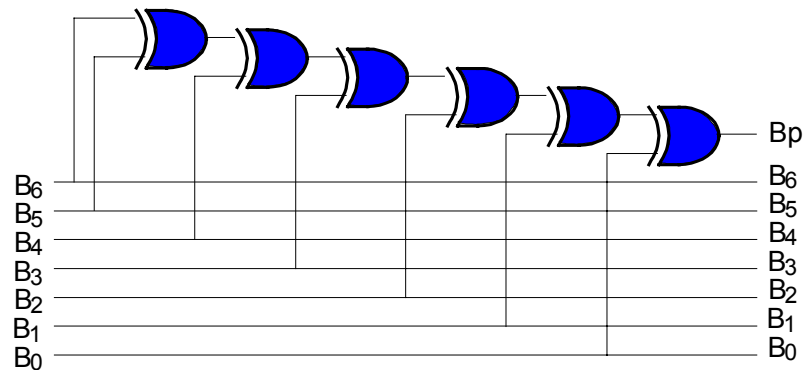
$$\begin{aligned}
 B_p &= \bar{B}_2 \bar{B}_1 B_0 + \bar{B}_2 B_1 \bar{B}_0 + B_2 \bar{B}_1 \bar{B}_0 + B_2 B_1 B_0 = \\
 &= \bar{B}_2 (\bar{B}_1 B_0 + B_1 \bar{B}_0) + B_2 (\bar{B}_1 \bar{B}_0 + B_1 B_0) = \\
 &= \bar{B}_2 (B_1 \text{ X-OR } B_0) + B_2 \overline{(B_1 \text{ X-OR } B_0)} = B_2 \text{ X-OR } B_1 \text{ X-OR } B_0
 \end{aligned}$$



Even parity (Paridad Par)

Siguiendo el mismo razonamiento, para siete bits tendremos

$$B_p = B_6 \text{ X-OR } B_5 \text{ X-OR } B_4 \text{ X-OR } B_3 \text{ X-OR } B_2 \text{ X-OR } B_1 \text{ X-OR } B_0$$



Odd parity:

- Para conseguir la paridad impar, lo único que tenemos que hacer es tomar la salida Bp a través de un inversor.

3.7.4-PARITY CHECKER

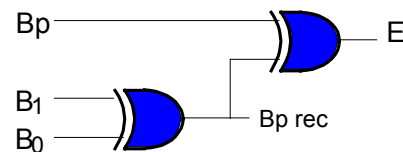
B_1	B_0	B_{PR}
0	0	0
0	1	1
1	0	1
1	1	0

La palabra del nuevo código es recibida y chequeada por el detector de paridad. Los bits que componen el dato, generan un nuevo BIT de paridad (B_{PR}), el cual se compara con el BIT de paridad enviado por el emisor (B_P), dando una señal de error E en caso de que no coincidan.

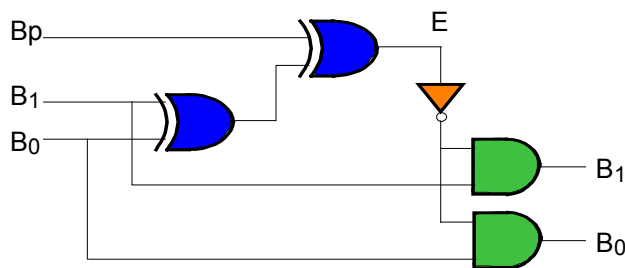
B_{PR}	B_P	E
0	0	0
0	1	1
1	0	1
1	1	0

$$E = \overline{B_{PR}} B_P + B_{PR} \overline{B_P} = B_{PR} \text{ X-OR } B_P$$

$$B_{PR} = \overline{B_1} B_0 + B_1 \overline{B_0} = B_1 \text{ X-OR } B_0$$

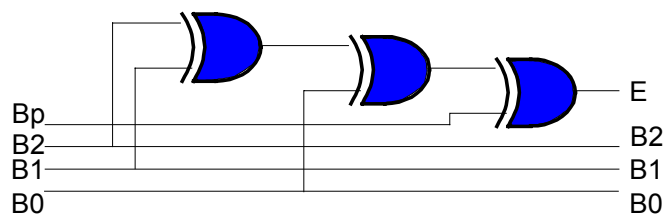


- Cuando $E=0$, indica que el BIT de paridad generado en origen con el dato enviado (B_P) es igual que el BIT de paridad generado en destino con el dato recibido (B_{PR}). Si $E=1$ el BIT de paridad generado en destino y el BIT de paridad generado en origen son diferentes,



indicándonos que existe un error en los datos recibidos. Esta indicación de error puede utilizarse para bloquear los datos recibidos además de las correspondientes acciones como indicar que se vuelvan a enviar etc.

Para tres bits tenemos



Estos tipos de detectores son los más simples que se pueden dar. Solamente detectan el cambio en un BIT y no indican que BIT es el erróneo por lo que no lo pueden corregir. Otro método utilizado es el *CRC (Cyclic Redundancy Check)*. Con este método la información es vista como un polinomio. La comprobación de errores se realiza a través de algoritmos matemáticos.

3.8-ARITMÉTICA. SUMADORES (ADDERS)

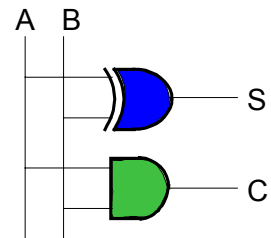
HALF ADDER. (Sumador)

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

□ Este tipo de sumador, no tiene en cuenta el Carry anterior.

$$S = \bar{A} B + A \bar{B} = A \text{ X-OR } B$$

$$C = A B$$



FULL ADDER. (Sumador completo)

C ₀	A	B	S	C ₁
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Este sumador si tiene en cuenta el Carry anterior.

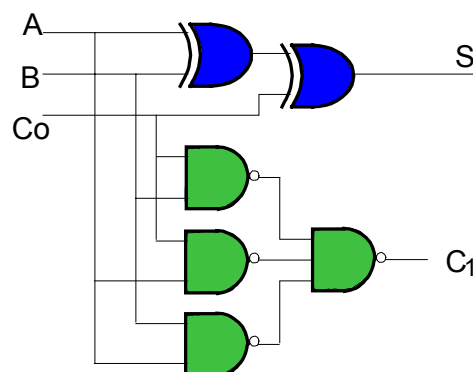
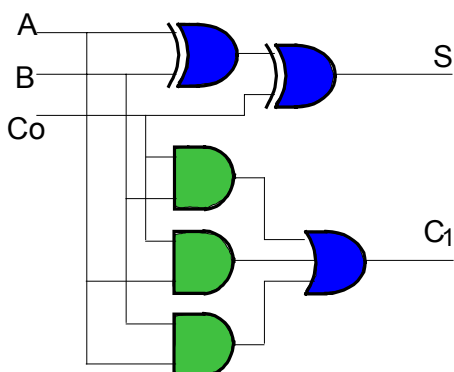
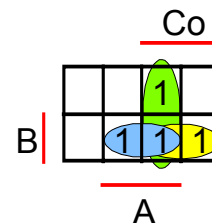
$$\begin{aligned} S &= \bar{C}_0 \bar{A} B + \bar{C}_0 A \bar{B} + C_0 \bar{A} \bar{B} + C_0 A B = \\ &= \bar{C}_0 (\bar{A} B + A \bar{B}) + C_0 (\bar{A} \bar{B} + A B) = \bar{C}_0 A \text{ X-OR } B + C_0 A \text{ X-NOR } B \\ &= C_0 \text{ X-OR } A \text{ X-OR } B \end{aligned}$$

$$\begin{aligned} C_1 &= \bar{C}_0 A B + C_0 \bar{A} B + C_0 A \bar{B} + C_0 A B = \\ &= C_0 (A \bar{B} + \bar{A} B) + A B (\bar{C}_0 + C_0) = C_0 A \text{ X-OR } B + A B \end{aligned}$$

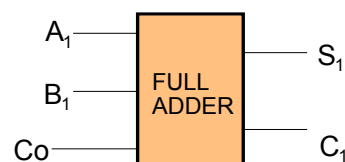
C₁ lo podemos tomar también como: C₁ = Σ₃ (3, 5, 6, 7)

$$C_1 = C_0 A + A B + C_0 B = C_0 (A + B) + A B$$

O también: C₁ = $\overline{\overline{C_0 A} \overline{C_0 B} \overline{A B}}$ (puertas NAND)



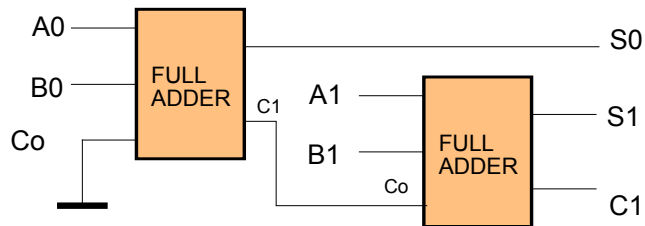
El circuito sumador completo se puede representar:



SUMA DE DOS PALABRAS DE DOS BITS:

A base de sumadores simples de un BIT, se pueden formar sumadores más complejos de palabras con un determinado número de bits.

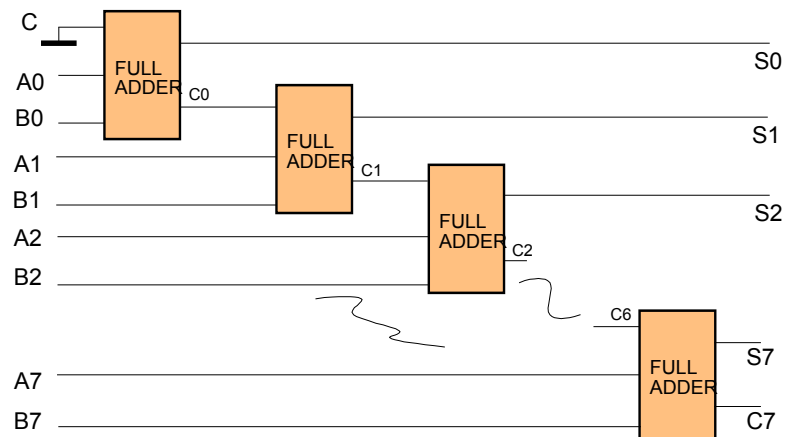
Sea la palabra A ($A_1 A_0$) y la palabra B ($B_1 B_0$):



SUMA DE DOS PALABRAS (A Y B) DE OCHO BITS:

A ($A_7 A_6 A_5 A_4 A_3 A_2 A_1 A_0$), B ($B_7 B_6 B_5 B_4 B_3 B_2 B_1 B_0$)

La forma de trabajo de este sumador es la llamada **TRABAJO EN SERIE**, pues cada etapa de las que se compone el mismo, debe esperar el carry de la etapa anterior (a excepción del primer sumador). El resultado será correcto cuando se hayan generado todos los carries de todas las etapas



3.8.1-SUMADOR PARALELO

- En este tipo de sumadores, el carry se genera simultáneamente,

C_0	A_1	B_1	S_1	C_1
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$S = C_0 \text{ X-OR } A_1 \text{ X-OR } B_1$$

C_1 lo podemos tomar como: $C_1 = \sum_3 (3, 5, 6, 7)$. Simplificando queda:

$$C_1 = C_0 A_1 + A_1 B_1 + C_0 B_1 = C_0 \underbrace{(A_1 + B_1)}_{\text{Suma (S)}} + \underbrace{A_1 B_1}_{\text{producto (P)}} = C_0 S_1 + P_1$$

Entonces.

$$C_1 = A_1 B_1 + C_0 (A_1 + B_1) = P_1 + C_0 S_1$$

$$C_2 = A_2 B_2 + C_1 (A_2 + B_2) = P_2 + C_1 S_2$$

$$C_3 = A_3 B_3 + C_2 (A_3 + B_3) = P_3 + C_2 S_3$$

$$C_n = A_n B_n + C_{n-1} (A_n + B_n) = P_n + C_{n-1} S_{n-1}$$

$$C_{n+1} = A_{n+1} B_{n+1} + C_n (A_{n+1} + B_{n+1}) = P_{n+1} + C_n S_{n+1}$$

En general:

$$C_i = A_i B_i + C_{i-1} (A_i + B_i) = P_i + C_{i-1} S_i$$

$$C_1 = P_1 + C_0 S_1$$

$$C_2 = P_2 + C_1 S_2 = P_2 + (P_1 + C_0 S_1) S_2 = P_2 + P_1 S_2 + C_0 S_1 S_2$$

$$C_3 = P_3 + C_2 S_3 = P_3 + (P_2 + P_1 S_2 + C_0 S_1 S_2) S_3 = P_3 + P_2 S_3 + P_1 S_2 S_3 + C_0 S_1 S_2 S_3$$

$$C_4 = P_4 + C_3 S_4 = P_4 + (P_3 + P_2 S_3 + P_1 S_2 S_3 + C_0 S_1 S_2 S_3) S_4 = P_4 + P_3 S_4 + P_2 S_3 S_4 + P_1 S_2 S_3 S_4 + C_0 S_1 S_2 S_3 S_4$$

$$C_n = P_n + P_{n-1} S_n + P_{n-2} S_{n-1} S_n + P_{n-3} S_{n-2} S_{n-1} S_n + \dots + P_1 S_2 S_3 S_4 \dots S_n + C_0 S_1 S_2 S_3 S_4 \dots S_n$$

□ De esta última expresión se puede observar que cualquier carry, va a depender únicamente de las entradas y es independiente de los carries anteriores.

□ Por otro lado:

$$S * P = (A + B) A B = A A B + A B B = A B + A B = A B = P$$

$$S + P = (A + B) + A B = [(A + B) + A][A + B] = (A + B)(A + B) = A + B = S$$

$$S * \bar{P} = (A + B) \overline{A B} = A \overline{A B} + \overline{A B} B = A (\bar{A} + \bar{B}) + B (\bar{A} + \bar{B}) = A \bar{B} + B \bar{A} = A \text{ X-OR } B$$

$$S * P = P$$

$$S + P = S$$

$$S * \bar{P} = A \text{ X-OR } B$$

SUMA EN PARALELO.

C	A ₀	B ₀	S ₀	C ₀
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$S_0 = \sum_3 (1, 2, 4, 7).$$

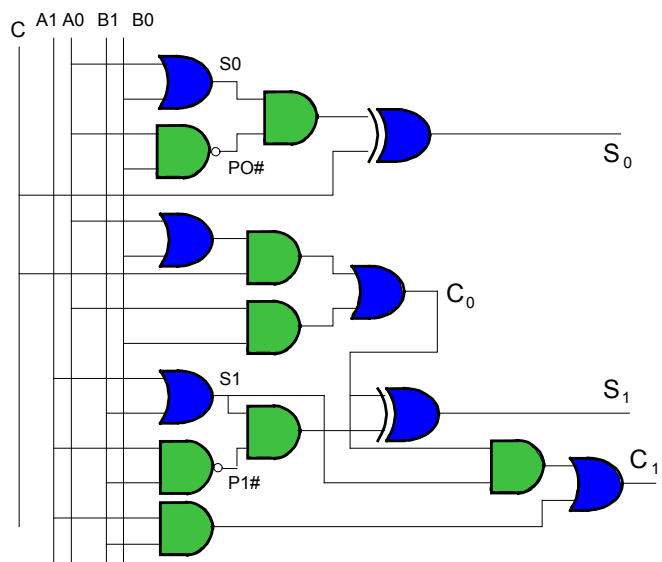
$$C_0 = \sum_3 (3, 5, 6, 7).$$

$$S_0 = C \text{ X-OR } A_0 \text{ X-OR } B_0 = C \text{ X-OR } S_0 \bar{P}_0$$

$$C_0 = P_0 + C S_0$$

$$S_1 = C_0 \text{ X-OR } S_1 \bar{P}_1$$

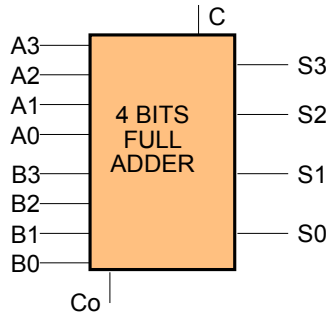
$$C_1 = P_1 + C_0 S_1$$



3.8.2-EJEMPLOS.

1.- Realizar un circuito sumador a partir de Full Adders IC-s, que sume dos palabras en BCD(N) y dé el resultado también en BCD(N)

- Hay que tener en cuenta que cada uno de los números puede variar entre 0000 (0) y 1001 (9). El valor de la suma de ambos números puede variar entre 0000 (0) y 10010 (18).



- Utilizando un sumador completo de cuatro bits para este fin, el resultado obtenido estaría dado en binario, siendo los bits del resultado:

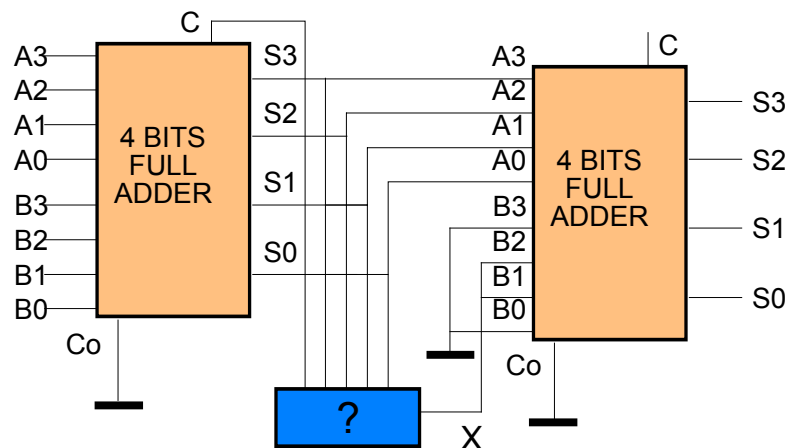
C (MSB) S₃ S₂ S₁ S₀ (LSB)

- Para obtener el resultado en BCD-N, hay que tener en cuenta que para valores superiores a 1001 (9) se debe codificar cada dígito decimal con cuatro bits. Para poder hacer esto, basta observar que si sumamos 0110 (6) a cualquier valor binario mayor de 1001 (9) obtenemos el equivalente a ese número binario en BCD-N.

1010 (10 binario) + 0110 (6 bin) = 1 0000 (10 BCD-N)
 1011 (11 bin) + 0110 (6 bin) = 1 0001 (11 BCD-N)
 10010 (18 bin) + 0110 (6 bin) = 1 1000 (18 BCD-N)

- De forma práctica, para obtener el resultado en BCD, tengo que diseñar un circuito combinacional que me permita sumar 0110 (6) al valor binario obtenido en la primera suma, siempre y cuando dicho valor sea superior o igual que 1001 (9).

El sumador de la derecha lo que hace es sumar al



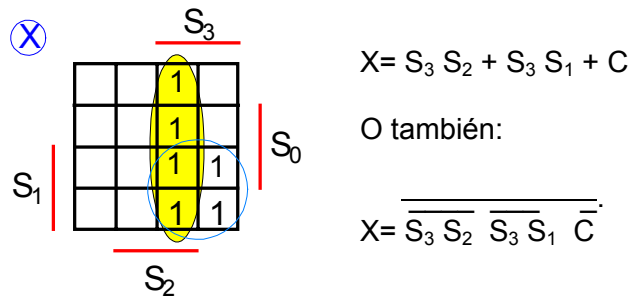
C	S ₃	S ₂	S ₁	S ₀
0	1	0	0	0
0	1	0	0	1
0	1	0	1	0
0	1	0	1	1
0	1	1	0	0
0	1	1	0	1
0	1	1	1	0
0	1	1	1	1
1	0	0	0	0
1	0	0	0	1
1	0	0	1	0

resultado de la primera suma, el valor 0110 (6) o 0000 (0) según que el resultado de dicha primera suma sea mayor o menor que 1001 (9). El circuito combinacional a diseñar, tendrá como entradas las salidas del sumador de la izquierda y como salida un uno cuando el valor de esas entradas exceda de 1001 (9).

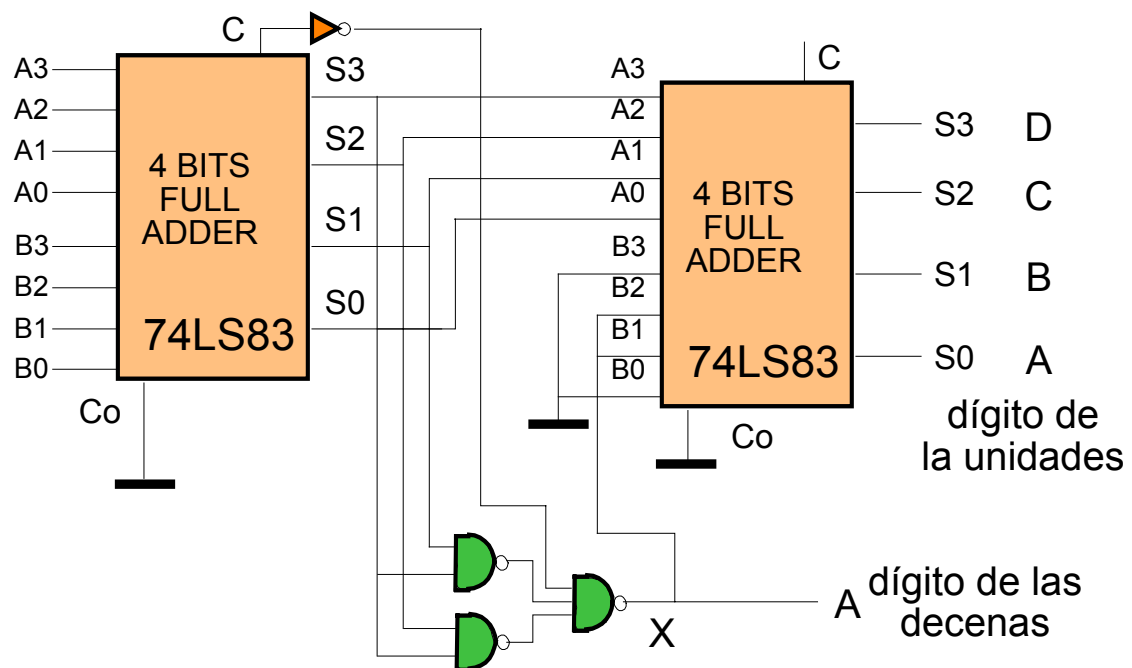
En la tabla se dan los números equivalentes binarios a partir del número 8. Observando el circuito y dicha tabla, vemos que X va a ser uno, cuando se cumplan los minterms desde 01010 (10) hasta 1111 (15) **OR** el valor de la C sea uno. Es decir:

$$X = S_3 \bar{S}_2 S_1 \bar{S}_0 + S_3 \bar{S}_2 S_1 S_0 + S_3 S_2 \bar{S}_1 \bar{S}_0 + S_3 S_2 \bar{S}_1 S_0 +$$

$$S_3 S_2 S_1 \bar{S}_0 + S_3 S_2 S_1 S_0 + C$$



El circuito será por lo tanto:



Hay que tener en cuenta que el carry (C) del sumador de la derecha no puedo utilizarlo como dígito para las decenas pues no se activa hasta el valor (16) y lo que necesito es que se active a partir del valor 9 (10, 11, 12, ...).

Pej: si la salida del sumador de la izquierda vale 10010 (C S₃ S₂ S₁ S₀), el sumador de la derecha sumaría 0010 + 0110 = 1000 con lo que el carry de este sumador no se activaría.

Debo utilizar la salida X como dígito para las decenas conectándolo a la entrada de la variable A de un decodificador BCD-N/ siete segmentos y poniendo las variables B, C y D a cero lógico.

1.- Realizar un circuito sumador a partir de Full Adders IC-s, que sume dos palabras en BCD(X3) y de el resultado también en BCD(X3)

Código BCD X3

D	C	B	A	
0	0	1	1	0
0	1	0	0	1
0	1	0	1	2
0	1	1	0	3
0	1	1	1	4
1	0	0	0	5
1	0	0	1	6
1	0	1	0	7
1	0	1	1	8
1	1	0	0	9

La suma va a variar entre

$$0011 (0) + 0011 (0) = 0110 (\neq 0) \text{ y } 1100 (9) + 1100 (9) = 11000 (\neq 18)$$

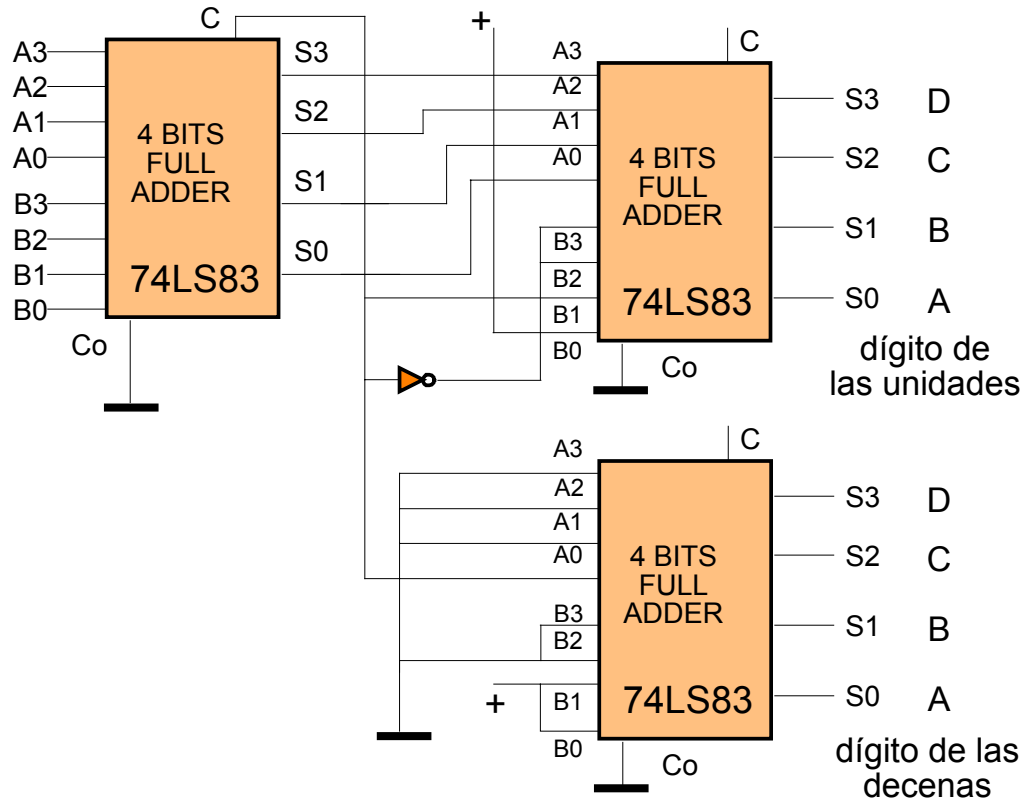
Vemos que los resultados necesitan ser corregidos, ya que el 0 en BCD-X3 es 0011 y el valor 18 es 0100 1011

Observando la tabla siguiente, vemos que para obtener los valores deseados de las unidades en BCD-X3, partiendo de los valores binarios que vamos a obtener del sumador, lo único que tenemos que hacer es restar el valor 0011 (3) binario a todos los valores que están entre el 0110 y el 1111 en la salida del sumador. De igual forma para conseguir los valores en código BCD-X3 debemos sumar el valor 0011 (3) binario a todos los valores que están comprendidos entre el 0000 y el 1000 en la salida del sumador.

Valor que obtengo del sumador					Valor que debo obtener en BCD-X3					
C	S ₃	S ₂	S ₁	S ₀	C	D	C	B	A	
0	0	1	1	0	0011	0	0	1	1	Restar el valor 0011
0	0	1	1	1	0011	0	1	0	0	
0	1	0	0	0	0011	0	1	0	1	
0	1	0	0	1	0011	0	1	1	0	
0	1	0	1	0	0011	0	1	1	1	
0	1	0	1	1	0011	1	0	0	0	
0	1	1	0	0	0011	1	0	0	1	
0	1	1	0	1	0011	1	0	1	0	
0	1	1	1	0	0011	1	0	1	1	
0	1	1	1	1	0011	1	1	0	0	
1	0	0	0	0	0100	0	0	1	1	Sumar el valor 0011
1	0	0	0	1	0100	0	1	0	0	
1	0	0	1	0	0100	0	1	0	1	
1	0	0	1	1	0100	0	1	1	0	
1	0	1	0	0	0100	0	1	1	1	
1	0	1	0	1	0100	1	0	0	0	
1	0	1	1	0	0100	1	0	0	1	
1	0	1	1	1	0100	1	0	1	0	
1	1	0	0	0	0100	1	0	1	1	

Para conseguir las decenas, lo que se debe hacer es sumar 0011 (3) binario al valor del carry del sumador. Para realizar la resta entre el valor obtenido en el sumador y el valor 0011, lo único que se debe hacer es sumar al resultado del sumador, el complemento a 2 de 0011, es decir 1101.

El circuito final es el siguiente.



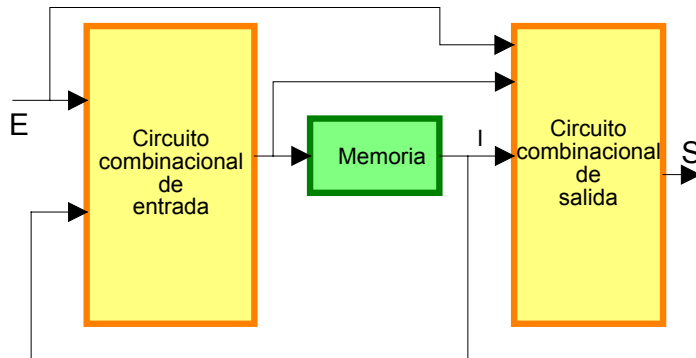


Circuitos Secuenciales

4

4.-ELECTRÓNICA SECUENCIAL.

- Los circuitos secuenciales son aquellos en los cuales la salida, en un instante determinado, no solamente depende del estado lógico de las variables de entrada, si no que es también función del estado interno (variables de estado interno) que posea el circuito.



Otra forma de expresarlo es que la salida no depende exclusivamente del valor de las variables de entrada, si no que, también va a depender del estado anterior de la salida.

La figura representa el esquema general de un circuito secuencial. (Autómata de Mealy)

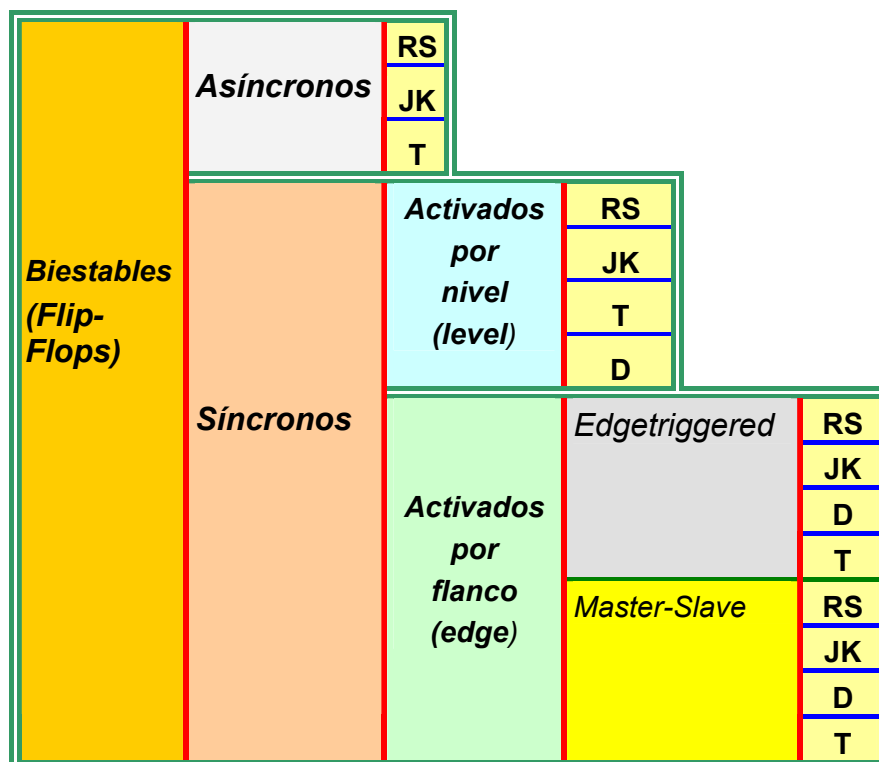
CIRCUITOS SECUENCIALES SÍNCRONOS:

- Estos circuitos necesitan una señal de control llamada *RELOJ (CLOCK)*, procedente de una fuente externa al circuito, la cual debe ser aplicada a este, para que se produzca una respuesta a la excitación. El clock se aplica a cada bloque integrado, en un terminal destinado para tal fin, de esta manera, sincroniza todos los módulos del circuito. La velocidad de clock, ha de ser tal que se adapte al bloque más lento del circuito.

CIRCUITOS SECUENCIALES ASÍNCRONOS:

- Cada módulo responde a los cambios de las variables internas o externas cuando existen variaciones en las entradas, sin necesidad de esperar el “permiso” del clock.

4.1-FLIP-FLOPS

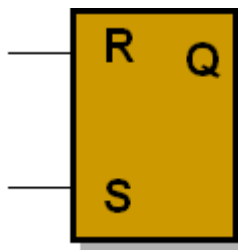


El **Flip-Flop** es el circuito básico secuencial. Tiene dos estados estables (**biestable**), es decir, puede estar en un estado u otro de forma indefinida. Para sacarle de uno u otro estado necesita un estímulo externo. Este circuito también es conocido por el nombre de **báscula**.

En la tabla adjunta se indican los tipos de básculas que existen.

4.1.1-FF ASÍNCRONOS RS.

RS NOR

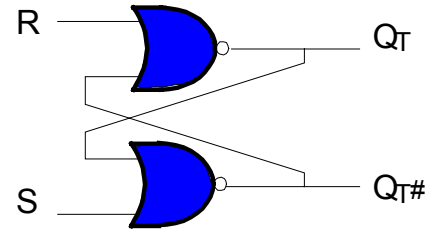


$$Q_{T+\Delta T} = R + \overline{Q_T}$$

$$\overline{Q_T} = S + \overline{Q_T}$$

$$Q_{T+\Delta T} = R + S + \overline{Q_T} = \overline{R} (S + \overline{Q_T})$$

$$Q_{T+\Delta T} = \overline{R} (S + \overline{Q_T})$$



$$\overline{Q_{T+\Delta T}} = S + \overline{Q_T} ; \quad \overline{Q_T} = R + \overline{Q_T} ; \quad \overline{Q_{T+\Delta T}} = S + R + \overline{Q_T} = \overline{S} (R + \overline{Q_T})$$

$$\overline{Q_{T+\Delta T}} = \overline{S} (R + \overline{Q_T})$$

TABLA DE LA VERDAD

R	S	Q_T	$Q_{T+\Delta T}$	$Q_{T+\Delta T} \#$
0	0	0	0	1
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	1
1	0	1	0	1
1	1	0	0	0
1	1	1	0	0

TABLA SIMPLIFICADA

R	S	$Q_{T+\Delta T}$	$Q_{T+\Delta T} \#$
0	0	Q_T	$Q_T \#$
0	1	1	0
1	0	0	1
1	1	0	0

COMBINACIÓN PROHIBIDA.

- En todas las combinaciones menos en una, se cumple que las dos salidas del FF son complementarias. A esta combinación en la cual las dos salidas toman el valor 0 se la llama

combinación prohibida. Para esta combinación se cumple que partiendo de ella, no se puede predecir la evolución que tendrán las salidas del FF cuando haya un cambio en las variables de entrada. El valor que tomen dichas salidas, va a depender de cual de las dos puertas que componen el circuito es la más rápida. Esta situación no se da para ninguna de las otras combinaciones, donde la evolución está perfectamente determinada.

TABLA DE TRANSICIÓN

R	S	Q_T	$Q_{T+\Delta T}$
X	0	0	0
0	1	0	1
1	0	1	0
0	X	1	1

La tabla de transición nos indica qué valores deben tener las variables de entrada R y S para que estando la salida del FF en un estado determinado en Q_T , evolucione hacia otro estado determinado en $Q_{T+\Delta T}$. Estas evoluciones son las indicadas en la tabla en la columna de la derecha.

En las columnas de R y S están dados los valores que deben tomar estas variables para que se cumpla la transición indicada en su correspondiente fila. El valor de X indica que no importa el valor que tome esa variable para la transición correspondiente. Por ejemplo la

primera fila se interpreta de la siguiente forma: “no importa el valor que tome la variable R, si la variable S toma el valor bajo y la salida del FF esta en estado bajo, la evolución de esta salida será mantenerse en el estado bajo”

Para la realización de esta tabla, se parte de la tabla de la verdad, tomando las columnas de Q_T y de $Q_{T+\Delta T}$ para ver las transiciones, y estudiando el valor de R y de S. Lógicamente la combinación prohibida no interviene en la confección.

RS NAND

$$Q_{T+\Delta T} = \overline{\overline{S} \overline{Q_T}}$$

$$\overline{Q_T} = \overline{Q_T \overline{R}}$$

$$Q_{T+\Delta T} = \overline{\overline{S} \overline{Q_T} \overline{R}} = S + (Q_T \overline{R})$$

$$Q_{T+\Delta T} = S + (Q_T \overline{R})$$

$$\overline{Q_{T+\Delta T}} = \overline{R} \overline{Q_T}; \quad Q_T = \overline{\overline{S} \overline{Q_T}}; \quad \overline{Q_{T+\Delta T}} = \overline{R} \overline{\overline{S} \overline{Q_T}} = R + (\overline{S} \overline{Q_T})$$

$$\overline{Q_{T+\Delta T}} = R + (\overline{S} \overline{Q_T}) = R + \overline{(S + Q_T)}$$

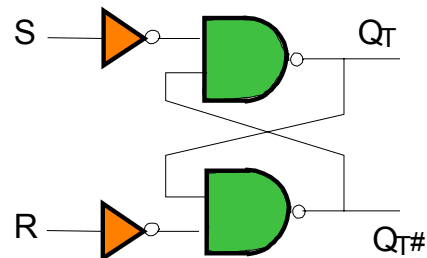
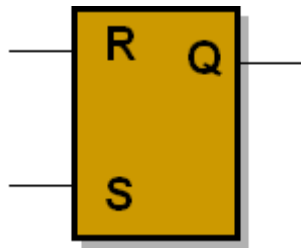


TABLA DE LA VERDAD

R	S	Q_T	$Q_{T+\Delta T}$	$Q_{T+\Delta T} \#$
0	0	0	0	1
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	1
1	0	1	0	1
1	1	0	1	1
1	1	1	1	1

TABLA SIMPLIFICADA

R	S	$Q_{T+\Delta T}$	$Q_{T+\Delta T} \#$
0	0	Q_T	$Q_T \#$
0	1	1	0
1	0	0	1
1	1	1	1

COMBINACIÓN PROHIBIDA.

- Para este circuito, la combinación prohibida es la de las dos entradas a 1, (este FF también es llamado por esta razón, FF de escritura prioritaria, a diferencia del FF NOR donde la

combinación prohibida pone las dos salidas a 0 y es llamado FF de borrado prioritario). El efecto sobre el circuito es el mismo que el expuesto para el FF NOR anterior.

R	S	Q_T	$Q_{T+\Delta T}$
X	0	0	0
0	1	0	1
1	0	1	0
0	X	1	1

TABLA DE TRANSICIÓN

- La tabla de transición es la misma que para el FF NOR, al no tener en cuenta la combinación prohibida en ninguno de los dos casos.
- Desde el punto de vista funcional, los dos circuitos tienen el mismo

comportamiento exceptuando el caso de la combinación prohibida.

FF NAND ACTIVO A NIVEL BAJO

$$Q_{T+\Delta T} = S \overline{Q_T}$$

$$\overline{Q_T} = R Q_T$$

$$Q_{T+\Delta T} = S R Q_T = S + (R Q_T)$$

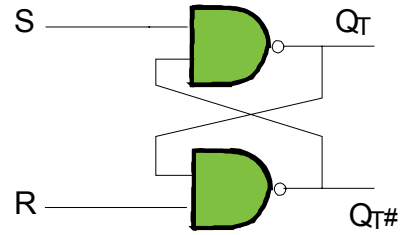
$$Q_{T+\Delta T} = \overline{S} + (R Q_T)$$

$$\overline{Q_{T+\Delta T}} = R Q_T$$

$$Q_T = S \overline{Q_T}$$

$$\overline{Q_{T+\Delta T}} = R S \overline{Q_T} = \overline{R} + (S \overline{Q_T})$$

$$\overline{Q_{T+\Delta T}} = \overline{R} + (S \overline{Q_T})$$

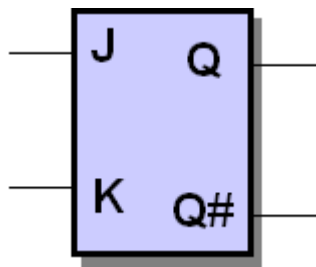


R	S	Q _T	Q _{T+ΔT}	$\overline{Q_{T+\Delta T}}$
0	0	0	1	1
0	0	1	1	1
0	1	0	0	1
0	1	1	0	1
1	0	0	1	0
1	0	1	1	0
1	1	0	0	1
1	1	1	1	0

R	S	Q _{T+ΔT}	$\overline{Q_{T+\Delta T}}$
0	0	1	1
0	1	0	1
1	0	1	0
1	1	Q _T	Q _T #

- Partiendo del hecho de que las letras R y S vienen a significar RESET (puesta a cero, puesta a inactivo) y SET (puesta a 1, puesta a activo) respectivamente, se ve en la tabla de la verdad que esto se cumple cuando los valores que se aplican a las entradas R y S son niveles bajos.

4.1.2-FF ASÍNCRONOS JK.



- En este tipo de flip-flop, se ha eliminado la combinación prohibida, de tal manera que cuando se dé la combinación que en los anteriores flip-flops se definía como prohibida, este circuito responderá con una salida definida, según la tabla de la verdad dada.

J	K	$Q_{T+\Delta T}$	$Q_{T+\Delta T}^\#$
0	0	Q_T	$Q_T^\#$
0	1	0	1
1	0	1	0
1	1	$Q_T^\#$	Q_T

El flip-flop JK se realiza a partir de un FF tipo RS, según el esquema de la figura. Se debe por lo tanto hacer una tabla de la verdad, para determinar el circuito combinacional cuyas salidas sean las entradas del RS. Hay que observar que este circuito tendrá una entrada que será la salida del circuito total, cumpliendo así la definición de circuito secuencial.

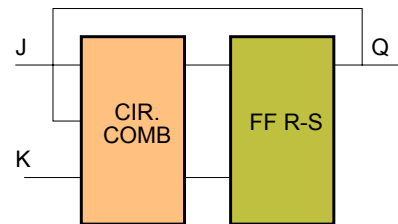


Tabla de transición RS

R	S	$Q_T \rightarrow Q_{T+\Delta T}$
X	0	0 $\rightarrow$ 0
0	1	0 $\rightarrow$ 1
1	0	1 $\rightarrow$ 0
0	X	1 $\rightarrow$ 1

- La tabla se obtiene considerando qué salida se desea en el nuevo circuito (FF JK) según las entradas al mismo (entradas al circuito combinacional a diseñar), y observando qué valores deben tener R y S para que lo anterior se cumpla.

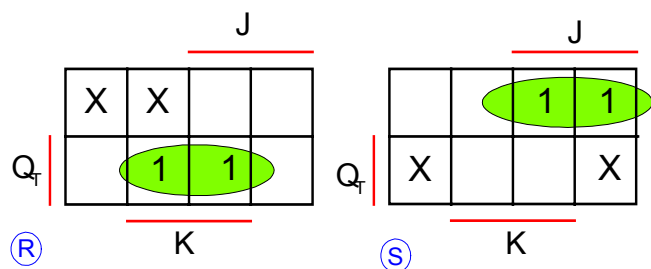
J	K	Q_T	R	S	$Q_{T+\Delta T}$	$Q_{T+\Delta T}^\#$
0	0	0	X	0	0	1
0	0	1	0	X	1	0
0	1	0	X	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	1	0
1	0	1	0	X	1	0
1	1	0	0	1	1	0
1	1	1	1	0	0	1

$$R = \sum_3 (3, 7) + d(0, 2)$$

$$S = \sum_3 (4, 6) + d(1, 5)$$

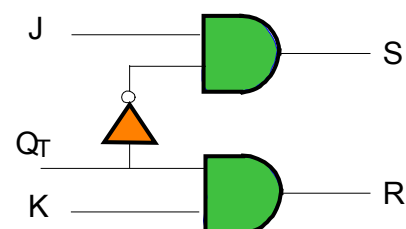
$$R = K Q_T$$

$$S = J \overline{Q_T}$$

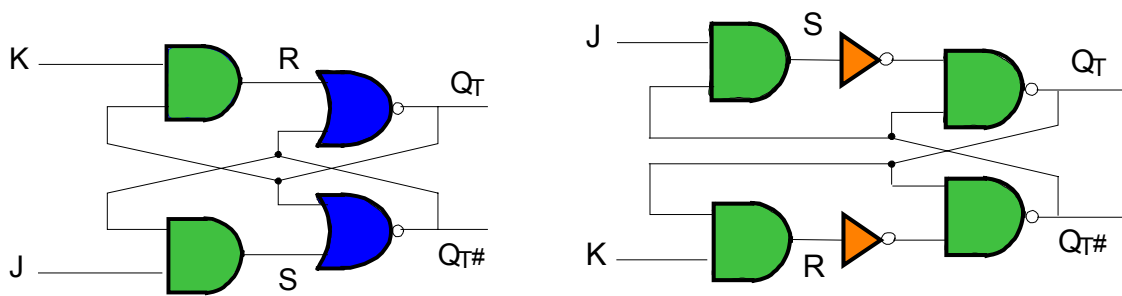


El circuito combinacional que se debía determinar, será por lo tanto:

Este circuito, una vez acoplado a los FF RS NOR o NAND resultará en un FF JK, el cual, no hará distinción funcional respecto al tipo de FF del cual partía.



El circuito completo con las básculas R-S realizadas a partir de puertas NOR y puertas NAND son por lo tanto:



Observar que este FF puede implementarse con cuatro NAND

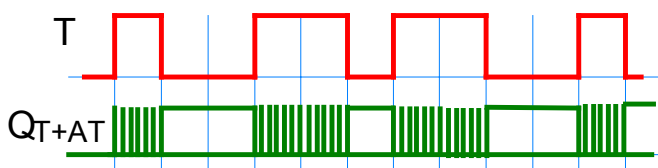
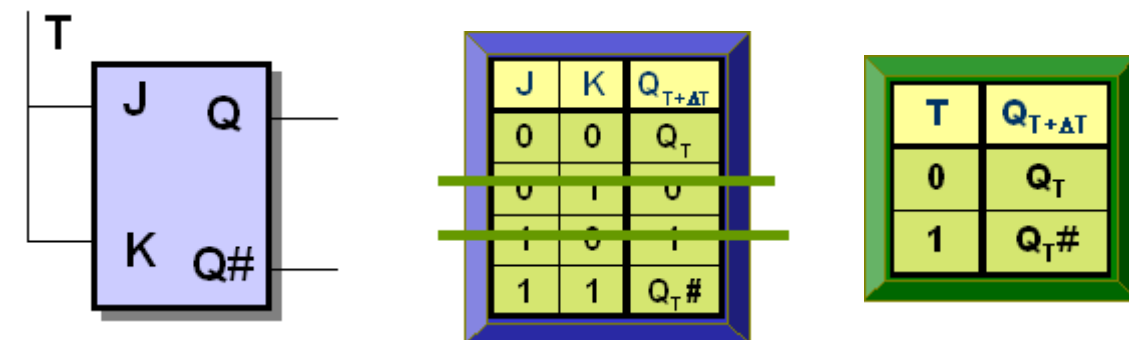
Las tablas de la verdad y la de transición del FF JK son las siguientes.

J	K	Q_T	$Q_{T+\Delta T}$	$Q_{T+\Delta T}^\#$
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	0	1
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	0	1

J	K	Q_T	$Q_{T+\Delta T}$
0	X	0	0
1	X	0	1
X	1	1	0
X	0	1	1

FF ASÍNCRONOS. T (TOGGLE)

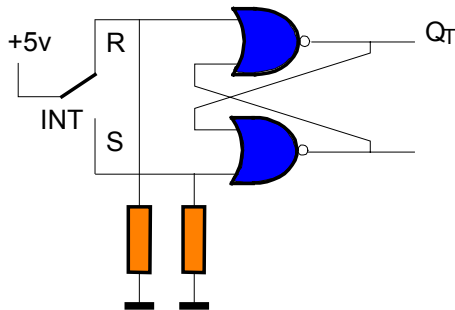
- Este FF parte de un JK en el cual las dos entradas están unidas.



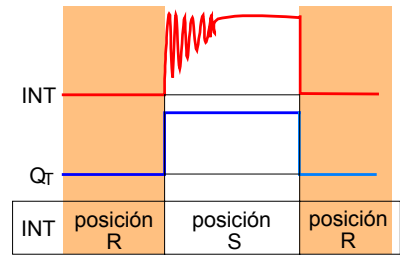
Como se deduce de la tabla de la verdad, cuando la entrada T vale 0, la salida permanece sin alterarse, pero si la entrada vale 1, la salida báscula hacia el valor complementario. En la gráfica se muestra este hecho. Cuando la entrada vale 1, la salida báscula continuamente. La frecuencia de oscilación va a depender del tiempo de retardo en la propagación del circuito. Cuando la entrada vale 0, la salida permanece en el último estado final de la oscilación. En la figura se representan los dos valores 0 y 1.

4.1.3-CONMUTADOR SIN REBOTES

- Cuando se acciona un conmutador mecánico, el cierre del circuito, no se realiza instantáneamente, es decir, el contacto golpea una vez y se separa de su posición de reposo, vuelve a ella y vuelve a separarse (rebota),....Una acción de este tipo en un circuito secuencial sería visto como una serie de impulsos, cuando



realmente lo que se pretende es un cambio de estado. Para evitar esto, los conmutadores mecánicos van conectados a través de un circuito antirrebotes, que lo que hace es eliminar esos estados transitorios del cierre del contacto mecánico. El circuito representa un ejemplo de un antirrebotes, para el caso de un FF tipo NOR. La forma de onda muestra la acción del conmutador sobre la posición S.

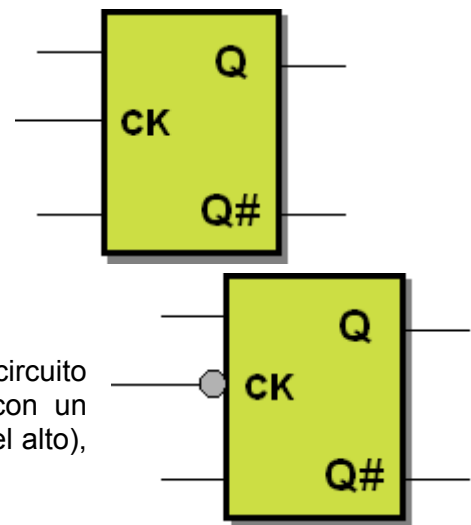
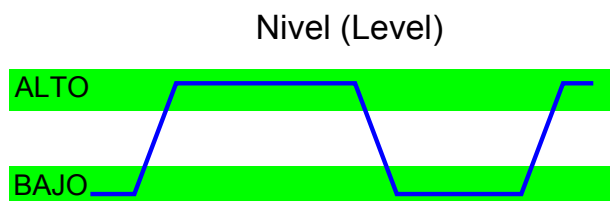


onda muestra la acción del conmutador sobre la posición S.

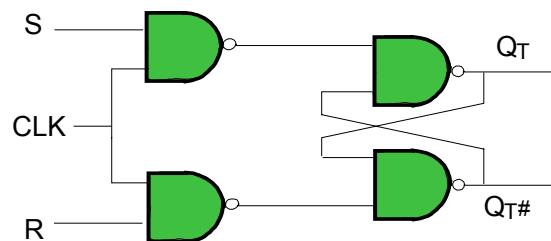
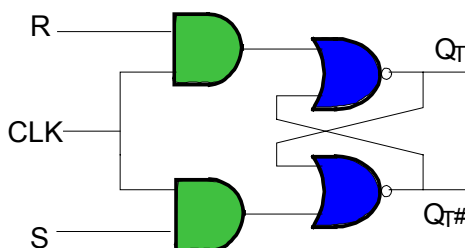
4.1.4-FF SÍNCRONOS

- El sincronismo de estos circuitos puede ser realizado atendiendo al nivel (alto o bajo) de la señal de sincronismo (clock) o atendiendo al flanco (ascendente o descendente) de esa señal.

-ACTIVADOS POR NIVEL.

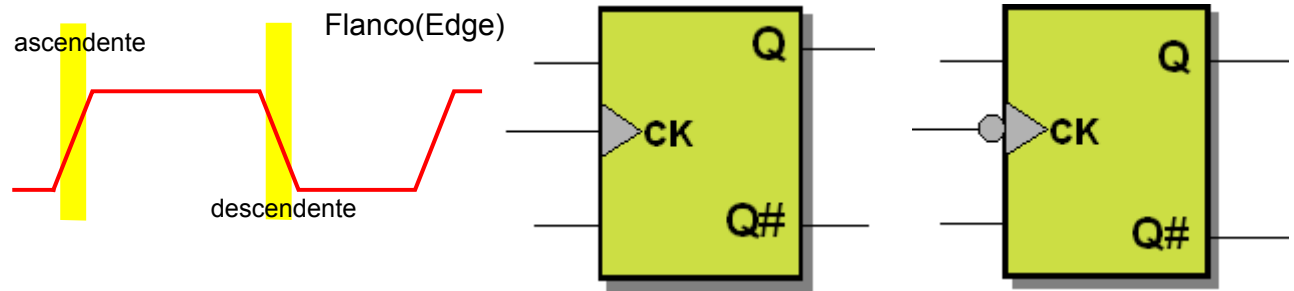


- Para indicar este tipo de sincronismo en un circuito determinado, el terminal de reloj (CK) se dibuja con un círculo (activo a nivel bajo) o sin círculo (activo a nivel alto), según figura.
- El circuito que hace que el reloj sea activo a un nivel u otro, se basa en el paso y bloqueo de entradas en puertas NAND y AND, según se representa en los esquemas siguientes.



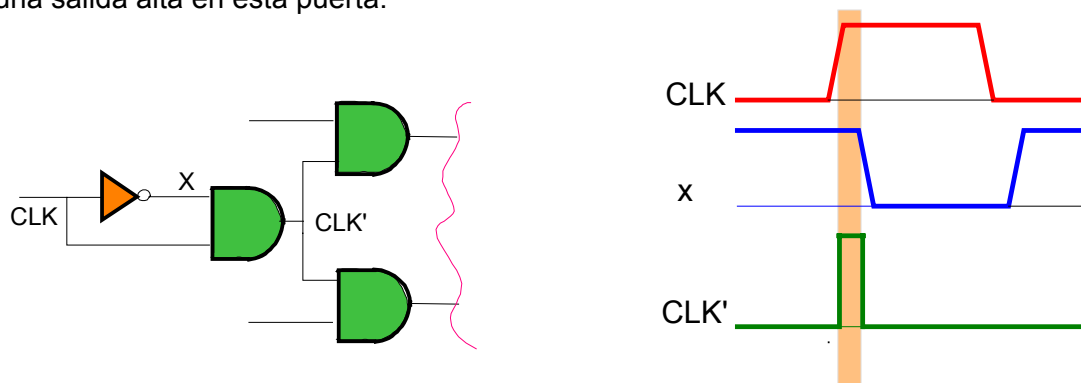
FF RS ACTIVADOS POR NIVEL

-ACTIVADOS POR FLANCO.

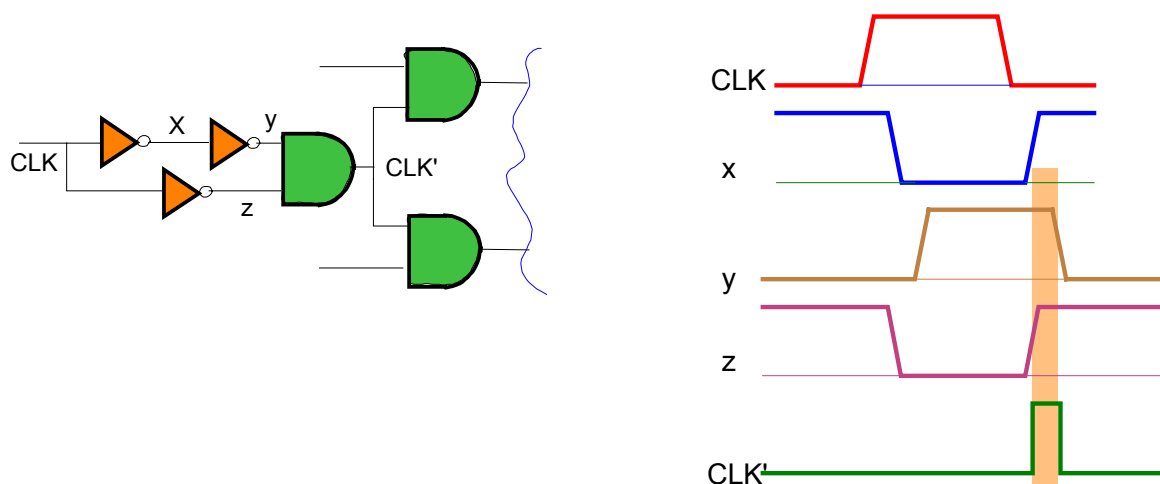


- Para indicar este tipo de sincronismo en un circuito determinado, el terminal de reloj se dibuja a través de un triángulo, con un círculo (activo a flanco de bajada) o triángulo sin círculo (activo a flanco de subida), según figura.
- El circuito que hace que el reloj sea activo a flanco ascendente o flanco descendente, se basa en el retardo que producen circuitos inversores, en una entrada de una puerta AND, respecto a la otra entrada.

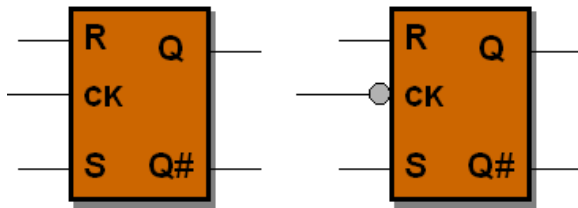
Para el caso de que el reloj sea activo al flanco ascendente de la señal de reloj, el inversor colocado en una de las entradas de la puerta AND, hace que durante el tiempo de retardo de dicho inversor, las dos entradas de la AND tengan un nivel alto, dando como resultado una salida alta en esta puerta.



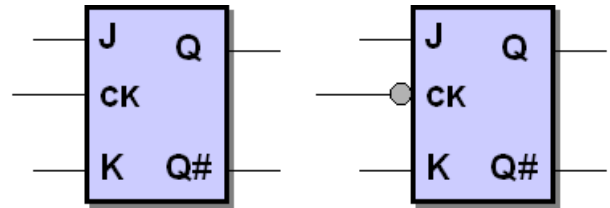
Para el caso de reloj activo al flanco descendente, el principio de funcionamiento es análogo al del caso anterior, trabajando ahora con tres inversores según la figura, para provocar el doble de retardo en una entrada de la puerta AND, respecto de la otra entrada.



RS ACTIVADO POR NIVEL

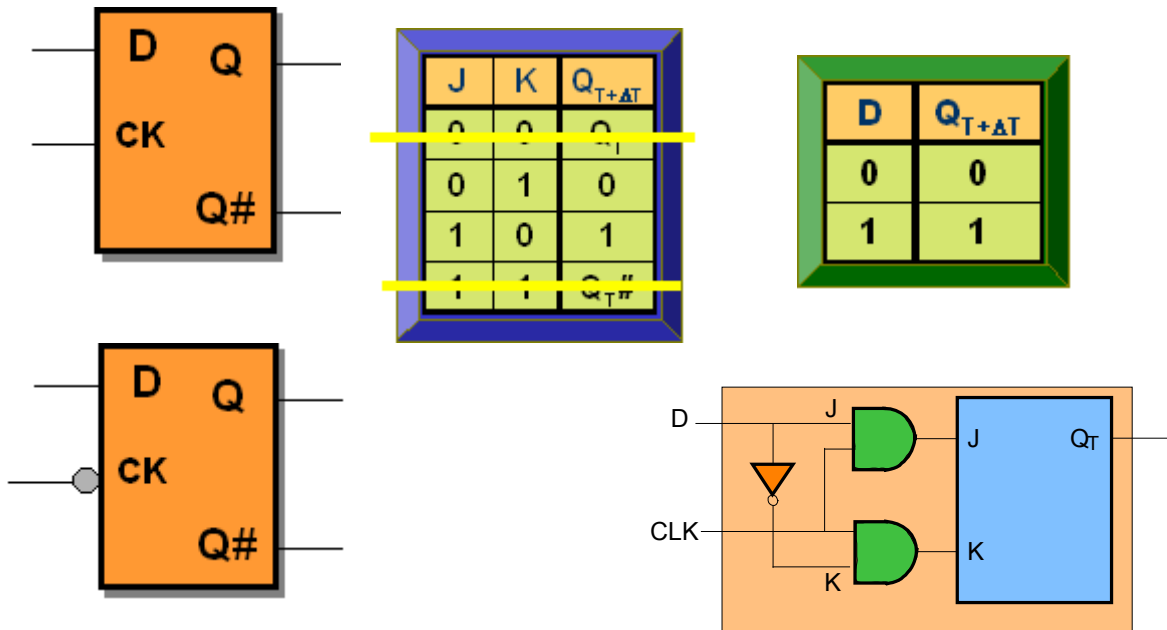


JK ACTIVADO POR NIVEL



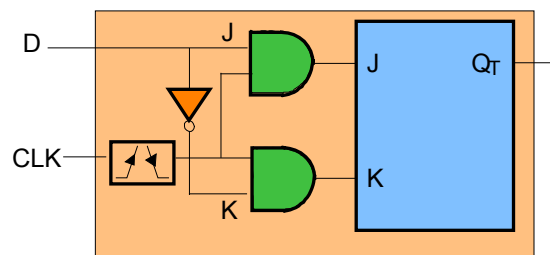
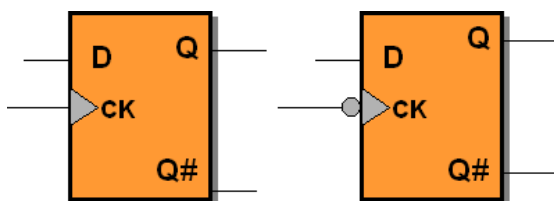
D ACTIVADO POR NIVEL (LATCH)

- Estos FF pueden ser realizados a partir de biestables RS o JK.



- La información en la salida del FF tipo D es la misma que hay en la entrada, pero en el instante que la señal de sincronismo lo permita, es decir, el reloj es el que dice cuando la señal de entrada pasará a la salida.

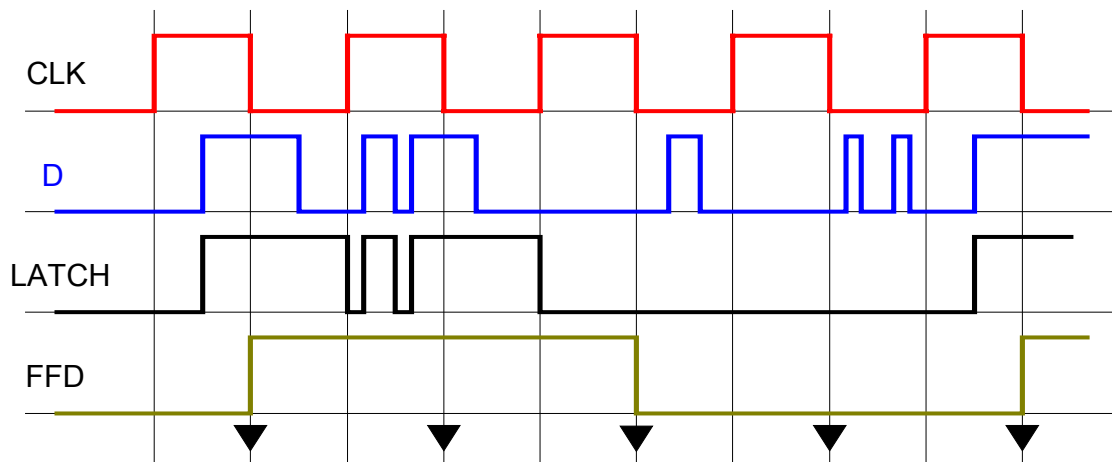
D ACTIVADO POR FLANCO



La figura de la derecha, representa un FF tipo D obtenido a partir de un biestable tipo JK. El rectángulo con las flechas representa un detector de flancos, bien ascendente o descendente.

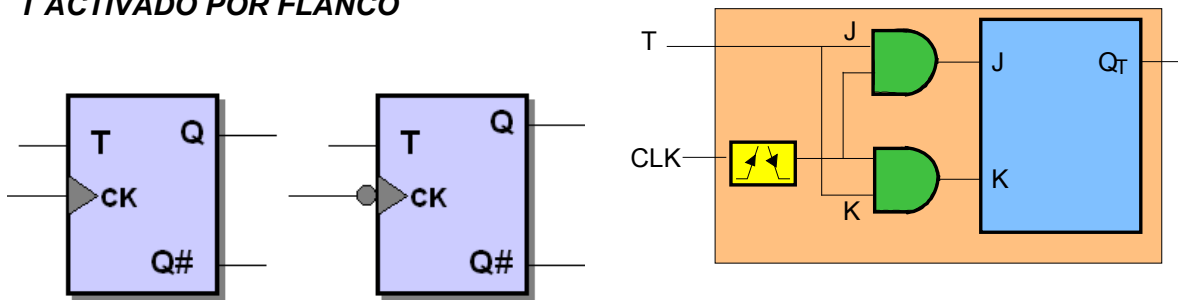
La diferencia entre este FF y el Latch está en el tiempo durante el cual la entrada va a tener un efecto en la salida. Generalmente, la activación en un nivel determinado se prolonga durante mas tiempo que la activación por flanco (los flancos tienen una menor duración que los niveles), por lo tanto, cabe la posibilidad de que cambios que se estén realizando en la entrada D, estén afectando en el mismo instante (teniendo en cuenta el retardo correspondiente) a la salida. Esta posibilidad cabe también para los FF activados por flanco, pero en menor grado. Esto se ilustra en la siguiente figura.

Respuesta de un FF tipo D y un Latch ante la misma señal de entrada en D.



Aquí se supone que el Latch es activo al nivel alto y que el FF tipo D es activo a los flancos descendentes de la señal de reloj.

T ACTIVADO POR FLANCO



- Estos FF se realizan a partir de FF JK síncronos, uniendo las entradas. Las figuras representan FF-s tipo T activados por flanco.

J	K	$Q_{T+\Delta T}$
0	0	Q_T
0	1	0
1	0	1
1	1	$Q_T\#$

T	$Q_{T+\Delta T}$
0	Q_T
1	$Q_T\#$

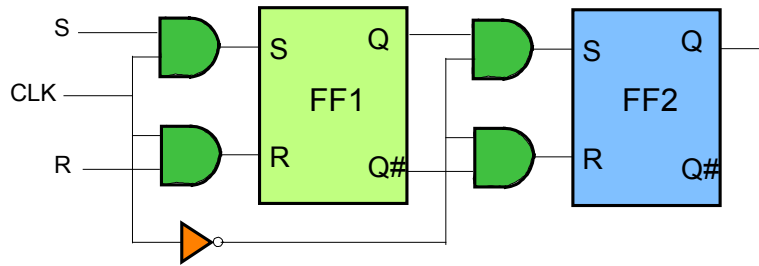
Con la entrada a nivel bajo, el reloj no provoca cambios en la salida. Con la entrada a nivel alto, hay un cambio en la salida por cada transición activa de la señal de reloj.

FF MASTER/SLAVE (MAESTRO/ESCLAVO)

- Aunque realmente su forma de funcionamiento interna, no corresponde a un FF que actúa según los flancos de la señal de reloj, desde el punto de vista funcional externo, sí que corresponde a un FF activo a los flancos.

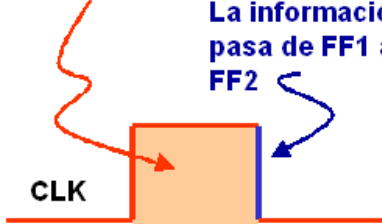
Los datos presentes en las entradas del FF, no afectan a la salida hasta que el reloj pasa a su estado bajo, en ese instante, los datos que están en el FF1 pasan al FF2. En el flanco ascendente, o nivel alto, los datos pasan de las entradas al FF1.

- Desde el punto de vista externo, el circuito se comporta como un FF activado a los flancos de bajada de la señal de reloj.



La información
pasa desde la
entrada a FF1

La información
pasa de FF1 a
FF2



- Antes de la aparición de los detectores de flancos, este era el método usado en los FF activados por flanco. Hoy en día esta prácticamente obsoleto.

4.1.5-ENTRADAS SÍNCRONAS Y ASÍNCRONAS.

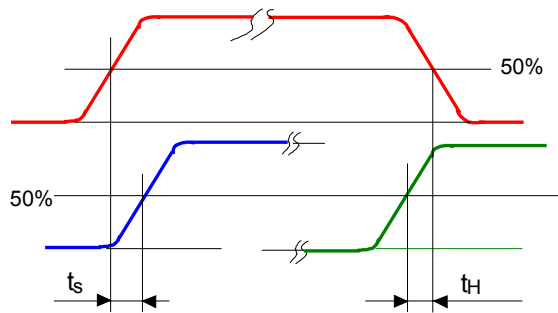


- En los FF sincronizados por clock, las entradas que se han tratado hasta ahora, no tienen su efecto en la salida hasta que no lo “permite” el clock, es decir, todo el proceso está sincronizado. A estas entradas se las llama entradas *síncronas* por su dependencia directa del reloj. Existen otras entradas que actúan sobre la salida, independientemente del estado del clock y de las entradas síncronas. A estas entradas se las llama *asíncronas* y son la entrada de *SET* o *PRESET* que ponen la báscula en estado alto cuando son activadas y la entrada de *CLR* o *RESET* que la ponen a estado bajo, también si son activadas. La tabla de la verdad para un circuito con este tipo de entradas corresponde a la dada en la figura superior.

Para el caso de que ambas entradas asíncronas estén activas al mismo tiempo, cada circuito tiene su propio modo de respuesta. En la tabla superior, no se indica ninguna acción en particular para este caso.

4.1.6-TIMMING EN FLIP-FLOPS.

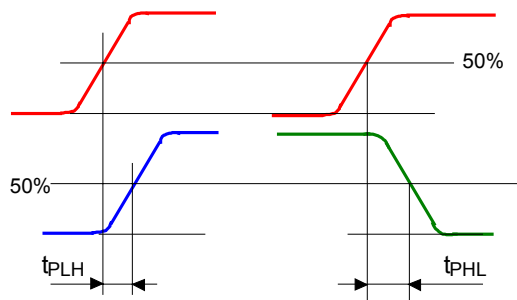
□ TIEMPOS DE ESTABLECIMIENTO Y RETENCIÓN (t_s y t_H)



Las entradas síncronas deben permanecer estables un tiempo mínimo t_s (t_{sample}) antes y un tiempo mínimo t_H (t_{hold}) después, de que se produzca la transición activa de la señal de reloj. Si estos tiempos no se cumplen, no se garantiza que el circuito actúe acorde con el valor de las entradas.

La figura representa una transición hacia un estado lógico alto de una entrada (señal superior). Esta ocurre un tiempo t_s antes de la transición activa de la señal de reloj (en este caso se supone que el circuito es activo a los flancos ascendentes de la señal de reloj) y vemos que permanece en el mismo estado un tiempo t_H después de que el reloj haya evolucionado en su estado activo.

□ RETARDOS EN LA PROPAGACIÓN (t_{PLH} , t_{PHL})

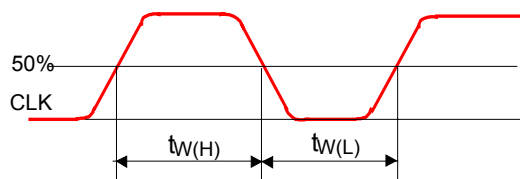


Estos valores especifican el máximo tiempo de retardo en la respuesta de un circuito a una excitación en la entrada del mismo. Si la respuesta debe ser una transición de nivel bajo a nivel alto, el tiempo de propagación viene dado como t_{PLH} . Si la respuesta es de nivel alto a bajo el tiempo será t_{PHL} .

□ FRECUENCIA MÁXIMA DE LA SEÑAL DE RELOJ (f_{MAX})

El fabricante, especifica la máxima frecuencia a la cual todos los dispositivos de la familia en cuestión funcionan correctamente cumpliendo su función.

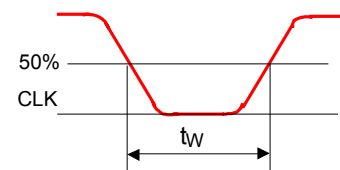
□ TIEMPOS EN ESTADO ALTO Y BAJO DE LA SEÑAL DE RELOJ ($t_{W(L)}$ y $t_{W(H)}$)



Valores mínimos de tiempo que debe estar el reloj en estado alto $t_{W(H)}$, y en estado bajo $t_{W(L)}$, para garantizar disparos fiables.

□ ANCHURA DEL PULSO ASÍNCRONO (PRE, SET, CLR, RESET) ACTIVO (t_W)

Tiempo mínimo de las entradas asíncronas para iniciar o borrar el FF. Si no se respeta este tiempo mínimo no queda garantizado que el circuito pueda iniciar o borrar la báscula.



□ TIEMPO DE TRANSICIÓN DE RELOJ (FLANCOS)

Estos tiempos no deben sobrepasar un tiempo máximo para que no se produzcan disparos erróneos. Se suele dar el tiempo máximo para toda la familia Ej: TTL <30ns; CMOS <150 ns

4.2-REGISTROS DE DESPLAZAMIENTO O ALMACENAMIENTO (SHIFT REGISTERS)

- ❑ Los registros son grupos de FF que sirven para almacenar bits (almacenamiento de información). Cada FF almacena un BIT (dos estados) por lo tanto, el número de FF empleados en la realización del registro determina la capacidad de registro.
- ❑ También sirven para mover o desplazar bits (transferencia de información). Clasificación:

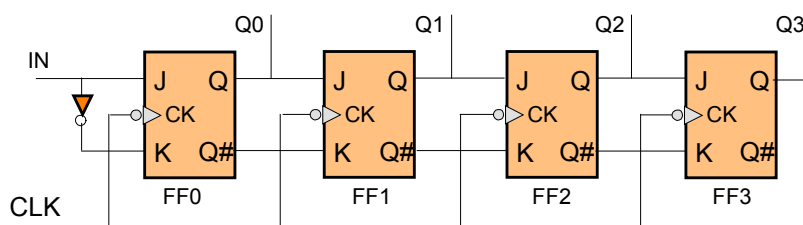
SEGÚN LA CARGA DEL REGISTRO	CARGA SERIE (SERIAL IN)
	CARGA PARALELO (PARALLEL ACCES)
SEGÚN LA LECTURA DEL REGISTRO	LECTURA SERIE (SERIAL OUT)
	LECTURA PARALELO (PARALLEL OUT)
SEGÚN EL DESPLAZAMIENTO DE LOS BITS	DIRECCIONAL
	BIDIRECCIONAL
	ALMACENAMIENTO

Algunas combinaciones comerciales de tipos de registros son:

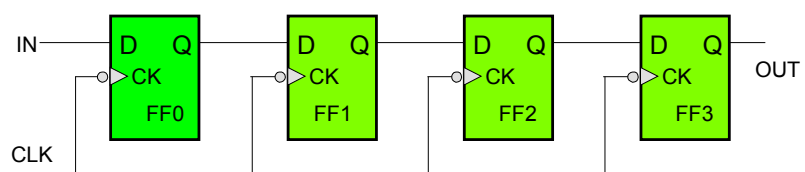
Serial-in serial-out shift register, serial-in parallel out shift register, parallel-in parallel-out shift register, bidirectional shift register, universal shift /storage register.

4.2.1-REGISTRO DE CARGA SERIE Y LECTURA SERIE.

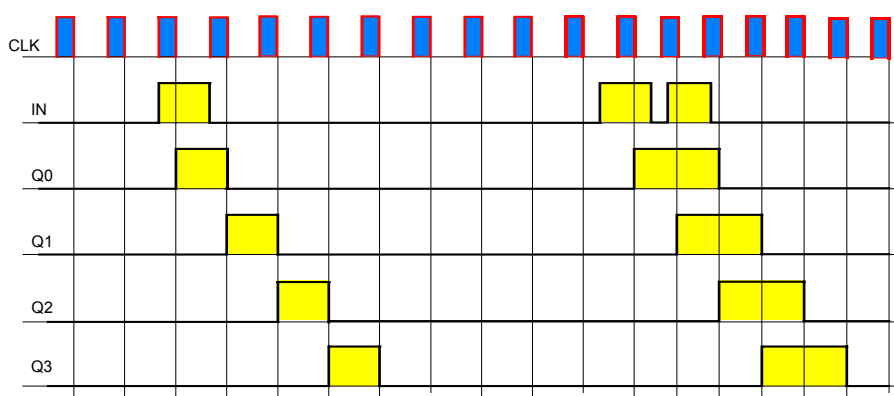
Cada impulso de reloj hace que la información en la entrada IN de FF0, pase a la salida del mismo; lo que hay en la salida del FF0 pase a la salida del FF1, lo que hay en la salida de este a la salida del FF2 etc. Para este caso necesitamos cuatro impulsos de reloj para cargar el registro.



Este registro también se puede representar:



FORMAS DE ONDA.



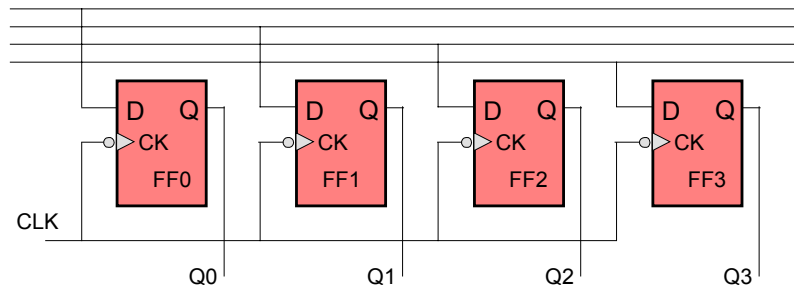
Se puede observar que en el tercer impulso (flanco descendente) del reloj, en la entrada IN del FF0 aparece un uno lógico, por lo tanto ese valor pasará a la salida de este FF.

Hay que tener en cuenta que aunque parece que en ese mismo instante del flanco descendente del tercer impulso de reloj, Q0 parece tener un uno lógico, eso no es así, pues ese uno lógico aparece en la salida de Q0 precisamente debido a ese flanco descendente, el cual afecta a todos los FF al mismo tiempo, por lo que en el instante en que FF1 estaba siendo activado, la salida del FF0 todavía tenía un cero lógico en su salida. Esto es así para todos los flancos activos de la señal de reloj.

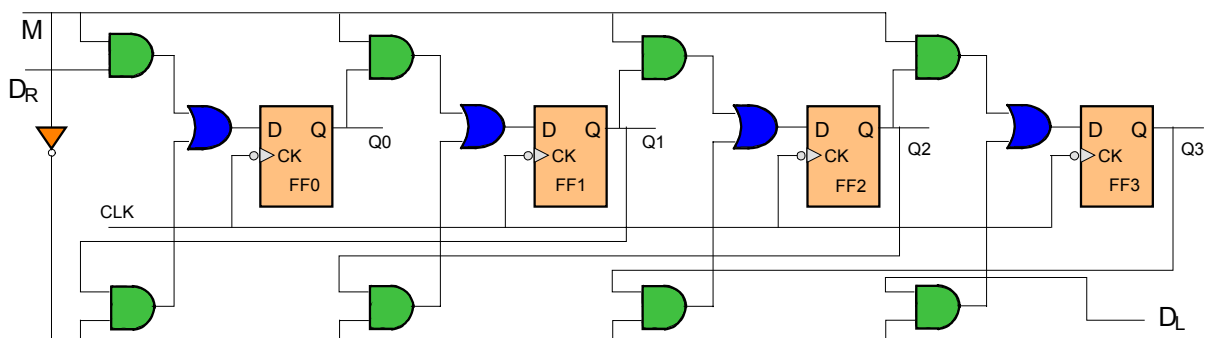
- También se puede observar que a la entrada del FF0 se aplica en un momento dado dos impulsos, sin embargo el registro los “ve” como uno de más larga duración. Esto se debe a que en el momento en que llega la transición activa de la señal de reloj, el valor en la entrada del registro es un valor alto, y ese es el valor que se va a introducir, el que está presente en la entrada en el momento de la transición activa. Lo que suceda entre una y otra transición es irrelevante para el registro.

4.2.2-REGISTRO DE CARGA PARALELO Y LECTURA PARALELO.

- En este registro, con una transición activa de la señal de reloj basta para cargar todo el registro. Esta información permanecerá almacenada hasta la siguiente transición activa del clock.



4.2.3-REGISTRO DE DESPLAZAMIENTO BIDIRECCIONAL, CARGA SERIE Y LECTURA SERIE O PARALELO.



D_R es la entrada de datos para desplazamiento de los bits hacia la derecha. D_L es la entrada de bits para desplazamiento a la izquierda. M es el modo de operación, el cual define el sentido del desplazamiento de la información; con $M=0$ el desplazamiento es hacia la izquierda, por lo tanto para este caso la entrada sería D_L . Para $M=1$, el desplazamiento sería hacia la derecha por lo tanto la entrada de los bits sería por D_R . El funcionamiento se basa en el paso y bloqueo de entradas de una puerta AND.

4.3-CONTADORES (COUNTERS)

4.3.1-MÓDULO DE UN CONTADOR.

- ❑ El módulo de un contador es el número de estados diferentes que se pueden dar en el mismo. Cuando el número de estados de un contador es 2^n , siendo n un número natural, se dice que el contador tiene un módulo binario.
- ❑ Los contadores se realizan a partir de FF-s. Teniendo en cuenta que un FF es de módulo 2 (0 y 1; dos estados), la relación entre el módulo del contador y el número de FF necesarios n , es:

$$\text{módulo} \leq 2^n; \log \text{módulo} \leq n \log 2; \quad n \geq \log \text{módulo} / \log 2.$$

n debe ser un número entero positivo, por lo tanto cuando el resultado es fraccionario, se tomará el número entero inmediatamente superior.

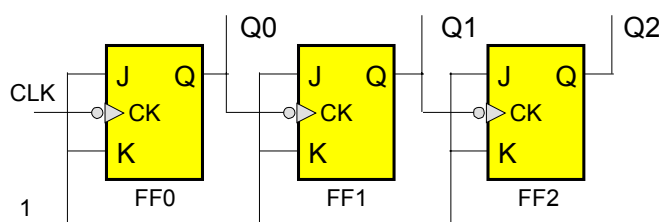
4.3.2-CONTADORES ASÍNCRONOS O DE RIZO. MÓDULO BINARIO.

- ❑ En este tipo de contadores, la señal de reloj (sincronismo) no se aplica de forma simultanea a todos los FF que componen el circuito.
- ❑ Para poder realizar estos contadores, necesitamos según el módulo, los siguientes FF:

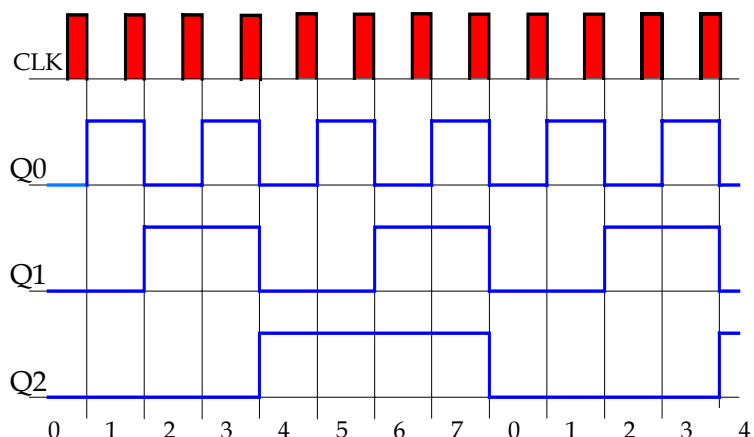
Mod 2	Mod 4	Mod 8	Mod 16	...	Mod 2ⁿ
1FF	2FF	3FF	4FF	...	n FF

- La conexión de los FF es tal que cada uno de ellos se convierte en un FF tipo T, es decir, las entradas J y K unidas. A cada entrada T se aplica un estado lógico alto. La señal reloj se aplica a la primera báscula (FF0). Para las restantes básculas del contador, la señal de reloj proviene de la salida de la báscula precedente.

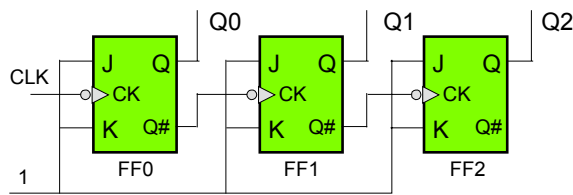
CONTADOR ASCENDENTE MÓDULO 8.



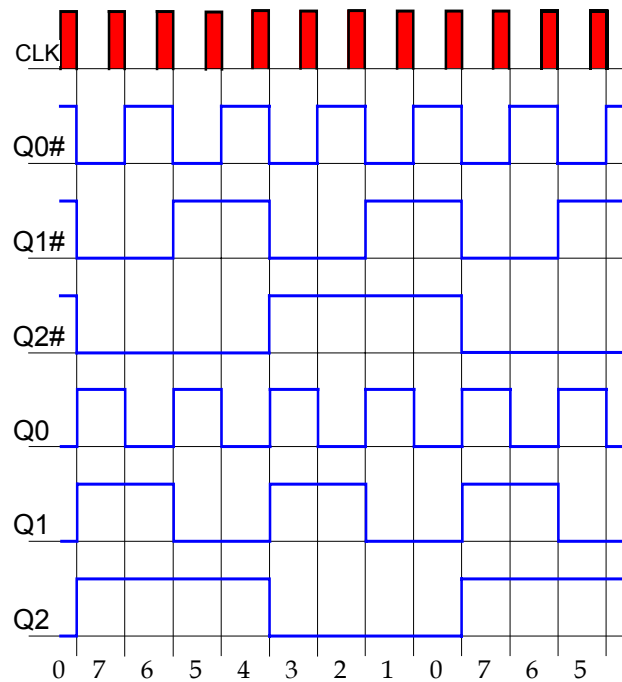
- El reloj es activo a los flancos de bajada, por lo tanto cualquier posible cambio en el valor de la salida de las básculas, solo podrá suceder en los flancos descendentes aplicados a las entradas de reloj de las respectivas básculas.



CONTADOR DESCENDENTE MÓDULO 8.



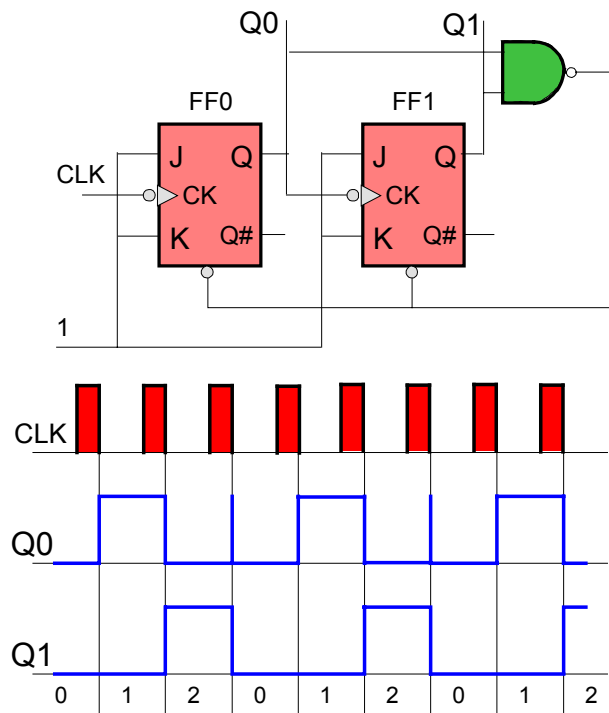
En este caso, la entrada de reloj de la FF1 y FF2 proviene de las salidas complementarias de la FF0 y FF1 respectivamente.



4.3.3-CONTADORES ASÍNCRONOS O DE RIZO. MÓDULO NO BINARIO.

- Para conseguir módulos no binarios, se debe actuar sobre el CLR# de las básculas. Cuando el contador alcance la cuenta deseada, según el módulo, las salidas del mismo, deben actuar sobre la entrada asíncrona de puesta a cero de los FF, con lo cual, a partir de ese momento, el contador comienza la cuenta.

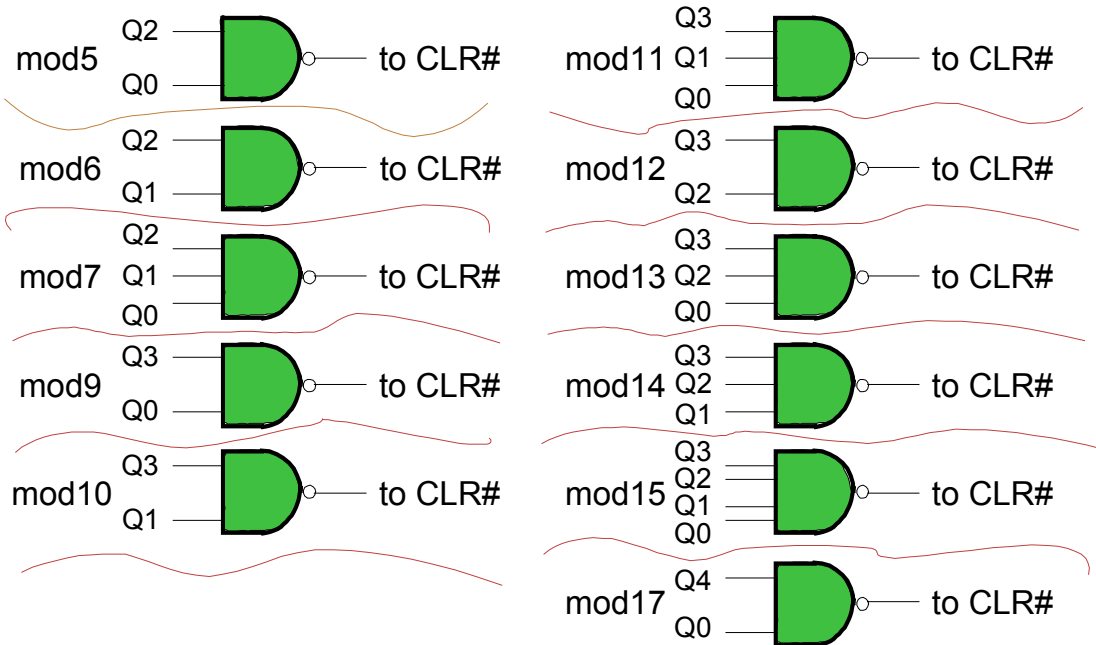
Q ₄	Q ₃	Q ₂	Q ₁	Q ₀	
0	0	0	0	0	0
0	0	0	0	1	1
0	0	0	1	0	2
0	0	0	1	1	3
0	0	1	0	0	4
0	0	1	0	1	5
0	0	1	1	0	6
0	0	1	1	1	7
0	1	0	0	0	8
0	1	0	0	1	9
0	1	0	1	0	10
0	1	0	1	1	11
0	1	1	0	0	12
0	1	1	0	1	13
0	1	1	1	0	14
0	1	1	1	1	15
1	0	0	0	0	16
1	0	0	0	1	17



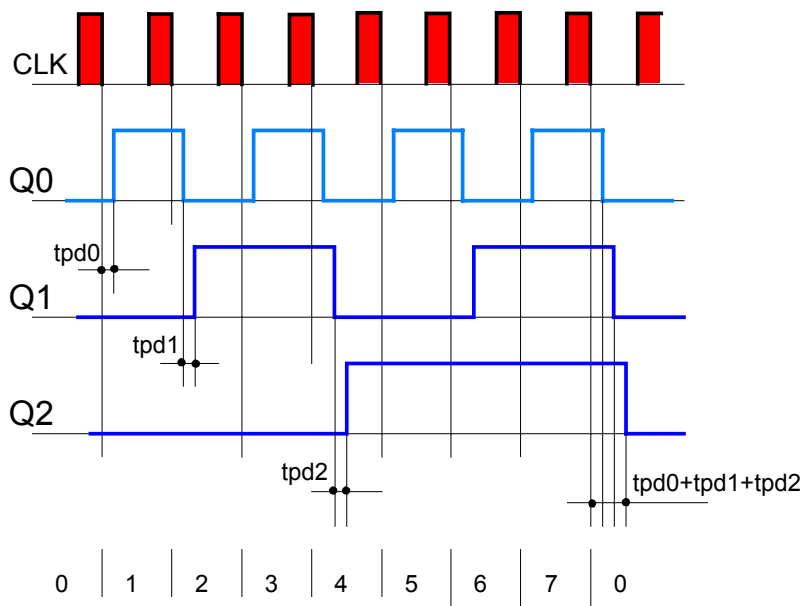
En este contador de módulo 3, (tres estados, 0, 1 y 2) vemos que cuando se alcanza el valor 3 ($Q_0=Q_1=1$), se activa la salida de la puerta NAND, de esta manera pone el contador a cero, iniciándose la cuenta de nuevo con la siguiente transición activa de la señal de reloj.

Hay que observar que de forma momentánea, aparece un estado mas en la cuenta, que los que corresponde al módulo del contador. Este estado es precisamente el que reinicia el contador.

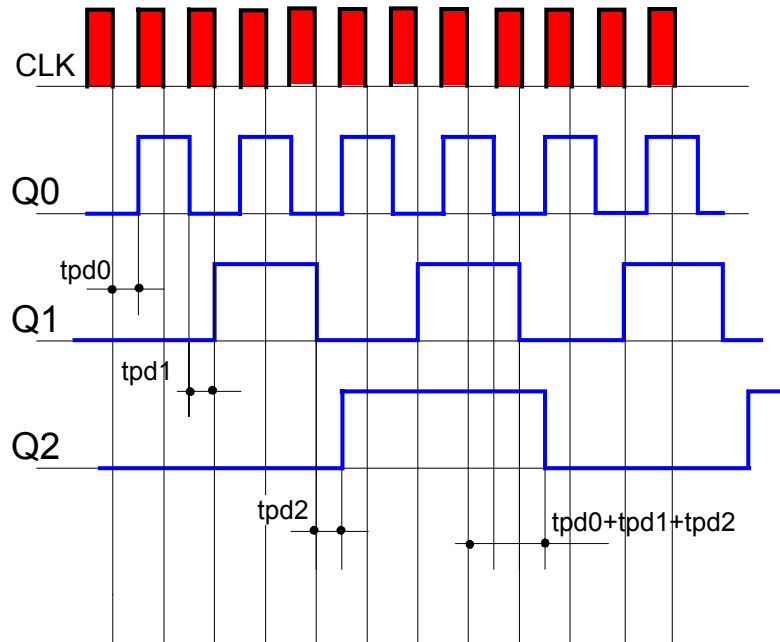
En la siguiente tabla se muestra las conexiones posibles a realizar sobre puertas NAND para obtener el módulo indicado.



TIEMPO DE PROPAGACIÓN Y FRECUENCIA



señal de reloj, estos tiempos de retardo pueden resultar despreciables. Pero si aumentamos la frecuencia del reloj (gráfica inferior), el periodo de la señal se hace más pequeño, mientras que los tiempos de retardo prácticamente se mantienen, no pudiendo despreciarse en este caso esos tiempos pues originan unos *estados transitorios* relativamente largos en el tiempo.



- Se debe tener muy en cuenta el tiempo de propagación de los FF a la hora de determinar la frecuencia de trabajo; esta siempre debe acomodarse al FF mas lento de todos los que componen un circuito determinado.

- En la figura puede verse como afecta la introducción de los tiempos de retardo de cada FF en un contador de módulo 8. Para una frecuencia relativamente baja (periodo relativamente alto) de la

Por ejemplo, se puede apreciar que para pasar del valor 011 (3) al valor 100 (4) la cuenta pasa por los siguientes estados transitorios 010-000-101. También se puede observar que cuanto menor es el periodo de la señal de reloj mas desordenada sale la cuenta.

Para evitar esto, el periodo de la señal de reloj ha de ser mayor que el retardo total del contador, es decir suma de los retardos individuales de cada FF que compone el circuito.

$$T_{CLK} > N t_{pd}$$



$$f_{CLK} < 1/(N t_{pd})$$

N= número de biestables de contador.
t_{pd} tiempo de propagación de uno de los FF.

4.3.4-CONTADORES SÍNCRONOS. MÓDULO BINARIO

- En este tipo de contadores, la señal de reloj (sincronismo) se aplica de forma simultanea a todos los FF que componen el circuito.

Q ₃	Q ₂	Q ₁	Q ₀
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

- Observando la tabla, se puede apreciar lo siguiente:

Q₀ debe cambiar de estado con cada transición activa de la señal de reloj lo cual implica que las entradas J y K deben estar conectadas a un uno lógico.

Q₁ debe cambiar de estado cuando se cumpla que Q₀=1 y hay una transición activa de la señal de reloj.

Q₂ debe cambiar cuando se cumple que Q₀=Q₁=1 y hay una transición activa de la señal de reloj.

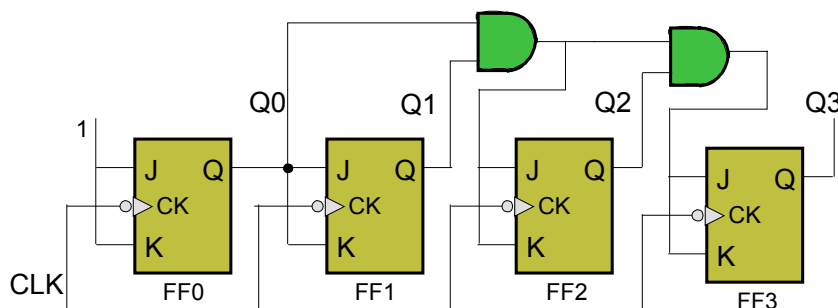
Q₃ cambia cuando se cumple que Q₀=Q₁= Q₂=1 y hay una transición activa del clock.

Q₄

La realización del circuito teniendo en cuenta lo anterior puede llevarse a cabo mediante una configuración de acarreo serie o configuración de acarreo paralelo (más rápido).

ACARREO SERIE

$$f_{MAX} = 1/[t_{PdFF} + t_{pd\&}(n-2)]$$



t_{PdFF} el retardo en la propagación de un FF

$t_{pd\&}$ retardo en la propagación de una puerta AND

n es el número de FF

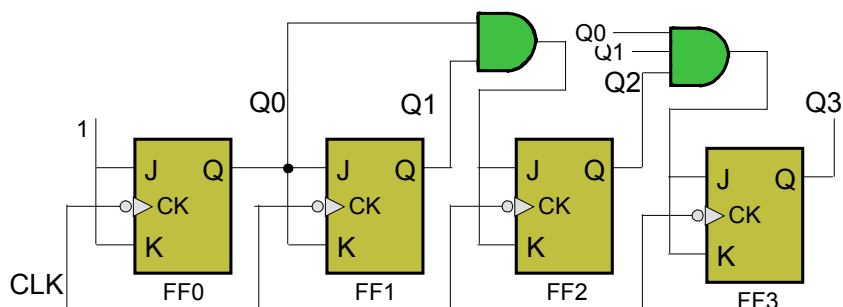
ACARREO PARALELO

$$f_{MAX} = 1/[t_{PdFF} + t_{pd\&}]$$

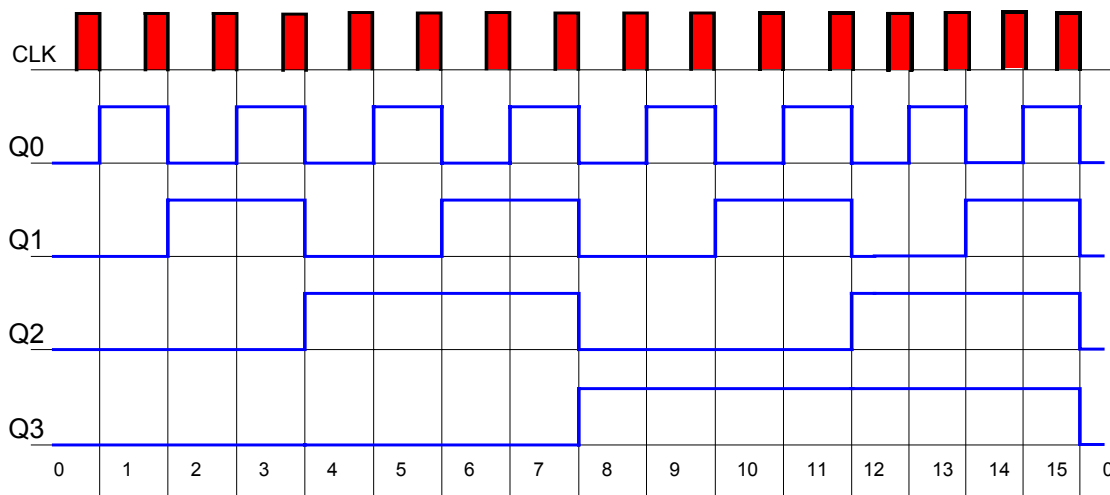
t_{PdFF} el retardo en la propagación de un FF

$t_{pd\&}$ retardo en la propagación de una puerta AND

n es el número de FF



ONDAS DE SALIDA.



4.3.5-CONTADORES SÍNCRONOS. MÓDULO NO BINARIO

J	K	Q_T	$Q_{T+\Delta T}$
0	X	0	0
1	X	0	1
X	1	1	0
X	0	1	1

- Para el diseño de estos contadores, partimos de la tabla de transición de los FF-s JK, pues los contadores serán realizados a partir de estas básculas.

CONTADOR SÍNCRONO MÓDULO 3.

- Para la realización de este contador, dado que tendrá tres estados, necesitamos 2 FF (FF_0 y FF_1)

estados	Q_1	Q_0	J_1	K_1	J_0	K_0
0	0	0	0	X	1	X
1	0	1	1	X	X	1
2	1	0	X	1	0	X
0	0	0	X	X	X	X

estados deseada. Esto nos lo dice la tabla de transición del FF JK.

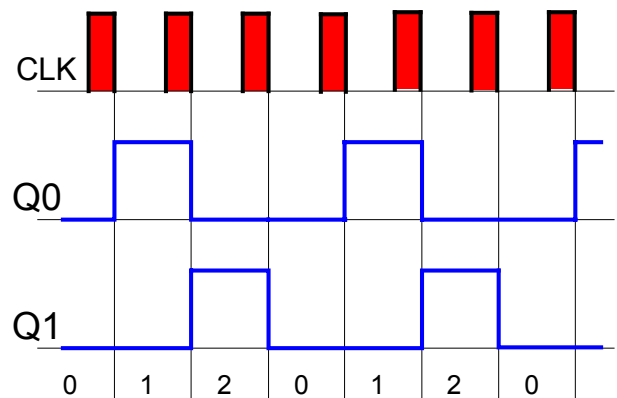
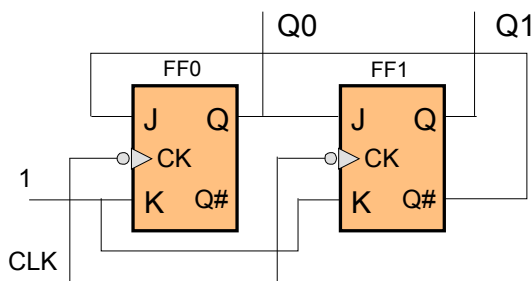
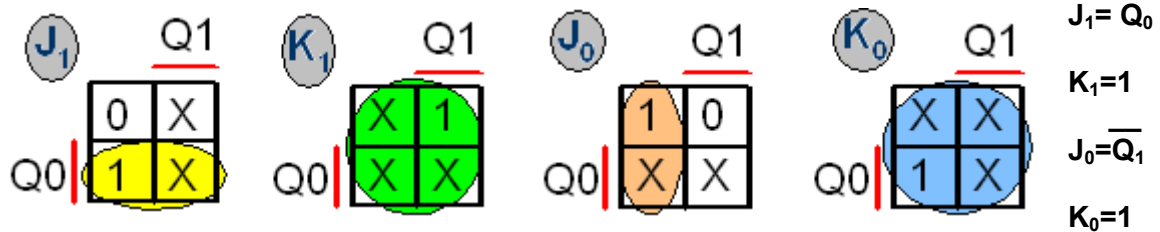
- El diseño parte de una tabla como la dada, en la cual se ponen todos los posibles estados que se van a dar en el contador, es decir todas las posibles salidas que van a darse en los FF_0 y FF_1 . Ahora lo único que necesitamos, es saber que condiciones tienen que cumplir la entrada J y la entrada K de cada báscula, para que las transiciones sean la que corresponden a la secuencia de

(Debemos observar que aquí partimos de los valores conocidos de la salida y debemos determinar las entradas que satisfagan esas salidas).

Por ejemplo, para determinar las condiciones que deben cumplir las entradas J_1 y K_1 para que su salida Q_1 cumpla que estando en 0, continúe en 0 (estado de 0 a 1), miramos en la tabla de transición y vemos que $J=0$ y $K=X$, valores que pasamos a la primera fila de la tabla en las columnas de J_1 y K_1 . Se pasa al siguiente estado (de 1 a 2), donde se ve que Q_1 estando en 0, debe pasar a 1. La tabla de transición nos dice que $J=1$ y $K=X$. Estos valores se pasan a la

segunda fila en las columnas de J_1 y K_1 . Esto hay que realizarlo para cada cambio de estado, y para cada uno de los FF que componen el contador.

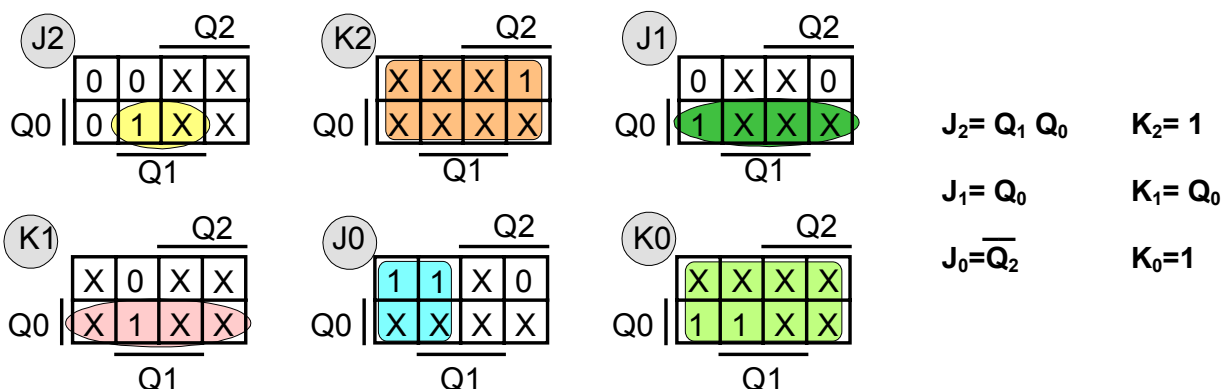
- Para simplificar debemos hacer uso de las combinaciones que no se llegarán a dar tomándolas como combinaciones "no importa". Para este caso la combinación $Q_1 = 1$ y $Q_0 = 1$ (3 decimal) no se dará.
- Se sacan los términos de la tabla de la verdad y se simplifican:

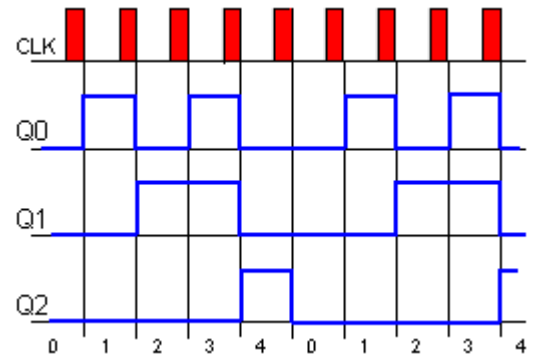
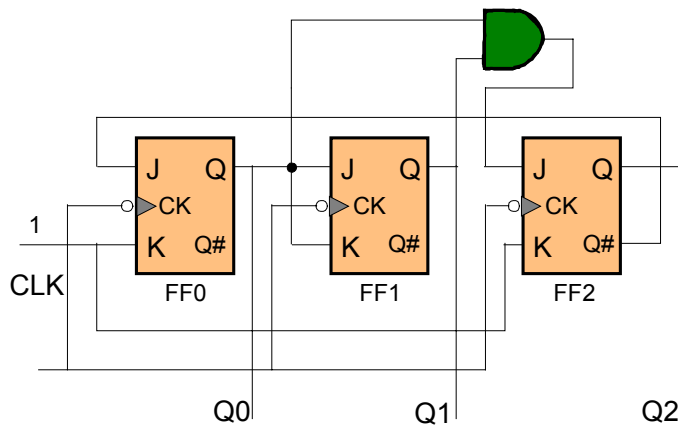


CONTADOR SÍNCRONO MÓDULO 5.

S	Q_2	Q_1	Q_0	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	0	X	0	X	1	X
1	0	0	1	0	X	1	X	X	1
2	0	1	0	0	X	X	0	1	X
3	0	1	1	1	X	X	1	X	1
4	1	0	0	X	1	0	X	0	X
0	0	0	0	X	X	X	X	X	X
				X	X	X	X	X	X
				X	X	X	X	X	X

J	K	$Q_T \rightarrow Q_{T+\Delta T}$
0	X	0 $\rightarrow$ 0
1	X	0 $\rightarrow$ 1
X	1	1 $\rightarrow$ 0
X	0	1 $\rightarrow$ 1

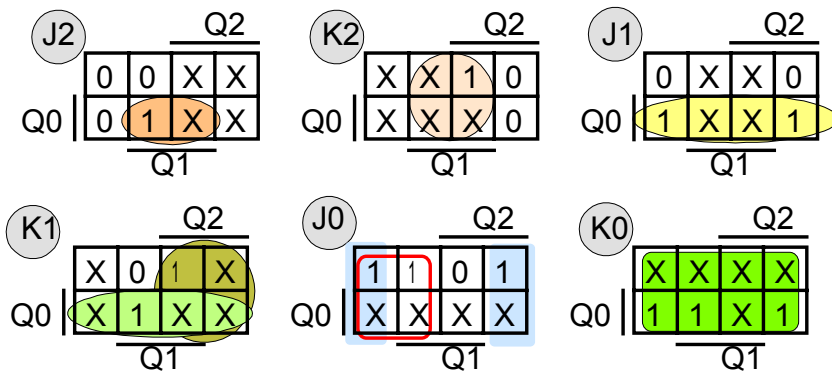




CONTADOR SÍNCRONO MÓDULO 7.

S	Q ₂	Q ₁	Q ₀	J ₂	K ₂	J ₁	K ₁	J ₀	K ₀
0	0	0	0	0	X	0	X	1	X
1	0	0	1	0	X	1	X	X	1
2	0	1	0	0	X	X	0	1	X
3	0	1	1	1	X	X	1	X	1
4	1	0	0	X	0	0	X	1	X
5	1	0	1	X	0	1	X	X	1
6	1	1	0	X	1	X	1	0	X
0	0	0	0	X	X	X	X	X	X

J	K	Q _T → Q _{T+ΔT}
0	X	0 → 0
1	X	0 → 1
X	1	1 → 0
X	0	1 → 1



$$J_2 = Q_1 Q_0$$

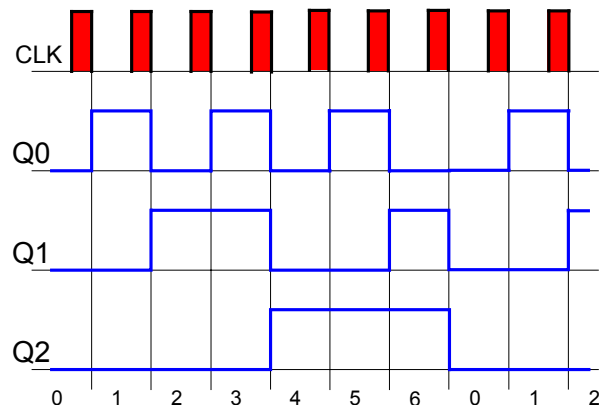
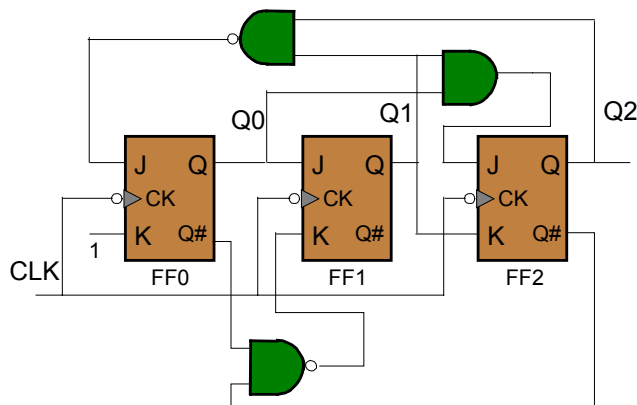
$$K_2 = Q_1$$

$$J_1 = Q_0$$

$$K_1 = Q_2 + Q_0 = \overline{Q_2} \overline{Q_0}$$

$$J_0 = \overline{Q_1} + \overline{Q_2} = \overline{Q_1} \overline{Q_2}$$

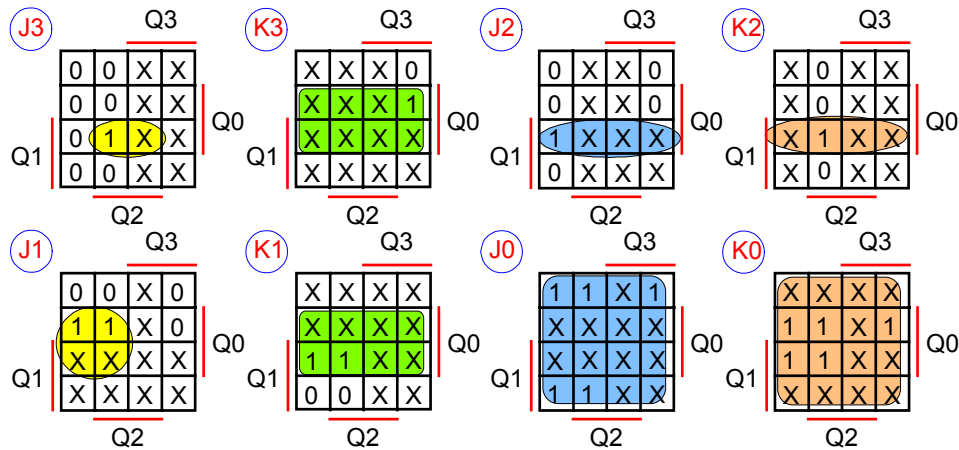
$$K_0 = 1$$



CONTADOR SÍNCRONO DE DÉCADAS.

S	Q ₃	Q ₂	Q ₁	Q ₀	J ₃	K ₃	J ₂	K ₂	J ₁	K ₁	J ₀	K ₀
0	0	0	0	0	0	X	0	X	0	X	1	X
1	0	0	0	1	0	X	0	X	1	X	X	1
2	0	0	1	0	0	X	0	X	X	0	1	X
3	0	0	1	1	0	X	1	X	X	1	X	1
4	0	1	0	0	0	X	X	0	0	X	1	X
5	0	1	0	1	0	X	X	0	1	X	X	1
6	0	1	1	0	0	X	X	0	X	0	1	X
7	0	1	1	1	1	X	X	1	X	1	X	1
8	1	0	0	0	X	0	0	X	0	X	1	X
9	1	0	0	1	X	1	0	X	0	X	X	1
10	1	0	1	0	X	X	X	X	X	X	X	X
11	1	0	1	1	X	X	X	X	X	X	X	X
12	1	1	0	0	X	X	X	X	X	X	X	X
13	1	1	0	1	X	X	X	X	X	X	X	X
14	1	1	1	0	X	X	X	X	X	X	X	X
15	1	1	1	1	X	X	X	X	X	X	X	X

J	K	Q _T	Q _{T+ΔT}
0	X	0	0
1	X	0	1
X	1	1	0
X	0	1	1



$$J_3 = Q_2 Q_1 Q_0$$

$$K_3 = Q_0$$

$$J_2 = Q_1 Q_0$$

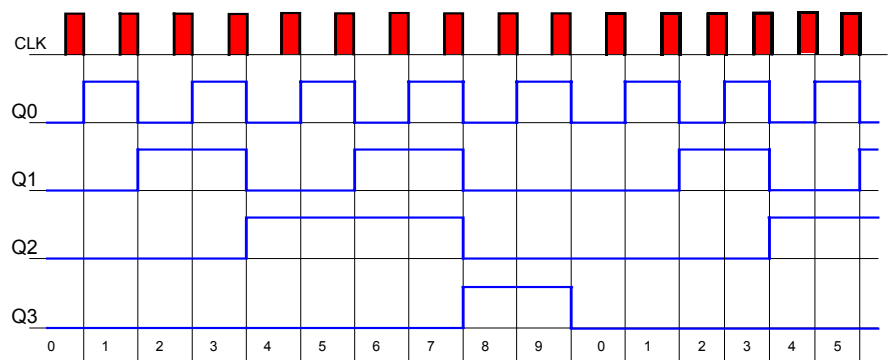
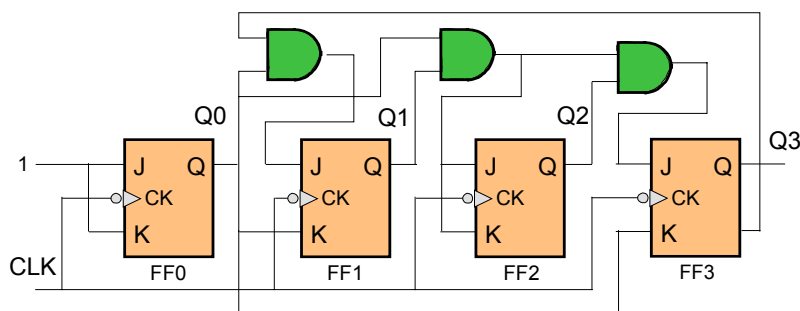
$$K_2 = Q_1 Q_0$$

$$J_1 = \overline{Q_3} Q_0$$

$$K_1 = Q_0$$

$$J_0 = 1$$

$$K_0 = 1$$



CONTADOR SÍNCRONO DESCENDENTE MÓDULO 10.

S	Q ₃	Q ₂	Q ₁	Q ₀	J ₃	K ₃	J ₂	K ₂	J ₁	K ₁	J ₀	K ₀
0	0	0	0	0	1	X	0	X	0	X	1	X
9	1	0	0	1	X	0	0	X	0	X	X	1
8	1	0	0	0	X	1	1	X	1	X	1	X
7	0	1	1	1	0	X	X	0	X	0	X	1
6	0	1	1	0	0	X	X	0	X	1	1	X
5	0	1	0	1	0	X	X	0	0	X	X	1
4	0	1	0	0	0	X	X	1	1	X	1	X
3	0	0	1	1	0	X	0	X	X	0	X	1
2	0	0	1	0	0	X	0	X	X	1	1	X
1	0	0	0	1	0	X	0	X	0	X	X	1
0	0	0	0	0	X	X	X	X	X	X	X	X
											*	*

Las combinaciones que faltan son salidas no importa

J	K	Q _T	Q _{T+ΔT}
0	X	0	0
1	X	0	1
X	1	1	0
X	0	1	1

* J₀ = K₀ = 1

J₃

1	0	X	X
0	0	X	X
0	0	X	X
0	0	X	X

K₃

X	X	X	1
X	X	X	0
X	X	X	X
X	X	X	X

J₂

0	X	X	1
0	X	X	0
0	X	X	X
0	X	X	X

K₂

X	1	X	X
X	0	X	X
X	0	X	X
X	0	X	X

J₁

0	1	X	1
0	0	X	0
X	X	X	X
X	X	X	X

K₁

X	X	X	X
X	X	X	X
0	0	X	X
1	1	X	X

J₀

1	1	X	1
X	X	X	X
X	X	X	X
1	1	X	X

K₀

X	X	X	X
1	1	X	1
1	1	X	X
X	X	X	X

$J_3 = \bar{Q}_2 \bar{Q}_1 \bar{Q}_0$

$K_3 = \bar{Q}_0$

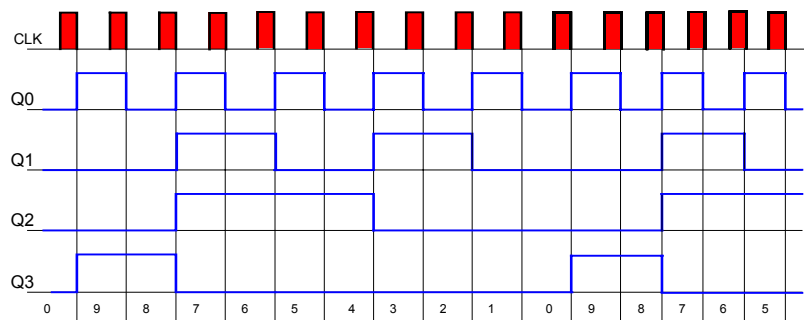
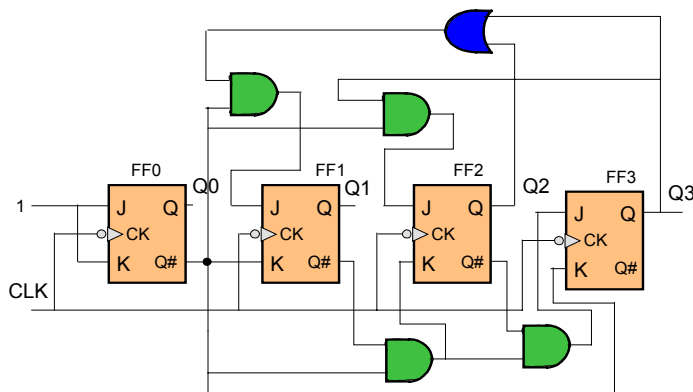
$J_2 = Q_3 \bar{Q}_0$

$K_2 = \bar{Q}_1 \bar{Q}_0$

$J_1 = Q_2 \bar{Q}_0 + Q_3 \bar{Q}_0$

$K_1 = \bar{Q}_0$

$J_0 = K_0 = 1$

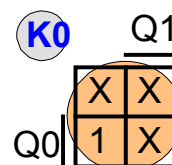
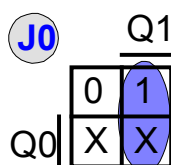
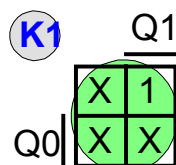
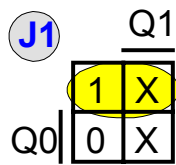


CONTADOR SÍNCRONO DESCENDENTE MÓDULO 3.

S	Q ₁	Q ₀	J ₁	K ₁	J ₀	K ₀
0	0	0	1	X	0	X
2	1	0	X	1	1	X
1	0	1	0	X	X	1
0	0	0	X	X	X	X
				*		*

* $K_1 = K_0 = 1$

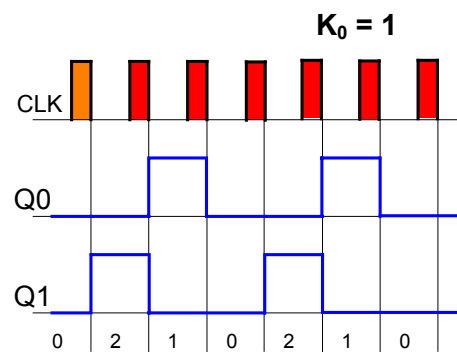
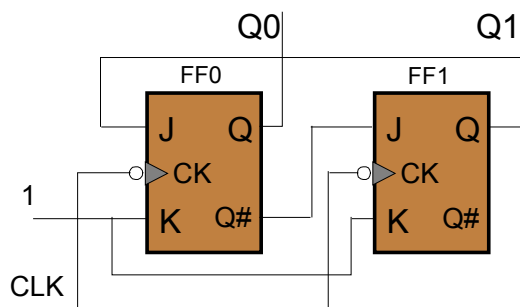
J	K	$Q_T \longrightarrow Q_{T+\Delta T}$
0	X	0 $\longrightarrow$ 0
1	X	0 $\longrightarrow$ 1
X	1	1 $\longrightarrow$ 0
X	0	1 $\longrightarrow$ 1



$$J_1 = \overline{Q_0}$$

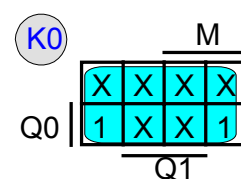
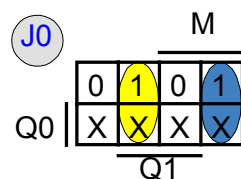
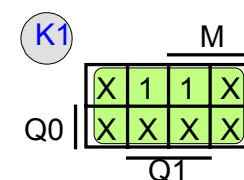
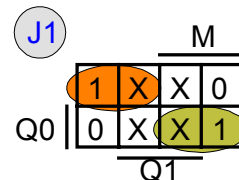
$$K_1 = 1$$

$$\mathbf{J}_0 = \mathbf{Q}_1$$



CONTADOR REVERSIBLE CONTROLADO POR UNA SEÑAL DE MANDO M (M=1 ascendente) MOD3.

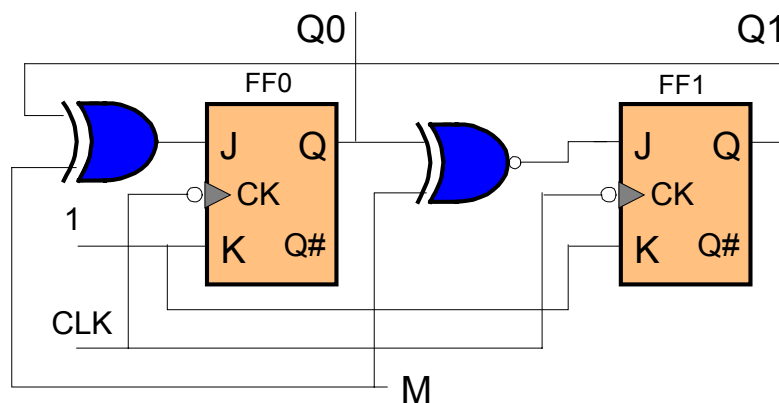
M	Q_1	Q_0	J_1	K_1	J_0	K_0
1	0	0	0	X	1	X
1	0	1	1	X	X	1
1	1	0	X	1	0	X
1	0	0	X	X	X	X
0	1	0	X	1	1	X
0	0	1	0	X	X	1
0	0	0	1	X	0	X
0	1	0	X	X	X	X



$$J_1 = \bar{M} \bar{Q}_0 + M Q_0$$

$$J_0 = \bar{M} Q_1 + M \bar{Q}_1$$

$$K_1 = K_0 = 1$$



También se puede realizar este contador de la siguiente manera:

M	Q ₁	Q ₀	J ₁	K ₁	J ₀	K ₀
1	0	0	0	X	1	X
1	0	1	1	X	X	1
1	1	0	X	1	0	X
1	0	0	X	X	X	X
				*		*

* K₁ = K₀ = 1

M	Q ₁	Q ₀	J ₁	K ₁	J ₀	K ₀
0	0	0	1	X	0	X
0	1	0	X	1	1	X
0	0	1	0	X	X	1
0	0	0	X	X	X	X
				*		*

* K₁ = K₀ = 1

J	K	Q _T	Q _{T+ΔT}
0	X	0	0
1	X	0	1
X	1	1	0
X	0	1	1

J1	M	Q0	Q1
		1	X
		X	0
		0	X
		X	1

Q0

K1	M	Q0	Q1
		X	1
		1	X
		X	X
		X	X

Q0

$$J_1 = \bar{M} \bar{Q}_0 + M Q_0$$

$$J_0 = \bar{M} Q_1 + M \bar{Q}_1$$

J0	M	Q0	Q1
		0	1
		1	0
		X	X
		X	X

Q0

K0	M	Q0	Q1
		X	X
		X	X
		1	X
		X	1

Q0

O también como dos contadores independientes.

ASCENDENTE

	Q ₁	Q ₀	J ₁	K ₁	J ₀	K ₀
0	0	0	0	X	1	X
1	0	1	1	X	X	1
2	1	0	X	1	0	X
0	0	0	X	X	X	X
				*		*

* K₁ = K₀ = 1

J1	Q1	Q0
	0	X
	1	X

Q0

K1	Q1	Q0
	X	1
	X	X

Q0

J0	Q1	Q0
	1	0
	X	X

Q0

K0	Q1	Q0
	X	X
	1	X

Q0

$$J_1 = Q_0 \quad K_1 = 1$$

$$J_0 = \bar{Q}_1 \quad K_0 = 1$$

DESCENDENTE

	Q ₁	Q ₀	J ₁	K ₁	J ₀	K ₀
0	0	0	1	X	0	X
2	1	0	X	1	1	X
1	0	1	0	X	X	1
0	0	0	X	X	X	X
				*		*

* K₁ = K₀ = 1

J1	Q1	Q0
	1	X
	0	X

Q0

K1	Q1	Q0
	X	1
	X	X

Q0

J0	Q1	Q0
	0	1
	X	X

Q0

K0	Q1	Q0
	X	X
	1	X

Q0

$$J_1 = Q_0 \quad K_1 = 1$$

$$J_0 = Q_1 \quad K_0 = 1$$

K₁ = K₀ = 1 independientemente de que esté en cuenta ascendente o descendente.

Para J₀ y J₁ se usa una señal de control para poder discriminar la cuenta ascendente y la descendente.

$$J_0 = M \bar{Q}_1 + \bar{M} Q_1$$

$$J_1 = M Q_0 + \bar{M} \bar{Q}_0$$

CONTADOR REVERSIBLE MÓDULO 5

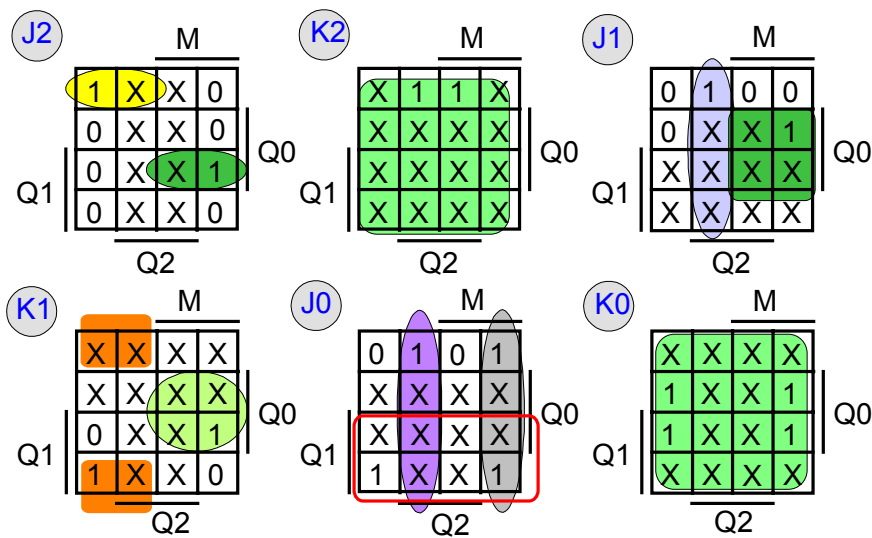
M=1 ascendente; M=0 descendente

M	Q ₂	Q ₁	Q ₀	J ₂	K ₂	J ₁	K ₁	J ₀	K ₀
1	0	0	0	0	X	0	X	1	X
1	0	0	1	0	X	1	X	X	1
1	0	1	0	0	X	X	0	1	X
1	0	1	1	1	X	X	1	X	1
1	1	0	0	X	1	0	X	0	X
0	0	0	0	X	X	X	X	X	X
					*				*

* $K_2 = K_0 = 1$

M	Q ₂	Q ₁	Q ₀	J ₂	K ₂	J ₁	K ₁	J ₀	K ₀
0	0	0	0	1	X	0	X	0	X
0	1	0	0	X	1	1	X	1	X
0	0	1	1	0	X	X	0	X	1
0	0	1	0	0	X	X	1	1	X
0	0	0	1	0	X	0	X	X	1
0	0	0	0	X	X	X	X	X	X
					*				*

$K_2 = K_0 = 1$



$$J_2 = \bar{M} \bar{Q}_1 \bar{Q}_0 + M Q_1 Q_0$$

$$K_2 = 1$$

$$J_1 = \bar{M} Q_2 + M Q_0$$

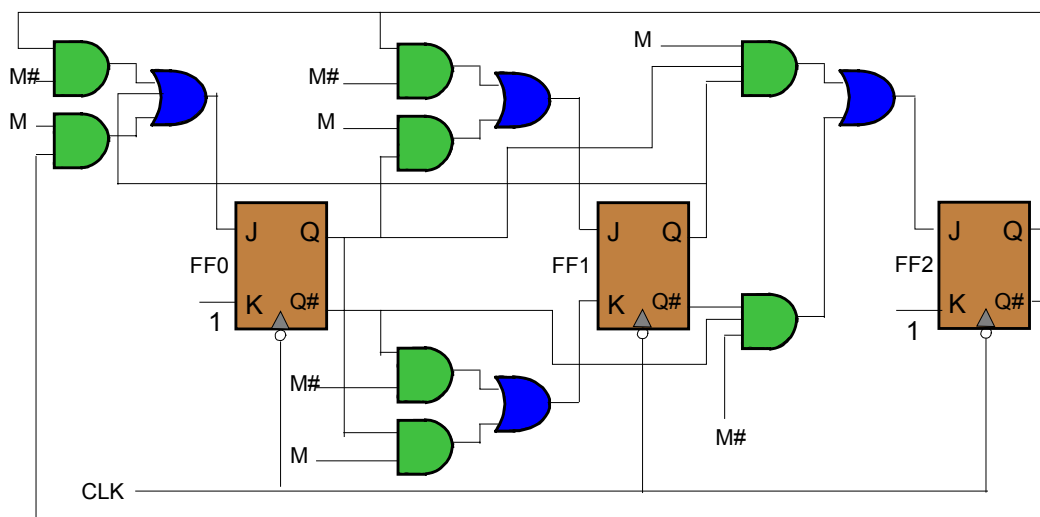
$$K_1 = \bar{M} \bar{Q}_0 + M Q_0 =$$

$$= M \text{ X-NOR } Q_0$$

$$J_0 = \bar{M} Q_2 + M \bar{Q}_2 + Q_1 =$$

$$= M \text{ X-OR } Q_2 + Q_1$$

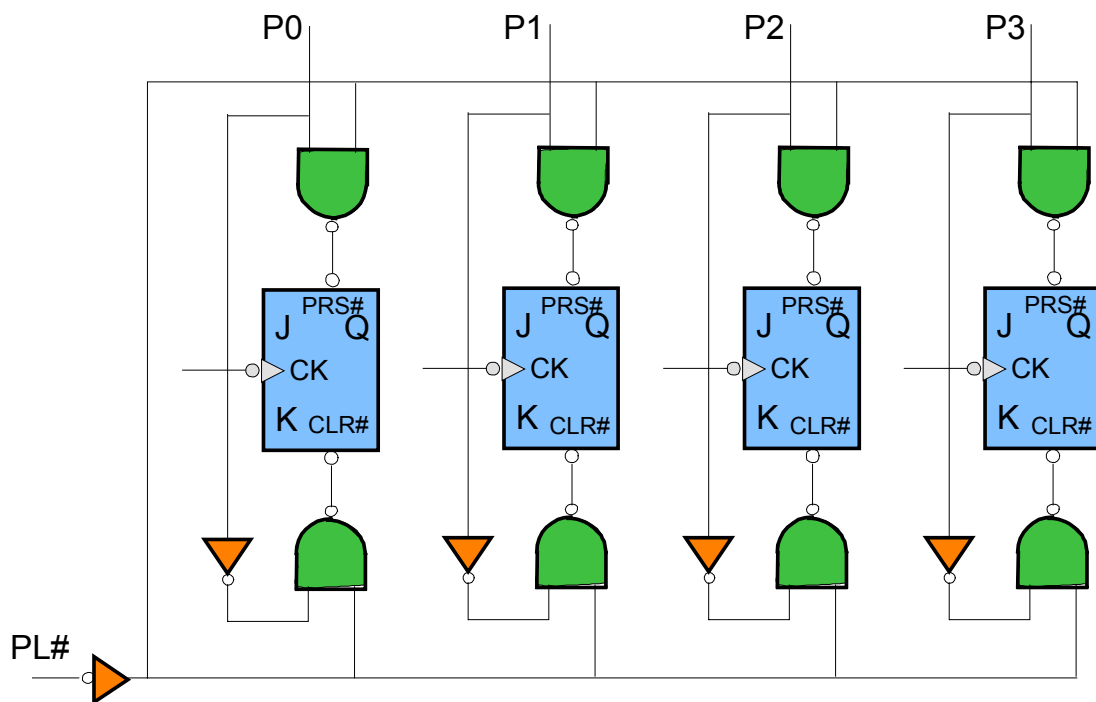
$$K_0 = 1$$



J	K	$Q_T \longrightarrow Q_{T+\Delta T}$
0	X	0 $\longrightarrow$ 0
1	X	0 $\longrightarrow$ 1
X	1	1 $\longrightarrow$ 0
X	0	1 $\longrightarrow$ 1

4.3.6-CONTADORES PREINICIABLES.

- ❑ Son contadores en los cuales el comienzo de la cuenta se puede establecer en un valor determinado.
- ❑ La puesta del contador a ese valor de cuenta, puede realizarse de forma asíncrona o de forma síncrona. En la forma asíncrona, la preinicialización es independiente de la señal de reloj, es decir, la entrada de preinicialización tiene prioridad sobre la señal de reloj (son contadores de este tipo 74XX190, '191, '192, '193). En la forma síncrona, el contador cargará el valor inicial durante la transición activa de la señal de reloj usada para el conteo (74XX160, '161, '162, '163).
- ❑ Tanto en la forma síncrona como en la asíncrona, el terminal PL# debe estar activado.



En el circuito de la figura, puede apreciarse que cuando se activa el terminal PL#, hace que todas las puertas NAND tengan una de sus entradas a nivel alto. Recordando lo que se decía sobre el paso y bloqueo de entradas en este tipo de puertas, esto hará que el valor presente en la otra entrada sea invertido en la salida.

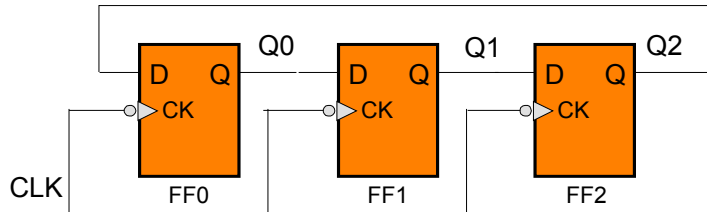
Las puertas superiores actúan sobre el PRS# (activo a nivel bajo) y las inferiores, actúan sobre el CLR# (también activo a nivel bajo).

Supongamos que se desea cargar el contador con el valor 1010 (P0, P1, P2, P3) para que la cuenta parta de este valor. Cuando se active el terminal PL#, en la salida de la puerta superior, correspondiente a P0 tendremos un cero lógico, activando por lo tanto el PRS# del FF de la izquierda, y poniendo su salida a un nivel alto. En la salida de la puerta inferior correspondiente también a P0, tendremos un nivel alto, no afectando por lo tanto a la entrada asíncrona de CLR#. Para las puertas correspondientes a P1, la salida de la puerta superior será un nivel alto, no afectando por lo tanto al terminal PRS# del FF, y en salida de la puerta inferior tendremos un nivel bajo, activando la entrada CLR# que pondrá el FF correspondiente a un nivel bajo. Para las demás entradas se sigue el mismo razonamiento.

4.3.7-CONTADORES CÍCLICOS O CONTADORES DE REGISTRO DE DESPLAZAMIENTO.

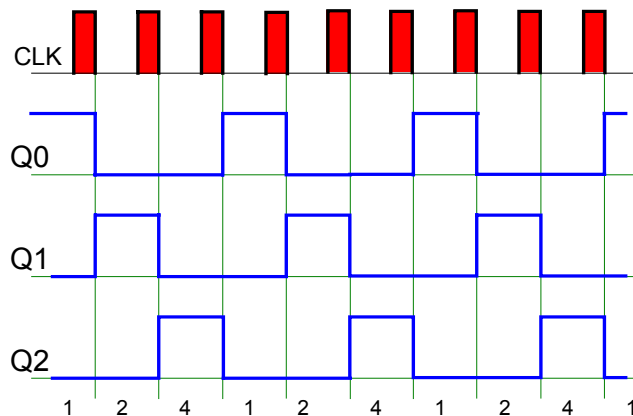
- Este tipo de contadores está retroalimentado.

CONTADOR EN ANILLO.

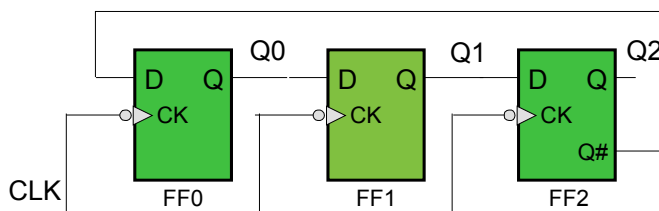


Generalmente en este tipo de contadores, solamente existe un 1 en el registro y se le hace circular por el mismo. Este valor, se coloca en el registro actuando sobre el *preset*.

El módulo lo determina el número de FF. Si tenemos n FF, el contador es de módulo n.



CONTADOR JOHNSON.



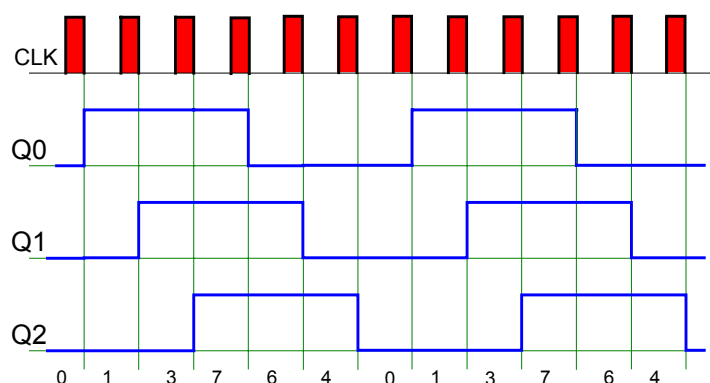
- Este contador realiza la secuencia de conteo según el código Johnson.

Duty Cycle= 50%

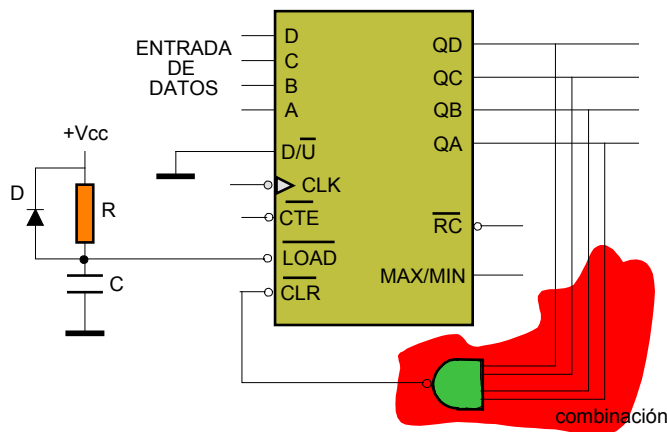
Módulo = 2 * núm de FF

DEC	Q ₂	Q ₁	Q ₀
0	0	0	0
1	0	0	1
2	0	1	1
3	1	1	1
4	1	1	0
5	1	0	0

Código Johnson para tres bits



4.3.8-CONTADORES COMERCIALES. VARIACIÓN DE LA CUENTA.



- En estos circuitos se puede hacer que el contador empiece en un valor de cuenta determinado y finalice en otro valor determinado.
- En el momento de energizar el sistema, mediante el circuito RC conectado al terminal de LOAD#, se hace que durante un determinado tiempo que dependerá de la constante de tiempo del circuito, ese terminal esté a nivel bajo, es decir, sea activo, cargando por lo tanto el

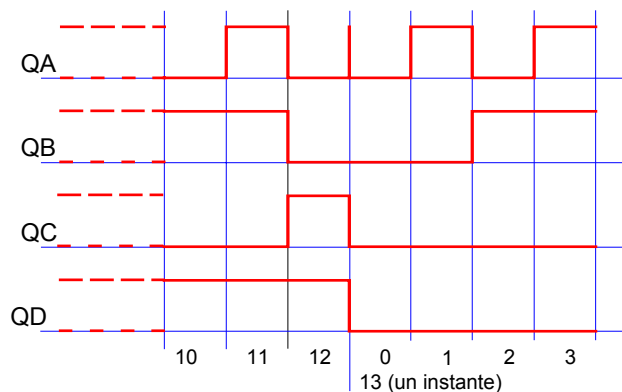
valor presente en las entradas de datos. Si se desea que la cuenta comience en 0, en las entradas de datos deberemos colocar el valor 0000. El diodo en paralelo con la resistencia del circuito RC, asegura una descarga rápida del condensador en caso de falta de energía eléctrica, de esta manera, se asegura una cuenta concreta de inicio en caso de falta momentánea de la alimentación.

- Para determinar el final de conteo o cuenta más alta, mediante un circuito combinacional se detecta cuando el valor de la cuenta ha llegado al valor determinado, en ese momento, la salida de ese circuito combinacional activa la entrada CLR# del contador, poniéndolo a 0. La cuenta se vuelve entonces a iniciar con el siguiente impulso de reloj.
- La combinación que reinicializa la cuenta, va a depender de si la entrada CLR# es asíncrona o síncrona.

Asíncrona: Para el caso de que sea asíncrona (no depende del clock, es prioritaria), esta entrada se verá afectada en el instante (mas retardos en la propagación del circuito combinacional) en que tengamos la combinación de salida del contador. Entonces, si queremos un contador de módulo M (conteo desde 0 hasta M-1) la combinación que debe activar el circuito combinacional ha de ser la que equivale en binario al valor del módulo M. En esta combinación se producirá un estado transitorio hasta que se active el CLR#.

	Q _D	Q _C	Q _B	Q _A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

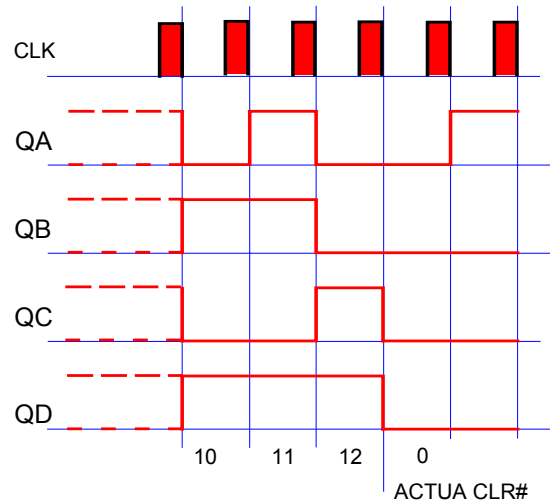
Ej: Para un contador de módulo 13.



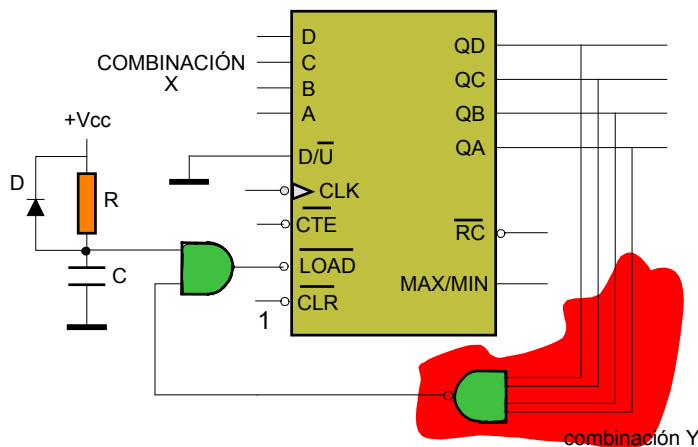
Las entradas de la puerta NAND serán la Q_D, Q_C, y Q_A. Cuando se dé la combinación 13 (1101) se activará el CLR# del contador reiniciándose la cuenta.

Síncrona: Para el caso de que sea síncrona (depende del clock, no es prioritaria), la combinación que se debe tomar para la acción sobre el terminal de CLR# será M-1, pues esta acción, no tendrá efecto hasta la siguiente transición activa de la señal de reloj; cuando esta transición se produzca, estando el terminal CLR# activado, pondrá el contador en estado de inicio.

Las entradas de la puerta NAND serán Q_D y Q_C como se deduce de la tabla de estados anterior.



4.3.9-ACOTACIÓN DE LA CUENTA ENTRE DOS VALORES X E Y

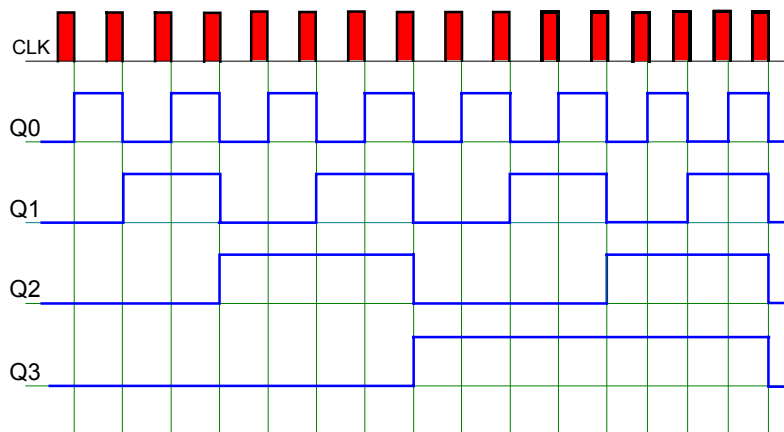


En este caso al energizar el circuito, la cuenta inicial será el valor X. Cuando se alcance el valor de Y, el circuito combinacional que detecta esta combinación, actuará sobre la entrada de carga (LOAD#), poniendo el valor de X en el contador y continuando la cuenta a partir de este valor.

Para realizar conteos descendentes, $D/U\# = 1$. Para que estos conteos estén acotados, se procede de forma análoga al caso anterior.

También se puede trabajar con las salidas de MAX/MIN, RC#, etc. Para cada problema habrá que tener en cuenta el tipo de contador utilizado y obrar en consecuencia.

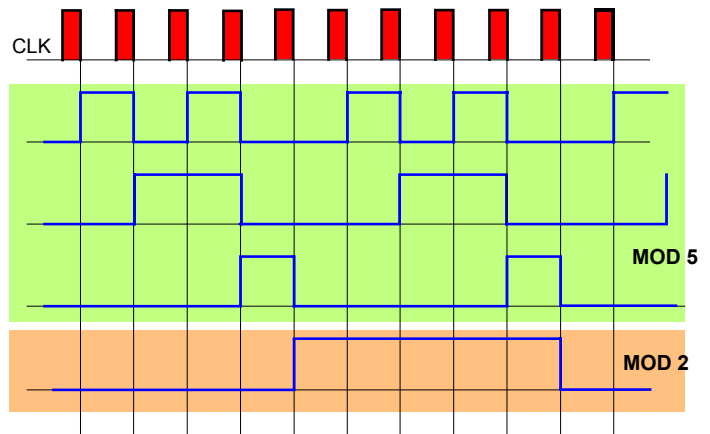
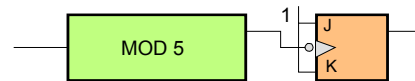
4.3.10-DIVISORES DE FRECUENCIA



□ Un contador de módulo M, es un divisor de frecuencia por ese modulo, es decir, en la salida del último FF, obtendremos la frecuencia de entrada al contador dividida por el módulo de ese contador.

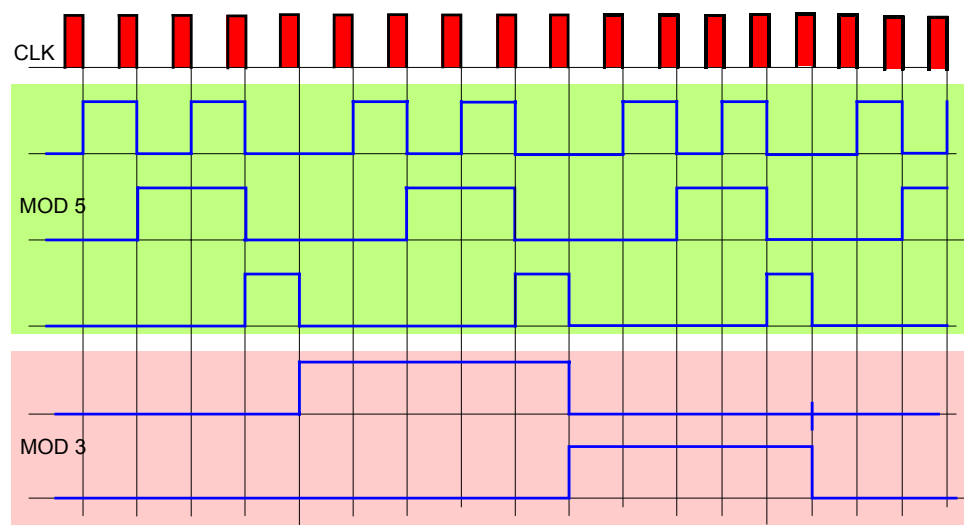
□ En los contadores con módulo binario, la señal con la frecuencia dividida, tiene un DC (Duty Cycle) del 50%. En los de módulo no

binario no se cumple este porcentaje. Para estos casos, en aquellos contadores de módulo par, se puede conseguir un DC=50%. Para ello, debemos dividir el contador en dos contadores diferentes, teniendo uno de ellos un módulo potencia de 2. Así para un contador de módulo 6, tomamos uno de módulo 3 y otro de módulo 2. Para módulo 10, tomamos módulo 5 y módulo 2, etc. En estos casos la simetría de la señal se consigue a costa de la secuencia de la cuenta.



Ej: Div de frecuencia por 10 con DC=50%.

Divisor de frecuencia por 15





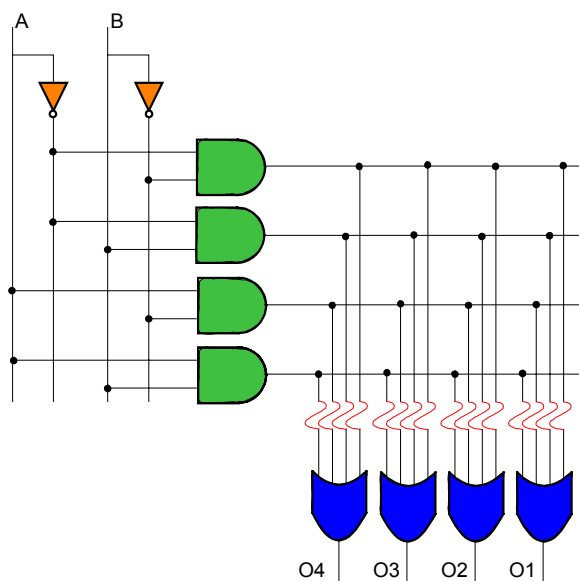
Dispositivos Lógicos Programables

5

5.-PLD-s (PROGRAMMABLE LOGIC DEVICES)

- ❑ Las exigencias del mercado actual imponen a los sistemas electrónicos, niveles muy altos de operatividad, reducción de tamaño, coste, fiabilidad (conexiones), etc. En muchos casos estos niveles son difíciles de alcanzar con IC-s estándar, por lo que muchos fabricantes de estos, optan por soluciones AS/C (Aplication Specific Intergrated Circuit).
- ❑ Entre los IC-s estándar y los ASIC se encuentra los IC-s de lógica programable. La idea de este sistema es integrar en una pastilla todos los posibles elementos necesarios para la realización de circuitos digitales. Esta operación de integración en un chip, de los elementos necesarios, es realizada por el fabricante de los PLD-s. El usuario debe configurar (programar) ese chip para que realice una tarea determinada en la aplicación.

5.1-IDEA BÁSICA:



Sea un circuito integrado, constituido por una serie de puertas lógicas ordenadas en forma de matriz de puertas AND (matriz AND) y matriz de puertas OR (matriz OR), según se muestra en la figura. Se compone también, de dos inversores y de una serie de fusibles conectados cada uno de ellos, a una entrada de las puertas OR. La idea de programación de este circuito, consiste en la fusión de los fusibles correspondientes para obtener una función lógica determinada. En el caso concreto de este ejemplo, podemos obtener cuatro funciones diferentes, de dos variables cada una.

Las salidas de las puertas AND serán las siguientes (comenzando por la superior):

$\bar{A} \bar{B}$, $\bar{A} B$, $A \bar{B}$, y $A B$.

Si los fusibles quedan intactos tendremos que las cuatro funciones O1, O2, O3 y O4 serán exactamente iguales, y de valor:

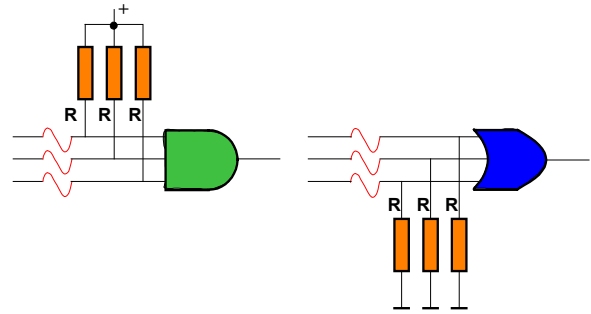
$$O1 = O2 = O3 = O4 = \bar{A}\bar{B} + \bar{A}B + A\bar{B} + AB = 1.$$

- ❑ Entonces, dependiendo del fusible o fusibles que se eliminen en cada una de las cuatro funciones, lo que haremos será eliminar uno o varios de los términos que componen cada una de las funciones.

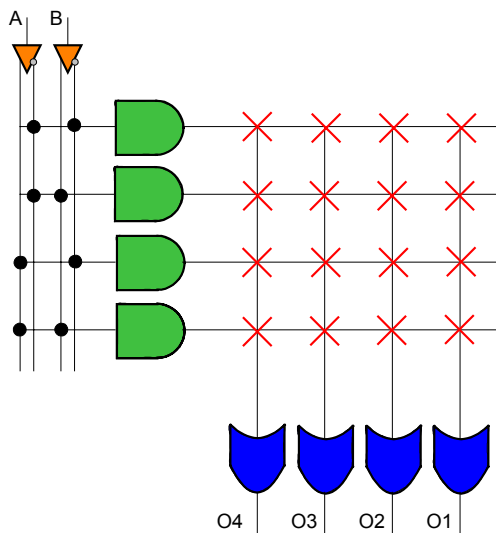
Por ejemplo, si eliminamos en la puerta OR correspondiente a la función O3 los dos fusibles de los extremos, esta función O3, tendrá valor:

$$O3 = \bar{A}B + A\bar{B} = A \text{ X-OR } B.$$

El hecho de eliminar fusibles en las entradas, no tiene que afectar para nada al comportamiento del circuito en cuanto al resultado final de la función de salida, la cual, debe ser función única y exclusivamente de las entradas cuyos fusibles quedan sin fundir. Para ello, los PLD-s, están diseñados de tal forma que la eliminación de un elemento fusible hace que la entrada correspondiente a la puerta OR, sea un cero lógico. Así, la función solo depende del valor de las otras entradas. Para el caso de una matriz con fusibles en las puertas AND el diseño sería tal que la fusión del fusible, dejaría la entrada correspondiente, a nivel lógico alto.



5.2-SIMBOLOGÍA.



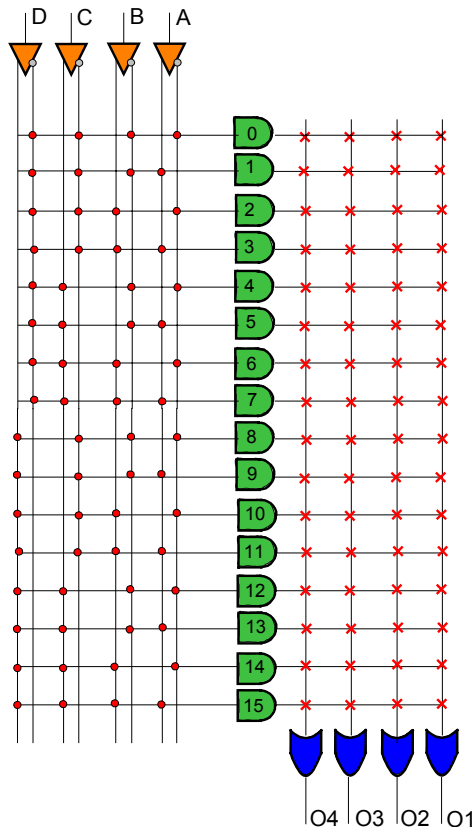
- El circuito anterior queda representado según la simbología utilizada para PLD-s de la forma siguiente:

● = conexión alamburada. Esta es una conexión fija en la estructura y por lo tanto no es programable.

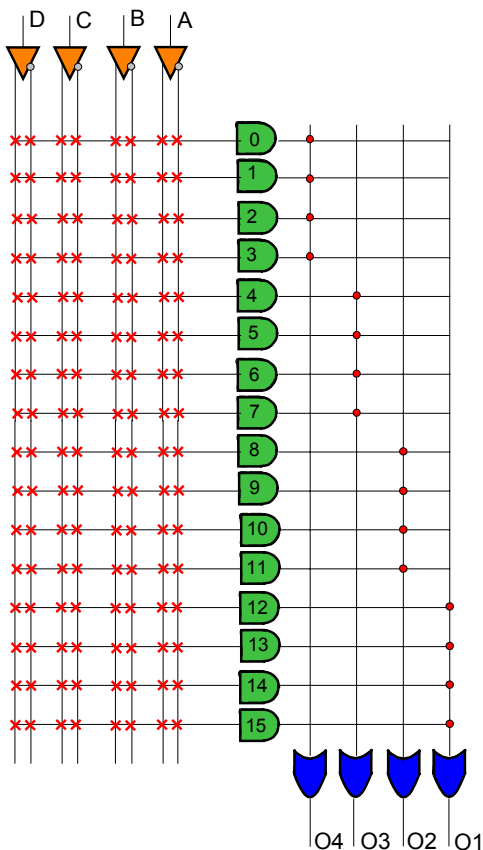
X= conexión fusible intacta. Esta es una conexión programable.

5.3-ESTRUCTURAS LÓGICAS BÁSICAS.

5.3.1-PROM (PROGRAMMABLE READ ONLY MEMORY)



- Una memoria PROM, puede ser utilizada para la realización de circuitos combinacionales. En este caso, las entradas de dirección son las variables de entrada de la función, y cada una de las líneas del bus da datos, representa una función.
- La PROM de la figura es una memoria de 16 X 4, pero desde el punto de vista de un PLD, es también un circuito integrado programable con el cual se pueden obtener cuatro funciones (O1, O2, O3 y O4) de cuatro variables (A, B, C, D).
- La matriz AND es fija, como corresponde a un decodificador de direcciones, y la matriz OR es la que puede ser programada.
- Esta estructura, es conveniente cuando se requiere un gran número de términos producto. No es conveniente cuando el número de variables es elevado y el número de términos producto que se requieren no lo es, pues por cada variable se duplica el tamaño de la matriz AND, desaprovechando la capacidad de la misma.



5.3.2-PAL (PROGRAMMABLE ARRAY LOGIC)

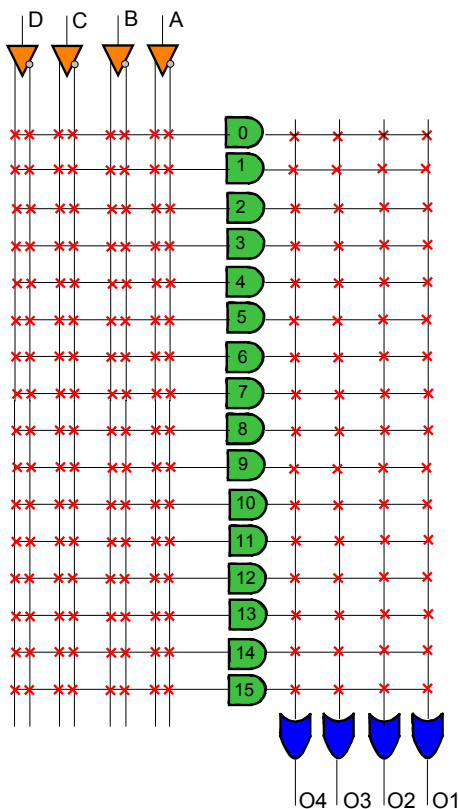
- La matriz AND es programable y la matriz OR es fija.
- Cada una de las funciones (O1, O2, etc.) tiene un número definido de términos, con lo cual aunque el número de variables sea grande, esto no significa haber doblado el tamaño del circuito por cada una de las variables incorporadas, como sucedía con la PROM.
- Las PAL, pueden ser de tipo bipolar o CMOS. Las de tipo CMOS pueden ser borradas mediante luz UV y también eléctricamente y pueden ser reprogramadas. Las borrables eléctricamente reciben el nombre de GAL. (Generic Array Logic).
- En este ejemplo de PAL, se pueden obtener cuatro funciones, con cuatro términos cada una como máximo, y con cuatro variables cada término también como máximo.
- Si en la función de salida se requieren menos de cuatro términos, se deben programar las entradas

de las puertas AND correspondientes a los términos que se quieran anular, de tal manera que la salida de estas puertas sea un cero, de esta manera, al efectuar la suma con los demás términos la función no se vea afectada. Para conseguir esto, lo único que se debe hacer es no fundir los elementos fusibles, así la salida de una AND en estas condiciones será:

$$\bar{D} D \bar{C} C \bar{B} B \bar{A} A = 0$$

Existen PAL con elementos biestables, lo que permite la realización de circuitos secuenciales.

5.3.3-PLA o FPLA. (PROGRAMMABLE LOGIC ARRAY o FIELD PROGRAMMABLE LOGIC ARRAY)



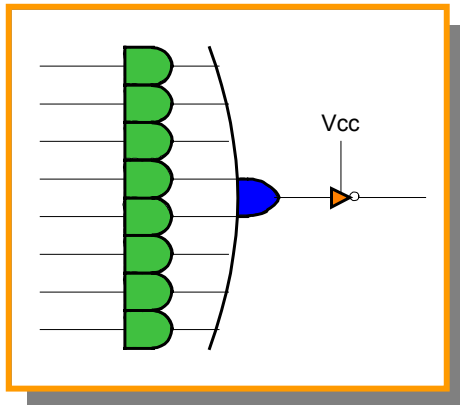
- Tanto la matriz AND como la matriz OR son programables. Permite por lo tanto mas flexibilidad en el diseño que los otros dos tipos, pero a costa de la complicación de la estructura, lo cual se nota en el precio.
- Recomendables en aplicaciones en las que se requiera una gran complejidad a nivel lógico.

5.4-CONFIGURACIONES DE SALIDA DE LAS PAL.

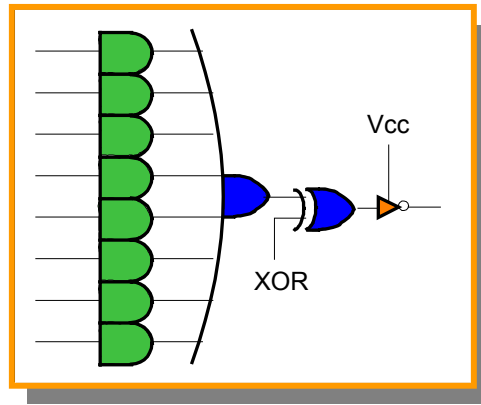
- La constitución de los circuitos de salida puede tomar diferentes formas, desde salidas simples tomadas directamente de las puertas de la matriz OR, pasando por terminales que funcionan como salidas y entradas, hasta salidas procedentes de lo que se llama macroceldas, que son circuitos compuestos internamente por los elementos necesarios para realizar funciones que se pueden programar, del tipo combinacionales de multiplexión, codificación, etc., y funciones secuenciales.

Las figuras siguientes muestran un ejemplo de diferentes arquitecturas de salida.

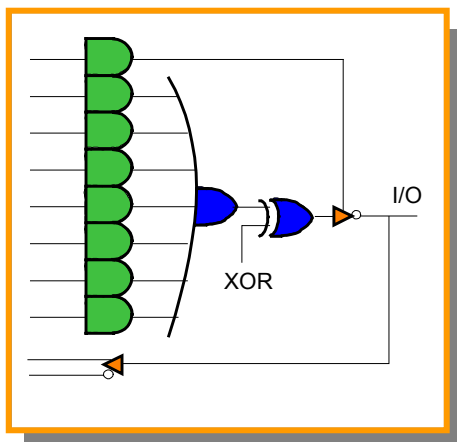
Combinacional



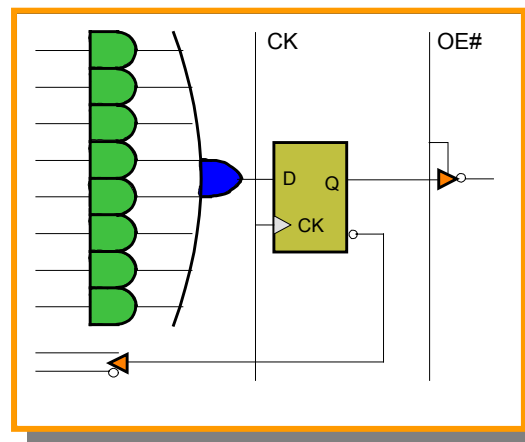
Combinacional con selección de nivel activo



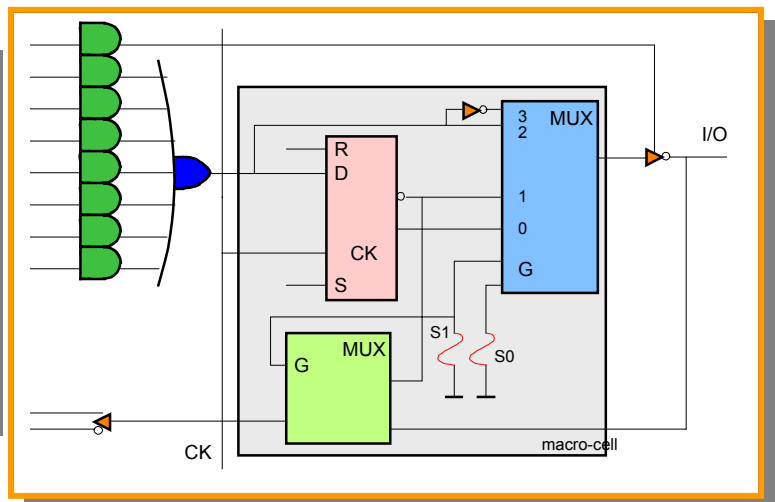
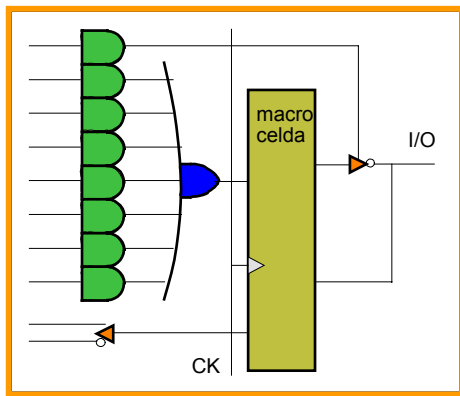
I/O Combinacional con selección de nivel activo.



Por registro realimentada.



Por macro-celda



5.5-NOMENCLATURA.

TIBPAL20L8-5C
TIBPAL20R4-5C
TIBPAL22V10-7C

- Las primeras letras suelen identificar al fabricante e indicar el tipo de PLD: PAL o GAL.

- Los siguientes datos nos muestran el número de entradas y salidas que tiene el circuito y tipo de salida. Los terminales que pueden funcionar como entradas o salidas, se cuentan para los dos casos.

20L8 significa que la PAL puede tener hasta 20 entradas y hasta 8 salidas activas a nivel bajo (L).

20R8 significa lo mismo que el anterior, pero en este caso las salidas son por registro. (Otras letras que pueden ir entre las dos cifras que indican las entradas y las salidas son: H, salidas activas a nivel lógico alto; V: salida variable).

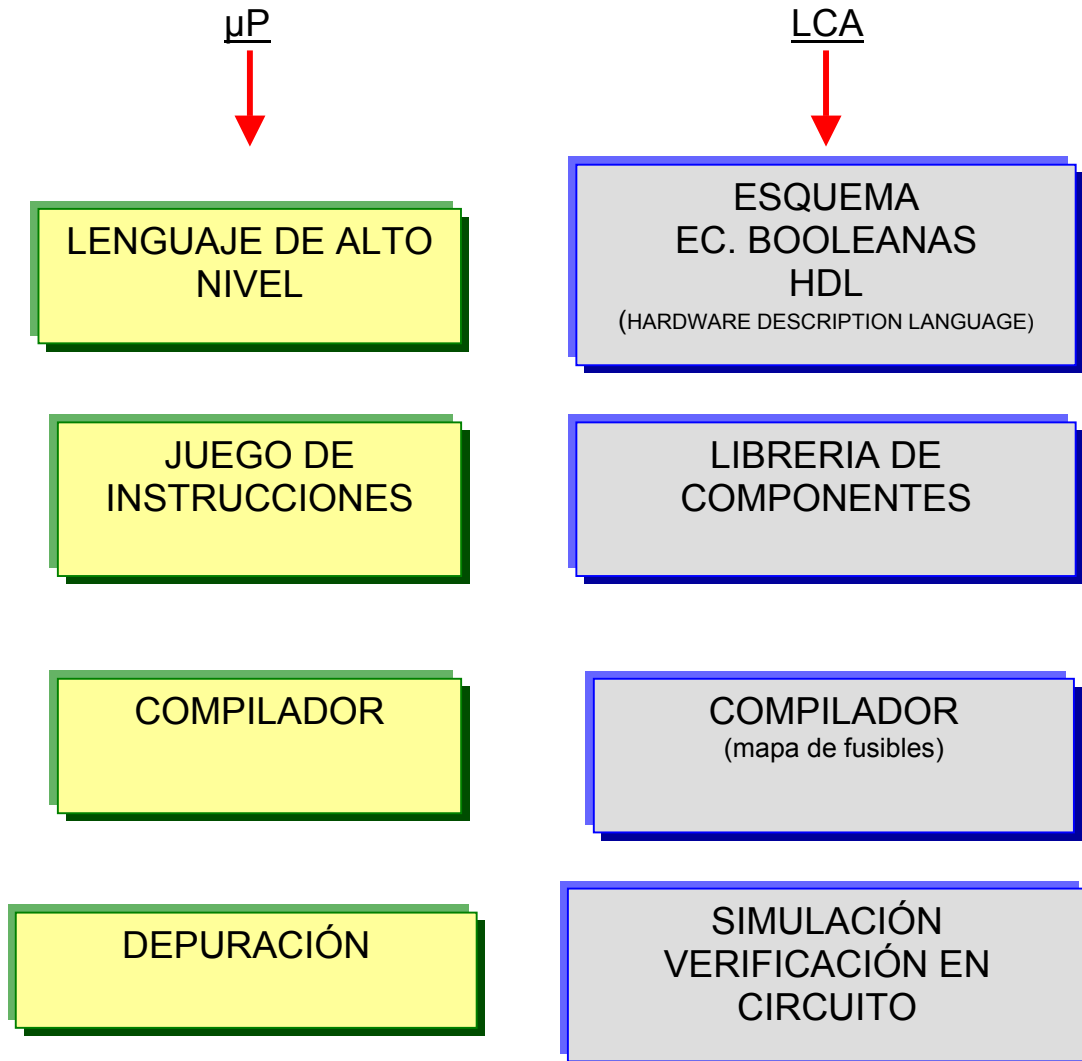
16R8-15: Cuando aparece un guión y después un número, éste indica el retardo entre entrada y salida en ns.

5.6-ACRÓNIMOS.

ASIC	APPLICATION-SPECIFIC INTEGRATED CIRCUIT
ASPL	APPLICATION-SPECIFIC PROGRAMMABLE LOGIC
E²PROM	EEPROM
EEPROM	ELECTRICALLY ERASABLE PROGRAMMABLE ROM
EPLD	ELECTRICALLY PROGRAMMABLE LOGIC DEVICE
EPROM	ELECTRICALLY PROGRAMMABLE ROM
FLEX	FLEXIBLE LOGIC ELEMENT MATRIX
FPGA	FIELD-PROGRAMMABLE GATE ARRAY
FPLS	FIELD-PROGRAMMABLE LOGIC SEQUENCER
GAL	GENERIC ARRAY LOGIC
LCA	LOGIC CELL ARRAY
MAX	MULTIPLE ARRAY MATRIX
MPLD	MASK PROGRAMMABLE LOGIC DEVICE
PAL	PROGRAMMABLE ARRAY LOGIC
PIC	PROGRAMMABLE INTEGRATED CIRCUIT
PLA	PROGRAMMABLE LOGIC ARRAY
PLD	PROGRAMMABLE LOGIC DEVICE
ROM	READ ONLY MEMORY

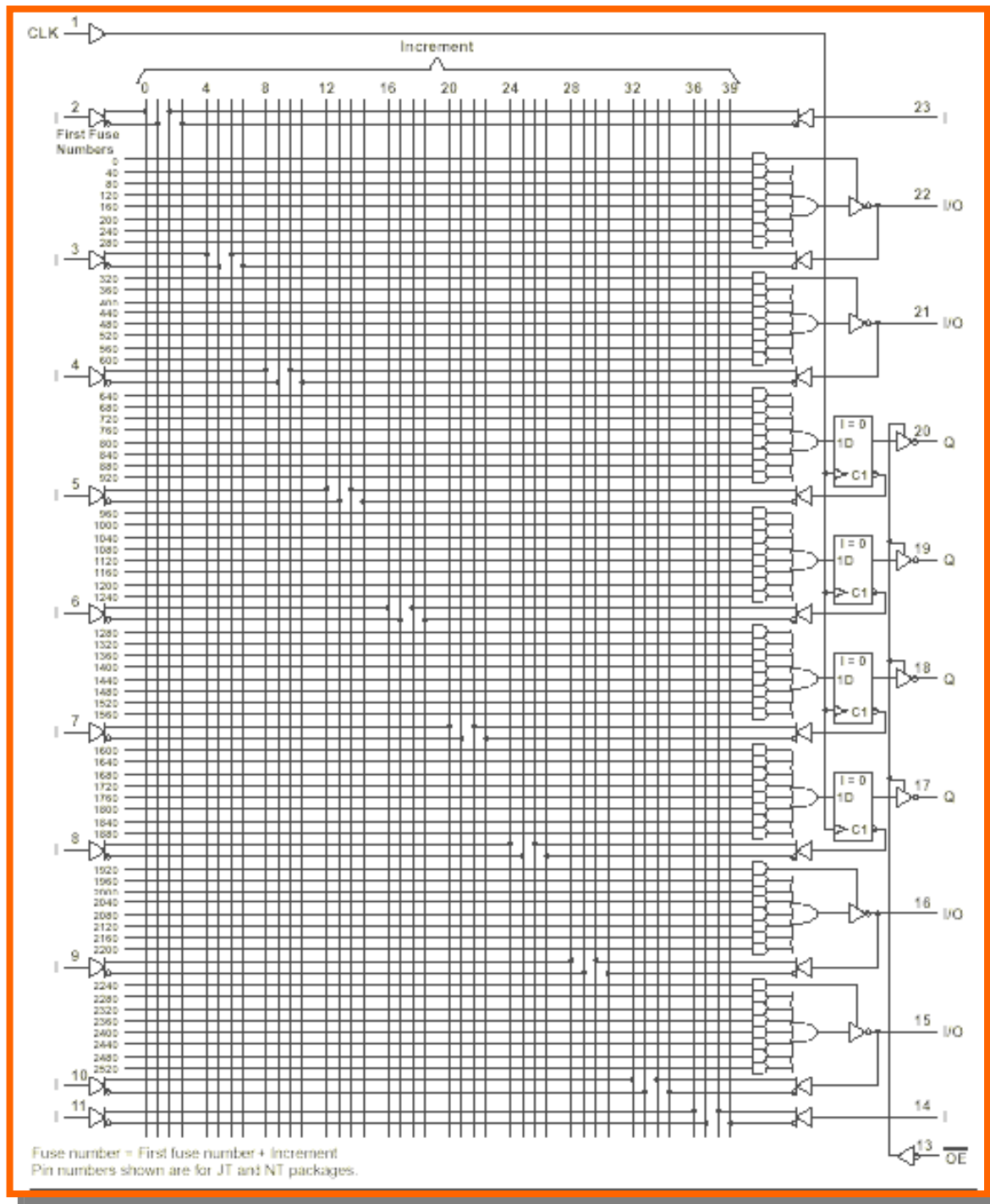
5.7-PROGRAMACIÓN

- Se muestra a continuación de forma resumida, los pasos a seguir en la programación de estos dispositivos, y su paso homologo en la programación de un microprocesador.



5.8-EJEMPLO PAL COMERCIAL.

- Se muestra un ejemplo de arquitectura PAL comercial con salidas de dos tipos: combinacionales y por registro realimentadas. Corresponde a una nomenclatura tipo 20R4



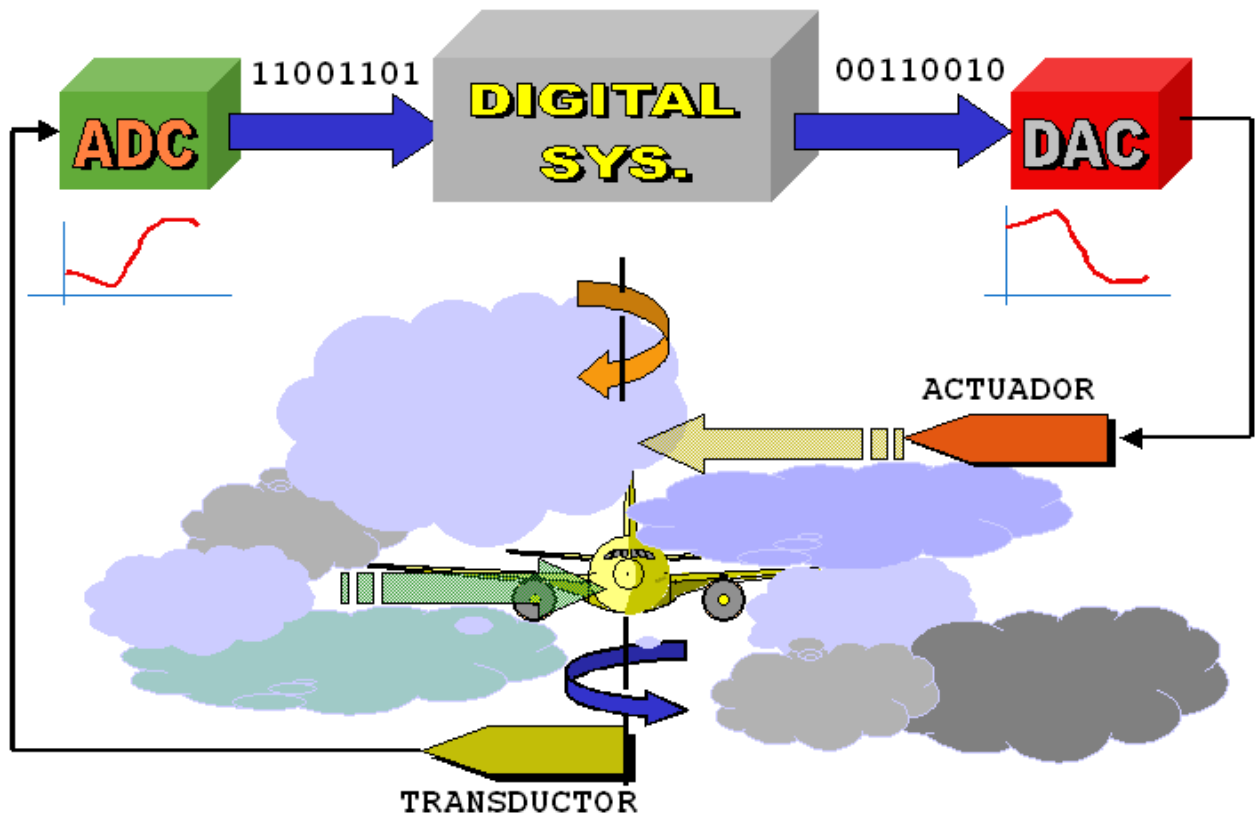


Conversión digital-analógica
Conversión analógica-digital

6

6.-ENTORNO ANALÓGICO Y DIGITAL.

- El tratamiento de las magnitudes físicas de naturaleza analógica, (velocidad, presión, temperatura, etc.), de forma digital, requiere primeramente de la conversión de dichas magnitudes a un entorno limitado a un número definido de valores, es decir, a un entorno digital. Este proceso es llevado a cabo en varias etapas e implica diversos elementos. En primer lugar, la magnitud física debe ser captada y convertida en una señal eléctrica (voltaje o corriente), debe ser tratada y posteriormente aplicada al convertidor analógico-digital.
- El elemento encargado de captar y convertir la magnitud física a eléctrica es el *transductor*. De forma genérica, un transductor es un dispositivo que toma una variable de entrada y produce una salida de naturaleza diferente. Para nuestro caso, la variable de entrada es una magnitud física y la salida una magnitud eléctrica. Los transductores podemos dividirlos en dos categorías según la función: Los que trabajan en paralelo con el flujo de la actividad y los que están implicados directamente con ese flujo. En el primer caso, si el transductor es eliminado, la actividad no sufre variación alguna. A este tipo pertenecen los transductores de medida. En el segundo caso, no existe actividad si se elimina el transductor.

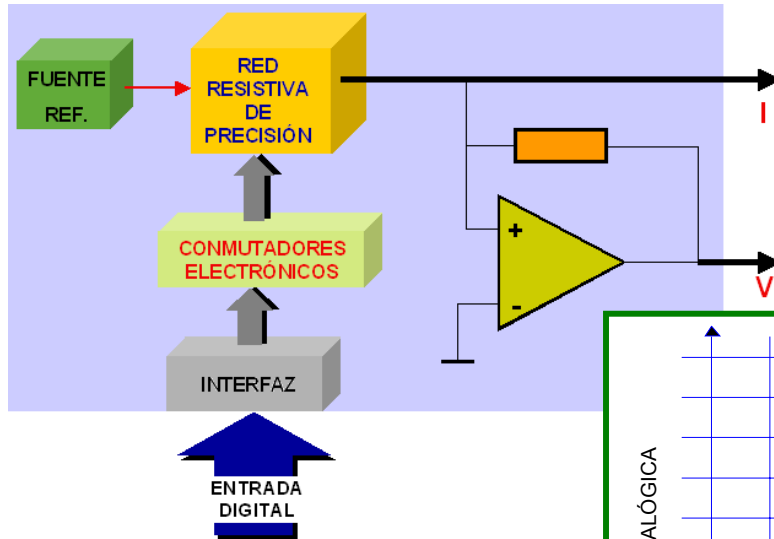


- La función de captación la realiza el transductor mediante un elemento llamado *sensor*. Un sensor es la parte del transductor que se comunica directamente con la variable que va a ser procesada, dando una respuesta que puede ser convertida en forma eléctrica.
- La señal eléctrica es tratada mediante el *acondicionador de señal* para que pueda ser aplicada al **ADC** (*Analog to Digital Converter*). Generalmente, las tareas que realiza el acondicionador son: suministrar una corriente o tensión de excitación alterna o continua al transductor, limitar o anular la señal de respuesta propia del elemento transductor, filtrar frecuencias indeseadas, etc.
- Las señales eléctricas obtenidas son convertidas a un formato digital por el ADC, de esta manera podrán ser procesadas por el sistema digital.

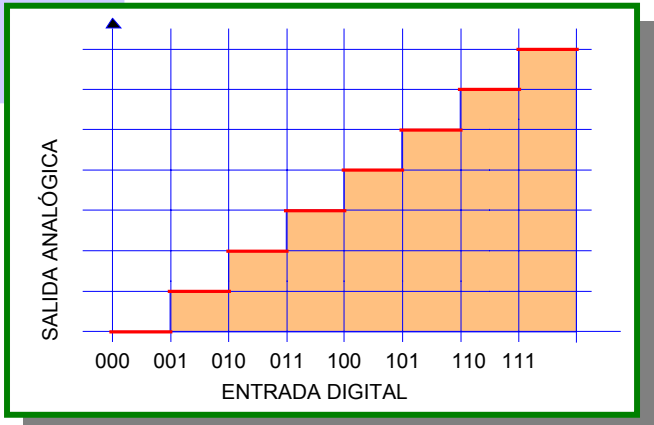
- El proceso inverso, consiste en la aplicación de una señal eléctrica digital, a un entorno analógico, es decir, donde las respuestas que se deben obtener de esa magnitud digital, son analógicas. Un **DAC** (*Digital Analog Converter*), convierte los valores digitales en una señal analógica. Esta señal eléctrica, puede ser aplicada a un *actuador eléctrico*, que es el elemento encargado de convertir la energía eléctrica que se le aplica, en otro tipo de energía (mecánica si es un motor, calorífica, luminosa, etc.)

6.1-DAC (DIGITAL TO ANALOG CONVERTER)

- La función de conversión digital a analógica, es convertir un código digital a un parámetro analógico mediante la conmutación de unos bits con una relación de pesos determinada, y recoger el resultado de esta acción en un punto en forma de suma.



- La figura muestra las partes características de un DAC. La diferencia fundamental entre un tipo de conversores y otro, reside en el tipo de red resistiva utilizada.

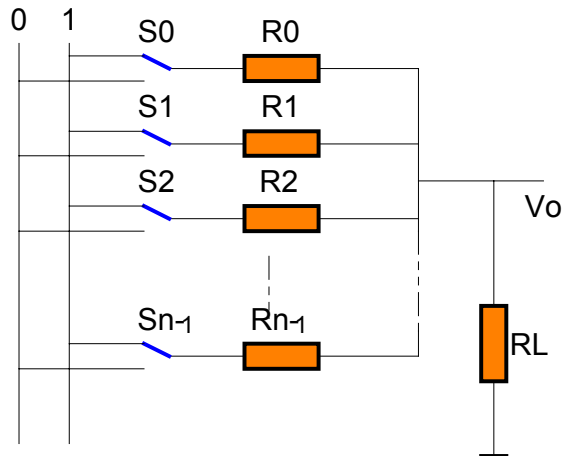


- El circuito de interfaz adapta los niveles lógicos estándar (TTL, CMOS,...) de entrada, al que requieren los circuitos que se encargan de gobernar los conmutadores electrónicos.
- La fuente de referencia es la que aporta el valor de tensión correspondiente al cero y uno lógicos, y hace que el valor de los pesos de cada BIT permanezca constante, independientemente de los valores de tensión de la información digital a convertir.
- Hay que observar que la salida analógica se obtendrá en forma de escalones, es decir, valores concretos, y que cada escalón corresponderá a un valor de la entrada digital. La variación mínima de valor entre escalones, dependerá del número de bits (combinaciones) del DAC. Para un número infinito de combinaciones, se obtendría un valor infinito de valores en la señal analógica.

6.1.1-DAC CON RESISTENCIAS PONDERADAS.

La figura siguiente, muestra la idea básica de un DAC de este tipo.

- El valor de cada resistencia es inversamente proporcional al peso del BIT. El BIT de más peso corresponderá con el conmutador por cuya rama circule mas corriente cuando este sea puesto a la tensión de referencia correspondiente al 1 lógico, es decir, la rama con la resistencia menor. Lo contrario ocurre con el BIT de menos peso.
- Lo más importante en esta configuración, es mantener la relación de pesos entre las resistencias para que la salida sea proporcional al código digital de entrada.



- Los valores en la entrada digital aplicados al DAC, no tienen que tener un valor definido sino simplemente un valor que identifique o diferencie un estado alto de uno bajo. Esto se debe a que esos valores son utilizados para actuar sobre los interruptores electrónicos. El valor efectivo con el que opera el DAC, viene dado por la fuente de referencia, que en la figura está representada mediante un 1 para el estado alto y un cero para el estado bajo.

ANÁLISIS.

- Tomamos un valor de R arbitrario. La relación entre la R elegida y las del circuito se establece como sigue:

$$R0=R/2^0=R; \quad R1=R/2^1; \quad R2=R/2^2; \quad R3=R/2^3; \quad \dots Rn-1=R/2^{n-1}$$

Para el análisis determinamos el circuito equivalente NORTON desde los terminales de la resistencia RL, considerando también el efecto de los conmutadores.

- Cálculo del circuito equivalente NORTON. (I_N y R_N)

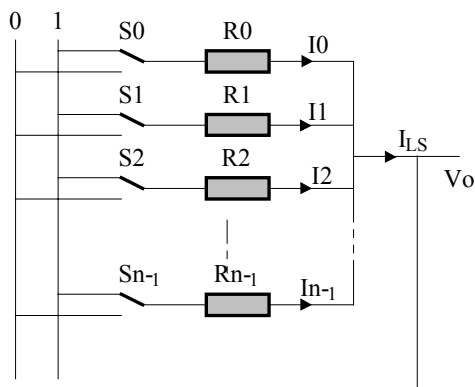
$R_L=0$ ($V_O=0$):

$$\begin{aligned} I_N = I_{LS} &= V_{REF} (S0/R0 + S1/R1 + S2/R2 + \dots + S_{n-1}/R_{n-1}) = \\ &= V_{REF} [S0/(R/2^0) + S1/(R/2^1) + S2/(R/2^2) + \dots + S_{n-1}/(R/2^{n-1})] = \\ &= (V_{REF} / R) (S0 \cdot 2^0 + S1 \cdot 2^1 + S2 \cdot 2^2 + \dots + S_{n-1} \cdot 2^{n-1}) \end{aligned}$$

V_{REF} : Tensión fija de referencia. Proporciona el 1 lógico.

I_{LS} : Corriente de cortocircuito:

S_i (para $i= 0, 1, 2, \dots$): muestran el estado del conmutador, cuando esta conectado a la tensión de referencia toma el valor 1, si está conectado a masa toma el valor 0.



$$R_N = r = 1 / (R0^{-1} + R1^{-1} + R2^{-1} + \dots + Rn-1^{-1})$$

Teniendo en cuenta que:

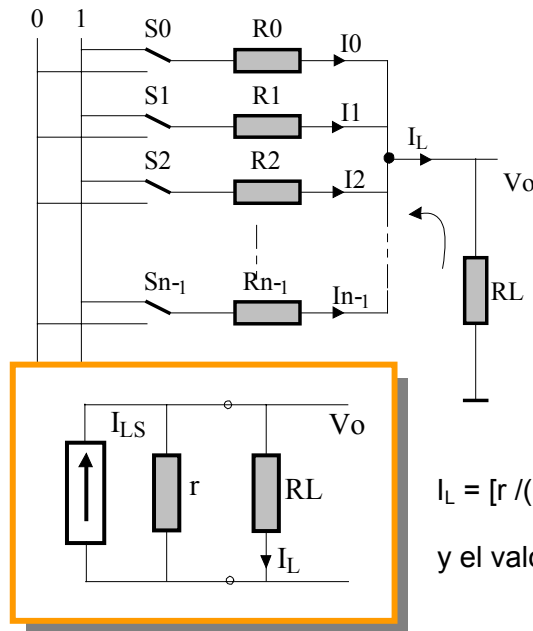
$$R0=R/2^0, \quad R1=R/2^1, \quad Rn-1=R/2^{n-1} \text{ queda}$$

$$r = R/(2^0 + 2^1 + 2^2 + \dots + 2^{n-1})^*$$

$$r = R/(1 - 2^n / -1) = R/(2^n - 1)$$

* (El paréntesis representa una progresión geométrica de razón 2)

$$[a + aq^1 + aq^2 + aq^3 + aq^4 + \dots + aq^{n-1} + \dots \quad S_n = (a - aq^n) / (1 - q)]$$



Trabajando con el circuito equivalente obtenido, se puede expresar el valor de la corriente por la RL de la siguiente forma

$$I_L = [r / (RL + r)] I_{LS}$$

Teniendo en cuenta el valor de I_{LS} , nos queda:

$$I_L = [r / (RL + r)] [(V_R/R) (S_0 2^0 + S_1 2^1 + S_2 2^2 + \dots + S_{n-1} 2^{n-1})]$$

y el valor de V_O :

$$V_O = RL * I_L = [(RL * V_{REF}) / (RL(2^n - 1) + R)] (S_0 2^0 + S_1 2^1 + S_2 2^2 + \dots + S_{n-1} 2^{n-1})$$

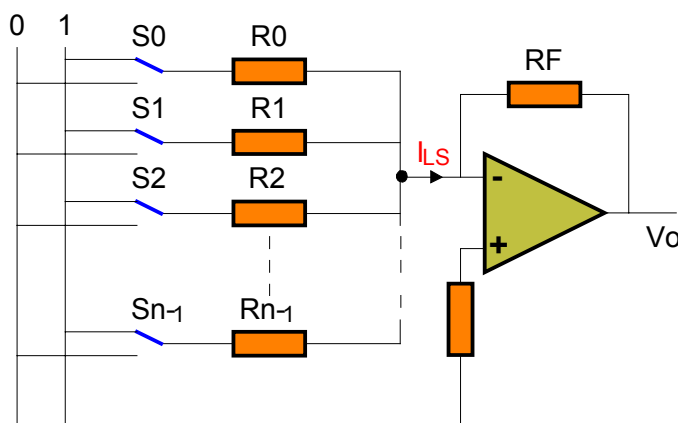
- El margen de tensiones varía entre $V_O = 0$ (cuando todos los S valen 0) y

$$V_O = [(RL * V_{REF}) / (RL(2^n - 1) + R)] (2^n - 1) = V_{REF} / [1 + (R/RL(2^n - 1))] = V_{REF} / [1 + (r/RL)]$$

cuando todos los S valen 1.

También se puede diseñar el circuito para obtener tensiones analógicas, positivas y negativas dependiendo de si el código de entrada es un valor positivo o un valor negativo (trabajando por ejemplo en complemento a 2). Para ello necesitamos una V_{REF} positiva y otra negativa.

CIRCUITO PRÁCTICO:



A la salida de la red de resistencias ponderadas conectamos un circuito convertidor I-V. La entrada inversora es una masa virtual, por lo tanto, la salida de la red de resistencias ponderadas estará a una tensión de cero voltios. El convertidor se encarga por lo tanto de convertir, la corriente de cortocircuito (o corriente NORTON) a una tensión de salida V_O cuyo valor es:

$$V_O = -(RF * I_{LS}) = -RF(V_{REF} / R) (S_0 2^0 + S_1 2^1 + S_2 2^2 + \dots + S_{n-1} 2^{n-1})$$

- El rango de tensiones que se puede obtener con este circuito, dependiendo de los valores de la V_{REF} es el siguiente:

(Hay que tener en cuenta que la máxima tensión de salida viene dada por la V de saturación del O.A.).

S	V_{REF}		V_O
	0	1	
todos a cero	0	V_R	0
todos a uno	0	V_R	$-(R_F/R) V_R(2^n-1)$
todos a cero	$-V_R$	V_R	$(R_F/R) V_R(2^n-1)$
todos a uno	$-V_R$	V_R	$-(R_F/R) V_R(2^n-1)$

- El número de bits de este tipo de convertidor, viene limitado por el rango de resistencias que hay que usar, ya que es difícil alcanzar de forma integrada, la precisión que se requiere por cada BIT, para el mantenimiento de los pesos y resultados satisfactorios.

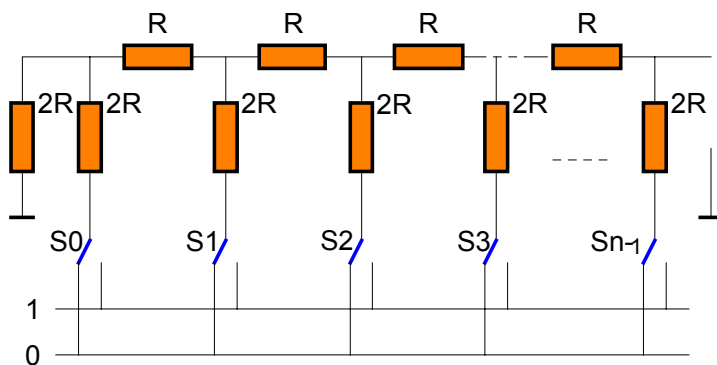
6.1.2-DAC CON RED EN ESCALERA R-2R.

- Este tipo de convertidor (también llamado convertidor por conmutación de voltaje), utiliza únicamente dos valores de resistencias por lo que es fácil de integrar en IC
- La resistencia que se “ve” desde la salida hacia el circuito es siempre del mismo valor y vale R. (*RESISTENCIA THEVENIN O NORTON*)
- Al igual que en la red de resistencias ponderadas, cada conmutador, según la posición en la que esté (0 o 1), y el peso que tenga asignado, hace una contribución en la salida.

La V_O en circuito abierto, viene dada por la expresión:

$$V_O = V_{REF} [(S_0 / 2^n) + (S_1 / 2^{n-1}) + (S_2 / 2^{n-2}) + \dots + (S_{n-1} / 2^{n-(n-1)})] =$$

$$= (V_{REF} / 2^n) (S_0 + S_1 2^1 + S_2 2^2 + \dots + S_{n-1} 2^{n-1})$$



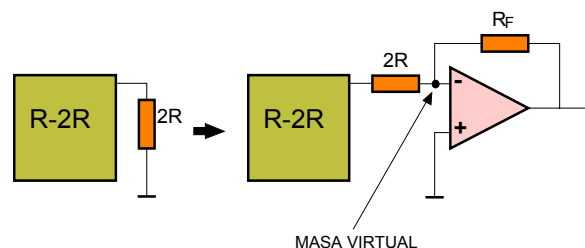
La contribución a la V de salida debida a S_0 es $V_{REF} / 2^n$, la debida a S_1 es $V_{REF} / 2^{n-1}$... etc., donde n es el número de bits (conmutadores) del convertidor.

- En un circuito con una carga RL tendremos:

$$V_O = [(R_L / (R + R_L))] (V_{REF} / 2^n) (S_0 + S_1 2^1 + S_2 2^2 + \dots + S_{n-1} 2^{n-1})$$

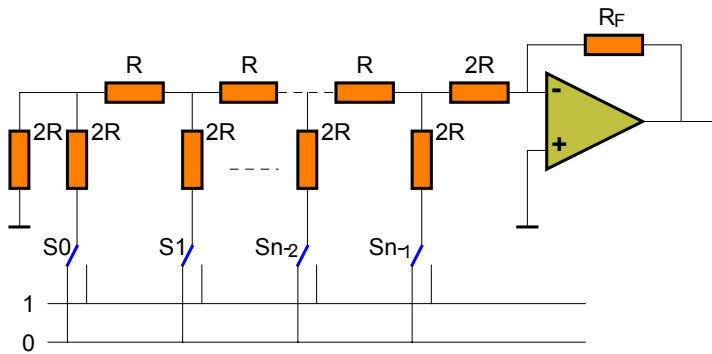
CARGA DE LA RED.

- Cargar la red, significa poner a la salida de la misma, una resistencia de valor $2R$, con lo cual se consigue, que la impedancia vista por el generador de referencia (V_{REF}), desde cualquiera de las entradas (S) sea siempre la misma, y de valor $3R$.

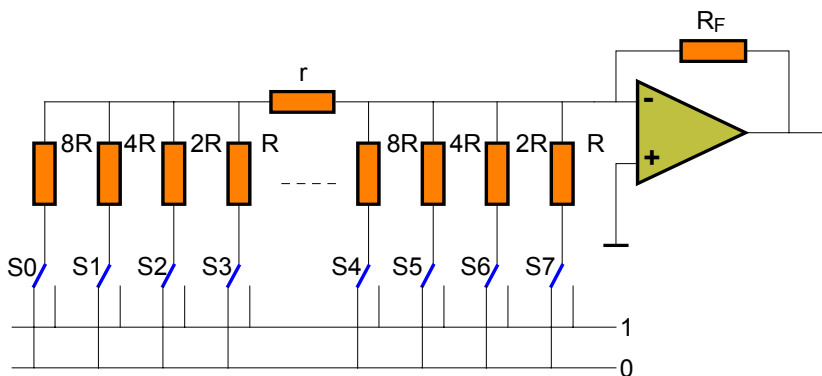


CIRCUITO PRÁCTICO

La V de salida del circuito es: $V_O = -(R_F/3R)(V_{REF}/2^n)(S_0 + S_1 2^1 + S_2 2^2 + \dots + S_{n-1} 2^{n-1})$



- En el convertidor digital analógico en escalera R-2R, los bits están ponderados mediante la provisión de caminos para la división de corriente, con las correspondientes atenuaciones sucesivas para los bits (conmutadores) de menos peso (bits menos significativos).



- En este convertidor, los bits están divididos en grupos de cuatro. Dentro de cada grupo, el esquema es el mismo que el de resistencias ponderadas.

- Entre el grupo de S_0, S_1, S_2, S_3 y el grupo de S_4, S_5, S_6, S_7 debe haber una atenuación de 16. (la aportación de corriente al convertidor I-V por el conmutador S_0 , debe ser 16 veces menor que la que se aporta por S_4 . Lo mismo debe cumplirse con S_1 y S_5 , con S_2 y S_6 y con S_3 y S_7). Esta atenuación se consigue tomando un valor de r :

$$r = 8R$$

- Esta disposición también se puede utilizar para la conversión de formato BCD. S_0, S_1, S_2 y S_3 representan el dígito decimal de las unidades. S_4, S_5, S_6 , y S_7 representan el dígito decimal de las decenas. Para este caso la atenuación debe ser de 10 entre los conmutadores correspondientes de un grupo y el otro. Esta atenuación se consigue para un valor de r :

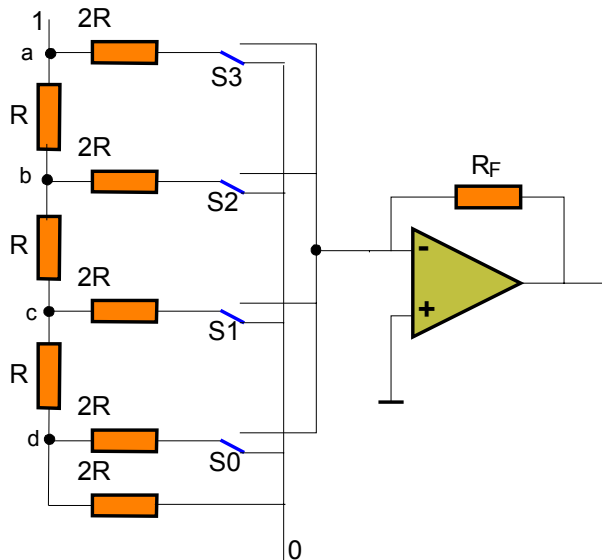
$$r = 4,8R$$

6.1.3-DAC CON ESCALERA INVERTIDA.

(Convertidor por Conmutación de Corriente)

- En esta configuración (ver figura en la página siguiente) se puede apreciar que la corriente que circula por cada una de las resistencias de la escalera, es independiente del estado de los conmutadores; Es decir, siempre circula la misma corriente, pues la diferencia de potencial en ellas permanece constante al tener conectado todas, uno de sus terminales a potencial cero (real o a la masa virtual). Entonces, la corriente suministrada por la fuente de referencia, V_{REF} (1) cuando todos los conmutadores están conectados a 0, es la misma que cuando todos están conectados a la entrada del operacional (convertor I-V) y vale:

$$I = V_{REF}/R$$



La tensión en el punto a, es la de la V_{REF} , ($V_a = V_{REF}$) por lo tanto la corriente I_{S3} es

$$I_{S3} = V_{REF} / 2R = I/2$$

$$V_b = R / (R+R) V_a = (1/2) V_a = (1/2) V_{REF}$$

(la resistencia equivalente desde el punto b hacia abajo es R)

$$I_{S2} = V_b / 2R = V_{REF} / 4R = I/4$$

$$V_c = R / (R+R) V_b = (1/2) (1/2) V_a = (1/4) V_a = (1/4) V_{REF}$$

$$I_{S1} = V_c / 2R = V_{REF} / 8R = I/8$$

$$V_d = R / (R+R) V_c = (1/2) (1/4) V_R = (1/8) V_{REF}$$

$$I_{S0} = V_d / 2R = V_{REF} / 16R = I/16$$

$$V_O = -[(R_F / 2R) V_a S_3 + (R_F / 2R) V_b S_2 + (R_F / 2R) V_c S_1 + (R_F / 2R) V_d S_0]$$

Haciendo $R_F = 2R$, la V de salida del circuito será:

$$\begin{aligned} V_O &= -[(2R/2R) V_a S_3 + (2R/2R) V_b S_2 + (2R/2R) V_c S_1 + (2R/2R) V_d S_0] = \\ &= -[V_{REF} S_3 + (1/2) V_{REF} S_2 + (1/4) V_{REF} S_1 + (1/8) V_{REF} S_0] \\ V_O &= - (V_{REF} / 2^3) (2^3 S_3 + 2^2 S_2 + 2^1 S_1 + 2^0 S_0) \end{aligned}$$

Para un convertidor de n bits:

$$V_O = - (V_{REF} / 2^{n-1}) (2^{n-1} S_{n-1} + 2^{n-2} S_{n-2} + \dots + 2^0 S_0)$$

- La principal ventaja de este tipo de convertidor aparte de que esté realizado solo con dos valores de resistencia, es la ya mencionada que las tensiones sean constantes en las resistencias y conmutadores. Esto hace que no haya retardos por la carga y descarga de capacidades parásitas, implicando una mayor velocidad de conversión y reduciendo fenómenos transitorios.

6.1.4-DAC. ESPECIFICACIONES

□ RESOLUCIÓN:

Se puede indicar especificando el número de bits con los que trabaja el convertidor.

Para n bits, el valor de estados diferentes que se pueden conseguir es 2^n , y el número de pasos en la salida del DAC será por lo tanto $2^n - 1$.

La resolución nos indica la mínima variación de corriente o tensión que puede ocurrir en la salida analógica como resultado del cambio mínimo que se puede dar en la entrada digital (valor igual al peso del LSB). Esta mínima variación en la salida se llama **TAMAÑO DE PASO**.

También se puede expresar la resolución como **PORCENTAJE DE SALIDA A ESCALA COMPLETA**:

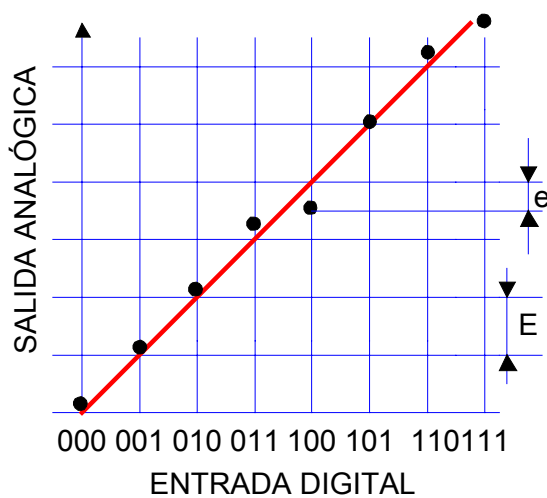
$$\% = (\text{tamaño de paso/escala completa}) * 100$$

o también

$$\% = [1/(2^n - 1)] * 100$$

$$(\text{tamaño de paso/FS}) = (\text{tamaño de paso}) / ((2^n - 1) * \text{tamaño de paso}) = 1 / (2^n - 1)$$

□ **PRECISIÓN:** La precisión del DAC, viene dada por diferentes parámetros:



***ERROR DE LINEALIDAD (E_L –Linearity Error):**

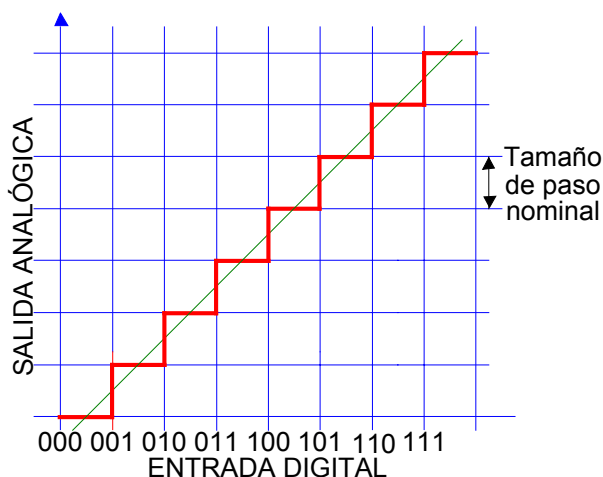
Nos indica la diferencia entre el valor real del paso y el valor nominal del paso después de que el error de offset y el error de ganancia hayan sido ajustados a cero. Si el convertidor fuera perfecto respecto a su linealidad, todos los puntos caerían en la línea recta gruesa.

e indica el error de linealidad.

E indica el tamaño de paso.

PEJ.

$$e = \pm 0.5 \text{ LSB} \longrightarrow |e| < 0.5E$$



***ERROR DE LINEALIDAD DIFERENCIAL (E_D –Linearity Error Differential).**

Nos indica la diferencia entre el tamaño de paso real y el tamaño de paso ideal. Un $E_D > 1 \text{ LSB}$, puede llevar al convertidor a la no monotonicidad.

***ERROR DE ESCALA COMPLETA. (E_{FS} – Full Scale Error)**

Este error nos indica la diferencia entre el valor real del paso y el valor nominal especificada en relación a la escala completa o fondo de escala.

PEJ. $\pm 0,01\% \text{ F.S.}$ $\longrightarrow$ si la salida FS son 9v quiere decir que la salida de este DAC puede variar en cualquier momento en una cantidad de:

$$(\pm 0.01\% / 100) * 9v = \pm 0.0009 v = \pm 0.9 \text{ mV}$$

***MONOTONICIDAD (Monotonicity).**

Es una propiedad de la función de transferencia, que asegura el consistente incremento o decremento de la salida analógica en respuesta a un incremento o decremento de la entrada digital.

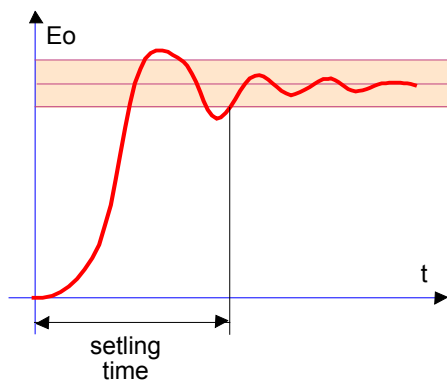
*ERROR DE GANANCIA (E_G – Gain Error).

Indica la desviación, en el voltaje de salida, del valor nominal cuando las entradas son tal que la salida debe dar fondo de escala.

*VOLTAGE DE OFFSET:

Es el voltaje que se obtiene en la salida analógica cuando todos los bits de entrada son iguales a 0. Muchos DAC tienen un ajuste externo de *offset* que permite reducir el voltaje de offset a 0.

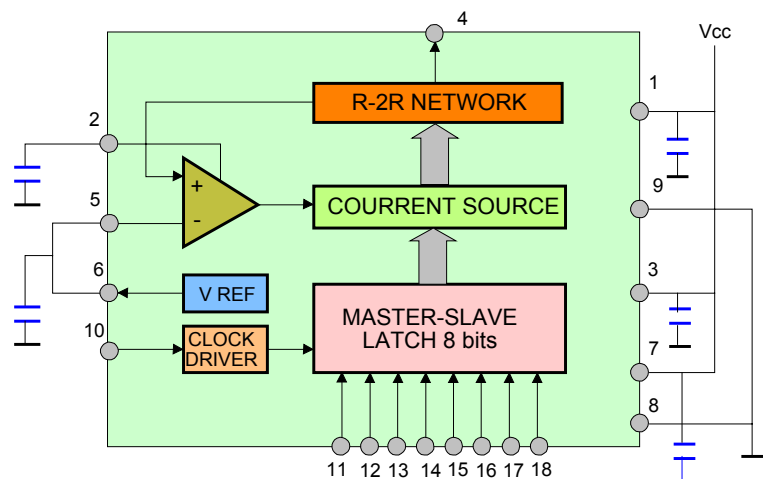
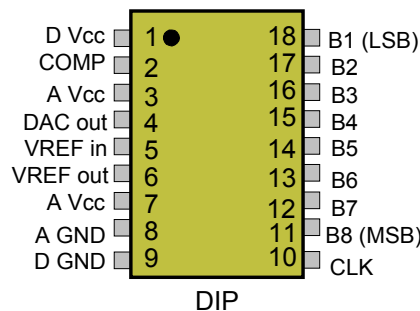
- ❑ ***SENSIBILIDAD A LA TEMPERATURA:*** Indican los efectos que los cambios de temperatura ejercen sobre la salida del conversor. Se especifica como cambio en % FS / °C.
- ❑ ***TIEMPO DE ESTABLECIMIENTO (t_s):***



Indica el intervalo de tiempo entre el instante cuando la entrada digital cambia y el instante cuando el valor de la salida analógica entra por última vez en una banda específica de error hacia su valor final.

Este tiempo se debe principalmente a los transitorios originados por los interruptores, y son mas o menos prolongados dependiendo de las capacidades e inductancias parásitas del circuito.

6.1.5-EJEMPLOS COMERCIALES.



- La entrada de la información digital está sincronizada con el CLOCK. Por cada flanco activo de dicho reloj, la información presente en la entrada digital es pasada al latch, y esa información se mantiene en dicho latch hasta otro flanco activo.
- La salida que vamos a obtener del circuito, es una salida de corriente, es decir, vamos a obtener una corriente proporcional a los valores digitales presentes en la entrada. Si lo que queremos es una tensión, debemos acoplar externamente un convertidor I-V a la salida del DAC (terminal 4).

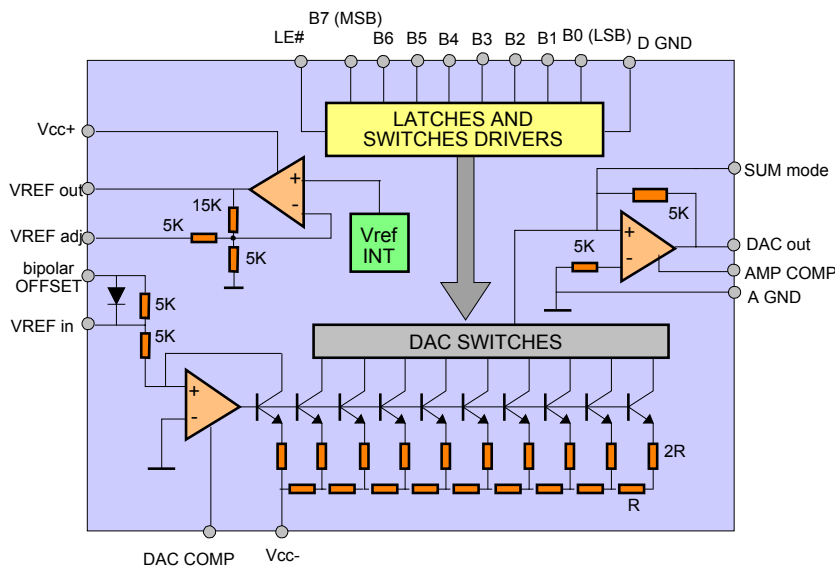
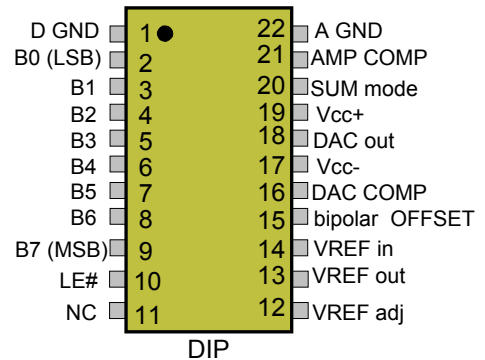
- La tensión de referencia puede ser suministrada por el propio circuito convertidor o puede ser una tensión externa. Esta es una característica muy común en los DAC.

Algunas características de este circuito son:

	SYMBOL	MIN	TYP	MAX	UNIT	COND. TEST
Resolution		8	8	8	bit	
Conversion Rate	f_{CLK}	40	-	-	MHz	
Linearity error	LE	-0.5	-	0.5	LSB	
Data setup time	t_{SU}	10	-	-	ns	$f_{CLK} = 40 \text{ MHz}$
Data hold time	t_h	10	-	-	ns	$f_{CLK} = 40 \text{ MHz}$
Reference input voltage	$V_{REF IN}$	2.0	3.0	4.0	V	
Reference output voltage	$V_{REF OUT}$	3.0	3.1	3.22	V	
Digital input voltage	V_{IH}	2.0	-	V_{CC}	V	
	V_{IL}	0	-	0.8	V	
Power supply voltage	V_{CC}	4.75	5.00	5.25	V	

5118

- En este ejemplo, las entradas digitales también van a un latch de entrada, pero en vez de estar el proceso sincronizado con una señal de reloj, está sincronizado con una señal de Latch Enable (LE#). Mientras esta señal es activa, el latch es transparente a la información digital, es decir, que si la información digital está cambiando, la salida analógica cambia siguiendo a la entrada.



Cuando LE#, pasa a su estado inactivo, la última información presente en la entrada, pasa al latch, y es mantenida ahí hasta una nueva transición activa de LE#. Por lo tanto, la salida queda estable en el valor analógico correspondiente a la información guardada en el latch. Mientras LE# esta inactivo, las entradas al latch se mantienen en estado de alta impedancia.

- Se puede observar también, que a diferencia del anterior IC, este necesita una tensión simétrica para trabajar (V_{CC+} y V_{CC-}).
- En las especificaciones se indica que el valor mínimo de la $V_{REF in}$, debe ser de 2.5 voltios, ya que la escalera necesita $500\mu A$ para que pueda operar correctamente en la zona lineal.

- La expresión de la tensión analógica de salida en función de la información digital de entrada viene dada por:

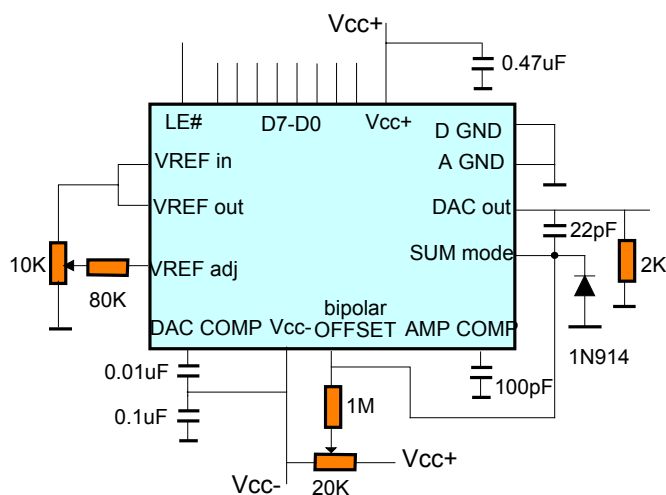
$$V_{OUT} = 2 V_{REF} (B7/2 + B6/4 + B5/8 + B4/16 + B3/32 + B2/64 + B1/128 + B0/256)$$

Los modos básicos de operación de este circuito son dos: unipolar y bipolar. Trabajando en el modo unipolar la salida puede variar entre 0 y +10 voltios. Para que trabaje en modo bipolar, el terminal denominado *bipolar OFFSET*, se debe unir con el terminal de *SUM mode*. En este caso el rango de salida es desde -5 a +5 voltios.

Se puede observar que cuando los dos terminales mencionados están unidos, el OA de la salida analógica, queda configurado como un amplificador inversor de ganancia unidad, cuya entrada de tensión es la aplicada al terminal V_{REF} in.

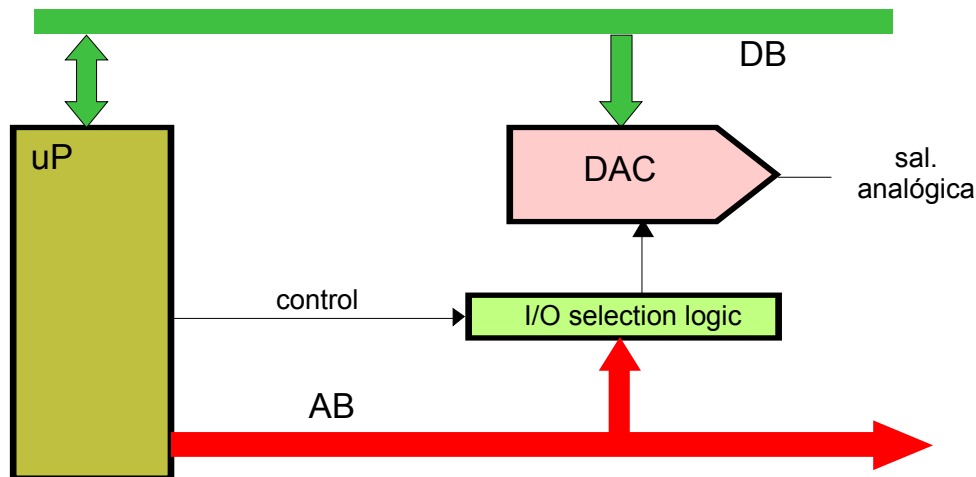
Cuando se aplica la tensión de referencia al terminal V_{REF} in, se producirá una corriente a través de la resistencia que une este terminal con el terminal bipolar OFFSET (sum OUT), la cual se sumará a la corriente de salida de la escalera, en el terminal de la entrada no inversora del OA de salida. Si consideramos por ejemplo, que la corriente de salida de la escalera es de 0 mA., y que la V_{REF} in es de +5v, tenemos que la corriente que va a circular por la resistencia de 5K que está en paralelo con el diodo, será de 1 mA., corriente hacia el OA de salida. Esta corriente da como resultado que en la salida de este OA, es decir, en el terminal DAC out se tenga una tensión de -5v. Si los conmutadores del DAC, son puestos para el máximo valor digital, la corriente de salida de la escalera será de 2 mA., con sentido desde el OA de salida hacia la escalera (observar que la escalera está alimentada con una tensión negativa V_{CC-} , respecto a masa).

La corriente neta en el OA será por lo tanto en este caso, de valor $2 - 1 = 1$ mA., y de sentido contrario al que tenía la corriente en el primer caso, dando como resultado una tensión de +5v a la salida del DAC.



El circuito de la figura muestra el circuito en la configuración para una aplicación típica de DAC bipolar de -5 a +5 voltios.

6.1.6-CONEXIÓN EN UN SISTEMA MICROPROCESADOR.

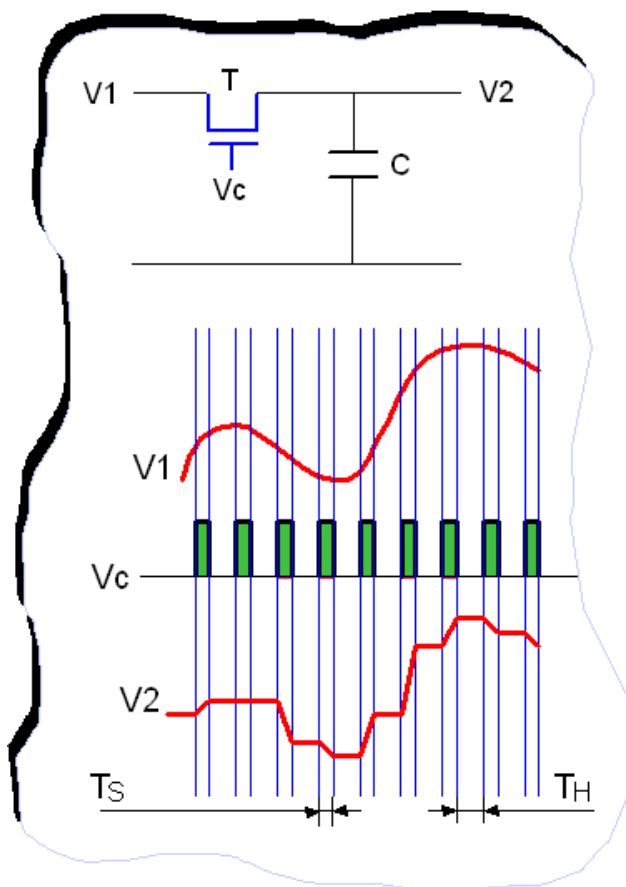


- ❑ En la figura se muestra la conexión básica de un DAC en un sistema microprocesador. Se debe tener en cuenta el tamaño del bus de datos del sistema cuando este no es coincidente con las líneas digitales del DAC, sobre todo a la hora de realizar la programación, de tal manera que la información destinada al convertidor, esté en las líneas del DB que están unidas al DAC.
- ❑ La lógica de selección se encarga de seleccionar el DAC en una dirección definida del mapa de memoria. Existen convertidores con registros internos que pueden ser programados para realizar diferentes funciones. En este caso, el chip ocupará un rango de direcciones del mapa de memoria, según el número de registros, de tal manera que cuando se quiera acceder a un registro en particular, este tendrá una dirección definida. Se debe procurar que todos los registros internos estén en direcciones consecutivas.
- ❑ En el caso del ejemplo anterior, el terminal de selección de chip es el de LE#. El microprocesador pone en el bus de datos el valor digital a convertir, e indica mediante el bus de direcciones el elemento a quien va dirigida esa información. Hay que tener en cuenta que todo este proceso debe estar muy bien sincronizado pues los valores de tensión en los buses, tarda un tiempo en estabilizarse, hasta que esa información puede ser tenida como válida.

6.2-CIRCUITO DE MUESTREO Y RETENCIÓN (SAMPLE & HOLD-S/H)

- ❑ Los circuitos S/H (circuitos de muestreo y retención) se encargan de tomar una muestra de una señal analógica (información analógica) y mantener esa muestra, durante el tiempo adecuado para poder convertirla en algún tipo de señal de control. Se trata pues de una especie de “memoria analógica”.
- ❑ El uso principal que se da a un circuito S/H, es como elemento de entrada en circuitos DAC. El circuito de S/H, es el encargado de tomar una muestra de la señal analógica a convertir, mantenerla estable y aplicar dicha muestra al ADC durante el tiempo que este necesita para realizar la conversión. La conversión A/D puede verse afectada si el voltaje analógico se aplica directamente a la entrada del ADC y varía durante el proceso. Un circuito S/H mejora la estabilidad de la conversión, manteniendo cada muestra en un valor estable, para que el ADC convierta cada una de ellas en un valor digital.
- ❑ En funcionamiento normal, la señal analógica de entrada de un ADC es *muestreada* (toma de una muestra) continuamente, a intervalos de tiempo (intervalos de muestreo, T_s) separados, que vienen definidos por una señal de control.
- ❑ Para que una señal analógica, pueda ser recompuesta a partir de una serie de muestras tomadas de ella, la frecuencia con la que se debe tomar las muestras, debe ser al menos el doble de la máxima frecuencia presente en la señal a muestrear (*TEOREMA DEL MUESTREO DE NYQUIST*).

IDEA BÁSICA:



V1: V a muestrear.

Vc: Impulsos de control.

T: Interruptor electrónico.

V2: V muestreada.

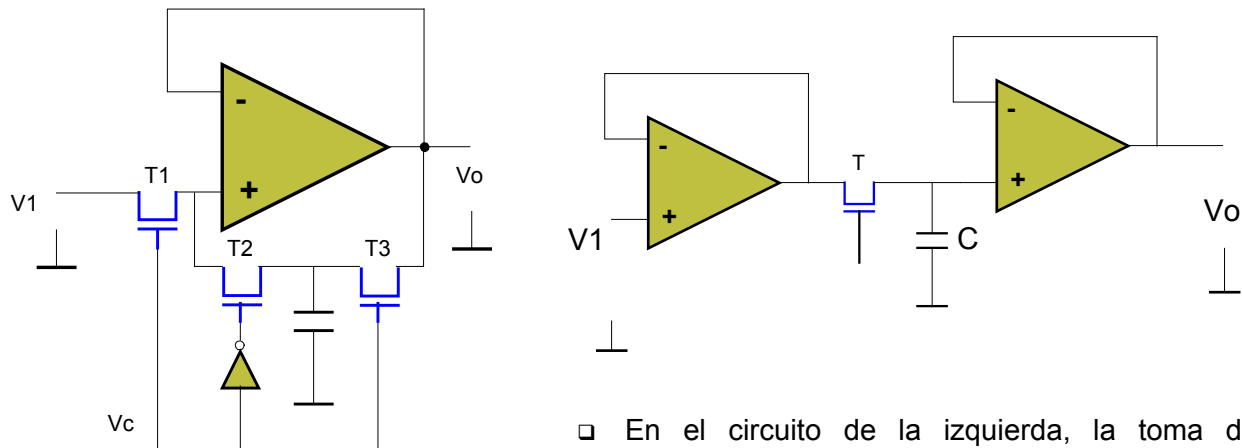
T_s : Tiempo de muestreo.

T_H : Tiempo de retención.

- ❑ Mediante unos impulsos de control o muestreo, se van tomando en tiempos definidos, valores de la señal analógica de entrada al circuito S/H.
- ❑ Cuando se aplica un impulso al transistor T, este conduce y el condensador se carga o se descarga según sea la V1 de entrada (muestreo). Cuando no se aplica impulso, T está en corte, por lo tanto el condensador mantiene la carga (retención).
- ❑ Cuando la señal muestreada, es

aplicada al convertidor analógico digital, solamente la parte de la señal que aparece en los tiempos de retención, será la que se convertirá en un equivalente digital. Se observa que durante la conversión, la señal a convertir (muestra) se mantiene constante.

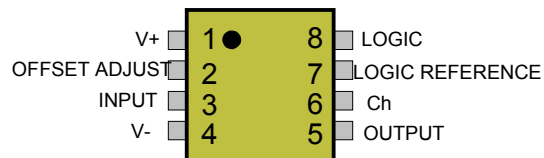
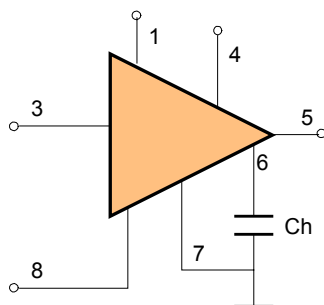
6.2.1-CIRCUITO PRÁCTICO.



- En el circuito de la izquierda, la toma de muestras es controlada mediante la señal de control Vc, de tal manera que cuando se activa esta para la toma de una muestra, el T1 y el T3 se ponen en ON, cargando el condensador a la tensión V1 presente en ese instante en la entrada. La influencia que este circuito tiene sobre la muestra de entrada es teóricamente nula pues la señal de entrada “ve” una impedancia infinita al estar aplicada al seguidor de tensión. Por otro lado, el condensador se carga a través de una baja impedancia (salida del OA), con lo que se gana en velocidad. Cuando la Vc es puesta a nivel bajo, entramos en el tiempo de retención, en el cual los transistores T1 y T3 son puestos en OFF y el T2 en ON, de esta manera la carga del condensador (muestra) es entregada a la etapa o circuito siguiente a través del seguidor de tensión.
- El circuito de la derecha tiene una forma de trabajo análoga. El transistor T funciona como un interruptor electrónico. En modo muestreo, se pone en ON dando paso a la señal de entrada hacia el condensador. En modo retención, el transistor se pone en OFF, haciendo que el primer amplificador queda aislado del segundo y del condensador. La tensión del condensador es aplicada a la salida del circuito a través del amplificador de ganancia unidad.

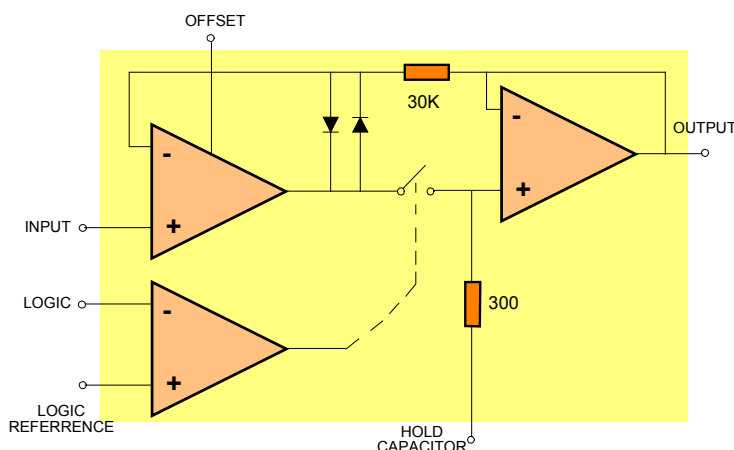
EJEMPLO COMERCIAL.

5537



Internamente, se basa en dos amplificadores operacionales que funcionan como amplificadores de ganancia unidad cuando el circuito está en el modo de muestreo, y un mecanismo de control de conmutación.

El primer amplificador de entrada de tensión analógica, tiene transistores bipolares en la entrada con lo cual se consigue un voltaje de offset mínimo. El segundo amplificador, posee transistores JFET con lo cual se consigue una impedancia de entrada elevada evitando así corrientes de fuga procedentes del condensador de retención (Ch). Las características indican que la corriente de fuga en el modo retención es de 6pA.



Cuando el interruptor de muestreo está abierto (modo retención), los diodos cierran el lazo alrededor del amplificador de entrada para evitar que se sature.

El control del interruptor es llevado a cabo por niveles lógicos de tensión externos. El terminal *logic reference* (7), puede ser puesto a diferentes valores de tensión, dependiendo del tipo de circuito al que está conectado el S/H. De esta manera se le hace

compatible con diferentes niveles lógicos. El dispositivo de conmutación trabaja a un voltaje umbral de 1,4 V. El modo de muestreo (interruptor cerrado), se consigue con un nivel alto de tensión en el terminal 8. Esto significa que la tensión en dicho terminal debe ser 1,4 v mayor que en el terminal 7. En la figura se muestra la conexión típica de este circuito.

6.2.2-TÉRMINOS S/H.

ACQUISITION TIME: El tiempo necesario para adquirir un nuevo voltaje analógico de entrada con un paso de salida de 10 v. (para el circuito del ejemplo es de 4μs)

APERTURE DELAY TIME: El tiempo que va desde la orden de retención hasta que el interruptor se abre.

APERTURE TIME: Es el retardo necesario entre la orden de hold y una transición analógica en la entrada para que dicha transición no afecte a la salida retenida.

BANDWIDTH: Es la frecuencia a la cual la ganancia baja 3 dB de su valor en DC. Se mide en el modo de muestreo con una pequeña señal senoidal que no exceda el límite de slew rate.

DYNAMIC SAMPLING ERROR: Es el error introducido en la salida retenida debido al cambio en la entrada analógica cuando la orden de retención es dada. Este error se expresa en mV para un valor de capacidad del Ch y slew rate de entrada dados.

EFFECTIVE APERTURE DELAY: Es la diferencia de tiempo entre la orden de retención y el instante en el cual la señal de entrada está en el voltaje de retención.

FIGURE OF MERIT: Es la relación entre la corriente disponible durante el modo de muestreo y la corriente de fugas durante el modo de retención.

GAIN ERROR: La relación entre la oscilación del voltaje de salida y la oscilación del voltaje de entrada en el modo de muestreo expresado como una diferencia porcentual. (0,002% para una carga de 2K)

HOLD MODE DROOP: Es el cambio de voltaje de salida por unidad de tiempo cuando el circuito está en modo retención. Suele representarse en V/s.

HOLD SETTING TIME: El tiempo necesario para que la salida se estabilice dentro de 1 mV de su valor final después de la orden de retención.

HOLD STEP: Es la variación de voltaje en la salida cuando se conmuta del modo muestreo al modo retención con una tensión analógica, continua y estabilizada de entrada. (0,5 mV para un Ch de 0,01µF).

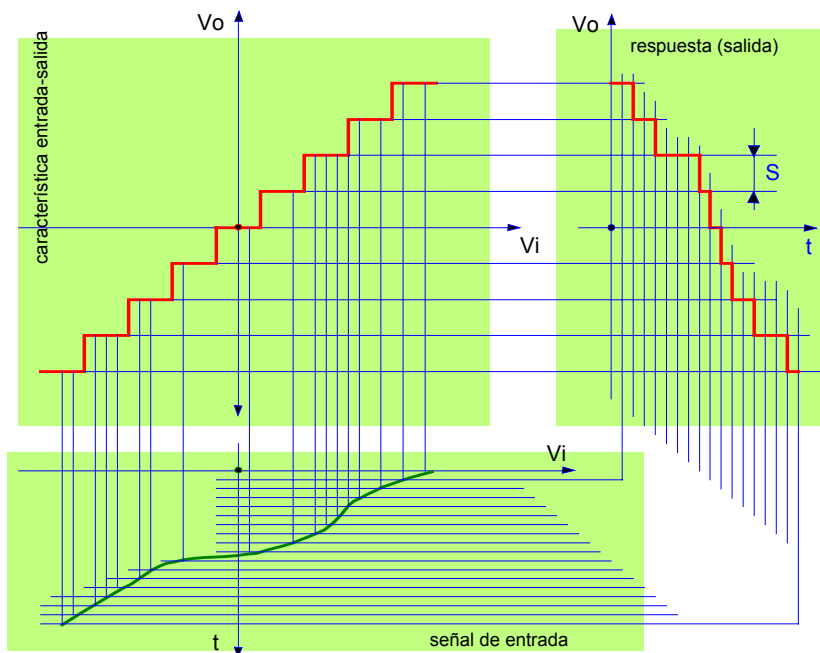
SAMPLE TO HOLD OFFSET ERROR: Es la diferencia en el voltaje de salida entre el tiempo en el que el interruptor comienza a abrirse y el tiempo cuando la salida se ha estabilizado completamente.

SLEW RATE: Es la máxima velocidad a la cual la salida del circuito puede cambiar (V/µs)

THRESHOLD LEVEL: Es el nivel que hace que el control del interruptor cambie de estado. (1,4 V)

6.2.3-CUANTIFICACIÓN:

- El proceso de cuantificación consiste en la asignación de un rango de valores infinito, a un rango de valores finito. Una señal analógica cuando sea convertida a digital, estará representada por un numero de valores concreto. Este número de valores dependerá del numero de bits que intervengan en la conversión; para un número n de bits, el número de valores diferentes que podemos obtener será de 2^n , por lo tanto, el rango de valores de la señal analógica será representado por los 2^n valores. Esto significa que a cada valor



concreto de la señal digital le va a corresponder un rango de valores de la señal analógica. Por lo tanto en el proceso de digitalización de muestras se debe efectuar una aproximación. A este proceso de aproximación se le llama cuantificación.

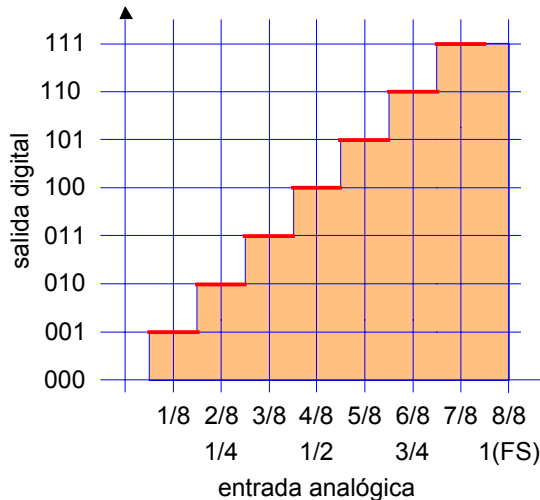
En la figura se puede observar este proceso. La característica entrada-salida del cuantificador es una

escalera. La señal analógica presente en la entrada del cuantificador, tendrá una forma escalonada como la dada en la figura en la gráfica "respuesta" una vez haya sido cuantificada. Esta forma se debe como ya se ha indicado, a que la salida del cuantificador permanece constante para un rango de valores de la señal analógica de entrada, es decir, mientras la tensión de entrada varía uniformemente en todo su margen, la señal cuantificada se mantiene en uno u otro de los posibles niveles fijos que puede tener.

INTERVALO DE CUANTIFICACIÓN.

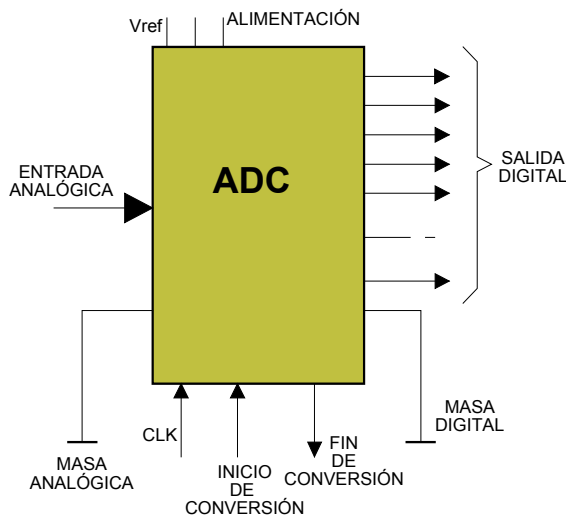
- El intervalo de cuantificación viene dado por la expresión $S = (V_{\max} - V_{\min}) / N$, donde $V_{\max}$ es el valor máximo de la señal analógica de entrada al cuantificador, $V_{\min}$ es el valor mínimo y N representa el número de valores con los que va a ser representada la señal analógica en la salida del cuantificador.

6.3-ADC (ANALOG TO DIGITAL CONVERTER)



- La figura muestra la curva de transferencia ideal para un ADC de 3 bits. Se puede apreciar que un valor digital de salida es válido sobre un pequeño rango de la señal analógica de entrada, no solamente en un punto; este rango se le conoce como anchura del código. Para un funcionamiento ideal, cada anchura de código (excepto en los extremos) debe ser de un LSB. El funcionamiento aceptable de un ADC estará para anchura de código que va desde $\frac{1}{2}$ LSB hasta $1\frac{1}{2}$ LSB.

TERMINALES CARACTERÍSTICOS.



- Los terminales característicos de un ADC son los representados en la figura. No todos los conversores van a tener todos estos terminales y algunos conversores, tendrán terminales diferentes.

INICIO DE CONVERSIÓN. Este terminal es usado por el microprocesador para indicar al ADC que debe comenzar la conversión de una muestra.

Cuando el ADC haya finalizado dicha conversión, se lo hará saber al μP mediante el terminal FIN DE CONVERSIÓN.

V_{ref} . Esta tensión puede venir de un circuito interno del propio chip. Del valor de esta tensión va a depender la equivalencia entre la

tensión de entrada y el valor digital en la salida.

En los casos en los que el ADC contenga circuitos secuenciales internos, deberá tener también una señal de reloj para poder gobernar dichos circuitos. Esta señal puede también venir del propio chip.

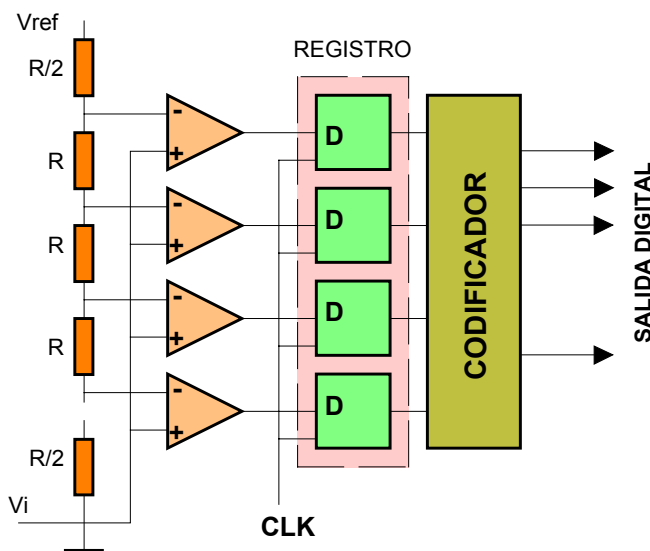
Internamente, las masas de la parte analógica y digital son diferentes. Generalmente estas masas deben estar unidas en un único punto en el circuito de aplicación.

- El proceso de conversión A/D es generalmente más complejo y largo que el de D/A, y lleva implícitos cuatro pasos:

6.3.1-MUESTREO – MANTENIMIENTO – CUANTIFICACIÓN – CODIFICACIÓN.

La operación de muestreo y mantenimiento es realizada simultáneamente por el circuito **S/H** (*Sample And Hold*).

La operación de cuantificación y codificación consiste en aproximar cada muestra, a un valor determinado dentro de un conjunto finito de valores del que se compone la información digital. La cuantificación es una operación intrínseca al propio funcionamiento del conversor.

TIPOS DE ADC.**1 .-PARALELO (FLASH)****2 .-REALIMENTACIÓN:**
CONTEO.
TRACKING (ASCENDENTE-DESCENDENTE).
APROXIMACIONES SUCESIVAS.**3 .-INTEGRACIÓN:**
RAMPA SENCILLA
RAMPA DOBLE
RAMPA TRIPLE**6.3.2-MÉTODO PARALELO. FLASH ADC.**

- Este tipo de convertidor es el que proporciona la mayor velocidad de los existentes actualmente, a costa de la resolución. La mayoría de los convertidores de este tipo están limitados a una resolución de 8 bits.

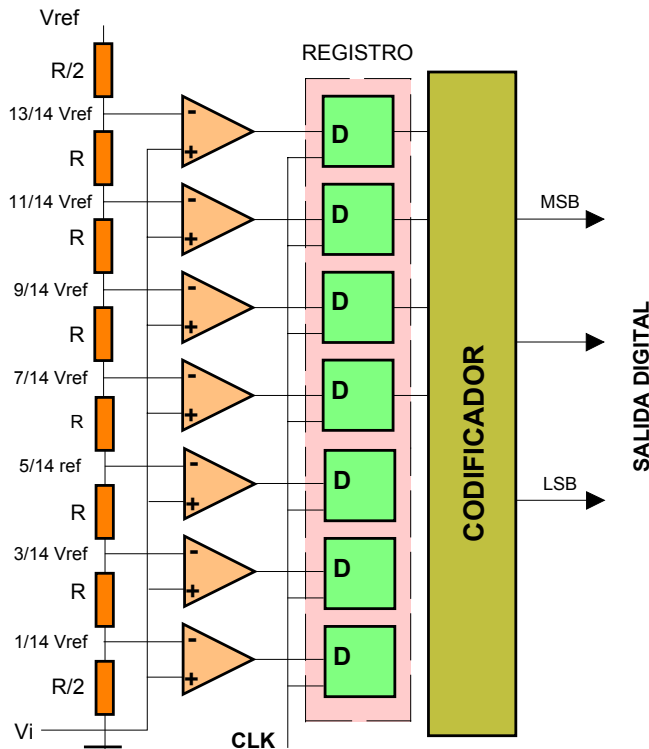
El FLASH ADC está formado por una red de resistencias y una matriz de $2^n - 1$ comparadores, siendo n el número de bits del ADC. El proceso de cuantificación se lleva a cabo mediante la comparación de la señal analógica a convertir, con unos valores fijos de tensión impuestos por el voltaje

de referencia. Estos valores fijos de tensión están aplicados a la entrada inversora de los comparadores, haciendo que estos mantengan su salida en un determinado valor. La señal analógica de entrada hará bascular unos u otros comparadores, según su valor esté por encima o debajo del valor que cada comparador tiene fijado como referencia. El valor de cero se da cuando ninguno de los comparadores ha basculado.

- Un codificador se encarga de convertir a un código digital conveniente de n bits, los $2^n - 1$ valores dados por los comparadores.
- Un registro mantiene la salida hasta que se acepta otra muestra. (Generalmente la muestra proviene de un *S/H CIRCUIT*, aunque si la señal analógica a convertir es de variación lenta con respecto al tiempo, puede aplicarse directamente).

Se puede observar en el circuito dado, que las dos resistencias de los extremos son de valor diferente al resto. Esto es una práctica común en este tipo de convertidores, y con ello se busca una anchura de código determinada en los extremos de la escala. Para el circuito dado vemos que las resistencias de los extremos son de un valor mitad que las demás por lo que el paso al primer valor digital partiendo del cero, ocurrirá para un rango de la tensión analógica de entrada mitad que el paso entre dos códigos consecutivos cualesquiera, pertenecientes al rango de entrada. Este rango de entrada viene determinado por la tensión de referencia.

Ej. Supongamos que tenemos un FLASH ADC de 3 bits.



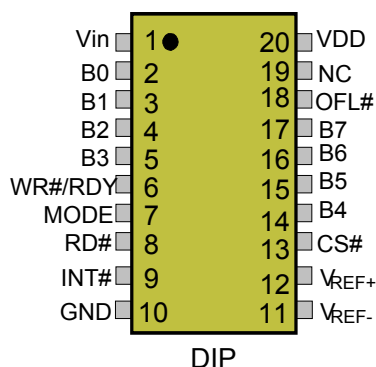
La salida digital correspondiente a una entrada analógica variable entre el valor de masa y la tensión de referencia es la dada en la siguiente tabla:

Tensión analógica V_i	Salida digital
$V_i < V_{ref}/14$	000
$V_{ref}/14 < V_i < 3/14 V_{ref}$	001
$3/14 V_{ref} < V_i < 5/14 V_{ref}$	010
$5/14 V_{ref} < V_i < 7/14 V_{ref}$	011
$7/14 V_{ref} < V_i < 9/14 V_{ref}$	100
$9/14 V_{ref} < V_i < 11/14 V_{ref}$	101
$11/14 V_{ref} < V_i < 13/14 V_{ref}$	110
$13/14 V_{ref} < V_i < 14/14 V_{ref}$	111

□ Como se ha dicho anteriormente, el problema que presenta este tipo de conversor es el límite en cuanto a su resolución. Esto se debe a que el hardware necesario, debe ser incrementado al doble por cada BIT adicional que se añade a su resolución. Así, para un conversor de 6 bits serán necesarios 63 comparadores, 64 resistencias, incremento en el registro, en el codificador, etc. Para un conversor de 8 bits, se necesitarán 255 comparadores, 256 resistencias...

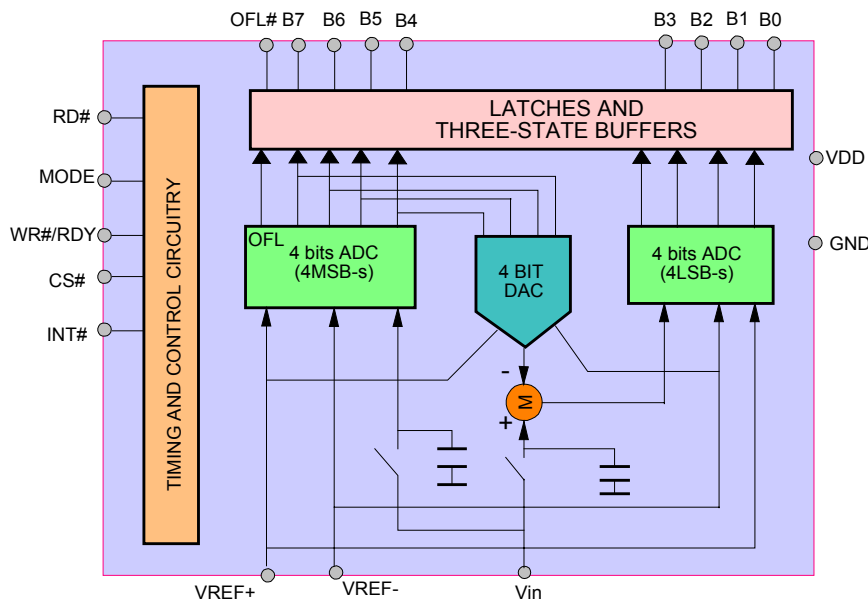
6.3.3-HALF-FLASH

□ Otro tipo de ADC de la familia es el que hace uso de técnicas *half-flash*. Este sistema consiste en conectar en cascada dos FLASH ADC-s de n bits para obtener un FLASH ADC de $2n$ bits, todo ello de forma integrada en un chip. Con esto se consigue un ahorro en el hardware, a expensas de la velocidad de ejecución.



□ Con este sistema, para conseguir un ADC de 8 bits se parte de dos ADC-s de 4 bits con lo cual en vez de necesitar los 255 comparadores y el incremento correspondiente del resto de la circuitería, se necesita solamente el doble de la que posee un ADC de 4 bits.

Tomando por ejemplo el caso del ADC de 8 bits de la figura, cada uno de los FLASH ADC está constituido por 15 comparadores que comparan la tensión analógica de entrada con la escalera de referencia (los diferentes valores de tensión en los diferentes puntos de la red de resistencias), para dar un resultado de 4 bits. Para obtener los 8 bits, un conversor ADC (el MS), realiza la conversión de la tensión de entrada, para obtener los 4 bits más significativos del resultado total. Un DAC interno cuya entrada son los valores de los 4 MSB (Most Significant Bits) y con un tamaño de paso igual al del convertidor MS, reconstruye una aproximación al voltaje de entrada. Esta tensión procedente del DAC, es restada de la tensión analógica de la entrada, y el resultado es convertido por un segundo ADC de 4 bits (el LS), dando este último el valor de los cuatro bits menos significativos del resultado.



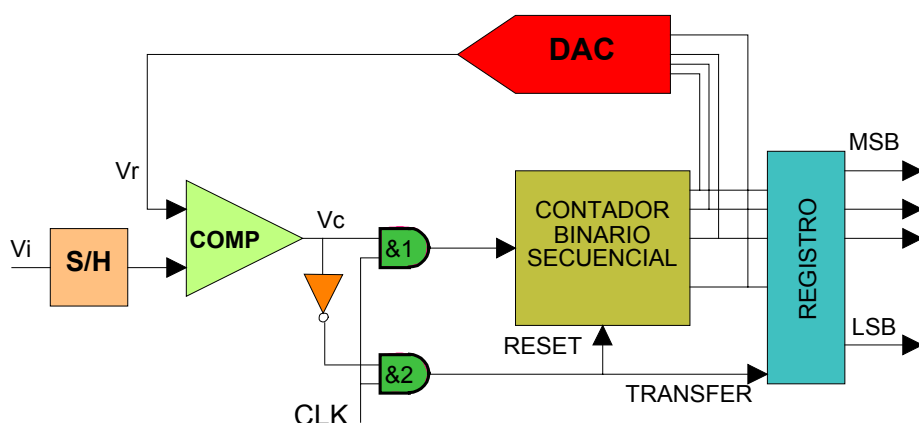
El tamaño de paso del convertidor MS, es 16 veces más grande que el tamaño de paso del convertidor LS; este último convertidor lo que realmente hace es convertir a digital, el valor de la tensión analógica de error que ha cometido el convertidor MS.

El funcionamiento de forma general, sin entrar en matices en cuanto al ajuste de la escala para disminuir el error, sería el siguiente: si

consideramos que el margen de entrada de la tensión analógica es de 0 a 5 voltios, el tamaño de paso del ADC será de 19,6 mV. Este tamaño de paso es el mismo que para el convertidor LS. Para el convertidor MS el tamaño de paso será de 313,6 mV. De esta manera, para una tensión de entrada de 2,456 voltios, la salida del convertidor MS será 0111, que corresponde a un valor analógico a la salida del 4 BIT DAC interno de 2,195V. Por lo tanto el valor a transformar por el convertidor LS será de $2,456 - 2,195 = 0,261$ voltios, que dará como resultado 1101. Tomando el resultado de los 8 bits del ADC tendremos el valor digital 01111101 (125) que equivale a una entrada analógica de $125 \times 19,6 \text{ mV} = 2,45$ voltios.

- El ADC comercial de la figura, posee también otros terminales para facilitar la conexión a un sistema basado en microprocesador. También un terminal que indica cuando la tensión analógica de entrada ha rebasado un valor definido por la V_{REF+} .

6.3.4-ADC DE RAMPA DIGITAL. CONTEO.

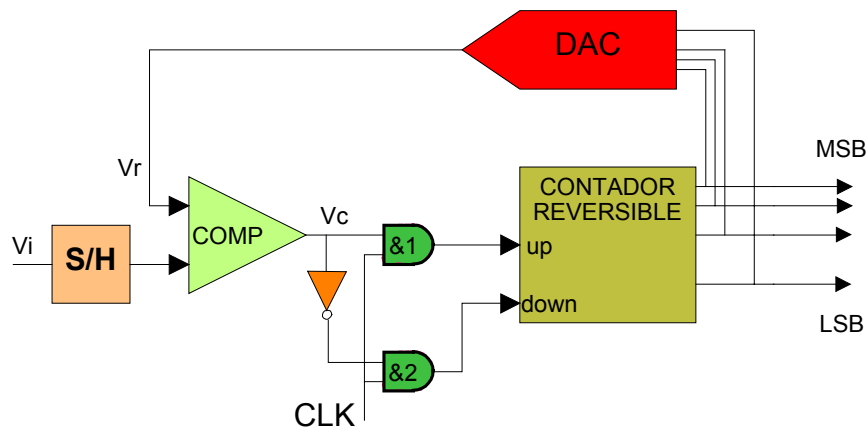


El funcionamiento de este ADC está basado en un contador binario ascendente. Con cada muestra analógica a convertir, el contador se inicializa y comienza la cuenta ascendente. Esta cuenta va a

continuar hasta que su valor, una vez pasado a forma analógica mediante un DAC interno se iguale a la muestra analógica de entrada. Un comparador se va a encargar de vigilar cuando el valor de la cuenta dada por el contador y la muestra de entrada, son iguales. En ese momento, el valor del contador es pasado a un registro de salida para dar el equivalente digital de la muestra analógica de entrada, y posteriormente, el contador se inicializa para la conversión de una nueva muestra.

- El proceso de conversión esta controlado por un reloj. Hay que observar que el tiempo de conversión va a ser función de la muestra a convertir.

6.3.5-CONTEO ASCENDENTE DESCENDENTE.

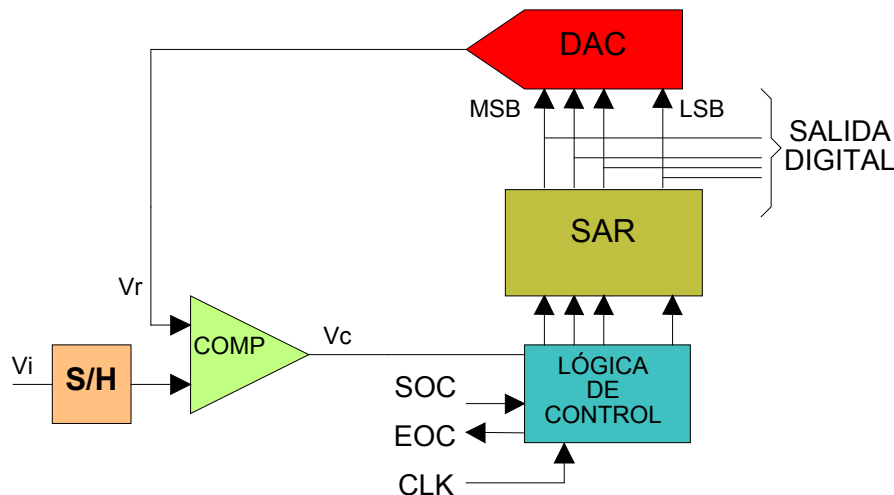


Este ADC está basado en el mismo principio de funcionamiento que el ADC anterior. El contador cuenta en sentido ascendente o descendente, según que el valor de la muestra de entrada sea mayor o menor respectivamente, respecto de la salida del contador una vez convertida en valor analógico. Con este método se consigue

acelerar el proceso de conversión respecto del anterior esquema, pues el contador no necesita partir siempre desde cero para la conversión de las muestras analógicas.

- Hay que observar que también en este caso el tiempo de conversión, es función directa del valor de la muestra a convertir.

6.3.6-ADC DE APROXIMACIONES SUCESIVAS.



El núcleo de este convertidor es un registro de control, también llamado registro de aproximaciones sucesivas (SAR). En el comienzo del ciclo de conversión (START OF CONVERSION, SOC) el MSB del registro, se carga con un 1, y los bits restantes se ponen a 0, es decir, el SAR queda 1000....0. Este

valor contenido en el registro en ese instante es la mitad, del número máximo de valores que se puede tener en el SAR con el número de bits dados:

$n \text{ bits} \Rightarrow 2^n \text{ posibles valores.}$

Valor contenido en el registro en el inicio de conversión = $2^n / 2$

Este valor contenido en el SAR, es convertido a analógico (Vr) mediante un DAC interno. Si el valor de la muestra es menor que Vr (a), el MSB del SAR se pone a 0 (valor del registro en ese instante: 0000...0), en caso contrario (b) permanece en 1 (1000....0). El siguiente impulso de reloj pone a 1 el BIT de peso inmediatamente inferior al MSB, [(a) 0100....0, (b) 1100...0], y se procede a realizar una nueva comparación de la muestra con el valor actual del SAR una vez

pasado este valor a tensión analógica (V_r). La secuencia de operaciones que sigue es la misma que la explicada para el MSB.

- La conversión se completa cuando se procesan todos los bits del registro, independientemente del valor de la señal analógica, por lo que este conversor tiene un tiempo fijo de trabajo: $t_c = n * T$

[t_c = TIEMPO DE CONVERSIÓN; n = N° DE BITS; T = PERIODO DEL CLOCK]

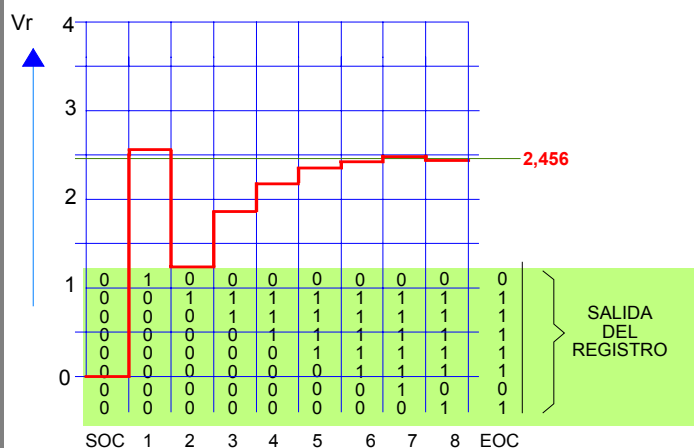
- El proceso anterior llamado ciclo de conversión, debe realizarse para cada muestra de la señal analógica de entrada.
- Este tipo de ADC es el más utilizado.

EJEMPLO.

Sea un ADC de 8 bits, con un margen de la tensión analógica de entrada $0 \leq V_i \leq 5$ v. Sea una tensión de entrada al conversor $V_i = 2,456$ voltios.

El número de pasos es $2^n - 1 = 2^8 - 1 = 255$. El tamaño de paso es $5 \text{ v} / 255 = 19,6 \text{ mV}$. Vamos a prescindir de los posibles ajustes de escala para compensación de errores, es decir, en toda la escala el tamaño de paso es el mismo. Esto quiere decir que mientras no exista una variación superior a $19,6 \text{ mV}$ en la entrada, la combinación permanecerá estable.

SAR	V_r	V_c	CLK
00000000	0	$V_r < V_i$	SOC
10000000	2,508	$V_r > V_i$	1
01000000	1,254	$V_r < V_i$	2
01100000	1,881	$V_r < V_i$	3
01110000	2,195	$V_r < V_i$	4
01111000	2,352	$V_r < V_i$	5
01111100	2,430	$V_r < V_i$	6
01111110	2,469	$V_r > V_i$	7
01111101	2,45	$V_r < V_i$	8
01111101			EOC
SALIDA DIGITAL 01111101 (2,45)			



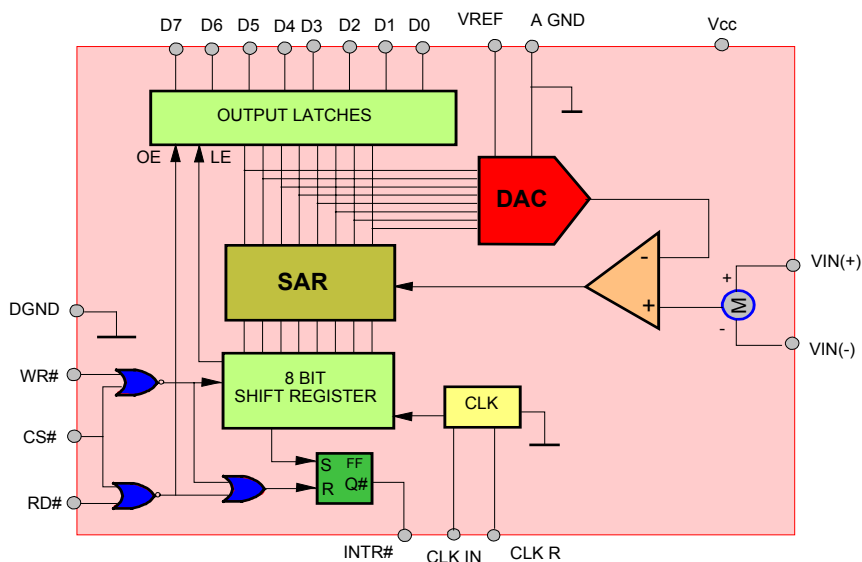
$V_r = \text{valor digital} * 19,6 \text{ mV}$

$$V_r = 128 * 19,6 \text{ mV} = 2.508 \text{ v}$$

$$V_r = 64 * 19,6 \text{ mV} = 1,254 \text{ v}$$

$$V_r = \dots$$

EOC = End Of Conversion.

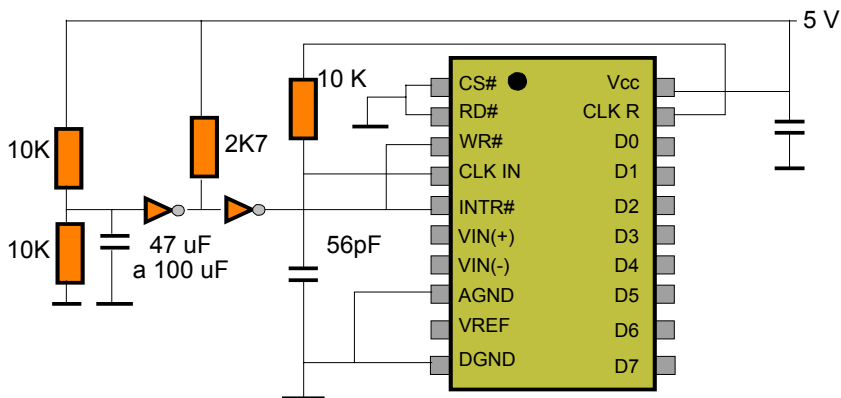


La figura muestra un ADC de tipo comercial. La conversión comienza cuando aparece un flanco activo en el terminal de WR# si el CS# es activo. Mientras las dos señales permanezcan en estado bajo, el ADC estará en estado de reset, es decir, el SAR inicializado, el terminal de salida INTR# en estado alto y el registro de 8 bits puesto a cero.

La conversión de una muestra comenzará desde uno a ocho periodos de reloj cuando cualquiera de las señales WR# o CS# haga una transición hacia el estado alto. Cuando la conversión finalice, el terminal de INTR#, se activará. Para obtener el resultado digital, se activará el terminal RD# (con CS# activo). Esta acción devolverá la línea de INTR# a su estado inactivo, y habilitará los latches de salida. Una conversión puede ser interrumpida por la acción de una nueva orden de WR#. La combinación de estos terminales hace posible una fácil conexión de este ADC a un sistema microprocesador.

El esquema de la figura muestra la configuración del circuito para una conversión continua.

CS# y RD# puestos a su nivel activo, aseguran por un lado que el circuito esté activo y por otro, que la información digital esté presente en la salida con cada conversión.

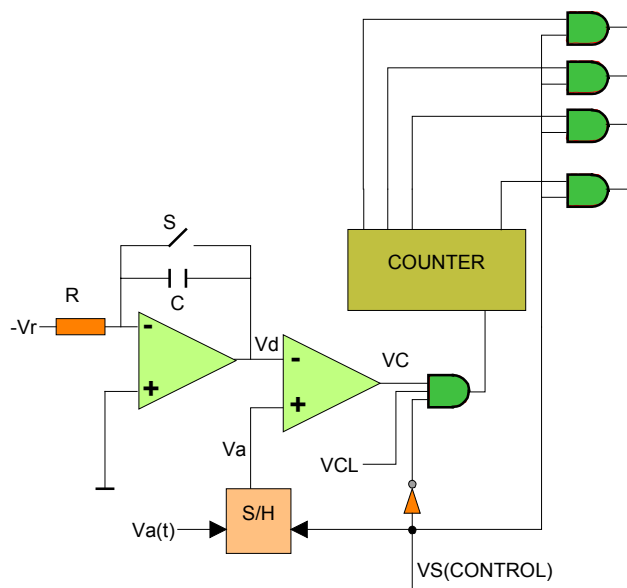


Un circuito de puesta a cero, asegura durante el inicio de operación del convertidor, que el terminal WR# se ponga a cero durante cierto tiempo para dar comienzo a la conversión. Después, es

el terminal de INTR# el que se encarga de iniciar una nueva conversión cuando termina la anterior. Hay que tener en cuenta que la activación de INTR# al final de una conversión, activa WR# para iniciar una nueva conversión, y que la activación de WR# desactiva INTR# y a la vez, esta desactivación de INTR# trae consigo la desactivación de WR#.

6.3.7-ADC DE SIMPLE RAMPA. (CONVERSIÓN TENSIÓN-TIEMPO)

El método usado por este circuito, es contar los ciclos de un reloj de frecuencia fija, durante un periodo de tiempo que varía según la tensión analógica que queremos convertir, es decir, se va a obtener una cuenta que va a ser función directa del nivel de la muestra de la tensión analógica de entrada.



Mientras el circuito está en el modo muestreo, el interruptor S del integrador, permanece cerrado descargando el condensador y dando por lo tanto una V_d igual a cero.

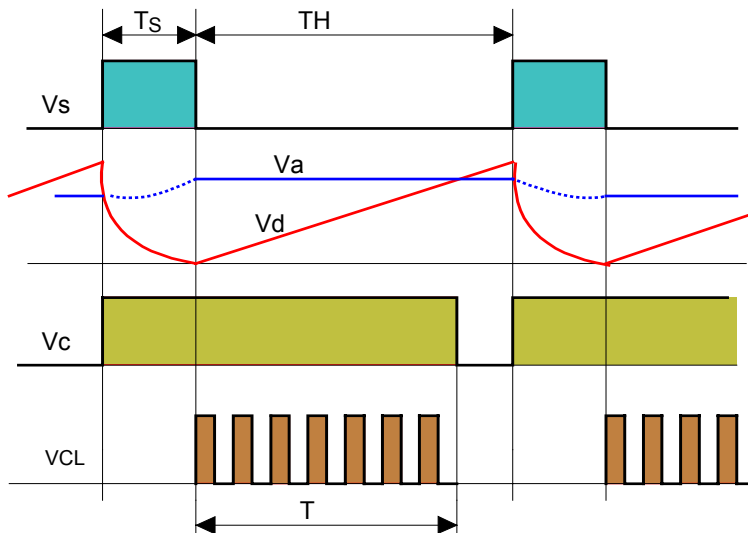
En el modo retención o conversión de la muestra V_a , S se cierra comenzando la integración de una tensión de referencia negativa ($-V_r$), que aplicada a la entrada inversora del integrador, nos dará a la salida de éste una rampa positiva (V_d).

Mientras V_d sea menor que V_a , la salida del comparador será alta, haciendo que la señal de reloj VCL, incremente la cuenta de un contador. Cuando la rampa V_d alcance el valor de la muestra V_a , el comparador cambiará de estado, haciendo que la puerta AND quede inhibida, con lo

cual no deja pasar más impulsos de reloj hacia el contador. Esto hace que el contador quede enclavado en una cuenta proporcional a la tensión V_a .

Todo el proceso está controlado por una señal de control que reinicializa el contador y el integrador después de cada conversión.

- En este tipo de convertidores, el tiempo que dura la conversión depende del valor de la muestra analógica a convertir.



*

$$\square V_d = -(t / \tau) (-V_r) = (t / \tau) V_r$$

$$\square \tau = RC$$

□ En un tiempo T,

$$V_d = V_a$$

$$V_a = (T / \tau) V_r.$$

□ La cuenta registrada en T será:

$$f * T = f * (\tau * V_a) / V_r = K * V_a$$

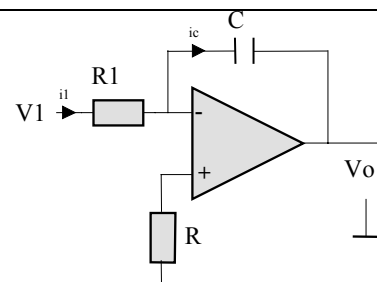
*

$$i_c = C (dv_c / dt); dv_c = i_c dt / C \quad v_c = (1/C) \int i_c dt$$

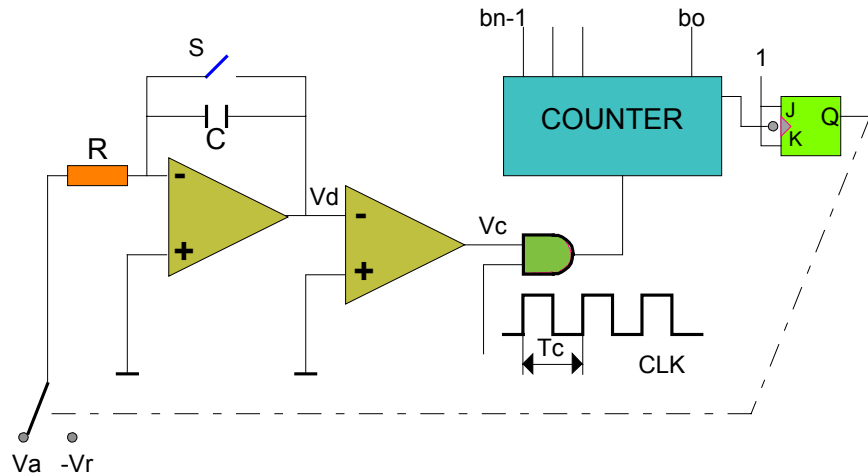
$$i_1 = i_c \Rightarrow V_1 / R_1 = C (dv_c / dt) = C (d(-V_o) / dt)$$

$$-[1/(R_1 C)] V_1 = d V_o / dt; \quad V_o = -1/(R_1 C) \int V_1 dt$$

$$\text{para } V_1 = \text{CTE} \Rightarrow V_o = -[V_1 / (R_1 C)] t$$



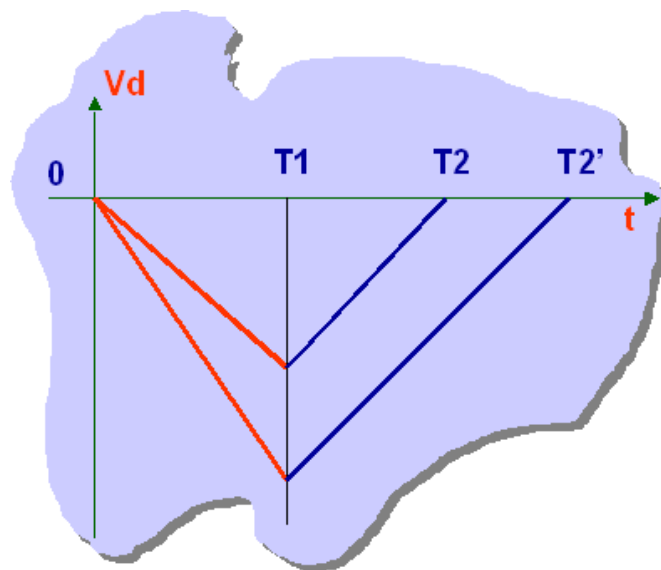
6.3.8-ADC DE DOBLE RAMPA.



La muestra de la tensión analógica a convertir (V_a), es aplicada durante un tiempo constante (T_1) a la entrada inversora de un integrador. Esta tensión, dará a la salida del integrador una rampa que tendrá más o menos pendiente según que el valor de la muestra sea mayor o menor.

Al finalizar T_1 , se aplica al integrador una tensión de referencia ($-V_r$) que dará una rampa con una pendiente determinada y constante, independiente del valor de la muestra.

La tensión de salida del integrador, es comparada con una tensión de cero voltios, dando a la salida del comparador un valor positivo siempre que V_d sea negativo. Este estado de salida del comparador sirve para habilitar o inhibir una puerta AND que dejará pasar o no, los impulsos de reloj a un contador.



El valor de la cuenta del contador, será proporcional a la muestra de la tensión analógica de entrada. Cuanto mayor sea ésta, más pendiente tendrá la tensión en la salida del integrador y por lo tanto más tiempo estará la salida del comparador en estado alto habilitando la puerta AND con lo cual mayor será la cuenta.

La figura muestra la tensión de salida del integrador para dos muestras diferentes.

□ $t = 0$: R está conectada a la muestra V_a .

□ $0 < t < T_1$: El contador da una vuelta completa a su cuenta. $T_1 = 2^n \times T_{\text{clock}}$

$$V_d = -(t / \tau) V_a.$$

□ $t = T_1$: $V_d = -(2^n \times T_{\text{clock}} / \tau) V_a$

El FF J-K, cambia de estado, haciendo que el conmutador cambie a $-V_r$. A partir de ese instante,

$$V_d = -[(t-T_1) / \tau] \cdot (-V_r) - (T_1 / \tau) \cdot V_a = [(t-T_1) / \tau] \cdot V_r - (T_1 / \tau) \cdot V_a$$

□ **T1 < t < T2:**

El contador comienza una nueva cuenta hasta que Vd tienda hacia un valor positivo. En este momento se inhibe el reloj. El valor que tiene el contador en esta segunda cuenta es proporcional a Va.

□ **t = T2:**

$$V_d = [(T_2 - T_1) / \tau] * V_r - (T_1 / \tau) * V_a = 0$$

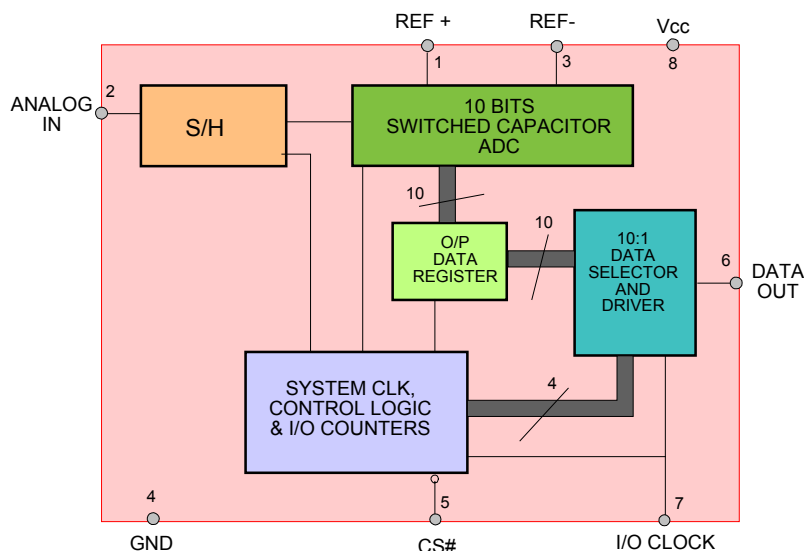
$$[(T_2 - T_1) / \tau] * V_r = (T_1 / \tau) * V_a$$

$$(T_2 - T_1) = (V_a / V_r) T_1 = (V_a / V_r) 2^n T_{\text{clock}}$$

El valor de la cuenta en T2 será: $f_{\text{clock}} \times (T_2 - T_1) = (T_2 - T_1) / T_{\text{clock}} = (V_a / V_r) 2^n = K V_a$

- En este tipo de conversores, el tiempo que dura la conversión es función de la muestra de la señal analógica a convertir. Cuanto mayor sea ésta, mayor es el tiempo de conversión.

6.3.9-ADC SERIE.



Existen en el mercado convertidores cuya forma de trabajo es serie.

El método de conversión utilizado es el de aproximaciones sucesivas, pero en este caso es un sistema de aproximación analógico basado en la carga y descarga de una matriz de condensadores y la distribución de esta carga en toda la matriz.

Cada una de estos condensadores, puede ser puesto a una tensión de

referencia positiva, negativa o puede ser puesto a la tensión de entrada mediante una serie de interruptores.

FUNCIÓN DE LOS TERMINALES		
NOMBRE	I/O	DESCRIPCIÓN
ANALOG IN	I	Entrada para la tensión analógica a convertir
CS#	I	Chip Select. Una transición hacia el estado activo, resetea los contadores internos, controla y habilita los terminales DATA OUT y I/O CLOCK. Una transición hacia su estado inactivo, inhabilita I/O CLOCK.
DATA OUT	O	Esta salida permanece en estado de alta impedancia cuando el terminal CS# está en estado alto, y es activa cuando CS# está en estado bajo. En su estado activo, DATA OUT pasa de estar en alta impedancia a tomar el nivel correspondiente al MSB de la conversión anterior. El siguiente flanco de bajada del I/O CLOCK, lleva el nivel del siguiente MSB a la salida DATA OUT, así van apareciendo los demás bits hasta el LSB que aparece en DATA OUT con el noveno flanco de bajada de I/O CLOCK. En el décimo flanco de bajada de I/O CLOCK, DATA OUT es llevado a un nivel lógico bajo.
GND Vcc	I	La masa para la circuitería interna y la alimentación positiva.
I/O CLOCK	I	Reloj del convertidor. Realiza las siguientes funciones:

		-En el tercer flanco de bajada del I/O CLOCK, el voltaje analógico de entrada comienza a cargar la matriz de condensadores y continua hasta el décimo flanco de bajada. -Desplaza los nueve bits restantes de la conversión previa a DATA OUT. -Transfiere el control de la conversión al controlador de estados interno en el décimo flanco de bajada
REF+		El voltaje máximo de referencia (generalmente Vcc). El máximo rango de voltaje a aplicar en la entrada, está determinado por la diferencia entre REF + y REF -
REF-		El voltaje mínimo de referencia (generalmente masa)

6.3.10-ESPECIFICACIONES ADC

❑ RESOLUCIÓN.

La resolución puede venir dada por el número de bits de la palabra digital de salida. De forma indirecta, también se refiere al número de códigos discretos de salida en que se puede dividir la señal analógica de entrada.

También se puede expresar como la variación de la tensión analógica de entrada necesaria para variar la salida entre dos códigos adyacentes (tamaño de paso).

❑ RANGO ANALÓGICO DE ENTRADA.

Valores entre los que puede estar comprendidos la tensión analógica de entrada.

❑ TIEMPO DE CONVERSIÓN.

Indica el tiempo transcurrido entre el instante en que se da la orden de conversión y la aparición a la salida del convertidor de la completa representación digital del valor analógico de entrada.

❑ SALIDA DIGITAL. Triestado, TTL, MOS, ECL...

❑ PRECISIÓN: La precisión del ADC, viene dada por diferentes parámetros:

*ERROR DE LINEALIDAD (E_L –Linearity Error): Nos indica la diferencia entre el valor real del paso y el valor nominal del paso después de que el error de offset y el error de ganancia hayan sido ajustados a cero.

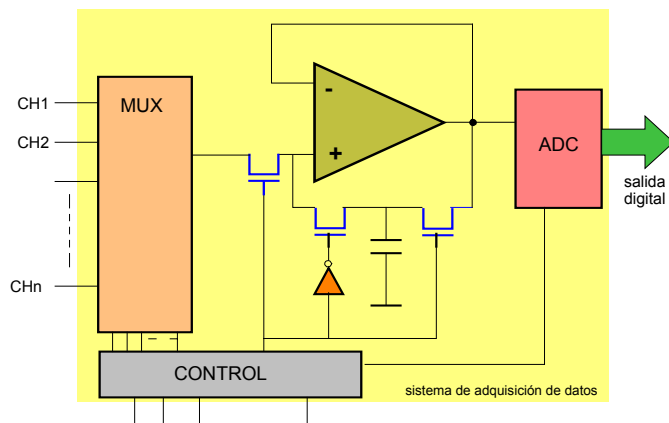
*ERROR DE ESCALA COMPLETA. (E_{FS} – Full Scale Error) Este error nos indica la diferencia entre el valor real del paso y el valor nominal especificada con relación a la escala completa o fondo de escala.

*MONOTONICIDAD (Monotonicity). Es una propiedad de la función de transferencia, que asegura el consistente incremento o decremento de la salida digital en respuesta a un incremento o decremento de la entrada analógica.

*ERROR DE CUANTIFICACIÓN. El error de cuantificación es un error inherente al proceso de conversión y se debe a que la resolución está dentro de un margen finito de valores, a los cuales deben ser aproximadas las muestras de entrada analógica.

❑ SENSIBILIDAD A LA TEMPERATURA: Indican los efectos que los cambios de temperatura ejercen sobre la salida del conversor. Se especifica como cambio en % FS / °C.

6.4-SISTEMA DE ADQUISICIÓN DE DATOS.

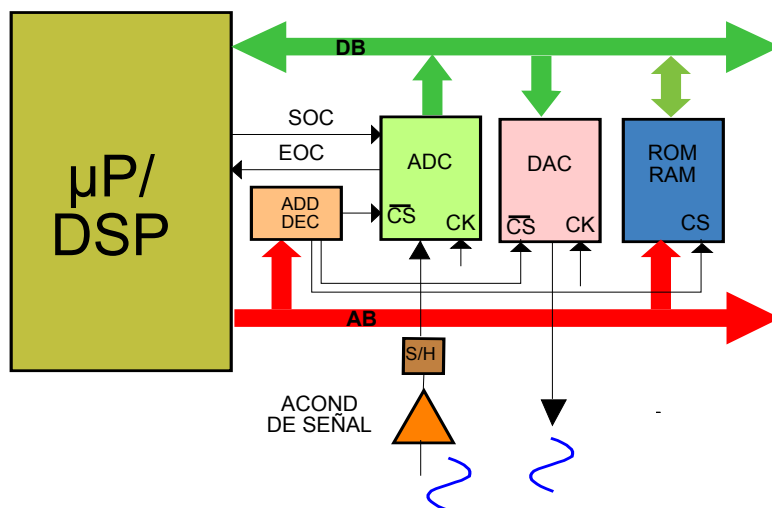


- Un sistema de adquisición de datos, puede ser definido como un convertidor analógico-digital, precedido por un multiplexor analógico para dar entrada a múltiples canales y un amplificador S/H. La misión fundamental de un sistema de estas características es la de tomar de forma periódica diferentes datos de su entorno analógico y convertirlos a un valor digital para su procesamiento para devolver después otra vez en forma analógica.

- Generalmente la información de entrada a estos sistemas, se compone de variables de carácter físico por lo que su variación es relativamente lenta. De esta manera puede ser utilizado un único conversor y un único amplificador S/H para todas las señales. El proceso de la adquisición consiste en ir multiplexando en el tiempo las diferentes señales de entrada, de tal manera que se toma una muestra de una de las señales (la que este seleccionada en ese momento), y se convierte a su valor digital. Durante el tiempo de esa conversión, es seleccionada otra de las señales de entrada para ser muestreada y convertida esa muestra a su correspondiente valor digital, y así sucesivamente. Todo el proceso debe estar perfectamente sincronizado para saber a que señal corresponde la muestra tratada en cada instante. Del proceso de sincronización y control se encarga el microprocesador DSP.

SISTEMA GENÉRICO.

DSP. (DIGITAL SIGNAL PROCESSING).



- “Ciencia que estudia la representación de señales mediante secuencias de números y el consiguiente procesamiento de dichas secuencias”.
- El núcleo de un sistema DSP es un microprocesador DSP, (*DIGITAL SIGNAL PROCESSOR*) cuya característica diferenciadora está en su mayor capacidad para procesar señales

en tiempo real respecto a otros μP.



Memorias

7

7.-MEMORIAS

VOLÁTILES (INTEGRADAS) <i>Pierden el contenido si no están energizadas (alimentadas)</i>	RAM (RANDOM ACCESS MEMORY)	SRAM	STATIC RAM
		DRAM	DYNAMIC RAM
		SDRAM RDRAM	SYNCHRONOUS DRAM
		RAM CMOS	
		NVRAM	NO VOLATILE RAM
NO VOLÁTILES <i>No pierden el contenido aunque no estén energizadas</i>	INTEGRADAS	ROM	READ ONLY MEMORY
		PROM	PROGRAMMABLE ROM
		EPROM	ERASABLE PROM
		OTPs	ONE TIME PROGRAMMABLE ROM
		FLASH	
		EAROM EEPROM	ELECTRICALLY ALTERABLE ROM ELECTRICALLY ERASABLE PROM
	MAGNÉTICAS	NÚCLEO MAGNÉTICO	(obsoletas)
		MAGNETIC BUBBLES	BURBUJAS MAGNÉTICAS
		DISCO	(HARD DISK, FLOPPY DISK)
		CINTA	STREAMER
	MAGNETO-ÓPTICAS	REGRABABLE	CD REGRABABLE (muchas veces)
	ÓPTICAS	CD-ROM	COMPACT DISK ROM
		WORM	RECORDABLE CD (una vez) WRITEABLE ONCE READABLE MANY
		DVCD (DVD)	DIGITAL VIDEO CD

7.1-CARACTERÍSTICAS GENERALES (Memorias integradas):

❑ DESCRIPCIÓN:

Este parámetro indica el tipo de memoria (SRAM, RAM CMOS, ROM, EAROM, etc.) al que hace referencia la hoja de datos. También suele indicar dentro del tipo de memoria, un modo especial de funcionamiento de ésta (entrada de datos en serie, modos de acceso...) o un modo particular de configuración externa de la misma (DIMM) o cualquier otra característica diferenciadora que el fabricante estime oportuno indicar en orden a destacar su producto respecto al de otros fabricantes.

❑ CAPACIDAD:

Indica el número de bits que pueden almacenarse en una memoria. Es común representar esta cantidad por dos números de la forma **A X B**, donde A representa el número de palabras (posiciones) que contiene la memoria y B, el número de bits de cada palabra.

La capacidad total de la memoria, dada en número de bits, es por lo tanto el producto de A x B.

❑ DISIPACIÓN DE POTENCIA:

Se da el consumo de potencia del chip, bajo diferentes modos de trabajo como puede ser el de lectura, escritura, Stand-By, etc. También suele darse el consumo máximo.

❑ VOLTAJES:

En este apartado se incluyen la tensión de alimentación del chip (máxima, típica, mínima), voltajes especiales que se necesiten para programación o borrado, voltajes correspondientes a los niveles lógicos tanto de entrada como de salida, etc.

❑ COMPATIBILIDAD:

Compatibilidad entre TTL y CMOS tanto en las entradas como en las salidas.

❑ MODOS DE OPERACIÓN:

Niveles lógicos que deben ser aplicados en los terminales correspondientes para conseguir un modo de trabajo determinado (modo normal, modo Stand-By, inhabilitación de las salidas por alta Z...).

❑ TIEMPOS EN EL CICLO DE LECTURA:

La operación de lectura (Fetch), es aquella mediante la cual, el contenido de una posición de la memoria es puesto en los terminales de salida de datos.

Suelen darse tiempos como el del ciclo de lectura, tiempos desde que se activa el chip hasta que se activan las salidas, tiempos desde que se activa la salida hasta que ésta es activa, etc. Estos parámetros, suelen ir acompañados de unos diagramas temporales para facilitar la comprensión de los mismos.

TIEMPO DE ACCESO (t_{ACC}):

Tiempo que transcurre entre la aplicación de una nueva dirección y la aparición de un dato válido de salida. Este tiempo suele darse siempre como característica general de una memoria, pues representa una medida de lo rápida que puede ser la misma.

❑ TIEMPOS EN EL CICLO DE ESCRITURA:

La operación de escritura, es aquella mediante la cual, una palabra puesta en los terminales de entrada de datos, va a ser "colocada" en una posición de la memoria.

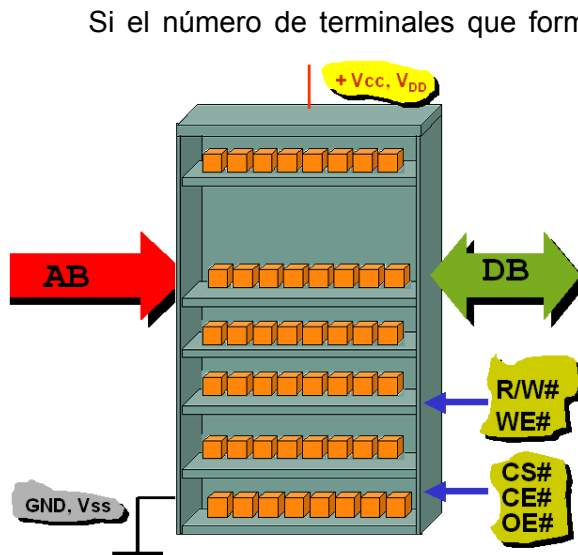
Los tiempos que se dan son similares a los anteriores, teniendo en cuenta la diferencia de operación, y también suelen ir acompañados de los correspondientes diagramas temporales.

7.1.1-VISIÓN GENERAL DE UNA MEMORIA INTEGRADA.

La memoria se puede ver como una estantería donde se coloca un número determinado de bits en cada una de sus baldas, las cuales están perfectamente identificadas por medio de un único número al que llamamos dirección.

□ AB: (ADDRESS BUS).

Conjunto de terminales que sirven para seleccionar posiciones concretas de la memoria. Una posición de la memoria, queda definida o identificada por una dirección (combinación) puesta en estos terminales.



Si el número de terminales que forman el bus es de n , el número de combinaciones posibles que podemos formar es de 2^n . Cada una de estas combinaciones está asociada a una posición de la memoria, la cual queda seleccionada al colocar en este bus la combinación correspondiente.

□ DB: (DATA BUS).

Conjunto de terminales por donde “salen” y/o “entran” los datos. El número de estos terminales nos indica cuantos bits podemos guardar en cada posición de memoria, es decir, el tamaño de la palabra de memoria.

NOTA: OBSERVAR EL SENTIDO DE LAS FLECHAS.

La relación que existe entre el número de bits, la organización de la memoria y el número de terminales de direcciones y datos es la siguiente:

Consideremos una memoria de 1Kb:

Para reducir el número de terminales en determinadas memorias, la información que se maneja en los mismos, está multiplexada en el tiempo.

ORGANIZACIÓN	AL	DL	TOT (AL, DL)
1K X 1	10	1	11
256 X 4	8	4	12
128 X 8	7	8	15

□ R/W#, WE#: (READ/WRITE, WRITE ENABLE).

Este terminal controla la operación de lectura o escritura en la memoria. La posición donde se escribirá o leerá, será aquella que este seleccionada por la combinación presente en el AB.

Un nivel lógico alto en el terminal R/W# hará que la memoria se ponga en modo de lectura, es decir, bajo ciertas condiciones, la memoria colocará en el DB el contenido de la posición que haya sido seleccionada mediante el AB.

□ CS#, CE#, OE#: (CHIP SELECT, CHIP ENABLE, OUTPUT ENABLE).

Estas entradas de control hacen operativo el chip, lo habilitan o inhabilitan. Puede haber más de una entrada de control para un mismo chip. Chip Enable se utiliza muchas veces para poner al chip en un estado de espera (*Standby*), durante el cual, se reduce el consumo de potencia.

- V_{CC} , GND; V_{DD} , V_{SS} .

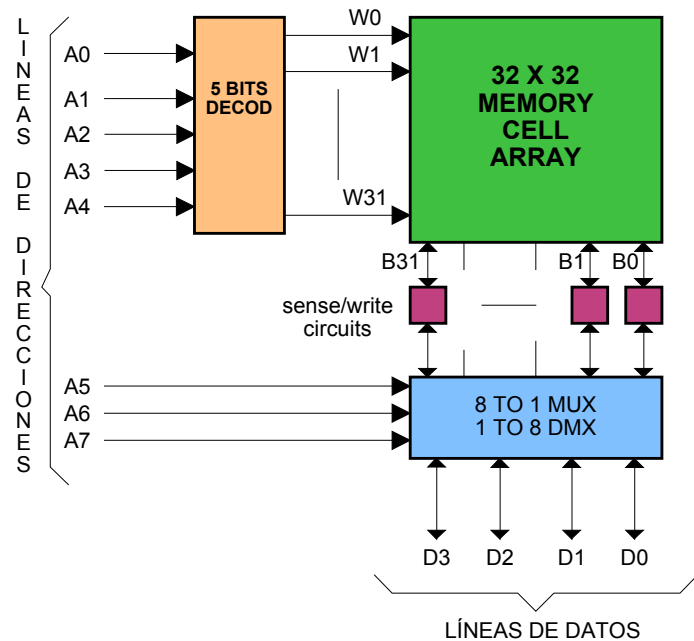
Son diferentes formas de denominación de los terminales de alimentación del chip. V_{CC} y GND son utilizados para el terminal positivo y negativo respectivamente en familia TTL. V_{DD} y V_{SS} , para el terminal positivo y negativo respectivamente, en MOS. Actualmente, V_{CC} y V_{DD} son utilizados de forma indiscriminada para el terminal positivo, y V_{SS} y GND para el terminal negativo.

7.1.2-ORGANIZACIÓN DE UNA MEMORIA.

Los bits se guardan dentro de la memoria en celdas; Una celda es el elemento básico de almacenamiento de un BIT, y puede estar formada por un elemento biestable (Flip-Flop), por un condensador (capacidad parásita), por una unión física de fila y columna, etc.

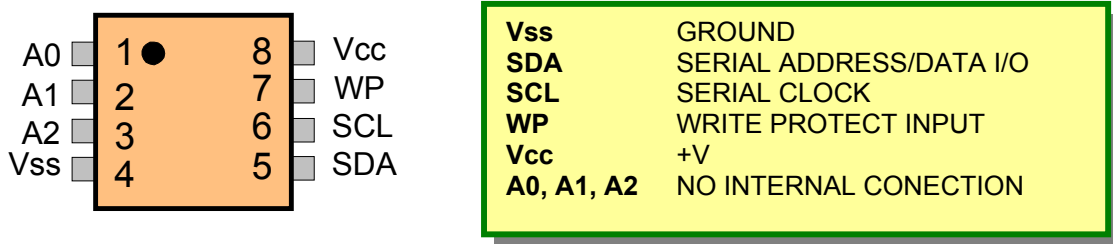
Una posición concreta de la memoria, se define mediante la activación de la fila y columna a la que pertenece dicha posición.

La organización interna de las celdas, se realiza en orden a una mejor utilización de la superficie disponible del chip, es decir, para conseguir el máximo número de celdas por unidad de superficie.



ORGANIZACIÓN DE UN CHIP DE MEMORIA DE 256 X 4

- Actualmente existe un gran número de EEPROM en las cuales la escritura y lectura de los datos se realiza de forma serie.



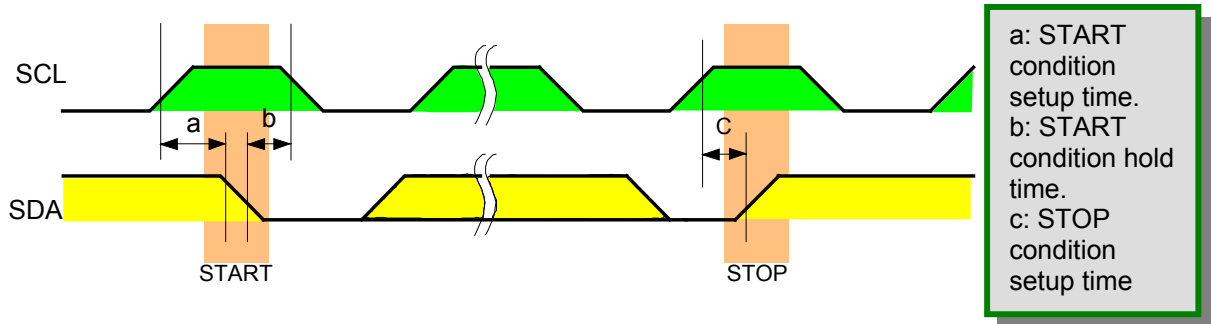
HN58X2416: 2048 X 8; $t_{WC} = 10ms$; $t_{ACC} = 100ns$ a $900ns$
24LC16B: 8 X 256 X 8; $t_{WC} = 2ms$

La escritura y lectura debe seguir un protocolo:

La transferencia de datos sólo puede ser iniciada cuando el bus no está ocupado.

Durante la transferencia de datos, la línea de datos debe permanecer estable siempre que la señal de reloj esté en estado alto. Cualquier cambio en la línea de datos cuando la línea de reloj está en estado alto, se interpreta como una condición de comienzo de transferencia o finalización de la misma (Start, Stop)

- La dirección donde debe efectuarse una operación, los datos de esa operación y, una palabra de control donde se incluye la operación a efectuar sobre esa posición, se aplica en un formato serie de secuencia de bits, precedidos por una condición de comienzo y seguidos por una condición de parada, al terminal SDA de la memoria.



7.5-FLASH MEMORY.

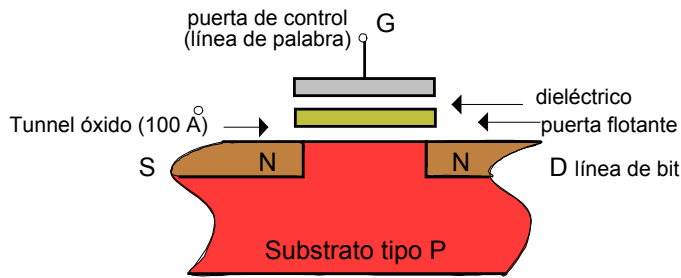
- Son una solución entre las EPROM y las EEPROM.

	EPROM	FLASH	EEPROM
PROGRAMACIÓN	FUERA DEL CIRCUITO	EN CIRCUITO	EN CIRCUITO
BORRADO	FUERA DEL CIRCUITO	EN CIRCUITO	EN CIRCUITO
	TODAS LAS CELDAS	POR BLOQUES O TOTAL	POR BYTES
	UV	ELÉCTRICO	ELÉCTRICO
COMPLEJIDAD	BAJA	MEDIA	ALTA
INTEGRACIÓN	≈FLASH	≈EPROM	<< FLASH O EPROM

- ❑ También deben ser borradas antes de ser escritas. En este caso, puede ser borrada entera (Bulk Erase Flash) o en bloques (Boot Block o Sector Erase Block Flash).
- ❑ Los dispositivos Boot Block, poseen bloques que pueden variar de 4 KB a 128 KB. Uno de los bloques (típicamente de 16KB), está protegido mediante hardware contra borrado. Este bloque puede servir para almacenamiento de firmware.
- ❑ Los dispositivos Sector Erase Block Flash, poseen bloques de igual tamaño. Suele utilizarse como almacenamiento de masa (grandes cantidades de información).
- ❑ Aunque el borrado se realiza en modo de bloques, las operaciones de lectura y escritura se realizan en modo aleatorio por bytes o por palabras.
- ❑ Toda la circuitería y lógica necesaria para poder borrar y escribir en la memoria, está implementada en el propio chip, descargando al sistema al que pertenece la memoria, del hardware y software necesario para la realización de estas tareas.

MT28F016S5: 2M X 8; t_{ACC} =90ns; borrrable en 32 bloques de 64 KB.

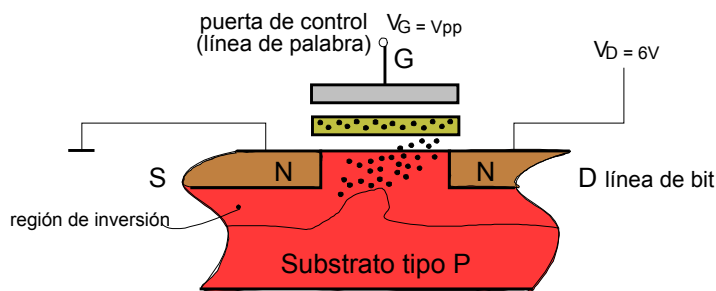
ESTRUCTURA DE LA CELULA FLASH.



Al igual que la celda EPROM, está formada por transistores de efecto de campo CMOS con puerta flotante. La puerta flotante, está aislada de la puerta de control por una capa de dieléctrico, y está aislada del sustrato por una fina capa de óxido metálico. El aislamiento permite a la puerta flotante almacenar carga. La puerta de control, se conecta a la línea de palabras.

La diferencia fundamental de la celda Flash respecto de la celda EPROM, es la mayor delgadez (≈ 100 angstroms $-10\text{nm}-$), de la capa de óxido que aísla el sustrato, de la puerta flotante. Esto permite el borrado de información (eliminación de la carga acumulada) mediante un efecto túnel (Fowler-Nordheim) de los electrones desde la puerta flotante.

ESCRITURA:

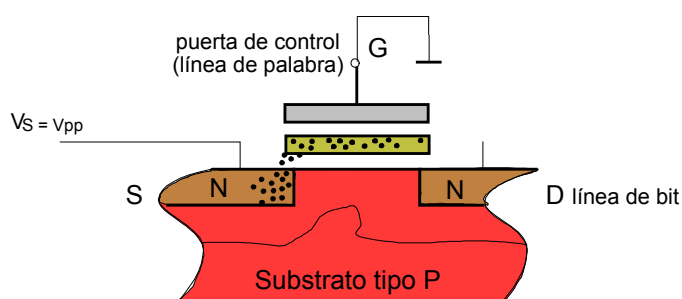


Durante la operación de escritura (WRITE OPERATION), un voltaje alto ($V_{pp} = 12\text{V}$) de programación, se aplica a la puerta de control, esto hace que se forme una región de inversión en el sustrato. El voltaje de drenador (D), se incrementa hasta aproximadamente la mitad del de la puerta de control, es decir, hasta los 6V , mientras que el voltaje de fuente (S), se pone a 0V . Con la

región de inversión formada, la corriente entre D y S aumenta. El alto flujo de electrones desde S hasta D, aumenta la energía cinética de los electrones, haciendo que éstos puedan pasar la barrera de óxido, hasta la puerta flotante. Una vez que la operación de escritura ha terminado, la carga negativa en la puerta flotante eleva el valor de la tensión umbral, por encima del valor lógico 1 de la línea de palabra.

Cuando la línea de palabra de una celda escrita se lleva a 1 lógico durante una operación de lectura, la celda no se activará. Los amplificadores que detectan la información contenida en la celda detectarán un valor insignificante de corriente, dando un 0 en aquellas celdas que estén escritas.

BORRADO.

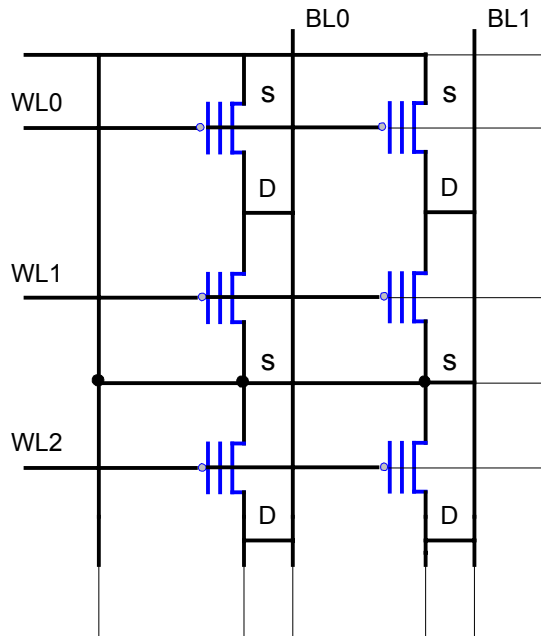


Para el borrado se conecta S a una tensión de $V_{pp} = 12\text{V}$, la G se pone a 0V y D se deja sin conectar. El mayor voltaje positivo en S comparado con el de G, atrae las cargas negativas desde la puerta flotante hasta S a través de la capa de óxido. En esta operación la corriente puesta en juego es mucho menor que en la operación de escritura debido a que el terminal de D está sin conectar.

Cuando la línea de palabra de una celda borrada se pone a 1 lógico durante una operación de lectura, el transistor se activará y conducirá mas corriente que una celda escrita por lo que los amplificadores sensores de corriente darán en este caso de celda borrada un 1 lógico almacenado.

ARQUITECTURA FLASH.

(arquitectura NOR)



LECTURA:

La línea de palabra seleccionada toma el valor lógico 1 ($>V_T$ de una celda borrada). Esta condición activa las celdas borradas, lo cual permite que la corriente circule desde D a S, mientras que para las celdas escritas, las mantiene en estado inactivo con una pequeña corriente circulando desde D a S.

ESCRITURA:

Al igual que la operación de lectura, la escritura se hace de forma aleatoria. La línea de palabra seleccionada se lleva a un voltaje de 12 voltios, la línea de BIT (D) se pone a 6 v y la S permanece a 0v. Todas las demás líneas de palabra del array permanecen en estado bajo.

BORRADO:

La S es común para todas las celdas dentro de un bloque de borrado. Durante la operación de borrado, las líneas de BIT (D), se dejan abiertas, las líneas de palabra se ponen a 0 v y S se lleva a 12v, borrando todas las celdas pertenecientes al bloque.

7.6-APLICACIONES DE LAS ROM.

FIRMWARE:

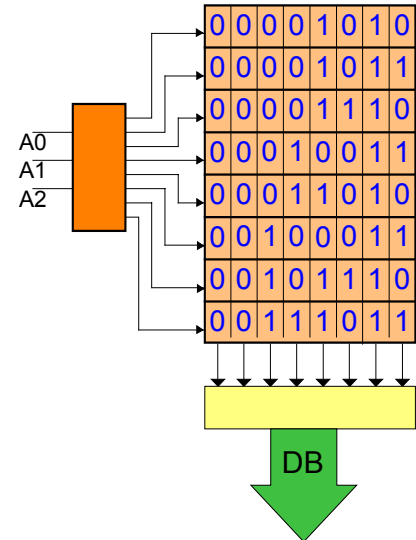
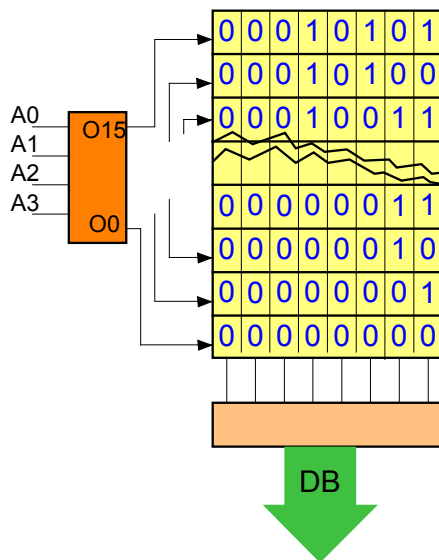
- Guardan programas de sistema operativo, modos de operación según el programa específico que se almacene, intérpretes de lenguaje; de esta manera se puede hacer uso inmediato de la computadora después de conectarse.

TABLAS DE DATOS:

- Tablas de funciones, trigonometría...

Ejemplo: memoria que contiene la función

$$Y = X^2 + 10$$



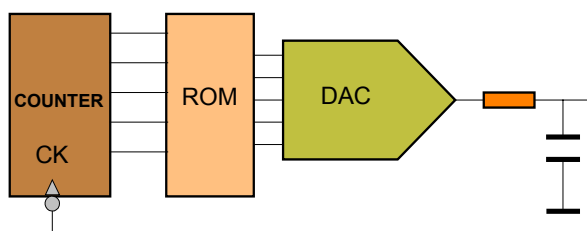
CONVERTIDORES DE DATOS:

Ejemplo: memoria convertidor BIN-BCD.

GENERADORES DE CARACTERES:

- Almacenan los códigos del modelo de puntos por cada carácter ubicado en una dirección. Esta dirección corresponde al código ASCII de ese carácter. Impresoras, monitores.

GENERADORES DE FUNCIONES:

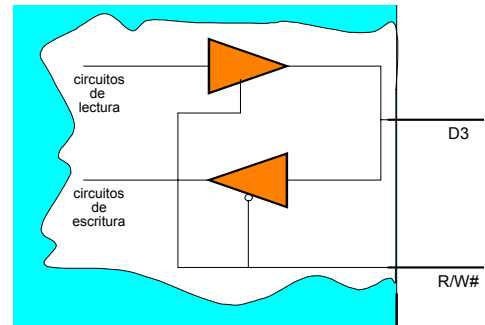


- El contador recorre de forma secuencial todas las posiciones de la memoria, entregando al convertidor el contenido de cada una de ellas. En la salida del convertidor, tendremos un punto de la señal analógica por cada posición de la memoria. El filtro

de salida "suaviza" los escalones producidos por el DAC.

7.7-RAM (RANDOM ACCESS MEMORY).

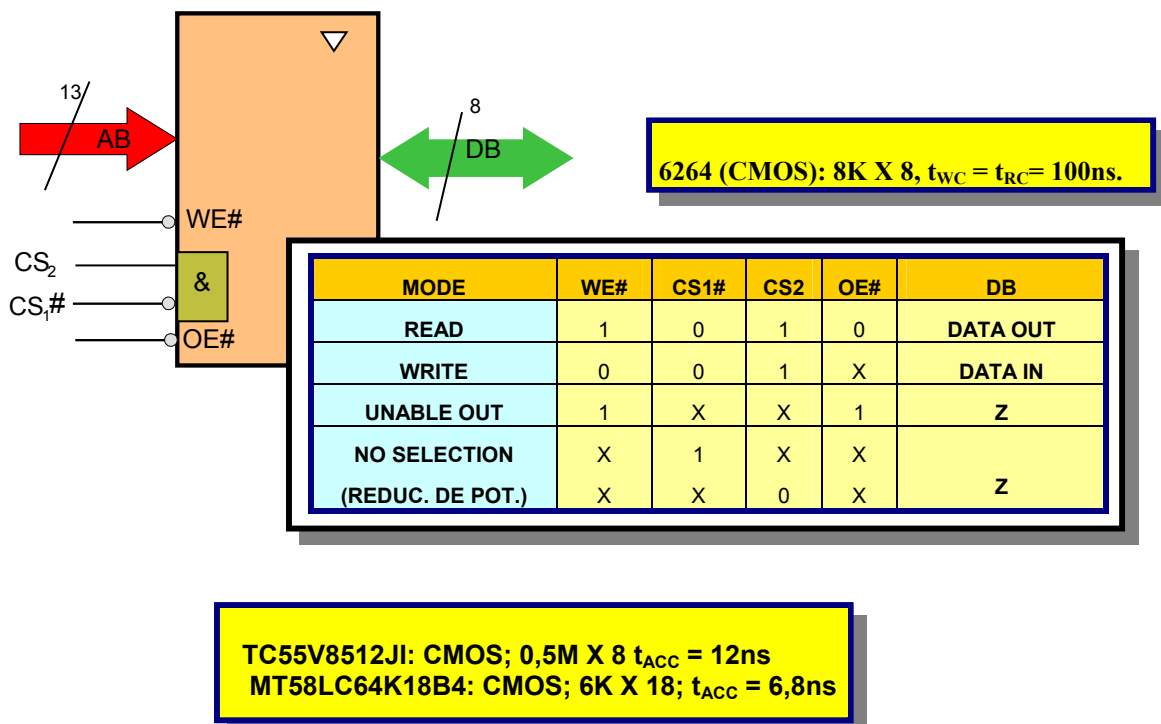
- Son memorias volátiles, es decir, la información almacenada se pierde cuando se deja de alimentar el chip de memoria.
- La operación de escritura en una posición determinada, destruye la información almacenada en dicha posición.
- Los terminales de datos son generalmente de entrada y salida, y están controlados principalmente por el terminal R/W#.
- Las RAM encuentran su aplicación en el almacenamiento de datos que cambian de forma frecuente durante la operación de un programa.
- Son fabricadas en tecnología bipolar y MOS.



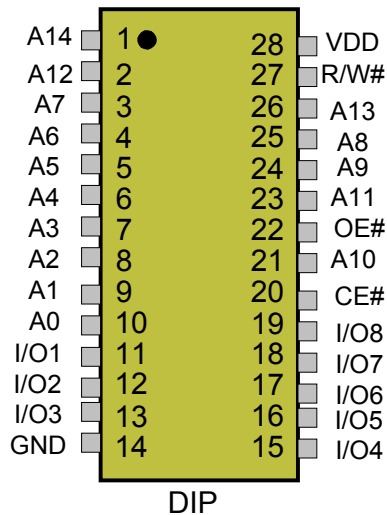
7.7.1-SRAM. (STATIC RAM)

- Las celdas están construidas, básicamente por Flip-Flop-s.
- Las RAM estáticas se fabrican principalmente en tecnología MOS (CMOS), debido a su mayor nivel de integración, y su menor consumo respecto de las SRAM bipolares. Además, la velocidad entre unas y otras no es un factor generalmente determinante, ya que se fabrican memorias CMOS con tiempos de acceso inferiores a los 5 ns.
- Generalmente, las SRAM se encuentran en aplicaciones donde se requieren bajas cantidades de memoria ($\approx 64k$), aplicación como memoria caché, o aplicaciones de gran velocidad.

MODOS DE OPERACIÓN BÁSICOS:

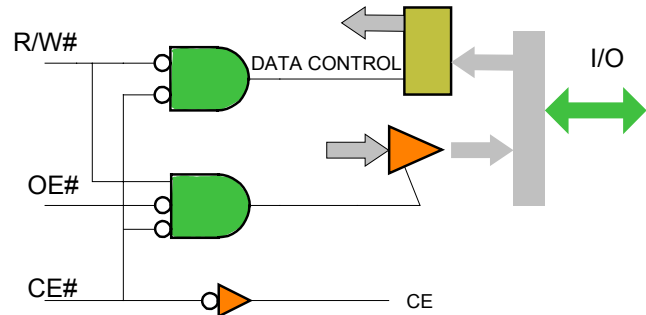


CARACTERÍSTICAS GENERALES.



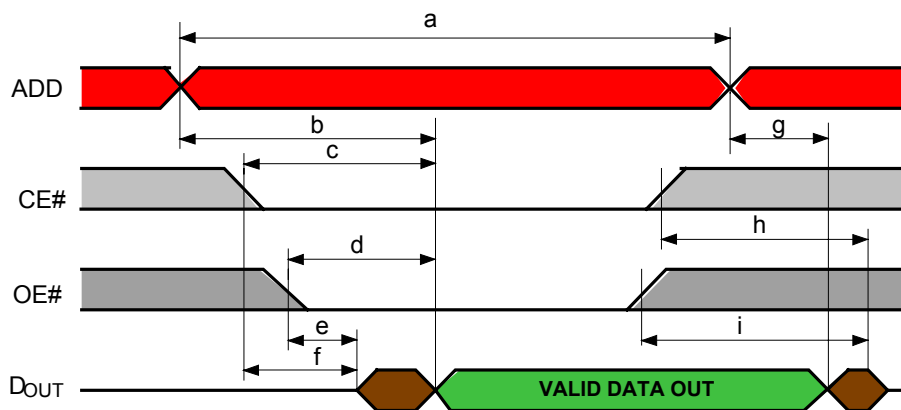
PIN NAMES	
$A_0 - A_{14}$	Address Input
$I/O_1 - I/O_8$	Data input/output
$OE\#$	Output Enable
$CE\#$	Chip Enable
$R/W\#$	Read/Write operation
VDD	Power Supply
GND	Ground

LÓGICA DE CONTROL



MODE	CE#	OE#	R/W#	DB
Read	L	L	H	Dout
Write	L	X	L	Din
Output disabled	L	H	H	High-Z
Standby	H	X	X	High-Z

TIMING.



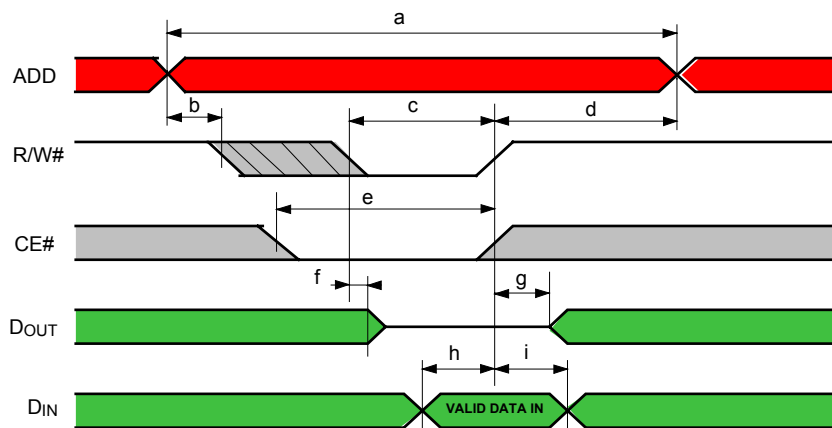
CICLO DE LECTURA

Durante el ciclo de lectura, $R/W\#$ permanece en estado alto

(D_{OUT} sombreado indica valores indeterminados)

a	t_{RC}	READ CYCLE TIME	55 ns
b	t_{ACC}	ADDRESS ACCESS TIME	55 ns
c	t_{CO}	CHIP ENABLE ACCESS TIME	55 ns
d	t_{OE}	OUTPUT ENABLE ACCESS TIME	30 ns
e	t_{OEE}	OUTPUT ENABLE LOW TO OUTPUT ACTIVE	10 ns
f	t_{COE}	CHIP ENABLE LOW TO OUTPUT ACTIVE	5 ns
g	t_{OH}	OUTPUT DATA HOLD TIME	20 ns
h	t_{OD}	CHIP ENABLE HIGH TO OUTPUT HIGH-Z	20 ns
i	t_{ODO}	OUTPUT ENABLE HIGH TO OUTPUT HIGH-Z	10 ns

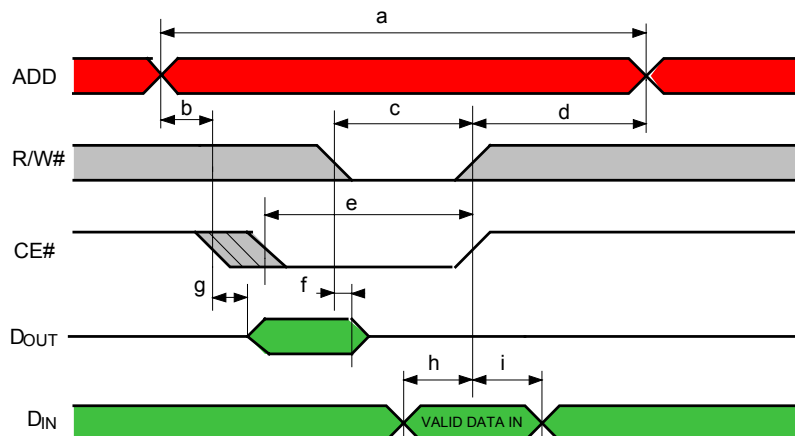
**CICLO DE ESCRITURA.
(R/W# CONTROLLED)**



Si CE# se activa coincidente con, o después de que R/W# baje, las salidas permanecerán en estado de alta impedancia. También, si CE# se desactiva coincidente con, o antes de que R/W# suba, las salidas permanecerán en estado de alta impedancia.

			min	max
a	t_{WC}	WRITE CYCLE TIME	55 ns	
b	t_{AS}	ADDRESS SETUP TIME	45 ns	
c	t_{WP}	WRITE PULSE WIDTH	50 ns	
d	t_{WR}	WRITE RECOVERY TIME	0	
e	t_{CW}	CHIP ENABLE TO END OF WRITE	0	
f	t_{ODW}	R/W# LOW TO OUTPUT HIGH-Z		20 ns
g	t_{OEWS}	R/W# HIGH TO OUTPUT ACTIVE	5 ns	
h	t_{DS}	DATA SETUP TIME	25 ns	
i	t_{DH}	DATA HOLD TIME	0	

(CE# CONTROLLED)

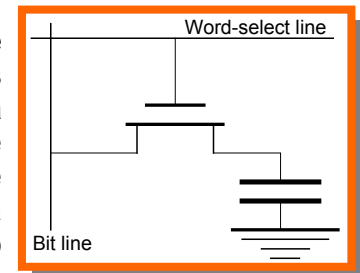


□ Si OE# es alto durante el ciclo de escritura, las salidas permanecerán en estado de alta impedancia.

			min	max
a	t_{WC}	WRITE CYCLE TIME	55 ns	
b	t_{AS}	ADDRESS SETUP TIME	45 ns	
c	t_{WP}	WRITE PULSE WIDTH	50 ns	
d	t_{WR}	WRITE RECOVERY TIME	0	
e	t_{CW}	CHIP ENABLE TO END OF WRITE	0	
f	t_{ODW}	R/W# LOW TO OUTPUT HIGH-Z		20 ns
g	t_{COE}	CHIP ENABLE LOW TO OUTPUT ACTIVE	10 ns	
h	t_{DS}	DATA SETUP TIME	25 ns	
i	t_{DH}	DATA HOLD TIME	0	

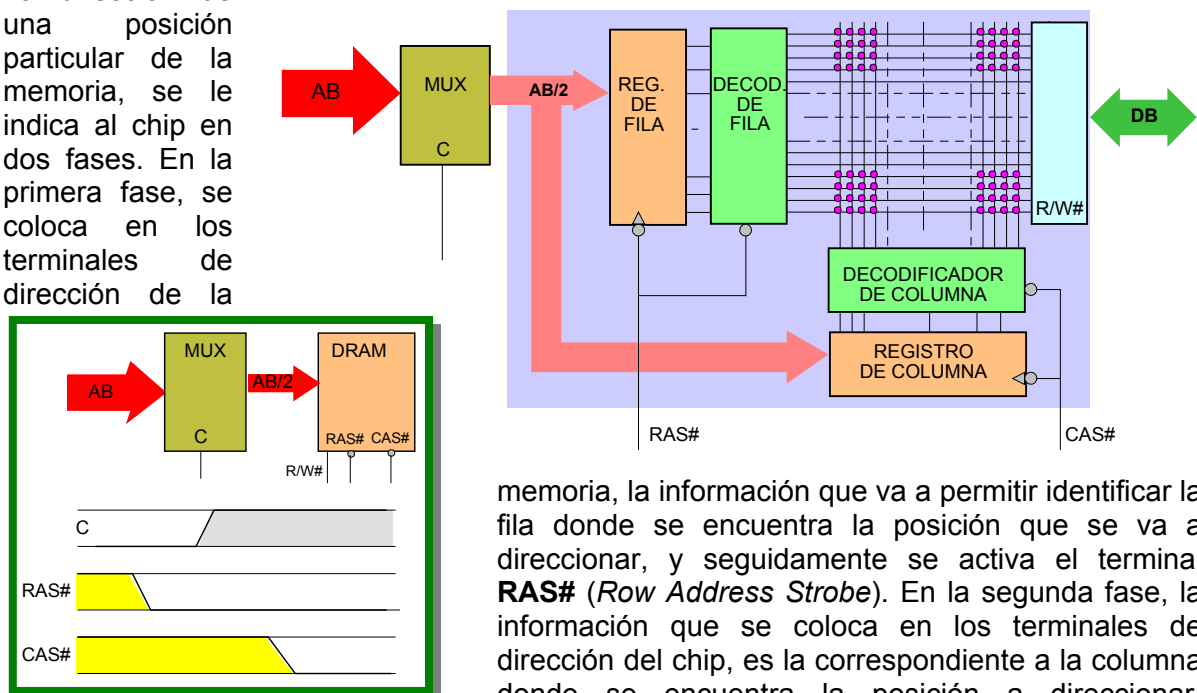
7.7.2-DRAM (DYNAMIC RAM).

- Realizadas con tecnología MOS, estas memorias tienen como características más importantes su gran capacidad de integración (≈ 4 veces más que una SRAM), y el bajo consumo de potencia ($1/6 \div 1/2$ de las SRAM). Sin embargo, la velocidad media es menor que las SRAM.
- Las celdas de almacenamiento no están construidas a partir de FF, sino por condensadores MOS formados por capacidades parásitas. Debido a las fugas de estas capacidades, la información tiende a perderse (aunque el circuito esté alimentado), por lo que se necesita la recarga periódica de esos "condensadores". Al fenómeno de la restitución de la información en las capacidades parásitas se le conoce como **REFRESCO**.
- El diseño del sistema de memoria basado en DRAM, necesita mayores requerimientos que las SRAM.
- Existen DRAM (**iRAM**) que incluyen circuitería de refresco en el propio chip. Este tipo de IC-s es utilizado en sistemas de memoria relativamente pequeña. Para requerimientos de memoria mayor se hace uso de los controladores DRAM, los cuales contienen toda la lógica necesaria para el refresco.
- Generalmente, las DRAM, se encuentran en aplicaciones donde se requieren grandes cantidades de memoria, o bajo consumo.



MULTIPLEXADO DE DIRECCIONES.

- Debido a la gran capacidad de integración de estas memorias, el número de líneas de dirección necesarias para poder direccionar todas las posiciones se hace relativamente grande, con el consiguiente aumento en el volumen del encapsulado. Para evitar este aumento de volumen, se recurre al mantenimiento del mínimo número de líneas de dirección, y recurrir al multiplexado en el tiempo de la información a aplicar a estas líneas. La dirección de una posición particular de la memoria, se le indica al chip en dos fases. En la primera fase, se coloca en los terminales de dirección de la

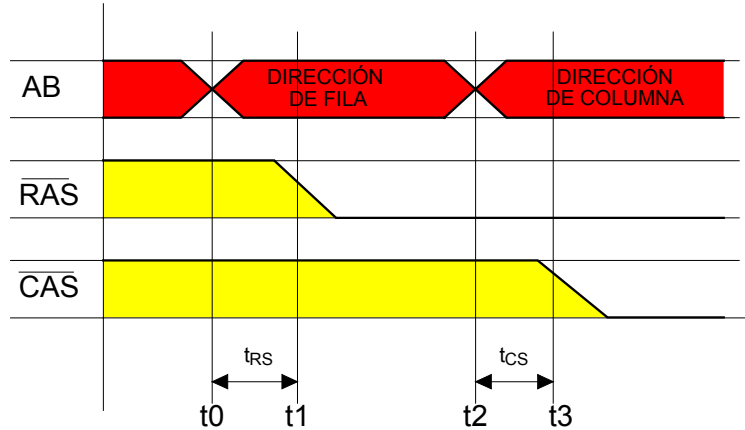


memoria, la información que va a permitir identificar la fila donde se encuentra la posición que se va a direccionar, y seguidamente se activa el terminal **RAS#** (*Row Address Strobe*). En la segunda fase, la información que se coloca en los terminales de dirección del chip, es la correspondiente a la columna donde se encuentra la posición a direccionar, indicando este hecho mediante la activación del terminal **CAS#** (*Column Address Strobe*).

- Para conseguir que la dirección presente en el AB del sistema, sea aplicada en dos tiempos a los terminales de la memoria DRAM, se necesita un multiplexor con un número mínimo de líneas de salida igual al de terminales de dirección del chip.

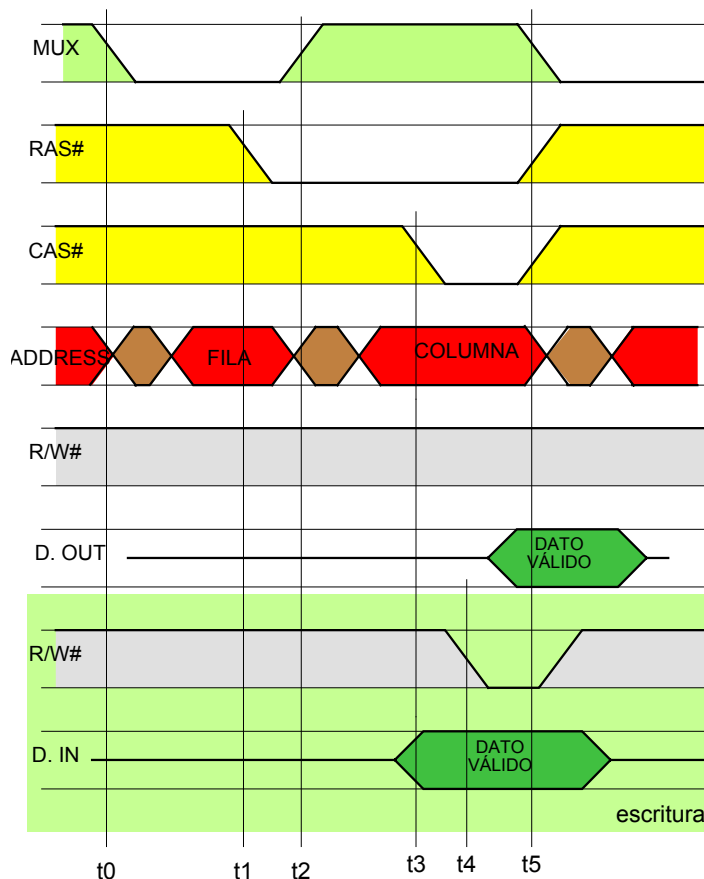
TIMING.

- **t_{RS}** : Tiempo de establecimiento de fila.
- **t_{CS}** : Tiempo de establecimiento de columna.
- **RAS#**: Selección de dirección de fila. (Row Address Strobe).
- **CAS#**: Selección de dirección de columna. (Column Address Strobe).
- RAS# y CAS# hacen las funciones de CS. Cuando ambas están a nivel bajo se activa la celda.



CICLOS DE LECTURA / ESCRITURA.

- Los ciclos de lectura y escritura en modo de acceso normal, son los representados en la figura:



*En t_0 , el multiplexor, con una puesta a nivel bajo de su línea de selección, entrega a los terminales de la DRAM la combinación procedente del AB del sistema, que corresponde con la dirección de fila. Esta combinación después de un periodo de tiempo (zona oscura), se estabiliza.

*En t_1 , se indica a la DRAM, mediante la activación de RAS#, que la combinación que tiene presente en sus terminales de dirección, corresponde con la dirección de la fila.

*En t_2 , el multiplexor cambia el valor de su línea de control, tomando del AB del sistema la combinación correspondiente a la columna y entregándola a los terminales de dirección de la DRAM.

*En t_3 , mediante la activación de CAS# se indica a la DRAM que el valor que tiene en sus terminales de

dirección es el correspondiente a la dirección de columna.

A partir de este momento (RAS# y CAS# activados), el chip queda seleccionado. Según esté la línea de R/W#, la operación que se efectuará sobre la memoria será la de escritura (R/W# = 0) o la de lectura (R/W# = 1).

En t_5 , el chip queda inhabilitado, al volver RAS# y CAS# a su estado inactivo.

Además del acceso normal descrito, existen otros modos de acceso que dependen del tipo de DRAM:

FAST PAGE MODE (FPM).

Esta operación se inicia con la activación de la señal RAS#, seguida por la señal de CAS# igual que para el acceso a posiciones particulares de la memoria. Sin embargo, diferentes columnas dentro de la misma fila pueden ser accedidas activando y desactivando CAS#, mientras se mantiene RAS# a su nivel activo, e introduciendo nuevas direcciones de columna con cada ciclo de RAS#. Al desactivar RAS#, termina la operación FPM.

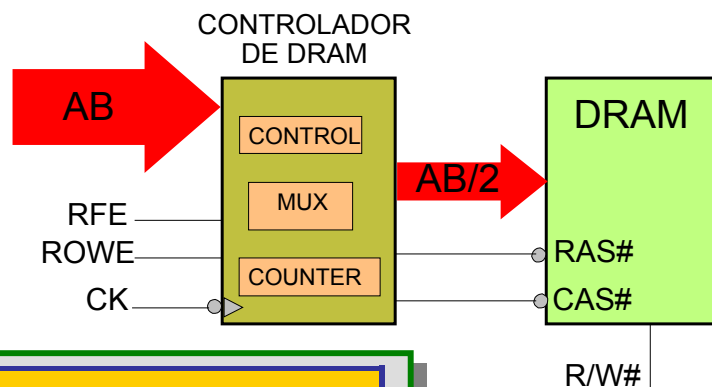
EDO (EXTENDED DATA OUT) PAGE MODE.

La operación básica es la misma que en FPM. La diferencia está en que bajo ciertas condiciones que deben cumplir algunos terminales de control de la memoria, los buffers de salida no se ponen en estado de Z con el flanco positivo de CAS#. En este modo, los datos son válidos después de que CAS# pase a su estado inactivo siempre que RAS#, y OE# sean activos y WE# esté a nivel alto.

REFresco:

- ❑ La celda se refresca cada vez que se hace sobre ella una operación de lectura o escritura. En esta operación también se refrescan todas las celdas que están en la misma fila que la celda seleccionada.

- ❑ El control de refresco suele realizarse mediante controlador DRAM. Este controlador recorre de forma cíclica todas las direcciones correspondientes a las filas para proceder al refresco de todas las celdas. La forma de recorrer las filas puede ser de



RFE	ROWE	AB/2
H	X	REGEN. DE LA DIRECCIÓN. (CONT. INTERNO)
L	H	DIRECCIONES DE FILA (AB)
L	L	DIRECCIONES DE COLUMNA. (AB)

forma continua (modo burst), que consiste en el refresco de todas las filas, una después de otra

desde la primera hasta la última, o de forma multiplexada, en la cual entre ciclo y ciclo de refresco, se realizan operaciones de acceso a memoria. En cualquier caso siempre debe cumplirse que la matriz sea refrescada de forma cíclica, dentro del tiempo especificado por el fabricante (una vez cada 64ms para la MT4LC4M16F5).

- ❑ El controlador determina (mediante las entradas de control) si las direcciones multiplexadas son para refresco o para lectura / escritura. Muchas memorias poseen internamente los contadores que direccionan de forma conveniente las diferentes filas de la matriz para su refresco.

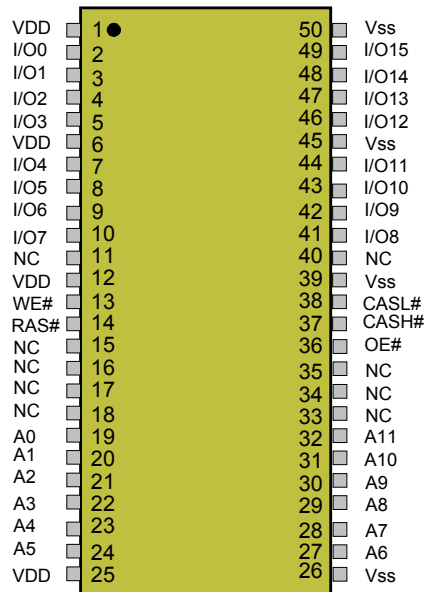
- Actualmente podemos encontrar tres modos de realizar los ciclos de refresco:

RAS# ONLY REFRESH: Consiste en el suministro periódico desde el AB del sistema, de la dirección de las filas a refrescar y validará con la activación de la línea RAS#.

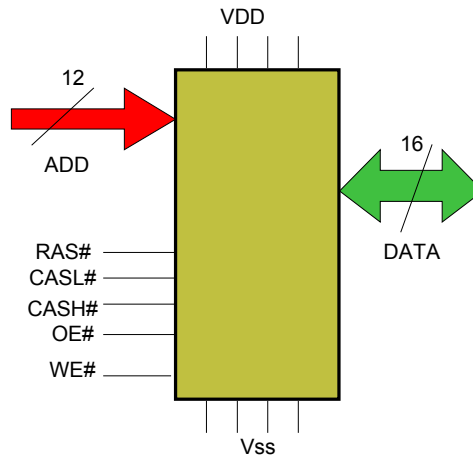
CAS# BEFORE RAS# REFRESH (CBR): Mediante la secuencia de activación de las señales de RAS# y CAS#, se activa en la memoria, la circuitería interna, cuya misión es generar las direcciones de las filas a refrescar, sin necesidad de que éstas tengan que ser suministradas desde el exterior, y sincronizar el proceso de refresco.

HIDDEN REFRESH: Con este método se simultanea el refresco con los accesos a la memoria.

**EJEMPLO.
(4M X 16)**

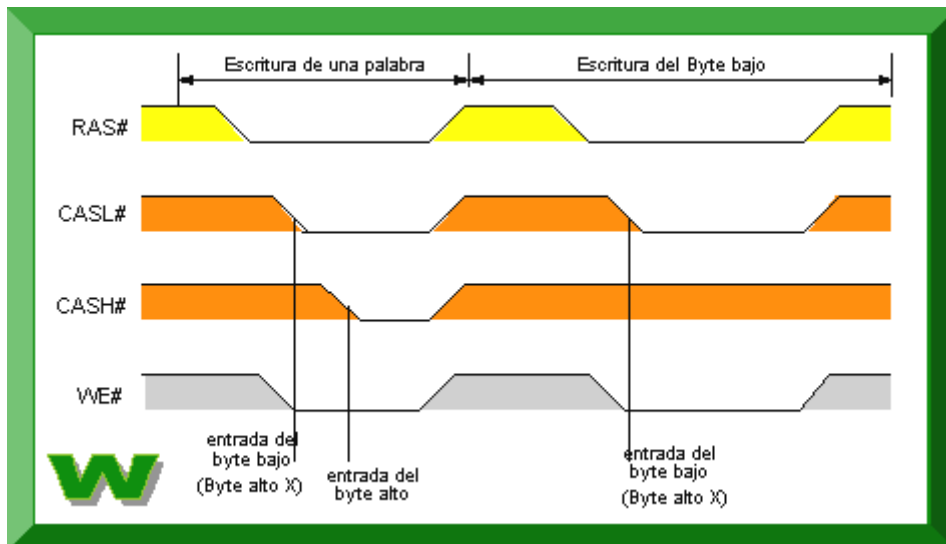


REPRESENTACIÓN FUNCIONAL

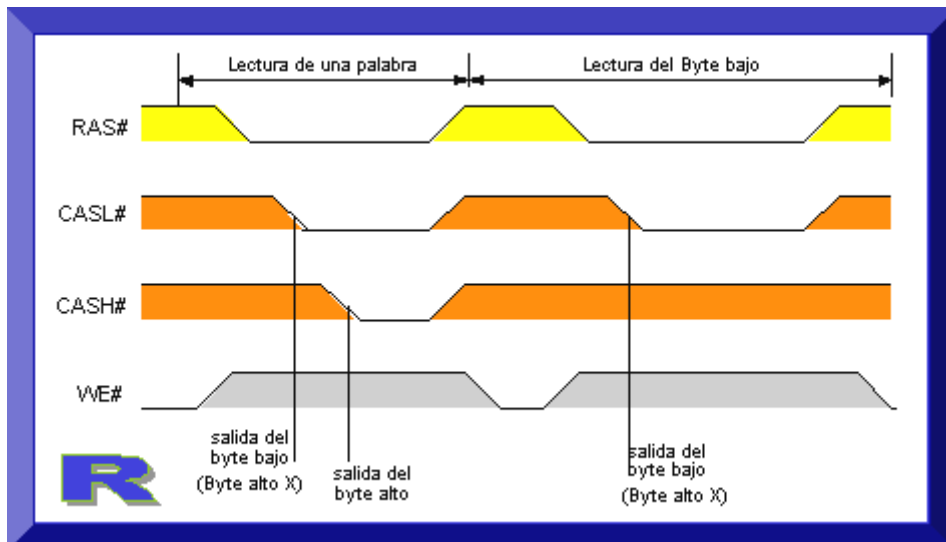


DISTRIBUCIÓN DE TERMINALES

EJEMPLO DE ESCRITURA DE PALABRA Y BYTE.



EJEMPLO DE LECTURA DE PALABRA Y BYTE.



7.7.3-SDRAM. (SYNCHRONOUS DRAM).

- Las memorias SDRAM siguen los mismos principios de funcionamiento que las DRAM vistas, pero con todas las señales de entrada y contadores internos, controladas por una señal de reloj, gobernada por el reloj del sistema. Actualmente la frecuencia de trabajo para aplicaciones comerciales se encuentra en los 133MHz.

MT2LSDT132U: 1,2M X 32; CMOS; t_{ACC} = 7,5 ns (100MHz); T refr= 64ms

7.7.4-RDRAM

- Es un tipo de memoria CMOS, enfocada básicamente a cubrir las necesidades de trabajo del procesador Pentium 4 de Intel®.

(Esta memoria, está en competencia con otro tipo de memoria llamada DDR -Double Data Rate- pero con un precio mucho mas alto que esta última).

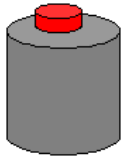
Las diferencias fundamentales respecto a la SDRAM son:

La transmisión de los datos se realiza de forma serie.

La velocidad de trabajo es notablemente mayor (600, 711 y 800MHz).

**3242 :CONTROLADOR DRAM DE 16K
MCM514256A (DARM): 256K X 4, t_{ACC} = 80 ns, T refr= 8ms.
MT4LC4M16F5 (DRAM): 4M X 16; CMOS; t_{ACC} = 50 ns; T refr= 64ms**

7.7.5-RAM CMOS

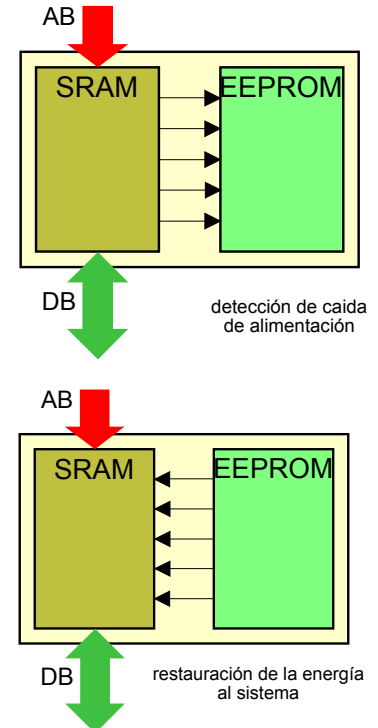


- Son SRAM de un bajo consumo que mantienen los datos después de una falta de energía en el sistema, gracias a que se mantienen alimentadas por una pequeña batería.

- Algunos chips RAM CMOS, incluyen una pequeña batería de litio sobre el mismo IC.

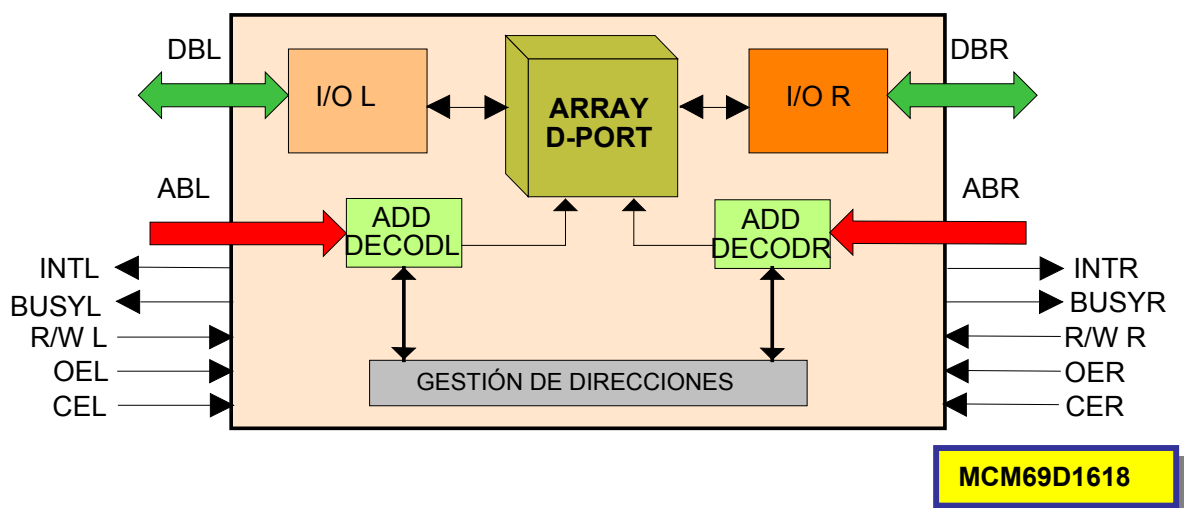
7.7.6-NVRAM (NO VOLATIL RAM)

- Están formadas por una matriz SRAM y una matriz EEPROM.
- Se combina la alta velocidad de operación de la SRAM con la no volatilidad de la EEPROM.
- A cada celda SRAM le corresponde una celda de la EEPROM.
- Cuando hay una falta de alimentación en el sistema los datos contenidos en la SRAM son transferidos a la EEPROM aprovechando la carga de los condensadores de la fuente de alimentación.
- Cuando el sistema vuelve a ser alimentado, la EEPROM transfiere de forma automática los datos a la SRAM.



7.7.7-RAM DUAL PORT

- Tiene dos conjuntos de terminales de dirección, de terminales de datos, y de señales de control mediante las cuales se pueden acceder a la matriz de datos, tanto para escribir como para leer desde ella de forma totalmente independiente.



- Los modos de acceso de cada puerto son iguales a los de las SRAM.
- Algunos chips de RAM DUAL PORT, tienen la posibilidad de que la información, pase de un puerto al otro, en cualquier dirección, sin necesidad de que sea previamente almacenada en la matriz (Pass-Through mode).

- En el caso de que exista coincidencia en la dirección en uno y otro puerto para realizar un acceso a la matriz, se siguen unas normas con el fin de evitar la pérdida de datos. Esta serie de normas es lo que define el protocolo: generalmente es común para todas las memorias de este tipo que si los dos puertos son leídos de forma simultanea, la operación transcurre de forma normal. Si un puerto es leído y el otro escrito, la lectura de los datos de la posición direccionada, tiene lugar antes de la escritura en esa misma posición de la matriz. Si los dos puertos son escritos, solamente tendrá efecto la escritura en la matriz por uno de los dos.
- Para la ejecución del protocolo, existen diferentes modos de arbitración para evitar los conflictos que puedan darse. Algunos de estos modos son:

ARBITRACIÓN POR HARDWARE: Al primer puerto que intenta acceder a la matriz, se le da prioridad. Durante el tiempo que dura la operación se activa una línea llamada BUSY# (ocupado) que posee el otro puerto que no ha podido acceder. Esta línea le indica al sistema conectado a este puerto, que debe esperar hasta que el acceso del otro puerto termine.

ARBITRACIÓN POR FLAGS: El acceso de un puerto a la matriz, debe ser comunicado al otro mediante la activación de unos determinados bits (flags) que posee la propia memoria. Antes de que un puerto quiera acceder, debe consultar estos flags para ver si es posible. Todas estas operaciones deben ser realizadas por software, es decir, es el programador el que debe indicar mediante programa, la consulta de los flags, la activación de los mismos, y la desactivación cuando se haya completado el acceso por parte de un puerto.

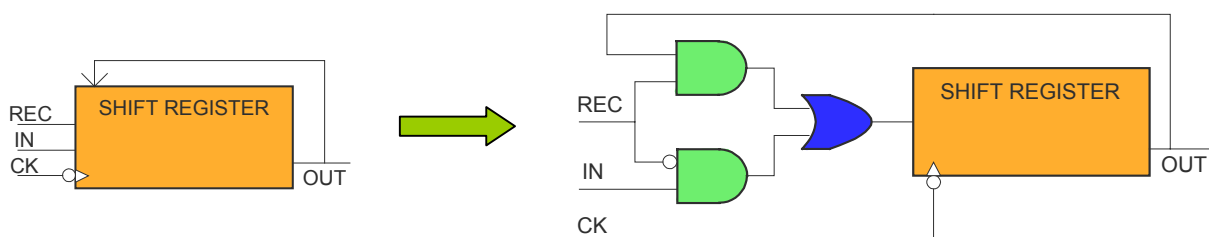
ARBITRACIÓN POR FLAGS (HARDWARE): El proceso es igual que el anterior, pero en lugar de que un sistema esté perdiendo tiempo, consultando los flags para ver si es posible el acceso, lo que se hace es proveer a la memoria con una línea (INT) que indica al sistema que está en espera, cuando el flag ha cambiado de valor. De esta manera, el sistema puede dedicarse a la realización de otras tareas hasta que se la avise mediante esta línea de interrupción.

- La memoria RAM DUAL PORT es utilizada para el paso de información entre sistemas interconectados. Es utilizada como VRAM en los sistemas de video, tomando un puerto para el flujo de datos desde el sistema microprocesador, y el otro puerto para el flujo de datos hacia el circuito de video. De esta manera el envío de información y el refresco, se hacen de forma independiente pero simultánea.

7.8-MEMORIAS SECUENCIALES INTEGRADAS.

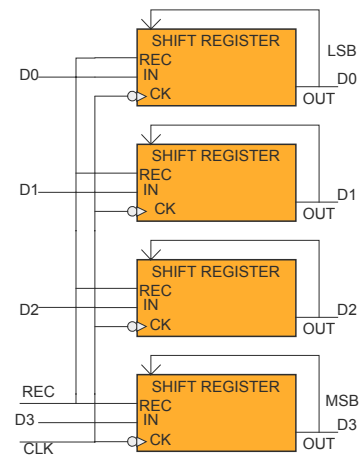
7.8.1-REGISTRO DE DESPLAZAMIENTO CON RECIRCULACIÓN.

- Basadas en registros de desplazamiento. El acceso a las posiciones se hace de forma secuencial. La velocidad de operación es inferior a la de las memorias de acceso aleatorio (RAM y ROM).
- Aplicación donde se necesiten datos repetitivos y secuenciales (refresco de imágenes).

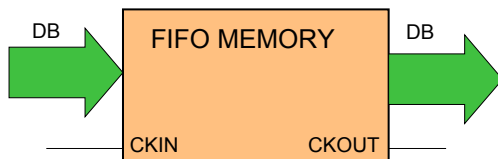


La línea de control REC, se encarga de dar paso a nueva información (cuando está a nivel lógico bajo), o dar paso a la información que ya estaba dentro del registro, es decir, recirculación de la información (cuando está a nivel lógico alto).

- ❑ Para conseguir datos repetitivos en paralelo de n bits, se conectan n bucles como el representado, trabajando en paralelo, con el mismo reloj y la misma entrada REC (recirculación).



7.8.2-FIFO MEMORY (FIRST IN-FIRST OUT)



- ❑ La memoria FIFO permite que de forma secuencial, se escriban datos en ella, y al mismo tiempo, se lean datos de ella.

FIFOS SÍNCRONAS.

En este tipo de memorias, el proceso de escritura no es independiente del proceso de lectura. Existe una relación estrecha de temporización entre el reloj de escritura y el de lectura para que la memoria pueda operar de forma correcta.

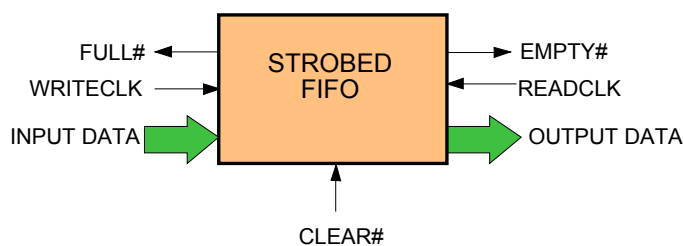
FIFOS ASÍNCRONAS.

No existe una dependencia entre la escritura y la lectura de datos. Apropia para la transferencia de datos entre sistemas que funcionan con velocidades diferentes.

- ❑ La computadora puede realizar otras operaciones mientras se llena la memoria (memoria intermedia).
- ❑ Dentro de las FIFOS ASÍNCRONAS tenemos: **STROBED FIFO-s** y **CLOCKED FIFO-s**

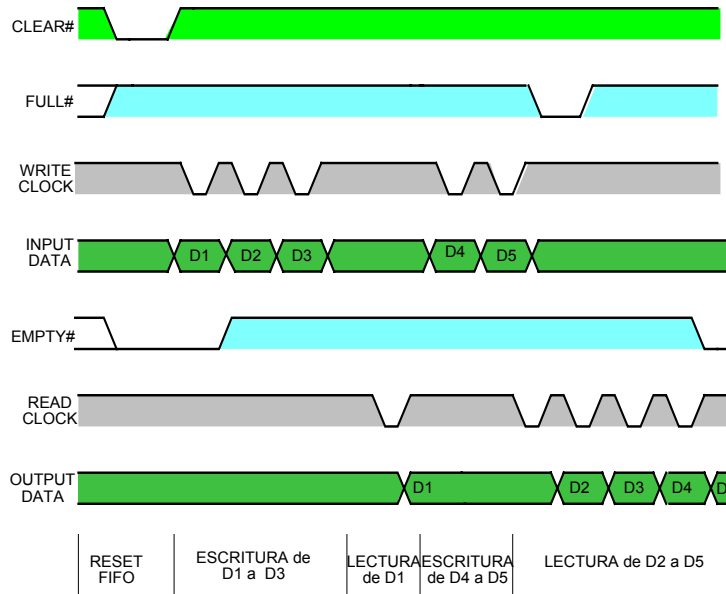
STROBED FIFOS.

- ❑ Las líneas de control *WRITECLK* y *FULL#* se usan para la transferencia de información hacia la memoria. Para realizar una escritura, primero se debe comprobar el estado de la línea *FULL#*, que indica si hay espacio libre para la escritura de un dato. Si hay espacio libre, el dato se aplica a la entrada de datos y se escribirá en la memoria en el flanco activo del reloj, *WRITECLK*.



- ❑ Para realizar una lectura, primero se comprueba que en la memoria hay datos, mediante la interrogación de la línea *EMPTY#*. Si es así, en el flanco activo de *READCLK*, aparecerá en la salida de la memoria la primera palabra de datos que esté en la cola.

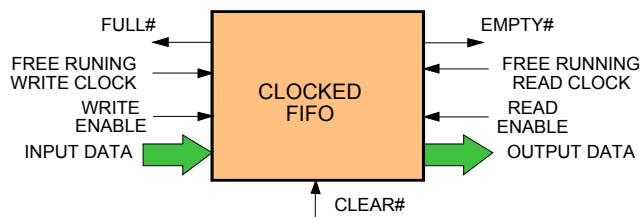
SN74ACT7804



Ejemplo de lectura y escritura de una FIFO de longitud 4 (cuatro palabras).

El ciclo comienza con un reset de la FIFO. La línea de EMPTY#, está activada mientras no existan datos en la memoria. La línea FULL# se activa cuando la memoria tiene escritas cuatro palabras. Observar que si a la vez que se escribe en la FIFO, se va leyendo de la misma, no podrá activarse la línea EMPTY#.

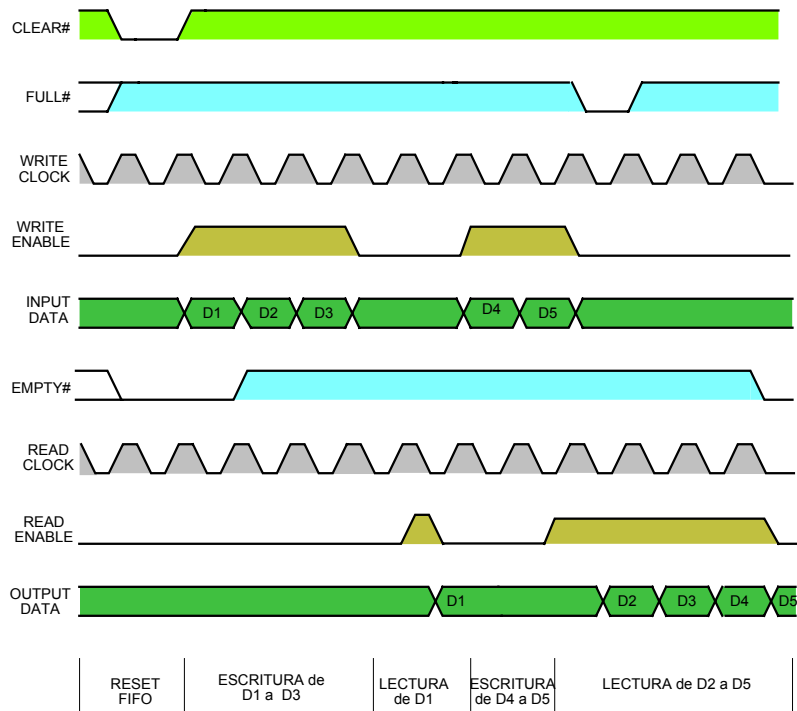
CLOCKED FIFOs.



En este caso, tanto el reloj para escritura como el de lectura, están continuamente trabajando, independientemente de que se lleve a cabo alguna acción o no. La operación de escritura se realiza bajo el control de la señal WRITE ENABLE, de forma síncrona con el reloj WRITE CLOCK. De forma análoga, la operación

de lectura se realiza bajo el control de la señal READ ENABLE, de forma síncrona con el reloj READCLK.

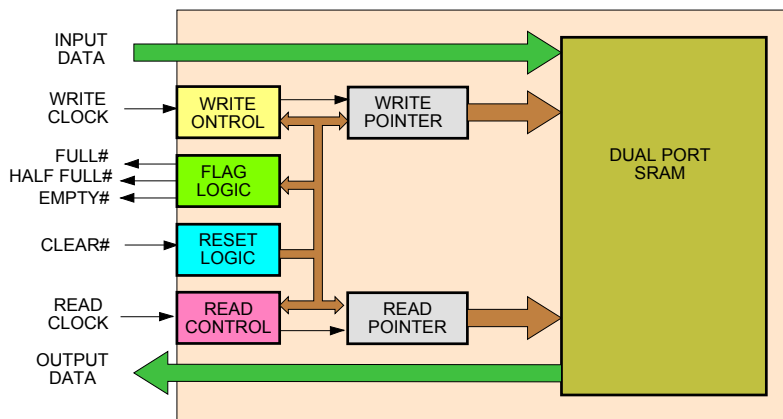
SN74ACT7803



Ejemplo de lectura y escritura de una FIFO de longitud 4 (cuatro palabras).

El ciclo comienza con un reset de la FIFO. La línea de EMPTY#, está activada mientras no existan datos en la memoria. La línea FULL# se activa cuando la memoria tiene escritas cuatro palabras. Para la escritura de un dato en la FIFO, no basta que ésta tenga espacio para el mismo, ni que el reloj aporte un flanco activo, además, es necesario que la señal WRITE ENABLE, sea activa. Algo análogo sucede con la operación de lectura y la señal READ ENABLE. Observar que si a la vez que se escribe, en la FIFO, se va leyendo de la misma, no podrá activarse la línea EMPTY#.

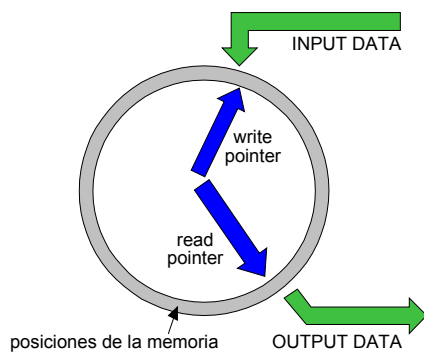
FIFOS CON SRAM



- Las direcciones de memoria donde los datos entrantes deben ser colocados, están en el puntero de entrada. Las de los datos que deben ser puestos en la salida, están en el puntero de salida.

- Después de un CLEAR#, ambos punteros, el de escritura y el de lectura, apuntan a la misma posición de la memoria.

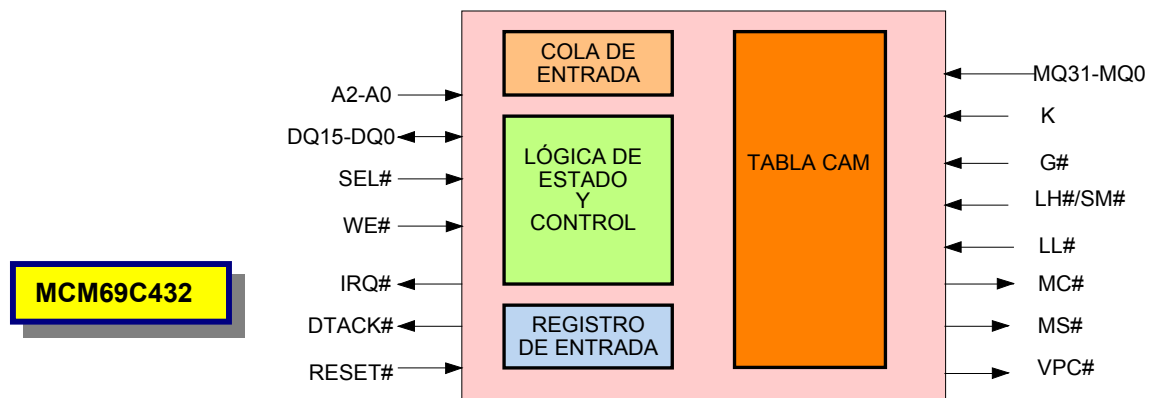
Después de cada operación de escritura, el puntero de escritura apunta a la siguiente posición de memoria. La lectura de una palabra de datos, sitúa el puntero de lectura en la siguiente posición de la memoria cuyos datos deben ser puesto en los terminales de salida. Por lo tanto, el puntero de lectura sigue constantemente al puntero de escritura. Cuando el puntero de lectura, alcanza al puntero de escritura, la FIFO está vacía. Si el puntero de escritura, alcanza al de lectura, la FIFO está llena.



- Los punteros toman la forma de contadores binarios.

7.9-CAM (CONTENT-ADDRESSABLE MEMORY)

- En su modo básico de operación, esta memoria lee una palabra de entrada a través de los terminales MQ, y la compara con todas las entradas que posee en la tabla CAM. El terminal MC# se activa siempre que la comparación haya terminado. Si se encuentra una coincidencia, el terminal MS#, se activa y los datos asociados con la entrada a comparar, son puestos en los terminales MQ. Si no se encuentra coincidencia, los terminales MQ permanecen en estado de alta impedancia.



7.10-MEMORIAS MAGNETICAS.

7.10.1-NUCLEO MAGNÉTICO:

- ☐ Las celdas son núcleos de ferrita.
- ☐ Son de escritura y lectura.
- ☐ Acceso aleatorio.
- ☐ No volátiles
- ☐ La lectura es destructiva.
- ☐ Se utilizaban como memoria principal. Obsoletas.

7.10.2-BURBUJA MAGNÉTICA: (MAGNETIC BUBBLES).

- ☐ Semiconductor. Los bits están constituidos por pequeños dominios (burbujas) magnéticos que están sobre una película magnética. La presencia o ausencia de una burbuja se interpreta como un uno o cero respectivamente.
- ☐ Son de lectura y escritura. Ambos procesos se realizan con la misma facilidad.
- ☐ Acceso secuencial. El movimiento de burbujas se realiza mediante campos magnéticos.
- ☐ No volátiles.
- ☐ Interfaz y control complicado y costoso.
- ☐ Utilizadas como memoria secundaria (no de trabajo) en aplicaciones específicas.

7.10.3-CINTA:

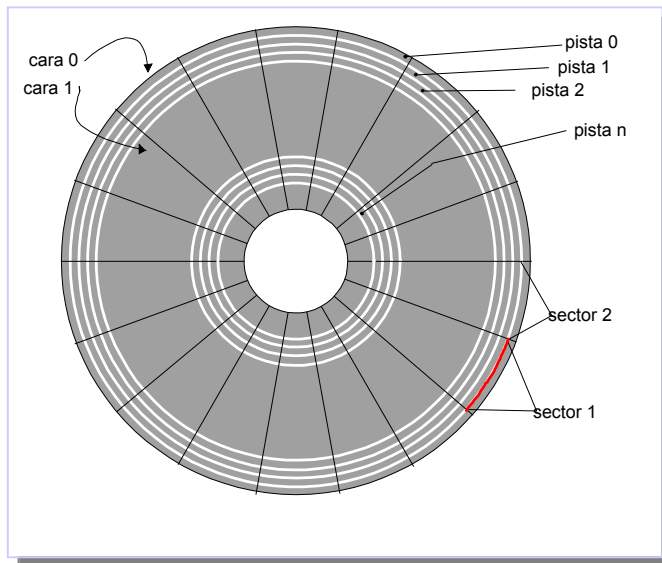
- ☐ Acceso secuencial.
- ☐ Lectura y escritura.
- ☐ No volátiles.
- ☐ Gran capacidad. Copias de respaldo.
- ☐ Partes móviles.

7.10.4-DISCO:

HARD DISK Y FLOPPY DISK.

- ☐ Tienen partes móviles.
- ☐ Son de escritura y lectura.
- ☐ Acceso aleatorio.
- ☐ No volátiles
- ☐ Utilizados como memoria secundaria.
- ☐ Capacidades muy grandes. (>100GB para los nuevos HD). Coste por BIT muy bajo.
- ☐ La lectura y escritura se realiza mediante cabezas magnéticas que polarizan en un sentido o en otro, pequeñas zonas (bits) de la superficie magnética del disco, dando como resultado el valor de 1 o 0 de estos bits.

FD:



Las dos caras del disco flexible (Floppy Disk) están impregnadas de una sustancia magnética. Cada una de estas caras está organizada según una serie de pistas concéntricas. Las dos pistas correspondientes en cada una de las caras forman un cilindro, el cual tiene el mismo número de identificación que el de las pistas que lo forman. Las pistas en cada cara, están divididas en sectores. La información (tanto datos de usuario como la información que necesita el sistema para el control de esos datos) se escribe en estos sectores. El tamaño de cada sector es de 512 bytes.

La numeración de las pistas comienza desde cero, la de las cabezas de lectura, también comienza en cero (cabeza 0 y cabeza 1), en tanto que los sectores comienzan desde uno.

El tipo de disco flexible más utilizado es el de 3^{1/2} pulgadas. Las características más importantes de formato de este disquete son:

Pistas por cara	Sectores por pista	Caras (cabezas)	Sectores totales	Capacidad
80	18	2	2880	1,44MB

HD:

En el disco duro (Hard Disk) los datos están organizados igual que en los diskettes según cabeza, cilindro y sector. Este tipo de discos está formado por una pila de varios platos que giran solidarios a un eje. Cada uno de estos platos es comparable a un disquete en cuanto a la organización de cada cara, lo que cambia es el valor de las características en cuanto al n°. de sectores, pistas, etc. valor que además, es muy variable de un HD a otro.

El diseño de las cabezas está realizado de tal manera que estas no tocan la superficie de los platos. Debido a la alta velocidad de giro de los platos, se forma una película de aire que mantiene las cabezas separadas de la superficie entre 0,1 y 1 micra.

El HD, nada más ser alimentado, comienza a girar, de esta manera se evitan retrasos en el acceso a los datos esperando que el disco adquiera la velocidad nominal.

7.11-MEMORIAS ÓPTICAS.

7.11.1-CD-ROM:

- Las celdas que contienen la información (bits), están formadas por pequeñas zonas con hoyos o quemaduras (pits) de 0,4 μm y lisas (sin hoyo), sobre una superficie circular que gira bajo un haz láser. Toda la información es guardada en una sola pista que tiene forma espiral. Cuando el haz choca con la superficie y no hay hoyo, el haz es reflejado hacia los sensores que toman esta acción como un 1 lógico. Si el haz choca con un hoyo, la luz se dispersa, no siendo detectada por los sensores ninguna luz de rebote. Esta acción es tomada como un 0 lógico.

- ❑ Gran capacidad (650 MB)
- ❑ Es de solo lectura.
- ❑ Acceso aleatorio.
- ❑ No volátiles
- ❑ Se utilizan como memoria secundaria.

7.11.2-RECORDABLE CD-ROM (WORM):

- ❑ Las características son las mismas que el CD-ROM, con la única diferencia de que son de lectura y escritura una sola vez por el usuario.
- ❑ Para la grabación se utiliza un haz Láser de alta energía que quema la superficie del disco creando los hoyos microscópicos en las zonas requeridas según la información a grabar.
- ❑ El proceso de lectura se realiza enfocando un haz Láser de baja energía sobre la superficie del disco, de la misma forma que en el CD de solo lectura.

7.11.3-DVCD, DVD

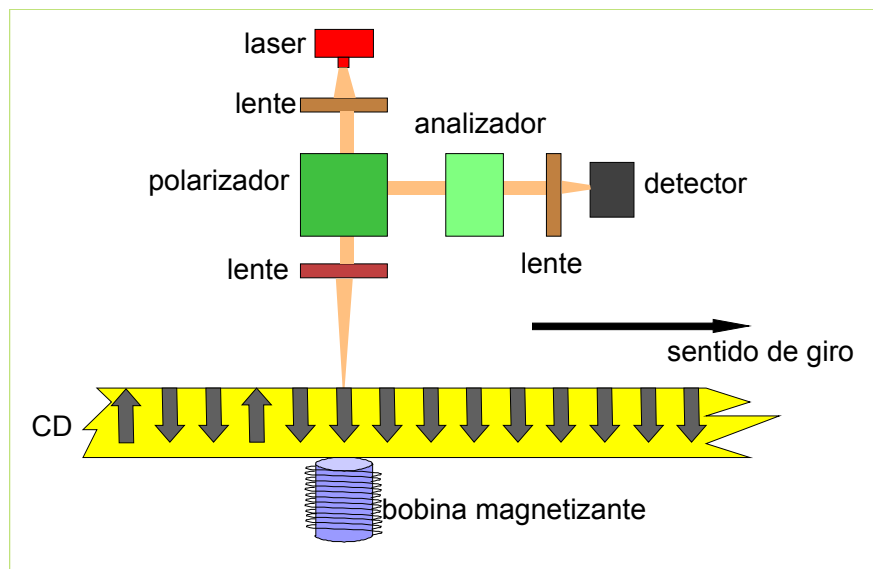
- ❑ Es un CD-ROM con mayor densidad de bits que éste.

7.12-MEMORIAS MAGNETO-ÓPTICAS.

7.12.1-RECORDABLE CD:

- ❑ Están basados en un sustrato de policarbonato o cristal, recubierto de fotopolímero y una capa de grabación uniformemente magnetizada.

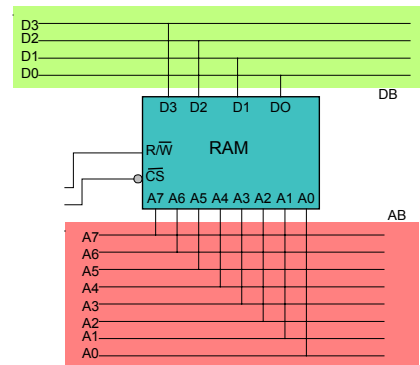
Para proceder a la grabación de los datos, un haz Láser de alta energía caliente (a alta temperatura) la zona (BIT) que debe ser grabada, de tal manera que borra la orientación magnética de las partículas existentes. En la cara opuesta a la de incidencia del Láser, una bobina genera un campo magnético que orienta las partículas del material según la información a grabar.



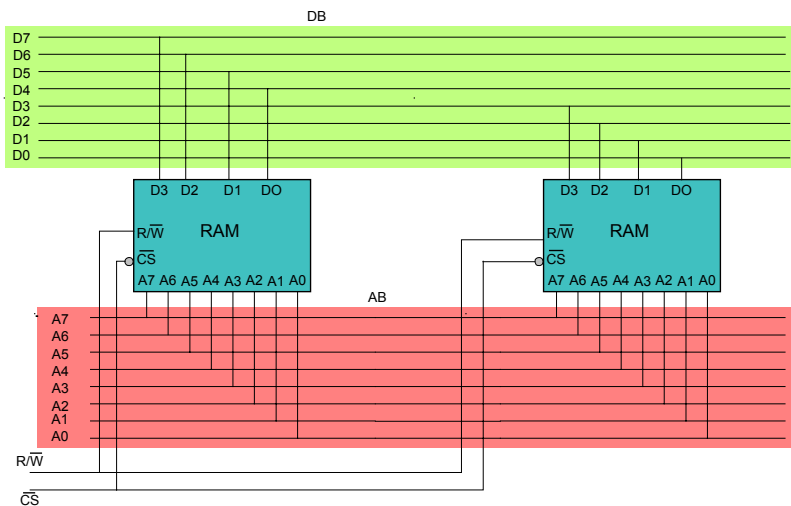
- ❑ La lectura se realiza de la misma forma que en los otros CD mediante un haz Láser de baja energía, sin intervención alguna de la bobina magnética. La reflexión va a depender del magnetismo del BIT.
- ❑ El formato es el de pistas y sectores de forma similar al de los FD y HD.

7.13-EXPANSIÓN DE MEMORIA INTEGRADA.

- ❑ *Expansión de palabra.* Partiendo de chips de memoria con un tamaño de palabra determinado, se trata de obtener un banco con una longitud de palabra que es múltiplo de la de los chips individuales. (p.e. partimos de la RAM de 256 X 4 de la figura)
- ❑ *Expansión de la capacidad.* Partiendo de chips de memoria con una capacidad determinada, se trata de obtener un banco con una capacidad que es múltiplo de la de los chips individuales.
- ❑ *Expansión de la longitud de palabra y de la capacidad.* En este caso se trata de conseguir la capacidad de la memoria y la longitud de palabra de la misma, a base de interconectar varios chips.



7.13.1-EXPANSIÓN DE LA LONGITUD DE PALABRA.

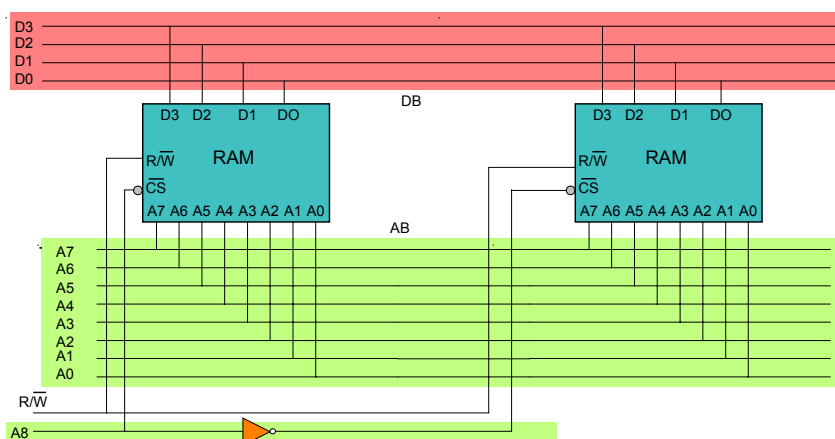


Los chips se interconectan de tal manera que las líneas del AB trabajen en paralelo, es decir cada una de las líneas que tienen la misma denominación (A0, A1,...) de cada uno de los IC-s están unidas entre si, y a la vez, a la línea del mismo nombre del bus de direcciones del sistema.

En cuanto a las líneas de datos, cada una se conecta a una línea diferente del bus de datos del sistema,

según la figura, de esta manera, cuando en el bus de direcciones del sistema se coloque una dirección, será accedida la misma posición en cada uno de los chips, pero el número de bits de datos será la suma del número de bits de cada posición. Así obtenemos un banco de 256 X 8.

7.13.2-EXPANSIÓN DE LA CAPACIDAD.



Los chips se interconectan de tal manera que las líneas del DB trabajen en paralelo, es decir cada una de las líneas que tienen la misma denominación (D0, D1, ...) de cada uno de los IC-s están unidas entre si, y a la vez, a la línea del mismo nombre del bus de datos del sistema. Las líneas del bus de

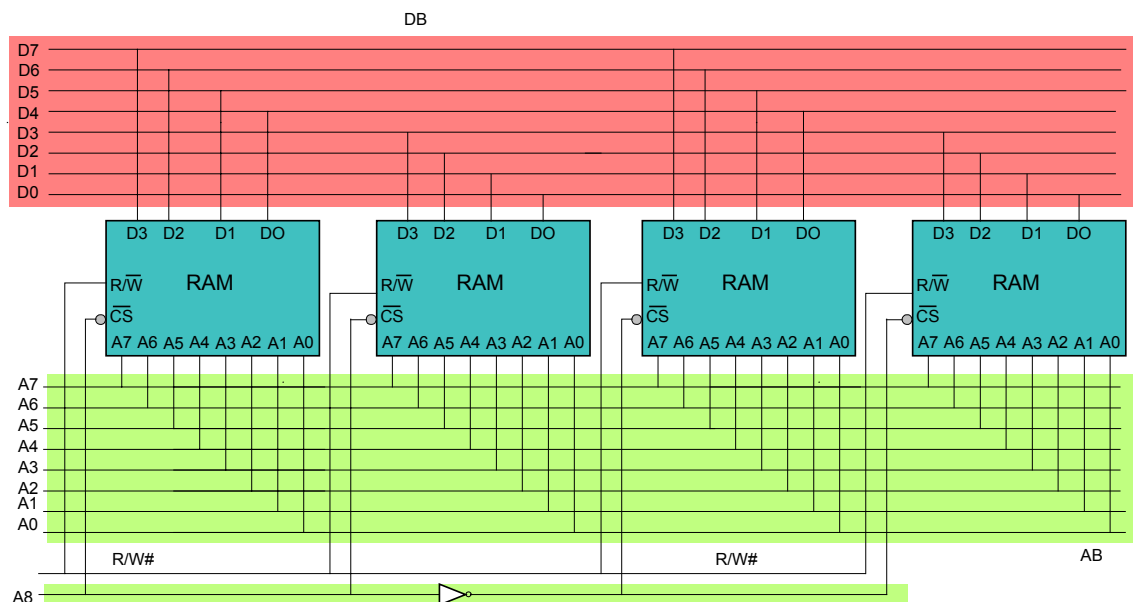
direcciones, se conectan de tal manera que trabajen también en paralelo.

Para ampliar la capacidad, se hace uso del terminal de CS# de los chips, y de una lógica o circuitos de selección que active un chip u otro, según la combinación puesta en las líneas del AB. Las líneas utilizadas en la lógica de selección, entran a formar parte del nuevo AB del sistema.

Cuando se pone una dirección en el AB, las líneas que están conectadas a los terminales de dirección de los chips, seleccionarán la misma posición en una y otra memoria, pero las líneas correspondientes a la lógica de selección de chip (que también forman parte del AB del conjunto formado), harán que sea uno u otro chip el que se active para realizar la transferencia de datos. Obtenemos así un banco de memoria de 256 X 8.

7.13.3-EXPANSIÓN DE LA LONGITUD DE PALABRA Y LA CAPACIDAD.

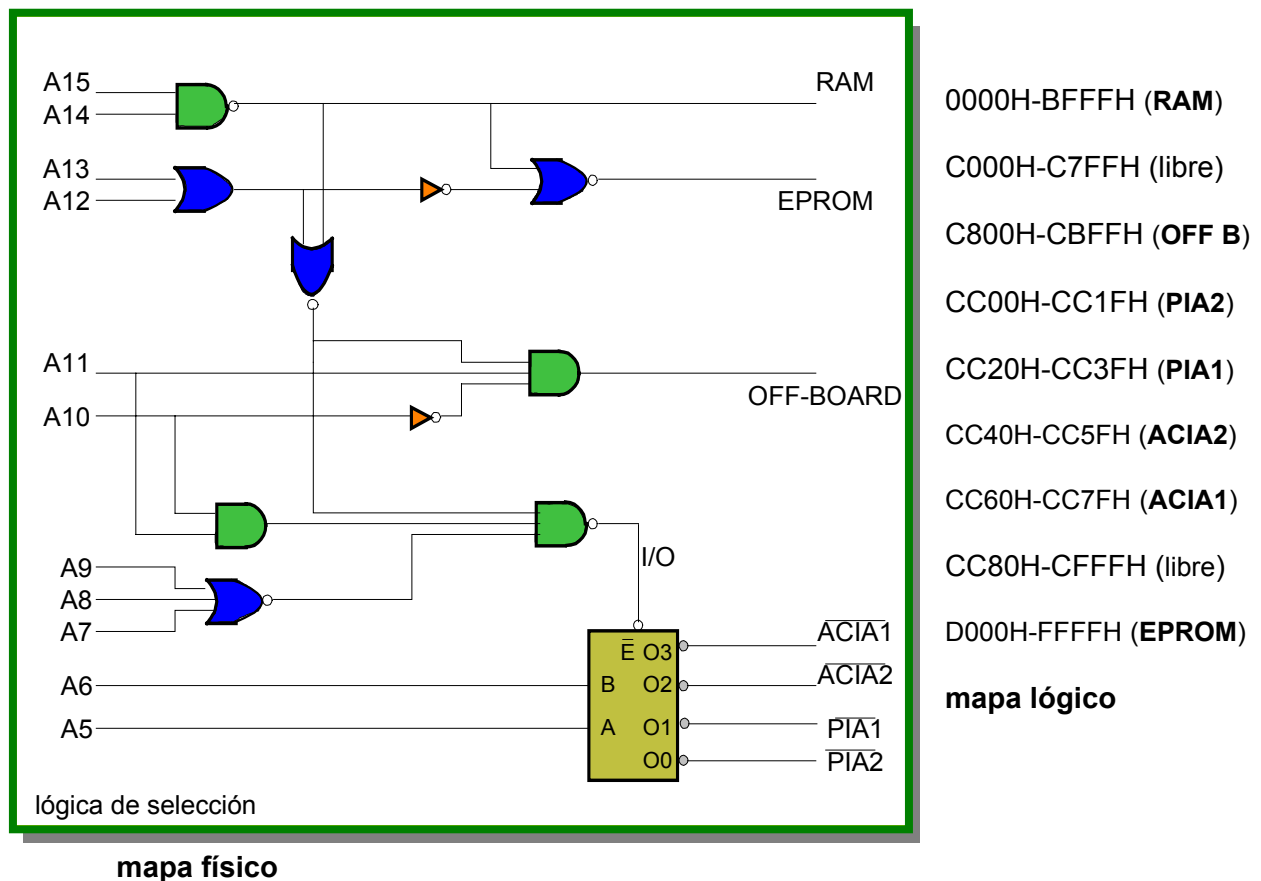
- La forma de conseguir la ampliación de los dos factores es una combinación de cada uno de los métodos anteriores.



Banco de 512 X 8

7.14-MAPA DE MEMORIA.

- ❑ El número de líneas de dirección en un sistema μ procesador, determina el margen de memoria direccionable (a la que se puede acceder); o de otra forma, el número de localizaciones posibles a las que se puede acceder.
- ❑ En ese margen no tienen porque estar instaladas todos los chips VLSI que cubran esa cantidad de posiciones o localizaciones direccionables. Puede haber espacios en los que no exista ningún elemento físico asociado a una determinada dirección (libre).
- ❑ También, puede haber espacios direccionables dentro de ese rango, que pertenezcan a chips de memoria, o a chips de I/O.
- ❑ La distribución que se hace de toda la capacidad de direccionamiento constituye el mapa de memoria.
- ❑ Esta distribución puede ser lógica o física. La distribución lógica representa el uso que el sistema μ procesador hace de cada una de las posiciones direccionables (*MAPA LÓGICO*). La distribución física representa la correspondencia entre las posiciones direccionables y el dispositivo físico que las soporta (*MAPA FÍSICO*).



Para determinar el mapa lógico a partir del mapa físico, debemos observar qué líneas del bus de direcciones mantienen seleccionado un chip de memoria, y qué líneas varían para seleccionar las diferentes posiciones de ese chip.

Determinemos por ejemplo, el rango de direcciones que abarca la PIA1# en el mapa físico de la figura.

Debemos observar que dicha PIA será seleccionada cuando la salida O1 del decodificador sea activa. Esto va a suceder cuando se cumplan las condiciones de que las

entradas B y A de dicho decodificador tengan el valor 0 y 1 respectivamente. Pero además, dicho decodificador debe ser activo (habilitado), por lo que su entrada de habilitación (E#) debe ser activa. Esta condición se cumplirá cuando la salida de la puerta NAND asociada a la entrada de habilitación del decodificador, sea activa.

Teniendo en cuenta todas las consideraciones anteriores, observamos que los valores que deben de tener las líneas del bus de direcciones para que se cumplan son:

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
1	1	0	0	1	1	0	0	0	0	1					

Mientras el valor de cada una de las líneas del bus de direcciones, desde la A15 hasta la A5 tengan el valor indicado en la tabla, la salida O1 del decodificador será activa, y por lo tanto, si esta salida está conectada al CS# de la PIA1, dicha PIA será el elemento seleccionado.

Para poder seleccionar los distintos registros dentro de la PIA, (o las distintas posiciones, en el caso de que el elemento seleccionado fuera un chip de memoria), tendrán que ser las líneas restantes que aún no han entrado en juego, es decir, desde la A4 hasta la A0, las que hagan dicha selección. No hay por qué utilizar todas esas líneas restantes, esto va a depender de la capacidad de la PIA (o de la memoria, en su caso).

Se puede entonces ver que con la siguiente combinación en el bus de direcciones, estará seleccionada la PIA1:

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
1	1	0	0	1	1	0	0	0	0	1	0	0	0	0	0

combinación que en hexadecimal representa el valor **CC20H**, y que con la siguiente combinación, también estará seleccionada la PIA1:

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
1	1	0	0	1	1	0	0	0	0	1	1	1	1	1	1

combinación que en hexadecimal representa el valor **CC3FH**. Pero, además, también estará seleccionada con todos los valores comprendidos entre los dos anteriores, dado que los valores de las líneas A15 hasta A5 no han variado.

Se puede decir por lo tanto, que el rango de direcciones donde se encuentra la PIA1 va desde la dirección **CC20H** hasta la **CC3FH**.



Sistema microprogramable

8

8.-SISTEMA MICROPROGRAMABLE. GENERALIDADES.

8.1-LÓGICA CABLEADA Y LÓGICA PROGRAMADA.

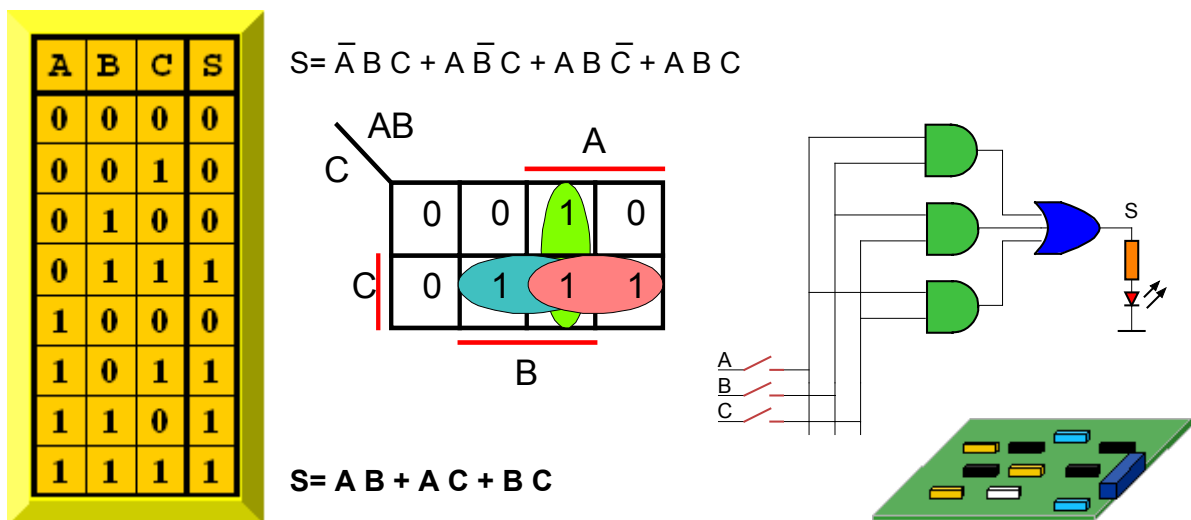
- La realización de un sistema mediante lógica cableada, requiere el diseño e implementación (construcción) de los circuitos necesarios para que el sistema cumpla con su función. En este caso, para cada función específica se requiere un diseño de circuito específico.
- Mediante lógica programada, una estructura física general compuesta por los circuitos electrónicos necesarios, se adapta a los diferentes requerimientos impuestos por las distintas funciones, que diferentes aplicaciones necesiten. En este caso, el cambio que debe ser realizado es el de una serie de instrucciones (programa), que gobiernan el funcionamiento de dicha estructura.

Ejemplo:

Aplicación 1: Se necesita un circuito que tenga tres interruptores como variables de entrada y que la salida de dicho circuito sea un nivel alto cuando la mayoría de los interruptores estén cerrados.

Diseño e implementación:

Mediante cableado:



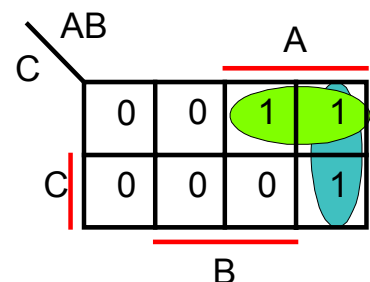
A	B	C	S
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

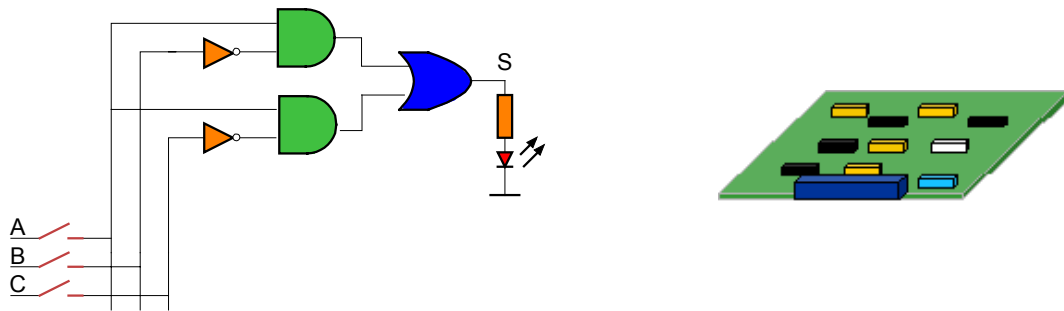
Aplicación2: Se necesita un circuito que tenga tres interruptores como variables de entrada y que la salida de dicho circuito sea un nivel alto cuando se cumpla que, estando el interruptor **A** cerrado cualquiera de los otros dos, esté abierto.

Diseño e implementación mediante cableado:

$$S = A \bar{B} \bar{C} + A \bar{B} C + A B \bar{C}$$

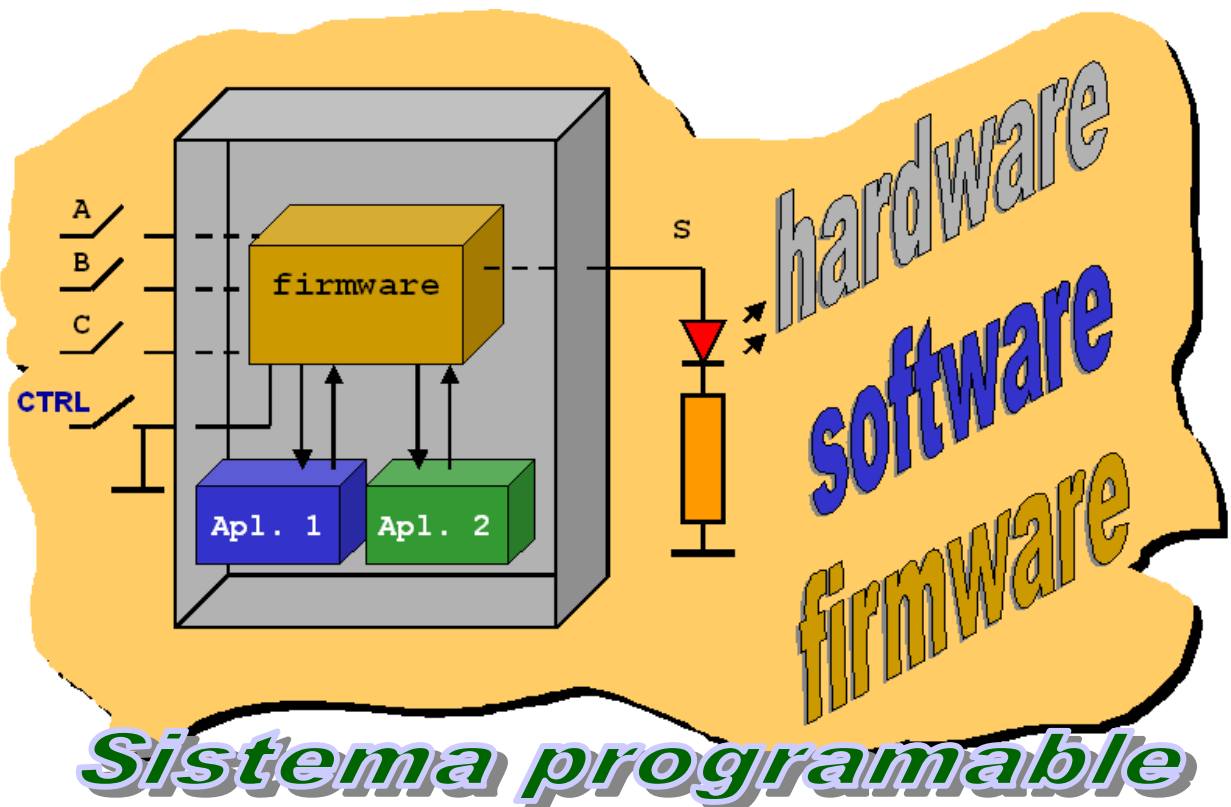
$$S = A \bar{C} + A \bar{B}$$





Para la realización de la aplicación 2 es necesario modificar o cambiar la estructura del anterior diseño de la aplicación 1.

Mediante lógica programada:



Una misma estructura electrónica (hardware), es capaz de realizar las dos funciones, simplemente debemos indicar al sistema, cual de las dos aplicaciones queremos que se lleve a cabo en un momento dado. Para ello, el sistema gestiona mediante el firmware, una serie de instrucciones que forman un programa (software) y que previamente han sido introducidas en dicho sistema. Que deseamos que se ejecute la aplicación 1, o la aplicación 2, se lo indicamos al sistema mediante el control externo CTRL.

Que en un momento dado ya no sirve cualquiera de las dos aplicaciones, o simplemente, queremos mejorar la eficiencia de las mismas, se introducen en el sistema de forma técnicamente fácil un nuevo conjunto de instrucciones y el sistema cambiará su forma de operar.

8.2-SISTEMA MICROPROGRAMABLE.

- Un sistema micro-programable es un sistema electrónico capaz de interpretar y ejecutar una serie de instrucciones (programa), con el fin de realizar una función determinada dictada por dichas instrucciones.

PARTES FUNDAMENTALES:

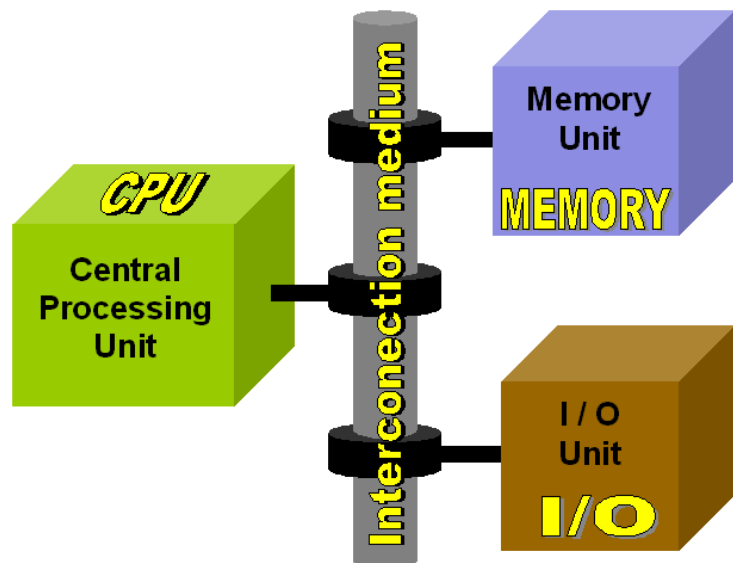
- **HARDWARE:** Es la parte electrónica y mecánica del sistema. Es decir, la parte física del sistema.
- **SOFTWARE:** Está compuesto por la serie de instrucciones que componen los programas, y por los lenguajes empleados en la realización de los mismos.
- **FIRMWARE:** Serie de instrucciones guardadas en memoria no volátil que prepara los elementos electrónicos para una configuración básica del sistema microprogramable. Generalmente, el firmware no forma parte de la aplicación específica que el sistema debe ejecutar, sino que prepara a éste para que dicha aplicación pueda ejecutarse.

PARTES FUNDAMENTALES HARDWARE.

- **CPU (CENTRAL PROCESSING UNIT):** La CPU es la parte principal del sistema microprogramable; en función de las instrucciones dadas por un programa, la CPU genera todas las señales de control y gobierno del sistema. La palabra CPU y microprocesador se utiliza para referirse al mismo componente, aunque en sistemas informáticos cuando se habla de CPU algunas veces se hace referencia al conjunto de los cuatro bloques mostrados en la figura.

- **MEMORY UNIT:** Este bloque se encarga de almacenar los bits que son tratados por la CPU. Estos bits, representan en unos casos las instrucciones (programa) que definen el comportamiento del sistema, y en otros casos la información que debe ser tratada según dichas instrucciones.

- **I/O UNIT:** Es el camino de paso del flujo de bits entre el sistema y los dispositivos que a éste se conectan.



- **INTERCONNECTION MEDIUM:** Es el medio que comunica los distintos bloques del sistema. Está compuesto por conductores que son compartidos por los distintos componentes del sistema. Este medio de interconexión se divide en grupos según la función que desempeña. Cada uno de esos grupos recibe el nombre de **BUS**.

- Una segunda descripción de los componentes fundamentales hardware es la que se representa en la figura. En ella vemos que la unidad de memoria se encuentra dividida en dos partes, según la memoria sea de solo lectura o de lectura y escritura. En la memoria de sólo lectura, generalmente irá guardado el programa que define la aplicación, y en la memoria de lectura y escritura se irá guardando la información que va procesando el programa.

- La unidad de entrada y salida, también está dividida en dos partes. Una de ellas se hace cargo del flujo de bits que va desde el interior del sistema a los dispositivos conectados a él y la otra parte se encarga del flujo que va desde los dispositivos al interior del sistema.

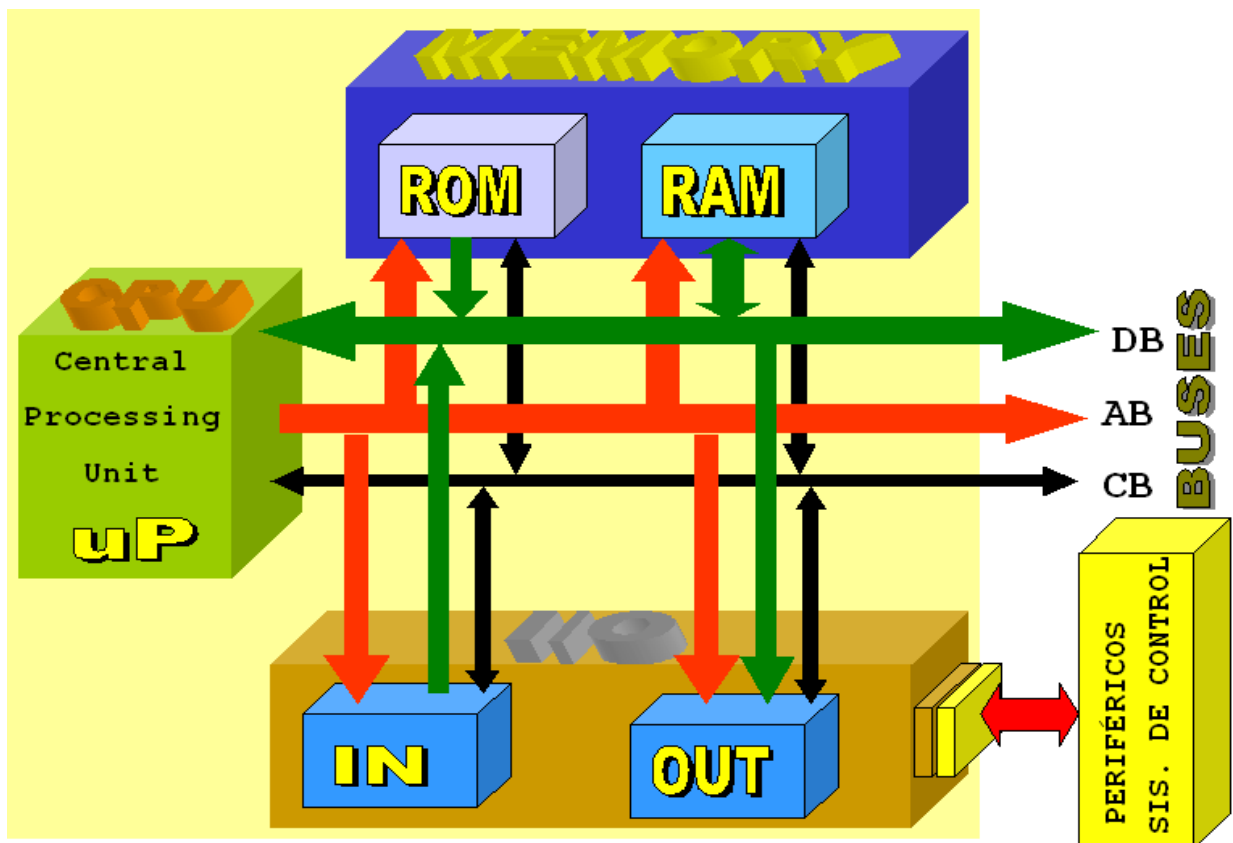
Los dispositivos que se conectan al sistema se pueden dividir en *periféricos* y *sistemas de control*. La diferencia entre ellos está en que los periféricos comunican al ser humano con el sistema.

- El medio de interconexión está dividido en tres grupos, que como ya se ha indicado se llaman buses. Tenemos el **AB** (*Address Bus*), el **DB** (*Data Bus*) y el **CB** (*Control Bus*).

El **AB** está compuesto por un conjunto de líneas cuyo número define el tamaño del Bus de Direcciones. En este bus, la CPU pondrá un valor binario, que será la dirección del elemento que es requerido para participar en una determinada función.

En el **DB** será puesta toda la información que deba ser procesada por la CPU.

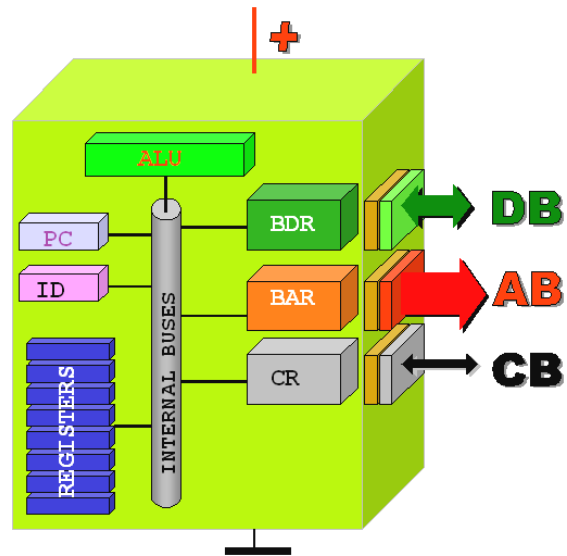
El **CB** está compuesto por aquellas líneas necesarias para poder sincronizar todos los elementos del sistema, líneas para indicar un tipo de operación particular, líneas mediante las cuales las distintas unidades indican a la CPU un estado particular, etc.



CENTRAL PROCESSING UNIT (CPU)

□ **BAR: BUS ADDRESS REGISTER.**

La necesidad de poder transferir información a través del BUS entre dos unidades conectadas a él, sin ningún tipo de interferencia por parte de otras unidades y la diferente velocidad a la que esas unidades puedan trabajar, hacen necesaria la incorporación de *BUFFERS* en la *INTERFAZ* de las unidades conectadas al BUS. Estos buffers pueden ser simplemente registros de almacenamiento, y suelen tener capacidad de corriente (FAN-OUT) suficiente para poder atacar un cierto número de dispositivos externos.



□ **BDR: BUS DATA REGISTER.**

El registro del bus de datos pone en contacto el bus interno de datos de la CPU y el bus de datos del sistema.

□ **ALU. ARITHMETIC AND LOGIC UNIT.**

Se encarga de realizar las operaciones aritméticas (suma y resta), con signo y sin él, con acarreo y sin acarreo, y lógicas (AND, OR, X-OR y NOT) que realiza la CPU.

□ **PC. PROGRAM COUNTER.**

Mediante este contador interno, la CPU controla en todo momento, de qué posición de memoria se debe obtener (*fetch*) la instrucción que va a ser ejecutada, sabiendo de esta manera donde tiene que dirigirse para buscar la próxima instrucción.

□ **ID. INSTRUCTIONS DECODER.**

Este decodificador, toma como entrada el código binario de la instrucción correspondiente y a partir de él, una vez interpretado, genera las señales de control, de datos o de dirección necesarias para la ejecución de dicha instrucción.

□ **REGISTERS. REGISTROS.**

Son un conjunto de registros que pueden tener un uso general o un uso específico (*ACCUMULATOR, STACK POINTER, STATUS REGISTER...*).

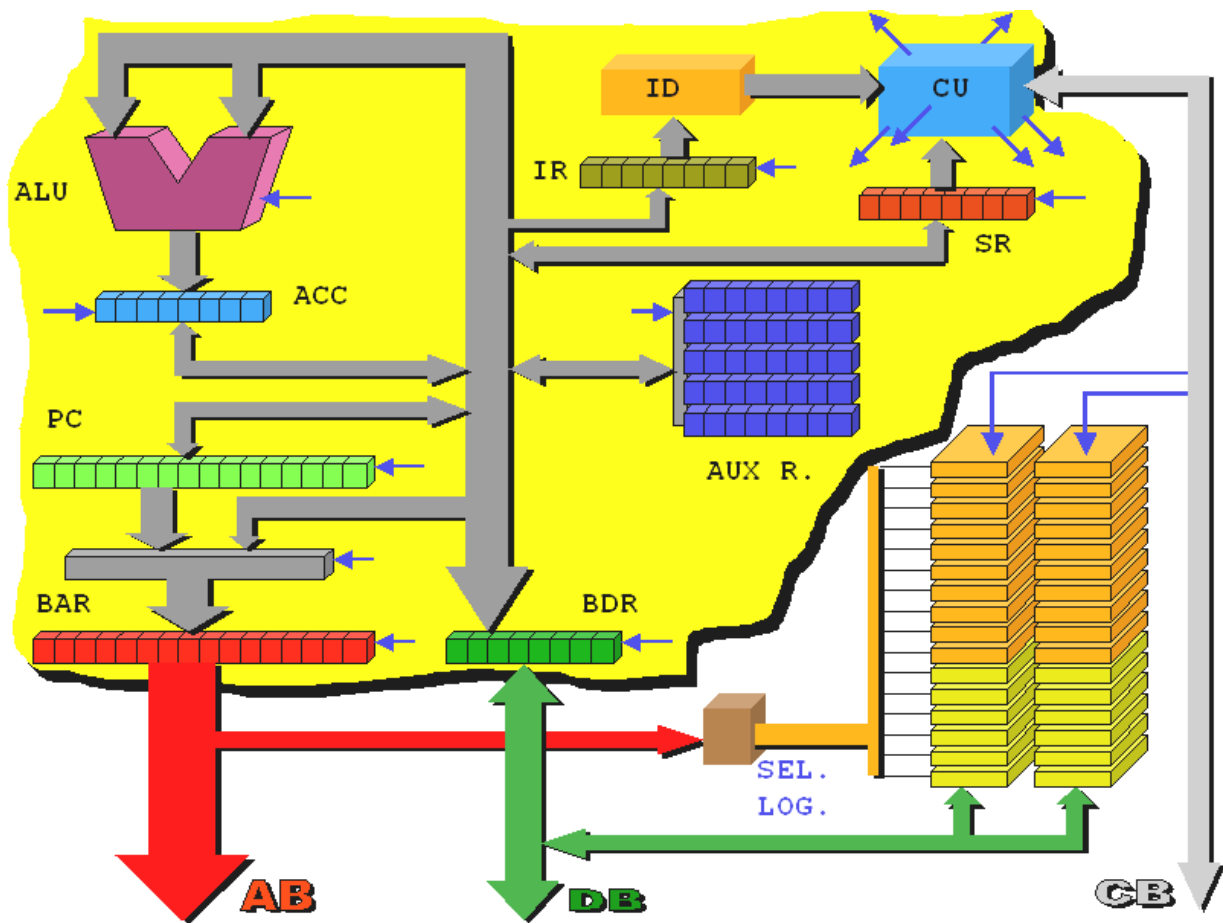
□ **CR. CONTROL REGISTER**

Este bloque, gobierna y coordina todas las operaciones que tienen lugar dentro de la CPU; No gobierna la operación de todo el sistema. En implementaciones reales, el control está dividido en diferentes unidades, cada una de las cuales posee los circuitos de control necesarios para que pueda trabajar adecuadamente dentro del sistema.

□ **INTERCONNECTION MEDIUM. BUSES.**

La unión de todos estos componentes se realiza mediante los buses internos, a los cuales vierten información, generalmente a través de salidas *TRI-STATE*, y toman información de ellos.

FUNCIONES BÁSICAS DE LA CPU.



Las operaciones básicas que realiza una CPU pueden clasificarse de la siguiente manera:

Transferir un dato entre dos registros.

Leer una posición de memoria y llevar el contenido a un registro.

Leer un registro y llevar su contenido a una posición de memoria.

Efectuar operaciones aritméticas (suma y resta) entre dos datos.

Efectuar operaciones lógicas (AND, OR, X-OR, NOT) entre dos datos.

Incrementar o decrementar el contenido de un registro.

Desplazar o realizar rotaciones sobre el contenido de un registro.

Alterar la secuencia de ejecución de las instrucciones.

- La realización de cada una de las anteriores operaciones requiere una serie de pasos o “microacciones” realizadas de forma sincronizada. Por ejemplo la operación de leer una posición de memoria y llevar el contenido a un registro se puede descomponer en los siguientes pasos:

a) *Buscar el código de la operación a realizar.*

- 1 Puesta en el AR de la dirección de memoria donde se encuentra la instrucción que se va a buscar y la habilitación del AR para la puesta de esa dirección en el AB.
- 2 Activación de una línea de control que indique una operación de lectura en la memoria. (Activación de la línea READ).
- 3 Activación del DR para permitir que los datos en el DB pasen a aquel.
- 4 Incremento del PC. (Observar que la instrucción en curso todavía no se ha terminado de ejecutar y el PC está ya apuntando a la siguiente posición de memoria).

b) Interpretar el código de la operación a realizar:

- 1 El dato pasa al bus interno y se carga en el IR (por ser una instrucción).
- 2 Se ordena la carga desde el IR al ID.
- 3 Decodificar el código de la operación para generar las señales de bus y control correspondientes.

c) Buscar el operando:

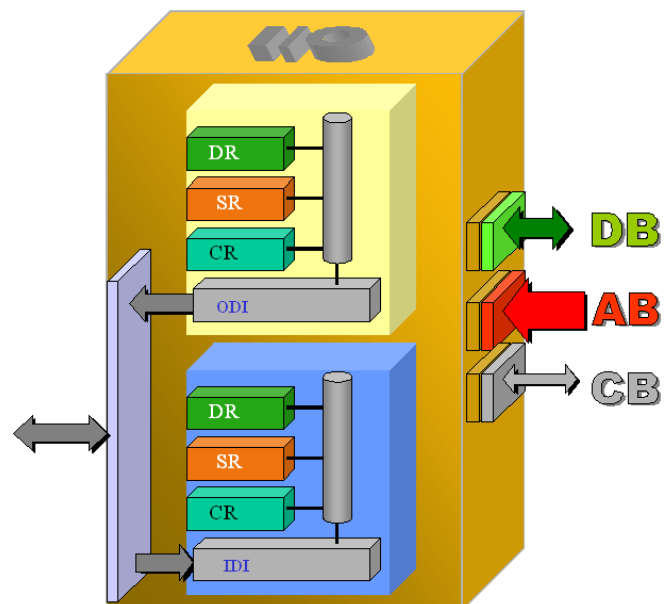
- 1 Poner la dirección del PC en el AR y autorizar la salida por el AB de la dirección correspondiente al operando.
- 2 Activar la línea de control (read) de lectura en la memoria para poner el operando de ésta al DB.
- 3 Activación del DR para permitir que los datos en el DB pasen a aquel.
- 4 Incrementar el PC.

d) Guardar el operando en el registro correspondiente:

- 1 El operando pasa al bus interno y se carga en el registro.

I/O UNIT. (UNIDAD DE ENTRADA / SALIDA)

- ❑ **DR: DATA REGISTER.**
Sirve como buffer para la transferencia de datos. Todos los datos en una y otra dirección pasan por estos registros.
- ❑ **SR: STATUS REGISTER.**
Guarda información sobre el estado en el que se encuentra el dispositivo externo o la propia unidad de entrada salida después de cada transferencia. El contenido de este registro puede variar de forma automática por la propia operación realizada o puede ser modificado por programa.
- ❑ **CR: CONTROL REGISTER.**
Guarda el modo de operación de la unidad y del dispositivo externo para que ambos puedan trabajar con los mismos criterios.
- ❑ **ODI, IDI (.OUTPUT DEVICE INTERFACE, INPUT DEVICE INTERFACE).**
Sincronizan la transmisión de información. Adaptan un medio a otro.



❑ PERIFÉRICOS Y SISTEMAS DE CONTROL.

El término periférico en computación de consumo, suele reservarse para el elemento que comunica al sistema con el usuario (teclado, display, impresora, plotter, modem, CD, FD, scanner...etc.). Bajo sistema de control se engloba a sensores, transductores y actuadores para automatización en general.

Las operaciones de I/O deben realizarse en momentos adecuados y precisos; para ello existen unas normas que tanto sistema μ procesador como periférico deben respetar. El conjunto de estas normas constituye el **protocolo**.

- ❑ Las unidades de I/O vienen dadas en forma integrada (chip) y dependiendo del fabricante tienen diferentes nombres:

INTERFACES UNIVERSALES DE I/O: PIA (*Peripheral Interface Adapter*), **VIA** (*Versatile Interface Adapter*)...

INTERFACE PARA COMUNICACIONES SERIE: **UART** (*Universal Asynchronous Receiver Transmitter*), **USART** (*Universal Synchronous Asynchronous Receiver Transmitter*)...

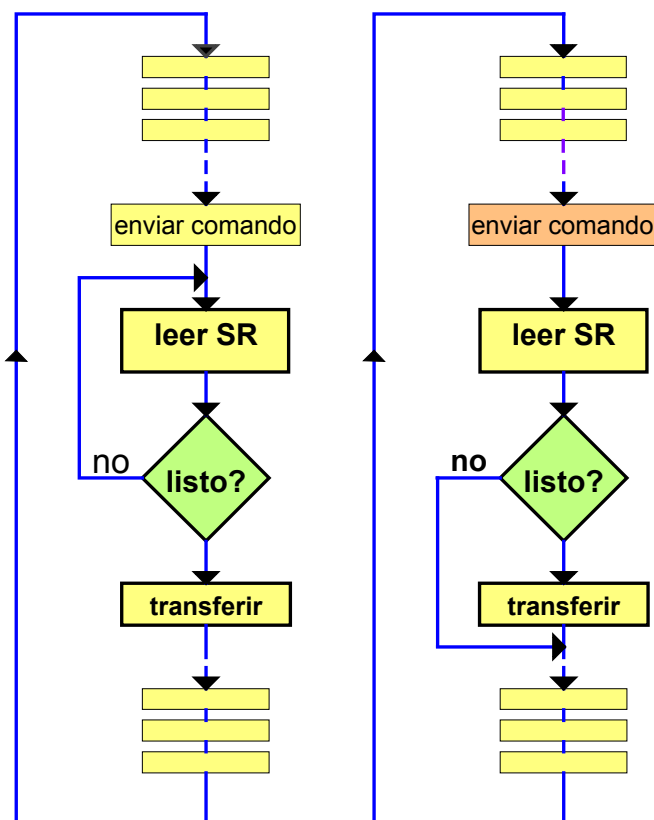
CRTC (*Cathode Ray Tube Controller*)

....

8.3-PROCEDIMIENTOS DE MANIPULACIÓN DE DISPOSITIVOS EXTERNOS.

-PROGRAM-CONTROLLED I/O.(ENTRADA/SALIDA CONTROLADA POR PROGRAMA).

- ❑ Con este procedimiento, el programa toma la iniciativa de la realización de una transferencia, con lo cual, esta tiene lugar de forma síncrona con la ejecución del programa.



- ❑ La **CPU** controla el estado de los dispositivos externos mediante la realización de consultas periódicas al SR (*STATUS REGISTER*). La lectura y escritura de datos se realiza a través del DR (*DATA REGISTER*) y el modo de operación a través del CR (*CONTROL REGISTER*).
- ❑ La secuencia se inicia con el envío de un comando (escritura en el CR) a la interfaz para definir el modo de operación. Después, se realizan consultas al SR y pueden darse dos casos (ver figura): (a) entrar en un bucle de lectura constante del SR hasta que el periférico esté disponible, en cuyo caso se realiza la transferencia y (b), si el SR no está disponible, seguir con la ejecución del programa hasta que se vuelva a realizar otra consulta. Si el SR está disponible, efectuar la transferencia.

Inconvenientes:

- ❑ Con este sistema el dispositivo externo (aunque necesite la atención por parte de la CPU), debe esperar hasta que la CPU realice la consulta periódica.
- ❑ Se pierde tiempo realizando las consultas periódicas. Esta pérdida es más acentuada cuanto menor es la actividad de I/O.
- ❑ Todo esto revierte en "más lentitud" debido a la intervención de la CPU en cada paso, pero a la vez revierte en seguridad al estar controlado todo el proceso desde que se diseña el programa.

-DIRECT MEMORY ACCESS (DMA). (ACCESO DIRECTO A MEMORIA)

- ❑ Mediante un controlador de DMA, el dispositivo externo (cuando lo solicita) manda datos a la memoria o viceversa de forma más eficiente que la CPU al ser aquel (DMAC) un dispositivo diseñado específicamente para ese fin. La intervención de la CPU se reduce solamente a iniciar el proceso e indicar ciertas pautas a seguir por el controlador, de esta manera se alcanzan grandes velocidades de transferencia.

-INTERRUPTS. (INTERRUPCIONES)

- ❑ Mediante este mecanismo, los dispositivos externos haciendo uso de una línea de interrupción, rompen la secuencia normal del programa para solicitar al μ procesador su atención. De esta manera, la CPU solamente atiende a los dispositivos externos cuando se necesita una transferencia y no se pierde tiempo en consultas periódicas.

8.4-MEDIOS PARA PROVEER EL ACCESO A LA INFORMACIÓN EN EL I/O INTERFACE.

Existen básicamente dos medios para acceder a la información de I/O:

- ❑ *Mediante instrucciones específicas para que la CPU las ejecute como parte de una rutina de I/O.* Este esquema es utilizado por los μ P-s de la familia Intel®
- ❑ *Haciendo uso de instrucciones de carácter general.* En este caso, a cada registro (DR, CR y SR) se le asigna una dirección dentro del mapa de memoria principal, de forma que sean accesibles de la misma manera que cualquier posición de la memoria del sistema. A este método se le llama *Memory-Mapped I/O (E/S mapeada en memoria)*, esquema utilizado por Motorola®.

8.5-BUSES TÍPICOS. TIMING

BUS SÍNCRONO.

- Todas las operaciones en el BUS se realizan tomando como referencia una señal de reloj de tal manera que todos los dispositivos del BUS sincronizan sus acciones con el reloj.

Entrada:

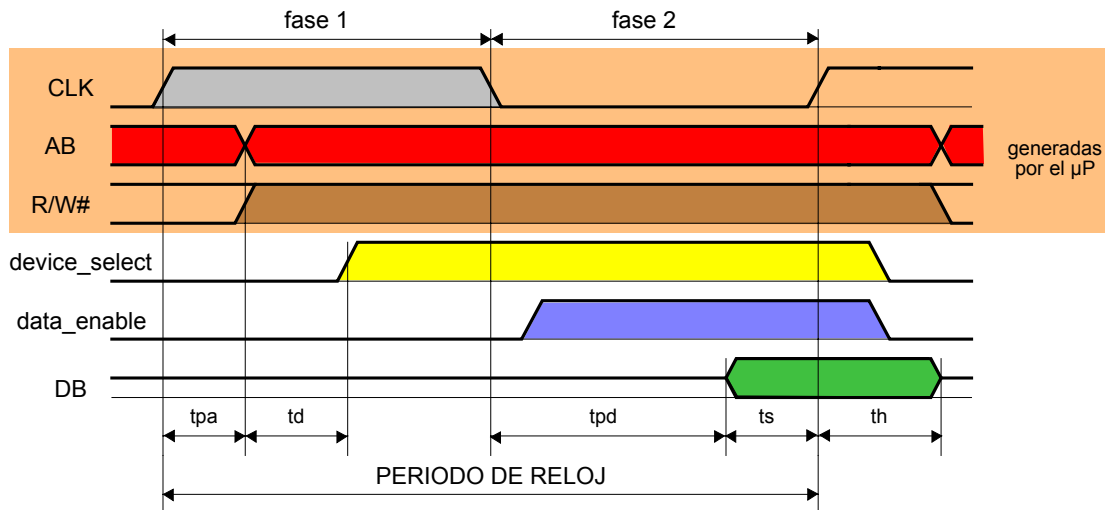


diagrama temporal para una operación de entrada

-El μ procesador pone la dirección del dispositivo en el BUS de direcciones, e indica la operación que desea realizar mediante la línea R/W#. Después de un retardo t_d , el decodificador de direcciones (device_select), selecciona el dispositivo sobre el que se va a realizar la operación.

Se puede observar que la operación de entrada está dividida en dos estados (fases) que se corresponden con el estado alto y bajo del reloj. En la fase 1, el μP transmite una nueva dirección y las señales de control necesarias. Esta información es recibida y decodificada por los dispositivos implicados. En la fase 2, tomando como referencia el flanco de bajada del reloj; el dispositivo seleccionado pone los datos en el BUS para que sean leídos por el μP .

Salida:

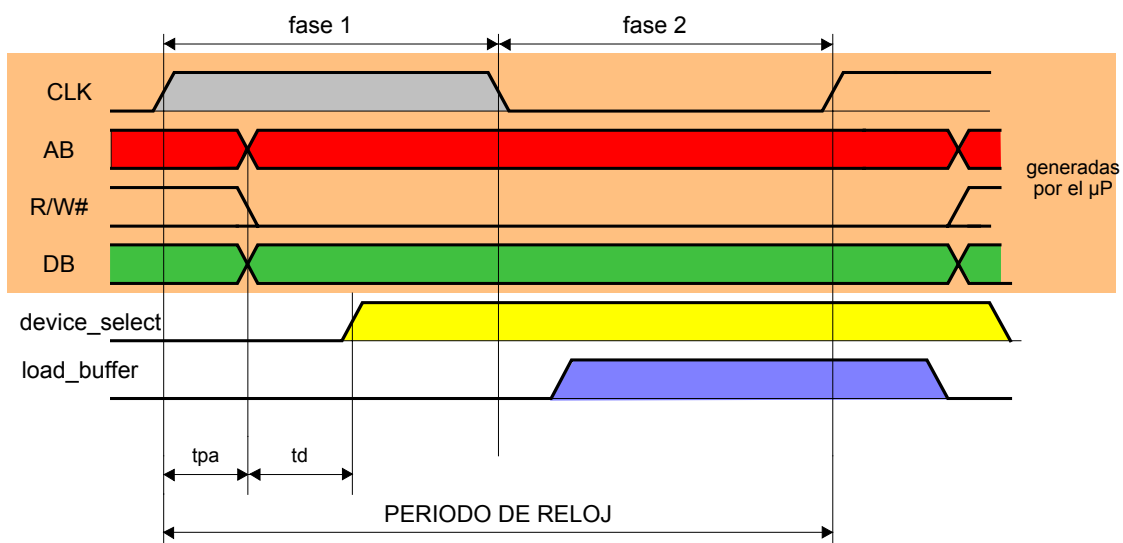


diagrama temporal para una operación de salida

-Al comienzo del periodo de reloj, el μP transmite la dirección del dispositivo a donde van a ir los datos, los cuales, también coloca en el DB (Data Bus). Al mismo tiempo, pone a 0 la línea R/W# indicando que la operación a realizar es de escritura. Al comienzo de la fase 2 todas las señales están estables, generándose por lo tanto la señal que pone los datos del BUS en el dispositivo seleccionado.

- Cuando varios dispositivos que trabajan a diferentes velocidades, están conectados a un sistema, se puede hacer que toda transferencia tenga lugar dentro de un periodo de reloj sin más que acomodar dicho periodo al dispositivo más lento.
- Para que un dispositivo que sea más lento, pueda ser conectado a un BUS sin que la velocidad de transferencia de los otros dispositivos se vean afectadas, existen dos alternativas: *Estados de Espera* (dentro del modo síncrono) y modo *ASÍNCRONO*.

ESTADOS DE ESPERA (WAIT STATES).

-A la interface del dispositivo se le permite saltarse uno o más ciclos de reloj antes de completar su parte de la operación de transferencia.

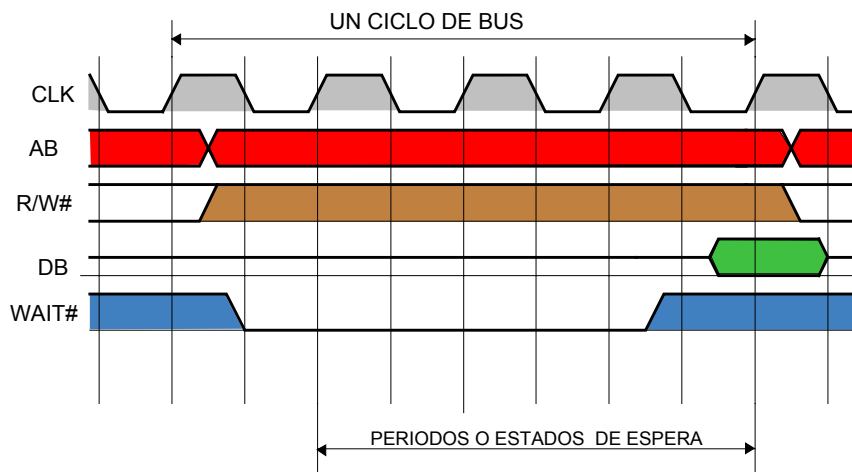


diagrama temporal para una operación de entrada

-Durante este periodo, el μP debe ser informado para que suspenda sus operaciones internas hasta que el dispositivo esté listo.

-La interfaz del dispositivo, una vez que ha sido seleccionado y sabiendo que no puede transmitir los datos de forma inmediata, activa la señal WAIT# con lo que el μP entra en un estado de espera.

Cuando el dispositivo está listo para transmitir los datos, desactiva WAIT# y pone los datos en el DB en lo que correspondería en tiempo a la fase 2 si no hubiera habido estados de espera.

-La señal WAIT# debe cumplir ciertos requerimientos de tiempos como es el de ser activada en la primera fase del periodo de reloj.

BUS ASÍNCRONO.

- El BUS ASÍNCRONO no depende de una señal común de reloj para sincronizar los diferentes estados de una operación de transferencia de datos. En su lugar, se utiliza un protocolo de intercambio de señales (*HANDSHAKE SIGNALING*).

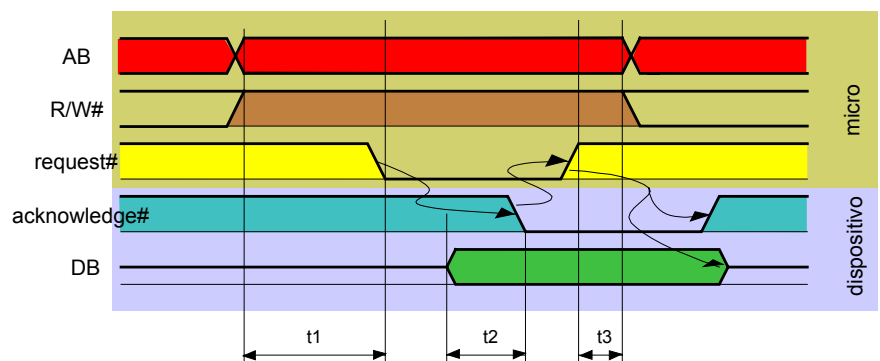


diagrama temporal para una operación de entrada en un bus asíncrono

-En lugar de una señal de reloj, se proveen dos líneas, *REQUEST#* y *ACKNOWLEDGE#*.

-El μP pone una dirección en el AB y especifica la tarea que quiere realizar mediante la línea *R/W#*; después de un retardo (t_1), activa la línea de *REQUEST#*. Cuando esta señal es recibida por el dispositivo direccionado, éste pone los datos en el DB y después de un retardo (t_2), activa la línea de *ACKNOWLEDGE#* indicando al μP que los datos pedidos están en el BUS. El μP carga estos datos en su buffer de entrada y desactiva la línea *REQUEST#*; después de un retardo (t_3), retira las direcciones y control del BUS. Finalmente, el dispositivo desactiva la línea de *ACKNOWLEDGE#* y retira los datos del BUS en respuesta a la vuelta al estado inactivo de la línea *REQUEST#*.

- Los retardos t_1 , t_2 y t_3 son introducidos por el μP y las interfaces de los dispositivos para asegurar una correcta transferencia e interpretación de la información.

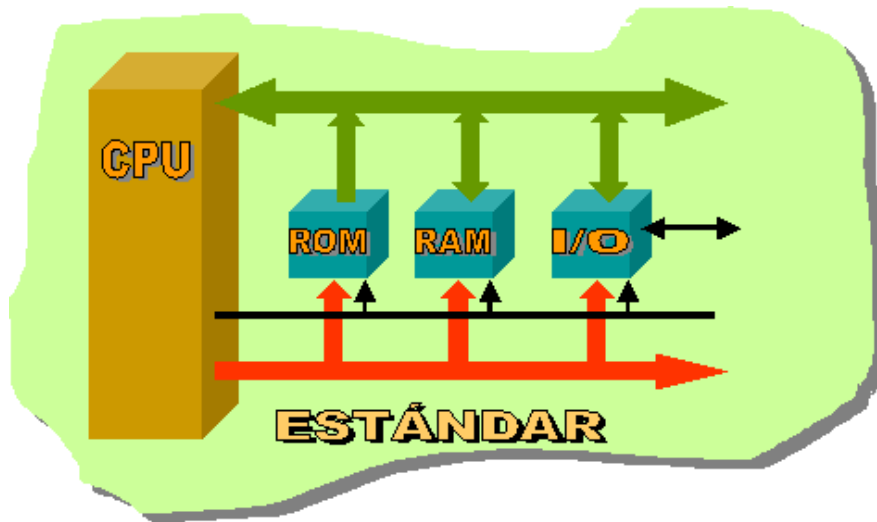
t_1 : el principal propósito de este periodo es compensar un retardo conocido como *BUS SKEW*, el cual representa la diferencia de los retardos en la propagación de las diferentes líneas de un BUS. También cubre los retardos producidos por el decodificador de direcciones.

t_2 : También sirve para compensar el *BUS SKEW*.

t_3 : Garantiza que la información en el AB y CB, vista por cualquier dispositivo del BUS, es válida tanto tiempo como la línea de *REQUEST#* permanece activa.

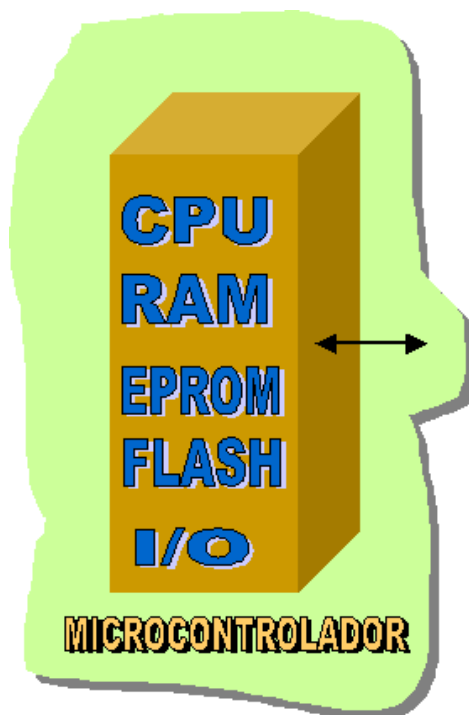
8.6-ARQUITECTURAS BÁSICAS DE SISTEMAS μ PROGRAMABLES.

ARQUITECTURA ESTÁNDAR.



- Esta arquitectura es la utilizada por los μ P de uso general. Se caracteriza por la flexibilidad en la integración de sistemas

ARQUITECTURA MICROCONTROLADOR



- En un mismo chip están integrados los elementos que componen la arquitectura estándar, es decir, la CPU, la memoria RAM, la memoria ROM o EPROM, y las interfaces de I/O. Tienen la posibilidad de expansión externa de la memoria.
- Tienen aplicación en sistemas relativamente simples y como apoyo en sistemas de arquitectura estándar.

8.7-PROGRAMACIÓN.

- ❑ PROGRAMA: Conjunto de instrucciones que definen la tarea que tiene que realizar el sistema.
- ❑ REPERTORIO (SET) DE INSTRUCCIONES: Conjunto de instrucciones que es capaz de procesar un sistema μ programable. El número máximo de instrucciones será menor que 2^n (n = número de líneas del DB) esto se debe a que una misma instrucción tiene códigos diferentes según el modo de acceder a los operandos.
- ❑ LENGUAJES: Conjunto de signos y reglas que definen la confección de un programa.

CLASIFICACIÓN DE LOS LENGUAJES DE PROGRAMACIÓN.

LENGUAJES		
<u>MÁQUINA</u>	<u>ENSAMBLADOR</u>	<u>ALTO NIVEL</u>
-Bajo Nivel	-Simbólico	
EJ: 1010 0111	EJ: LDA	EJ: PRINT

LENGUAJE MÁQUINA:

- ❑ El lenguaje máquina se compone únicamente de dos símbolos, el 0 y el 1. Este lenguaje es interpretable directamente por la CPU. Cada CPU posee su propio set de códigos (instrucciones) binarios, aunque en las CPU que pertenecen a una misma familia, es común a todas ellas un set básico de instrucciones.
- ❑ Este lenguaje es el más cercano a la máquina y por lo tanto el más alejado al humano.

Ejemplos:

0100 1110 0111 0000

0100 1110 0111 0101

LENGUAJE ENSAMBLADOR:

- ❑ En este lenguaje, cada código binario tiene asignado un conjunto de letras, generalmente iniciales o juegos de palabras inglesas, que indican de forma escueta, la tarea que realiza cada uno de esos códigos binarios a los que representa. Este conjunto de letras se llama **NEMÓNICO** y es el lenguaje con el cual se trabaja a la hora de programar.
- ❑ El lenguaje ensamblador, no es interpretable de forma directa por la CPU. Para que sea interpretable, debe ser convertido a código máquina, es decir a ceros y unos.
- ❑ Cada CPU posee su propio set de nemónicos.

Ejemplos:

NEMÓNICO:	CÓDIGO BINARIO ASOCIADO:	SIGNIFICADO DEL NEMÓNICO
RESET	(0100 1110 0111 0000)	Reset
NOP	(0100 1110 0111 0001)	No Operación
RTS	(0100 1110 0111 0101)	Return from Subroutine
RTE	(0100 1110 0111 0011)	Return from Exception

LENGUAJE DE ALTO NIVEL:

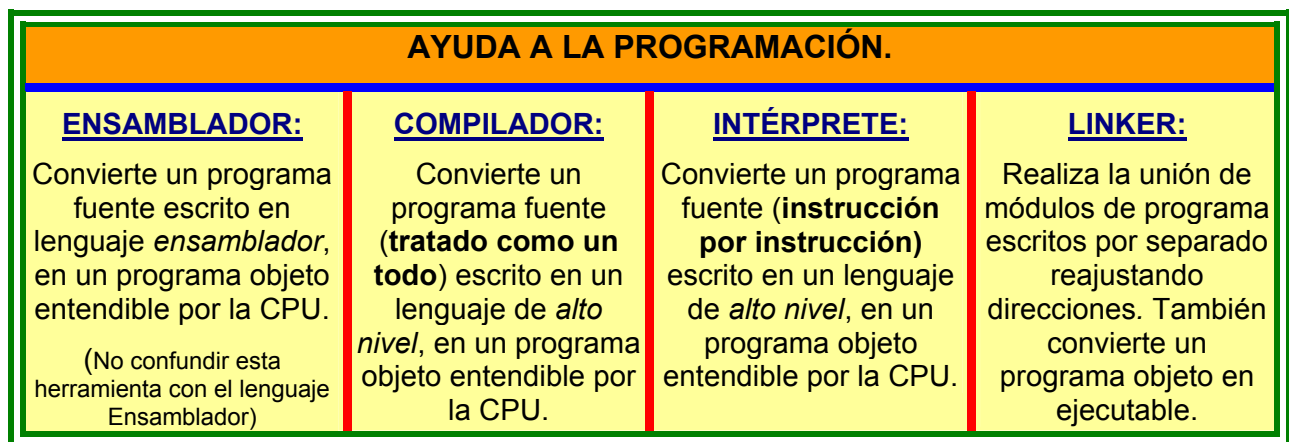
- El lenguaje de alto nivel es el más legible para los usuarios. El set de instrucciones que definen estos lenguajes, no depende de la CPU. Un programa escrito en este lenguaje puede ser ejecutado en CPU-s diferentes.
- Cada una de las instrucciones de un lenguaje de alto nivel se compone da varias instrucciones del lenguaje máquina, con lo cual se pierde el control sobre tiempos de ejecución de las instrucciones y ocupación del programa en la memoria.

Ejemplos:

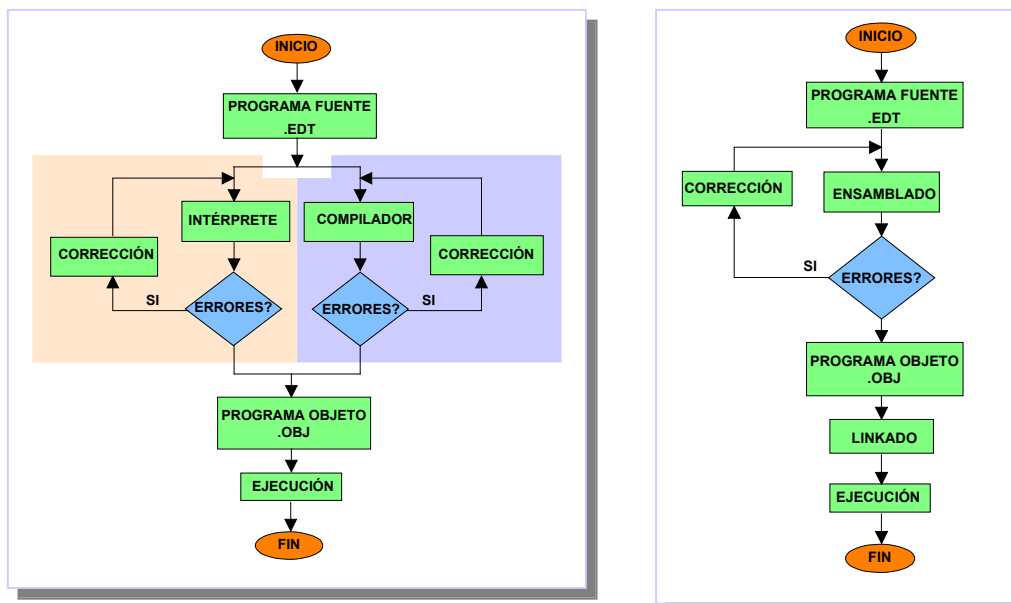
FORTRAN (1954), COBOL (1954), PL/1 (1960), BASIC (1963), ALGOL.(1957), PASCAL (1970), ADA (1980), C (AÑOS `70), LISP.

AYUDAS A LA PROGRAMACIÓN:

- Las ayudas a la programación son herramientas que sirven de puente entre el programador y la forma final del programa de aplicación que debe ser ejecutado. Estas herramientas son a su vez programas.

**PROCESO DE PROGRAMACIÓN.**

Se muestra de forma gráfica el proceso de realización de programas para el caso de programación en lenguaje de alto nivel (izda.) y para el caso de lenguaje ensamblador (dcha.)



PROGRAMACIÓN EN LENGUAJE ENSAMBLADOR.

Los pasos a seguir para realizar la programación en lenguaje ensamblador son los siguientes:

1º *Creación del programa fuente:* Mediante un editor de texto ASCII, se escribe el programa, teniendo en cuenta las normas que rigen para el programa de lenguaje ensamblador con el cual se trabaja. El archivo que se obtiene es un archivo de texto y es el llamado archivo fuente. La extensión de este archivo suele ser .EDT (de editor), .ASM (de ensamblador), etc.

2º *Ensamblado del fichero fuente:* El archivo fuente, debe ser ensamblado mediante un programa ensamblador. Este archivo procedente del ensamblado es llamado *archivo objeto* (con extensión .OBJ), y además de contener el código máquina correspondiente, contiene información adicional para el linker. Para realizar el ensamblado existen diferentes tipos de ensambladores:

- * Autoensamblador, Ensamblador Residente o Ensamblador: Cuando la CPU que realiza la conversión a código máquina, es la misma que aquella donde se va a ejecutar el programa resultante.
- * Ensamblador Cruzado: La CPU que realiza el ensamblado y linkado es diferente a aquella para la cual va destinado el programa.
- * Macroensamblador: Permite el uso de Macros.
- * Ensamblador de una pasada: Solo reconoce etiquetas de la parte del programa ensamblado.
- * Ensamblador de dos pasadas: Primero toma el valor de las etiquetas y después traduce las instrucciones.
- * Ensamblador condicional: Da como salida diferentes códigos objeto dependiendo de una condición.

El ensamblado puede detectar errores de sintaxis (códigos inexistentes, múltiple definición de etiquetas, direccionamiento no permitido).

3º *Linkado:* El linker toma el archivo objeto y lo convierte en archivo ejecutable, es decir, genera el código máquina. También realiza la distribución de direcciones cuando un programa ha sido escrito en diferentes módulos y ensamblado cada uno de ellos por separado.

4º *Ejecución:* La ejecución puede ser realizada en simuladores o en el propio sistema. En caso de que el resultado sea el esperado, el proceso de programación termina, en caso contrario se debe depurar el archivo fuente y repetir todo el proceso.

CREACIÓN DEL PROGRAMA FUENTE.

- ❑ Cuando se escribe un programa en lenguaje Ensamblador hay que seguir ciertas normas para que dicho programa pueda ser ensamblado sin ningún tipo de error de carácter sintáctico. Estas normas dependerán del Ensamblador utilizado y variarán de un fabricante a otro. A continuación se dan unas pautas y reglas que son comunes a la mayoría de los programas ensambladores.

NOMBRES, NÚMEROS Y CARACTERES.

- ❑ Un nombre puede contener cualquier carácter alfanumérico; letras, números y símbolos como &, %... Una restricción usual es que el primer carácter debe ser una letra.

- El número máximo de caracteres de un nombre puede estar limitado a un tamaño. El ensamblador generalmente ignora los caracteres que sobrepasan ese tamaño. En este punto hay que prestar atención cuando se utilizan nombres que comienzan de la misma forma.

Ej. Si el número máximo de caracteres está limitado a 8, **elemento21** y **elemento33** serán tratados como la misma referencia, es decir, como **elemento**.

- Es conveniente utilizar en el programa nombres que sugieran su propósito. Por ejemplo, para identificar una posición de memoria, la cual es el comienzo de una serie de posiciones, es mejor identificarla como DIRINICIO que como MARGARITA.
- Se pueden utilizar diferentes sistemas de numeración durante la programación. Para ello se debe anteponer al número un símbolo que indica el sistema que queremos utilizar. Pej: % para indicar binario, \$ para indicar hexadecimal (%001010111, \$F2A00).

FORMATO DE UNA LÍNEA DE INSTRUCCIÓN TÍPICA.

<u>etiqueta</u>	<u>operación</u>	<u>operando(s)</u>	<u>comentarios</u>
LOOP	Move	#3A, POINTERA	;carga el valor 3A en el registro POINTERA

- Los campos están delimitados por algún carácter, generalmente por uno o varios espacios. Los campos son opcionales.

ETIQUETA.

- Una etiqueta representa la dirección de la posición de memoria principal donde la instrucción etiquetada será cargada. Cuando una etiqueta aparece como operando en una instrucción, el ensamblador la reemplaza con la dirección equivalente.
- Una etiqueta debe ser única.

OPERACIÓN.

- La operación se especifica con los *nemónicos* correspondientes al ensamblador utilizado. Además de especificar una operación, un nemónico puede identificar un registro de la CPU.

OPERANDO(S).

- Este campo contiene un operando inmediato o información (modo de direccionamiento) que especifica donde puede ser encontrado dicho operando. El ensamblador reemplaza cualquier nombre que aparece en este campo, por su equivalente numérico.

COMENTARIO.

- Este campo es ignorado por el ensamblador. Puede contener cualquier carácter, incluyendo espacios. Cuando un comentario ocupa toda la línea, ésta debe comenzar con un carácter “;”. Cuando está al final de una línea de instrucción suele comenzar con “;”.
- Los comentarios facilitan la comprensión del programa.

8.8-DIRECTIVAS

- ❑ Las directivas representan información auxiliar que no forma parte de la operación del programa y que es necesaria para que el programa ensamblador pueda realizar el proceso de traducción (ensamblado). Se puede decir que representan ordenes que utiliza el ensamblador, pero que en ningún caso son ensambladas en el programa objeto.
- ❑ La estructura de las *directivas de ensamblador*, es similar a la usada por las instrucciones máquina.

Algunas directivas comunes son:

EQU (equate).

Asigna el valor numérico en el campo del operando al nombre o símbolo en el campo de la etiqueta.

Ej:

BUFFER	EQU	\$2EF0
SEVEN	EQU	7
NEWBUFF	EQU	BUFFER+\$8

En el ensamblado, cuando aparezca la palabra BUFFER, será sustituida por el valor hexadecimal 2EF0H, la palabra SEVEN por el valor decimal 7 y la palabra NEWBUFF por el valor hexadecimal 2EF8H.

ORG (origin).

Permite especificar al programador en que lugar de la memoria principal, serán guardados el programa o los datos.

Ej:

	ORG	1000	;el ensamblador emplazará
START	LDX	#250	;la instrucción LDX en la
*pos. de mem 1000 y asignará el valor 1000 al nombre START.			

DATA (definir datos).

Sitúa en la memoria datos específicos.

Ej:

	ORG	1000	;el valor 20 se guarda en la posición
VEINTE	DATA	20	;de memoria 1000 y
*en este caso al nombre VEINTE se le asigna el valor 1000.			
LIST	DATA	"ERROR FATAL"	
NUM	DATA	6,15,125,8	;carga en cuatro posiciones de memoria
* los números dados y asigna el nombre NUM a la primera de las posiciones.			

END (fin).

Indica al ensamblador donde termina el programa. Cualquier código después de esta directiva será ignorado.

8.9-CLASIFICACIÓN DE LAS INSTRUCCIONES:

Las instrucciones en el lenguaje ensamblador se pueden clasificar bajo las siguientes denominaciones:

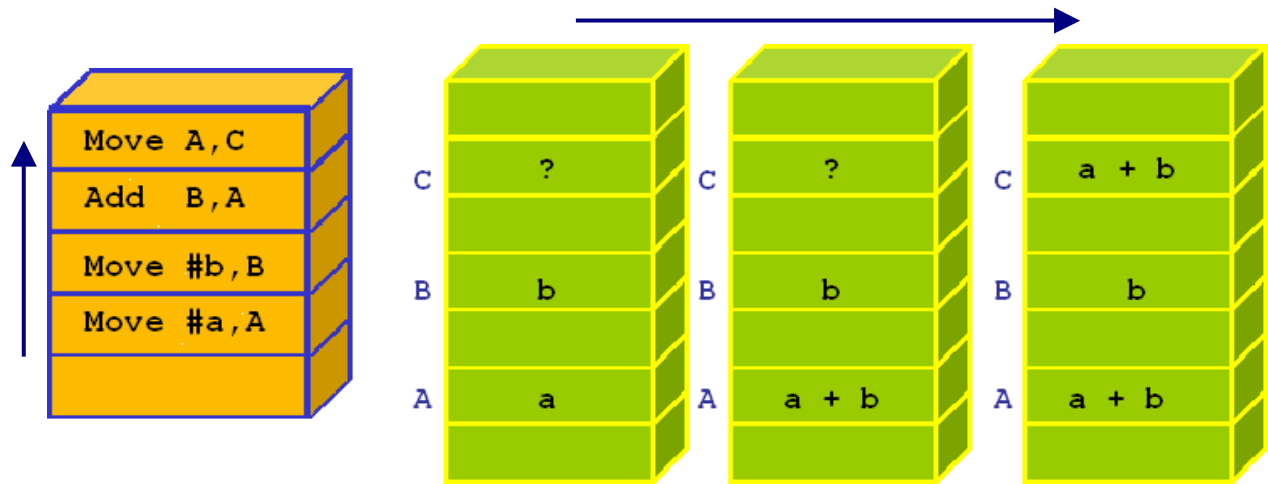
- ❑ **ARITMÉTICAS.**
Realizan operaciones aritméticas (suma y resta).
- ❑ **LÓGICAS.**
Realizan las operaciones AND, OR, NAND, NOR, X-NOR, etc.
- ❑ **DE TRANSFERENCIA.**
Transfieren datos entre los registros internos de la CPU. También en los dos sentidos, entre estos registros y la memoria o módulos I/O.
- ❑ **DE SALTO.**
Hacen que la ejecución secuencial de un programa se rompa mediante un salto a una dirección anterior o posterior a la actual, especificada por la instrucción, y continúe la ejecución a partir de esta nueva dirección.
- ❑ **DE LLAMADA A SUBROUTINA.**
Estas instrucciones hacen que la ejecución del programa salte a una dirección determinada donde se inicia la ejecución de otro programa (subprograma).
- ❑ **DE EMPLEO DE STACK POINTER.**
Mandan datos a la pila (stack) o recuperan datos de la misma.
- ❑ **DE I/O.**
Tratan directamente con módulos de I/O transfiriendo datos e información con estos.
- ❑ **ESPECIALES.**
Rotación, No Operación, Borrado, etc.

8.10-PRINCIPIOS DE PROGRAMACIÓN.

Se debe tener en cuenta que las instrucciones utilizadas en los ejemplos de programas que siguen en todo el resto del tema, sólo tienen carácter didáctico y que realmente no existen tales instrucciones expresadas de esa manera, aunque puede haber en algunos casos coincidencia con nemónicos existentes. Si que existen instrucciones correspondientes a cada μ Procesador que realizan las operaciones que en estos programas se dan.

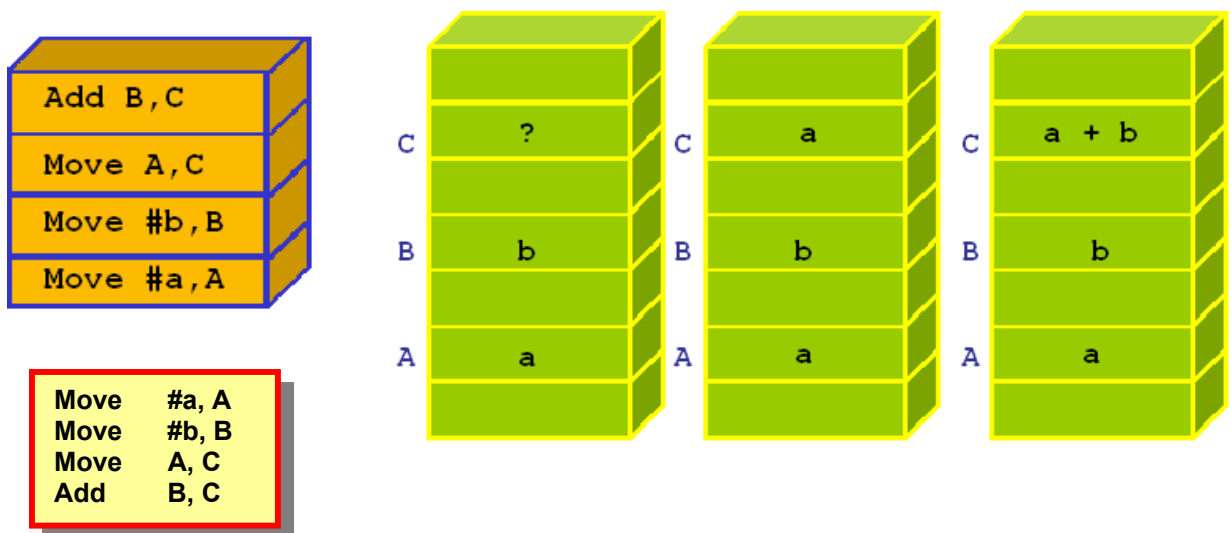
INSTRUCCIONES DE DOS OPERANDOS:

- ❑ Cuando se especifica una localización de donde se pueden obtener los operandos, esta puede ser de la memoria principal, de registros de la CPU o de posiciones direccionables de las unidades de I/O.
- ❑ La figura siguiente muestra la evolución de la memoria implicada en un programa ejemplo del uso de instrucciones con dos operandos. Se trata de la tarea de sumar el valor "a", que guardaremos en la posición de memoria A, con el valor "b", que guardaremos en la posición B, y el resultado "c" lo guardaremos en la posición C.
- ❑ De forma general, los operandos que aparecen siguiendo a una operación, están ordenados de tal manera que el primero que aparece, es el operando fuente, y el segundo, el operando destino donde se guardará, en su caso, el resultado de la operación.



Move	#a, A
Move	#b, B
Add	B, A
Move	A, C

- En las dos primeras instrucciones, el operando fuente que aparece con la almohadilla (#), es un valor concreto, que está situado en una posición de memoria de programa (memoria ROM) *inmediatamente* después de donde está guardado el código de la instrucción Move. Por lo tanto, este valor está guardado en la memoria de programa, y no tiene que ser buscado en ningún otro lugar. Los operandos que aparecen como destinos, (A y B) hacen referencia a posiciones de memoria donde debe ser guardado el valor con almohadilla.
- En la instrucción Add, tanto la fuente (B) como el destino (A), hacen referencia a posiciones de memoria, entendiéndose la instrucción como la suma del contenido de la posición de memoria B con el contenido de la posición de memoria A y el resultado de la suma guardarlo en esta última posición. La última instrucción indica una copia del contenido de la posición de memoria A en la posición de memoria C.
- Se observa que con este programa, el contenido original de la posición de memoria A, ha sido eliminado.
- Esta figura representa el mismo ejemplo que la anterior pero con instrucciones que salvan el contenido de las posiciones de memoria A y B.



Move	#a, A
Move	#b, B
Move	A, C
Add	B, C

Realicemos otro programa cuya tarea es sumar el valor "a" guardado en la posición A, con el valor "b" guardado en la posición B y al resultado, multiplicarlo por dos y sumarlo el valor "c" guardado en la posición. C. El resultado de toda la operación se guardará en C.

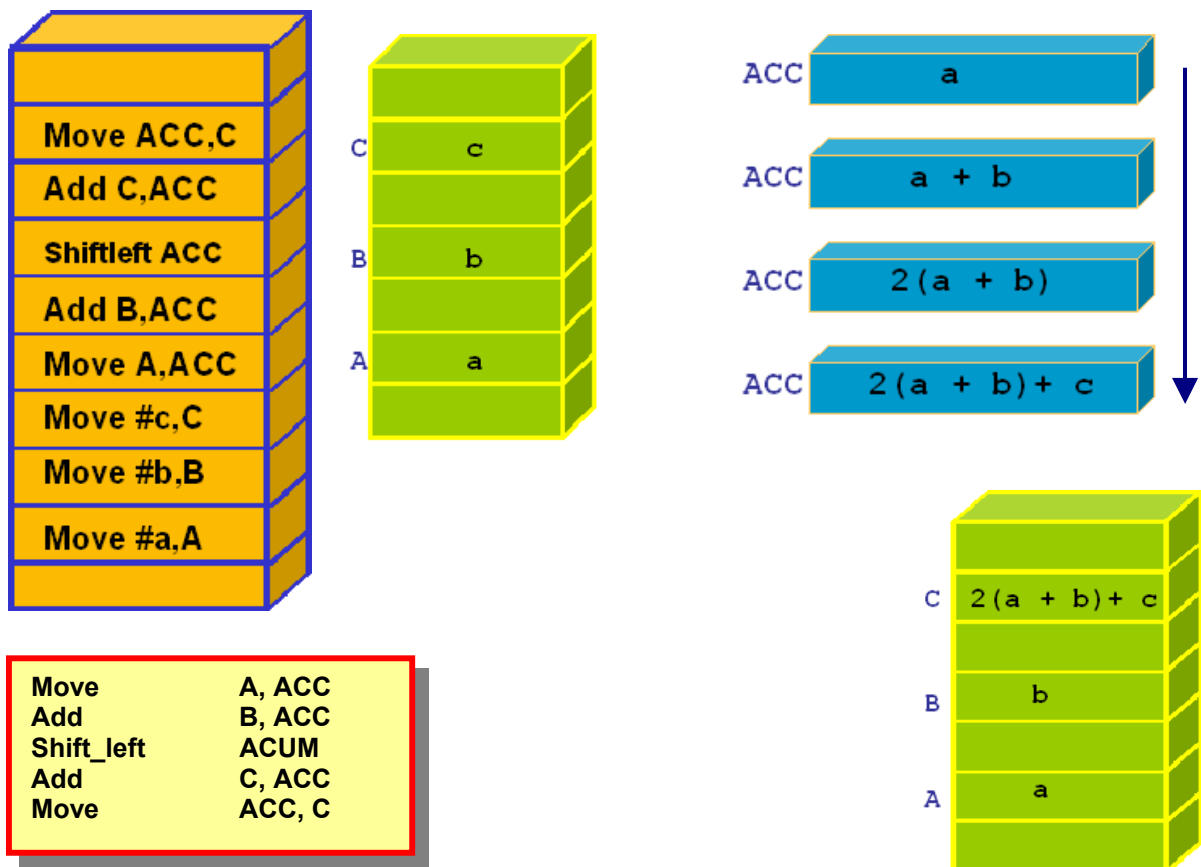
La operación que se quiere realizar se denota de la siguiente manera:

$$C \leftarrow 2([A] + [B]) + [C]$$

Hay que observar que la forma de representar la operación es diferente en este caso, del caso de los operandos en un programa. Aquí, el contenido de una posición de memoria se representa encerrando el nombre, que hace referencia a la posición de memoria, entre corchetes.

La figura representa la evolución del acumulador y de las posiciones de memoria implicadas. Se puede observar que en el programa existe una instrucción de un solo operando, el acumulador.

El Acumulador es un registro particular que poseen muchas CPU-s. Se accede a él con instrucciones propias y de forma mucho más rápida que a una posición de la memoria principal. Esto último se va a cumplir siempre para cualquier registro que esté situado dentro de la CPU ya que esta, no tiene que poner una dirección en el AB, activar la línea de control correspondiente, etc. para acceder a una posición de memoria.



INSTRUCCIONES DE UN OPERANDO:

Por ejemplo la tarea de escribir la palabra 00H en la posición A puede ser realizada con dos operandos o con uno. Lógicamente en este último caso se ganará en velocidad y en espacio ocupado en la memoria.

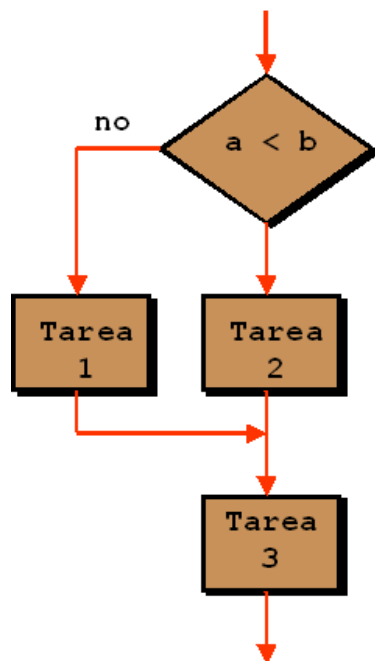
Subtract A, A. 2 OPERANDOS (resta el contenido de A de ese mismo contenido).

Move #00, A 2 OPERANDOS (pone el valor 0 en la posición de memoria A).

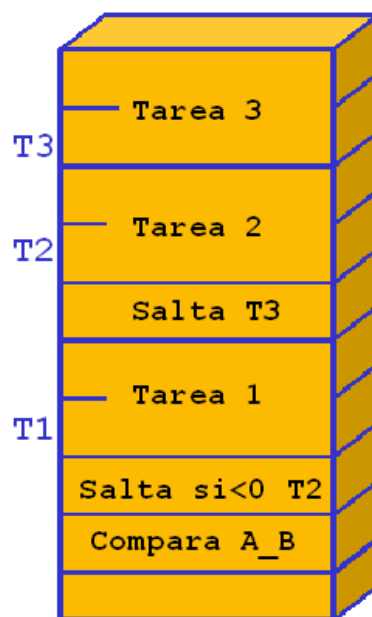
And	#00, A	2 OPERANDOS (realiza la operación lógica AND entre el contenido de A y el valor 0).
Clear	A	1 OPERANDO (de forma automática la CPU pone a cero la posición A)

INSTRUCCIONES DE CONTROL DE SECUENCIA.

- Controlan la secuencia de ejecución de un programa, permitiendo al programador especificar el orden en el cual las instrucciones que forman un programa serán ejecutadas.



En el ejemplo, el control de la secuencia se hace mediante el cumplimiento o no de una condición, ¿a<b?. Si se cumple, se realiza la TAREA2, si no se cumple, se realiza la TAREA1. En cualquier caso, se realizará después la TAREA3.



- La realización de una u otra tarea, se implementa mediante instrucciones de salto a una u otra parte de la memoria donde se guarda el programa de la tarea a realizar.

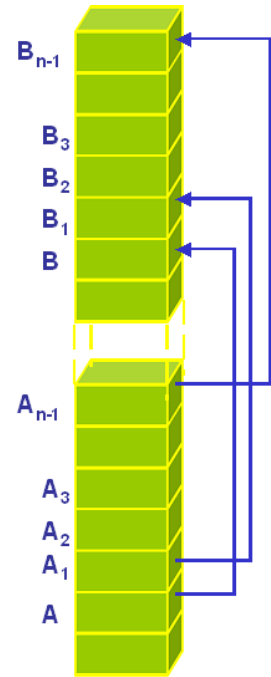
Normalmente, la comparación entre dos operandos se realiza restando dichos operandos (A-B) y consultando las banderas correspondientes para saber si el resultado de dicha resta ha sido positivo, negativo o cero.

INSTRUCCIONES DE BUCLE.

- Consideremos el caso en el que sea necesario copiar el contenido de n posiciones de memoria consecutivas ($A, A_1, A_2 \dots A_{n-1}$) en otra serie de n posiciones de memoria también consecutivas ($B, B_1, B_2 \dots B_{n-1}$).

Esta tarea se puede realizar de diferentes modos. Uno de ellos es mediante las instrucciones de copiar uno por uno el contenido de las posiciones A en las posiciones B . Está claro que solamente para el proceso de la copia, se necesitan tantas instrucciones *Move*, como lugares de memoria a procesar.

```
Move  A,  B
Move  A1, B1
Move  A2, B2
Move  A3, B3
...
Move  An-2, Bn-2
Move  An-1, Bn-1
```



- Otro método más práctico consiste en hacer uso de pocas instrucciones, pero haciendo que estas se ejecuten un número determinado de veces. Para ello, se introducen las instrucciones en un bucle cerrado de ejecución y se controla este bucle, con un contador y con una condición de salida del mismo.

```
Move      #n, N
Clear     NCOUNT
LOOP Move  A(NCOUNT), B(NCOUNT)
Increment NCOUNT
Compare   NCOUNT, N
Branch_if<0 LOOP
Halt
```

La forma práctica de realizar la tarea es la mostrada en el cuadro.

La traza o comportamiento del programa es el siguiente:

Move #n, N: Se guarda en una posición de memoria o registro de la CPU (N) el valor de la cantidad de posiciones de memoria a procesar, en este caso el valor es n .

Clear NCOUNT. Se inicializa (poniendo a cero su valor) un registro $NCOUNT$, que va a funcionar como contador de las veces que se ejecuta el bucle

Move A(NCOUNT), B(NCOUNT). La dirección efectiva del operando fuente, se obtiene sumando la dirección A con el contenido de $NCOUNT$. La dirección efectiva del operando destino, se obtiene sumando la dirección B con el contenido de $NCOUNT$. Con la ejecución de esta instrucción, se copia el valor de la posición de memoria A ($A+0$) en la posición de memoria B ($B+0$).

Increment NCOUNT. En la cuarta instrucción se incrementa el valor de $NCOUNT$ pasando a tener el valor 1, indicando con ello que el proceso de copia se ha ejecutado ya una vez.

Compare NCOUNT, N. Se realiza un chequeo para comprobar si se han procesado ya todas las posiciones de memoria, es decir, las n posiciones. Generalmente en este tipo de instrucciones, la CPU realiza una resta entre los operandos. Puede ser resta de fuente menos destino o al revés. En este caso, la resta que realiza de forma interna es fuente menos destino, es decir, el contenido de $NCOUNT$ menos el contenido de N .

Branch_if <0 LOOP. Si el resultado de la resta anterior es negativo (contenido de N mayor que contenido de NCOUNT), quiere decir que no se han procesado las n posiciones de memoria y por lo tanto la instrucción indica que la ejecución del programa debe continuar en la posición de memoria marcada con la etiqueta LOOP, es decir, en la línea Move A(NCOUNT), B(NCOUNT). En esta segunda pasada por esta línea, copiará el contenido de la posición A1 (A + 1) en la posición B1 (B + 1) ya que el contenido de NCOUNT se había incrementado. El programa continua a partir de aquí realizando los pasos como en el caso anterior hasta que no se cumpla la condición en la comparación, es decir, contenido de NCOUNT igual al contenido de N, en cuyo caso la ejecución del programa se detendrá (**Halt**).

En este cuadro se da otra posible solución para realizar la misma tarea con un ahorro en el número de instrucciones utilizadas. El funcionamiento es análogo al explicado para el caso anterior.

```

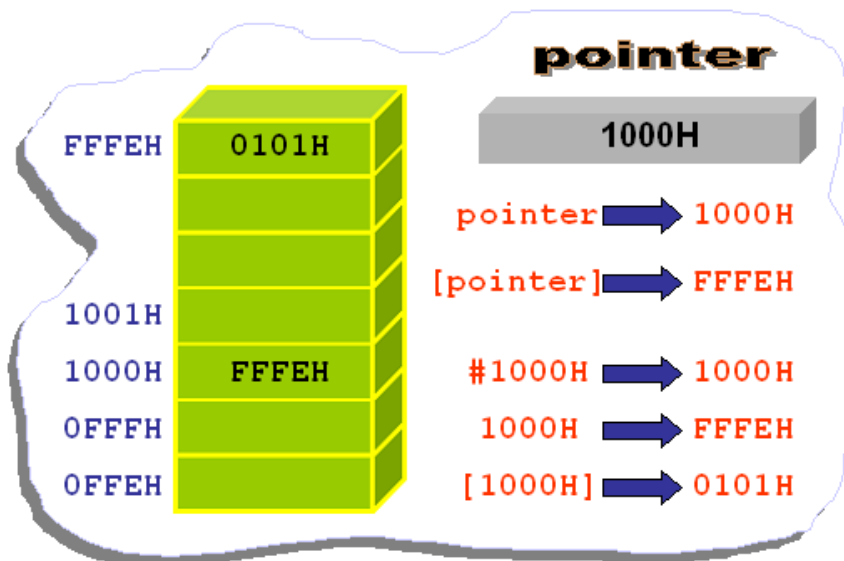
Move      #n, NCOUNT
Decrement NCOUNT
LOOP Move  A(NCOUNT), B(NCOUNT)
Decrement NCOUNT
Branch_if ≥0 LOOP
Halt

```

Otra solución para la realización de la tarea viene dada por la utilización en el programa de **PUNTEROS**. Un puntero es generalmente un registro de la CPU cuyo contenido es una dirección de memoria, se dice por lo tanto, que el registro apunta a una posición de memoria.

Para ver el modo de utilización de un puntero consideremos por ejemplo que a dicho puntero lo llamamos **POINTER**.

- Entonces una referencia a **POINTER**, se refiere al contenido de este registro, es decir al valor 1000H.
- Una referencia al registro, acotada, por ejemplo mediante corchetes **[POINTER]** hace referencia al contenido de la posición de memoria cuya dirección es contenido del registro POINTER, es decir, al valor FFFE.



Si se toma la posición de memoria 1000H como pointer tenemos que 1000H hace referencia al contenido de esta dirección, es decir al valor FFFE. [1000H] hace referencia a 0101H y #1000H hace referencia al propio valor que sigue a la almohadilla, es decir a 1000H.

Teniendo en cuenta lo anterior, el programa que se da en el cuadro adjunto no debería presentar en principio, ningún problema a la hora de realizar su traza.

```

Move      #n, NCOUNT
Move      #ADRESSA, POINTERA
Move      #ADRESSB, POINTERB
LOOP Move  [POINTERA], [POINTERB]
Increment POINTERA
Increment POINTERB
Decrement NCOUNT
Branch_if >0 LOOP
Halt

```

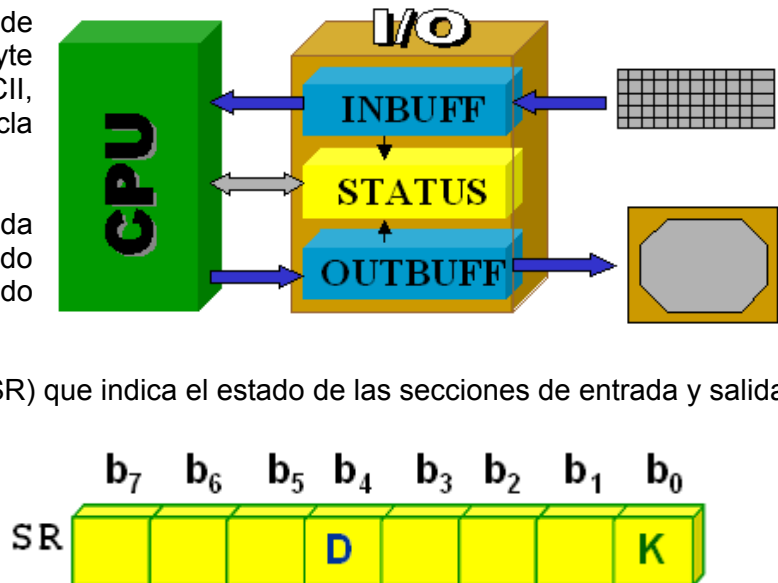
8.11-TAREAS DE ENTRADA / SALIDA. (I/O)

Consideremos que existen tres registros (INBUFF, OUTBUFF y STATUS) en el interfaz de I/O, el cual une un terminal con el BUS de la computadora.

- ❑ **INBUFF:** Es un registro de entrada que recoge el byte codificado en ASCII, correspondiente a la tecla pulsada en el teclado.

- ❑ **OUTBUFF:** Registro de salida que acepta el carácter codificado en ASCII que va a ser mandado al monitor.

- ❑ **STATUS:** Registro de estado (SR) que indica el estado de las secciones de entrada y salida del terminal:



b0 Indica el estado del teclado (KEYBOARD STATUS): 1 = buffer lleno. 0 = buffer vacío

b4 Indica el estado del monitor (DISPLAY STATUS): 1 = buffer listo. 0 = buffer ocupado.

- ❑ Supongamos que es el propio hardware el que se encarga de modificar el registro de estado de tal manera que cuando una tecla es pulsada, el b0 del registro de estado se pone a 1, y cuando el registro INBUFF es leído por el μ procesador, el b0 del registro de estado se pone 0. En este último caso, si no se realiza de forma automática, debería ser el propio programa (programador) el encargado de acceder al registro de estado para poner este b0 a 0 cuando hubiera realizado la lectura del INBUFF.

Para el caso del monitor, cada vez que este lee el dato que tiene en OUTBUFF, pone de forma automática en el registro de estado, el b4 a 1, indicando que ya está listo para leer otro dato. Mientras el dato en OUTBUF no haya sido leído, el b4 del registro de estado permanece en 0.

- ❑ Asociemos los nombres INBUFF, OUTBUFF y STATUS con las direcciones del mapa de memoria de los registros de I/O del interface. Usando el esquema de I/O controlado por programa (*PROGRAM-CONTROLLED I/O SCHEME*) las tareas de I/O pueden realizarse mediante los siguientes programas:

Lectura de un carácter del teclado.

La operación lógica AND del valor #1 (00000001) con el contenido del acumulador, hace que

```

WAIT Move      STATUS, ACCUM
  AND          #1, ACCUM
  Branch_if_0  WAIT
  Move         INBUFF, ACCUM
  HALT

```

se determine el valor del b0 del registro de estado, ya que si el resultado de la operación es 0, quiere decir que el b0 tiene de valor 0, en caso contrario el resultado de la operación AND daría como resultado un valor distinto de 0 (#1).

Muestra de un mensaje de una línea

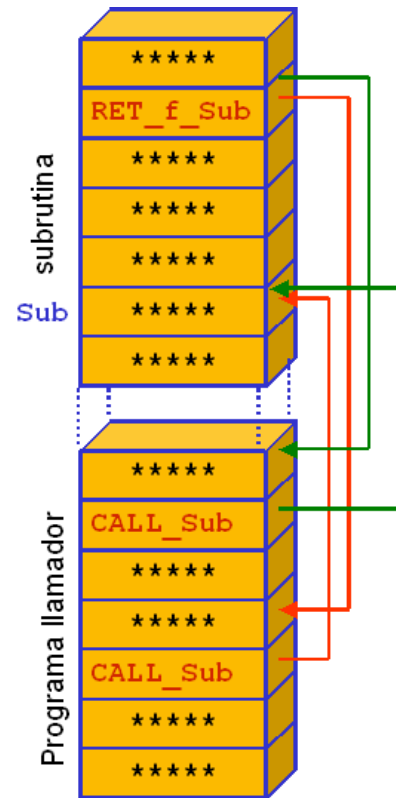
Para el chequeo del b1, del registro de estado se realiza la operación lógica AND con el valor #16 (00010000).

El programa enviará caracteres a la pantalla hasta que se encuentre con el carácter de control #CR (Carriage Return), que indica el final del mensaje. Lógicamente, en el comienzo del programa el programador habrá asociado mediante la directiva EQU el código ASCII correspondiente al retorno de carro (0001101), a la expresión CR.

```
      Move      #MENSAJE, POINTER
WAIT Move      STATUS, ACCUM
      AND       #16, ACCUM
      Branch_if_0 WAIT
      Move      [POINTER], OUTBUFF
      Compare   #CR, [POINTER]AUTOINC
      Branch_if_n0 WAIT
      HALT
```

8.12-SUBROUTINAS.

- ❑ Se puede definir la subrutina como el segmento de programa que se ejecuta en diferentes partes de otro programa al cual vamos a denominar *programa llamador*.
- ❑ La subrutina se va a ejecutar cuando se solicite dicha ejecución en el programa llamador. La solicitud, se realiza mediante instrucciones especiales clasificadas dentro del set, como grupo de *instrucciones de llamada a subrutina*.
- ❑ La vuelta desde la ejecución de una subrutina, a la ejecución del programa llamador se realiza mediante una instrucción especial del grupo de *instrucciones de retorno de subrutina*.
- ❑ Cuando se termina de ejecutar la subrutina, la ejecución debe continuar en el programa que hizo la llamada a esta. Como dicha subrutina puede ser llamada desde diferentes puntos del programa llamador, la dirección a la cual debe volver, no es única.
- ❑ Para poder volver de la ejecución de una subrutina a la ejecución del programa llamador, es necesario salvar la dirección a donde se debe volver, antes de que la subrutina sea ejecutada. La dirección que debe ser guardada, corresponde con la de la instrucción que sigue inmediatamente a la instrucción de llamada a la subrutina en el programa llamador. El modo de realizar esto es salvar el contenido del PC cuando comienza a procesarse la instrucción de salto, ya que el PC apunta a la dirección que sigue a aquella cuyo contenido está en proceso.
- ❑ Una vez salvada la dirección de vuelta de la subrutina, en el PC se carga la dirección de la primera instrucción de la subrutina que ha sido llamada para ser ejecutada.



En la figura se representa la ejecución de un programa (programa llamador), durante la cual, la subrutina SUB es llamada desde dos puntos diferentes. Como puede observarse la dirección de vuelta de la subrutina es diferente según el lugar del programa llamador desde donde se encuentra la instrucción de salto a la subrutina.

- ❑ El programa llamador puede “dar” datos o información de donde encontrarlos, a la subrutina para que ésta trabaje con ellos. Estos datos que se pasan a una subrutina se llaman *parámetros*.

Por ejemplo, volviendo al caso de copiar el contenido de n posiciones de memoria consecutivas en otra serie de n posiciones de memoria también consecutivas, el programa resultante haciendo uso de una subrutina tendría la forma mostrada en el cuadro.

	Move	#n, NCOUNT
	Move	#ADDRESSA, POINTERA
	Move	#ADDRESSB, POINTERB
	Call	SUBM
	next instruction	
	.	
SUBM	Move	[POINTERA], [POINTERB]
	Increment	POINTERA
	Increment	POINTERB
	Decrement	NCOUNT
	Branch_if >0	SUBM
	Ret_from_subr	

- El programa llamador pasa tres parámetros a la subrutina, estos son: el tamaño, es decir, la cantidad de posiciones a procesar (#n), la dirección de inicio de la zona de memoria a copiar (#ADDRESSA) y la dirección de inicio de la zona de memoria donde debe guardarse la información copiada (#ADDRESSB).

El siguiente ejemplo muestra el uso de una subrutina (MAX), para buscar el número más alto

	Move	#LISTADDRESS, POINTERA
	Move	SIZE, POINTERB
	Add	POINTERA, POINTERB
	Call	MAX
	next instruction	
		
MAX	Move	POINTERA, MAXLOC
	Move	[POINTERA], ACCUM
LOOP	Compare	ACCUM, [POINTERA]
	Branch_if $\geq$ 0	NEXT
	Move	POINTERA, MAXLOC
	Move	[POINTERA], ACCUM
NEXT	Increment	POINTERA
	Compare	POINTERA, POINTERB
	Branch_if<0	LOOP
	Move	MAXLOC, POINTERA
	Ret_from_subr	

en una lista, entendiendo como lista, una colección de posiciones de memoria con un contenido organizado o relacionado de cierto modo.

Los parámetros que se pasan a la subrutina son la dirección de comienzo de la lista (#LISTADDRESS), y la dirección de final de la misma, la cual está contenida en POINTERB. POINTERA y POINTERB son

registros de la CPU. La subrutina va a devolver al programa llamador, el valor mas alto encontrado en la lista. Este valor estará almacenado en el ACCUM. También va a devolver la dirección donde se encuentra dicho valor. Esta dirección estará almacenada en POINTERA.

	Move	#LISTADDRESS, POINTERA
	Move	SIZE, POINTERB
	Add	POINTERA, POINTERB
	Call	SORT
	next instruction	
		
SORT	Move	POINTERA, STARTLOC
SLOOP	Call	MAX
	Move	[POINTERA], TEMP
	Move	[STARTLOC], [POINTERA]
	Move	TEMP, [STARTLOC]
	Increment	STARTLOC
	Move	STARTLOC, POINTERA
	Compare	POINTERA, POINTERB
	Branch_if<0	SLOOP
	Ret_from_subr	
MAX	Move	POINTERA, MAXLOC
	Move	[POINTERA], ACCUM
LOOP	Compare	ACCUM, [POINTERA]
	Branch_if $\geq$ 0	NEXT
	Move	POINTERA, MAXLOC
	Move	[POINTERA], ACCUM
NEXT	Increment	POINTERA
	Compare	POINTERA, POINTERB
	Branch_if<0	LOOP
	Move	MAXLOC, POINTERA
	Ret_from_subr	

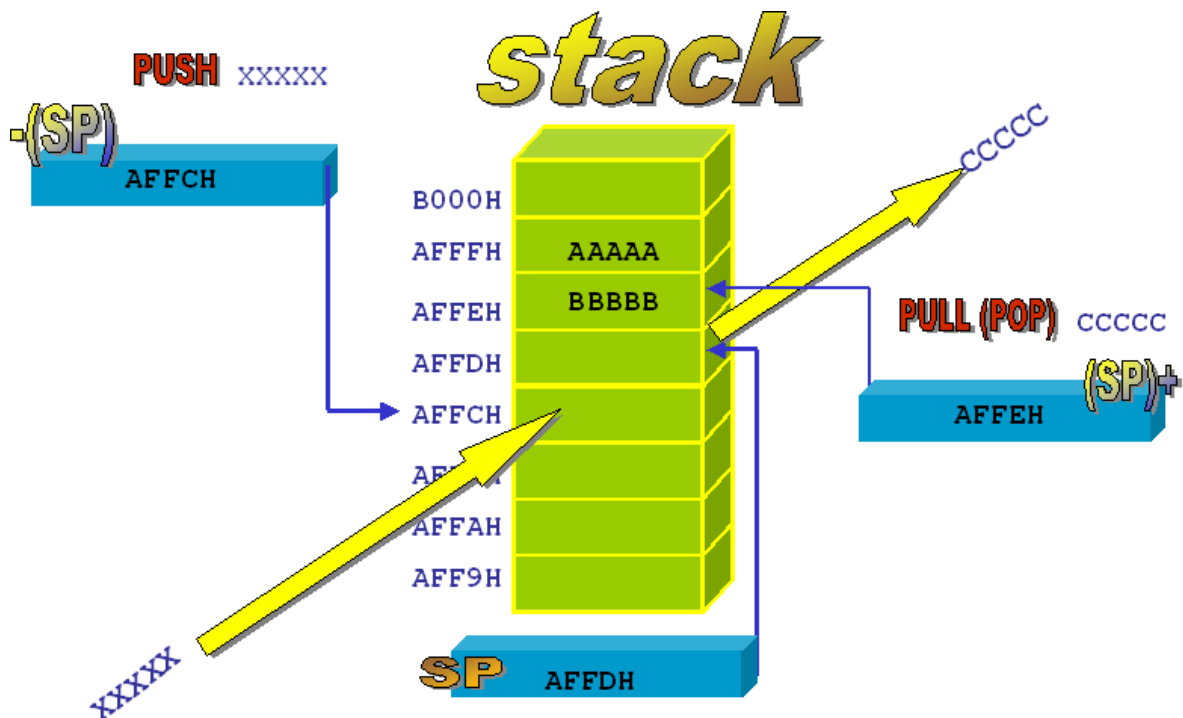
Este ejemplo muestra la ordenación de una lista de números en orden descendente. Observar el uso de subrutinas anidadas, es decir, subrutinas que son llamadas desde otras subrutinas. Como se puede apreciar, la subrutina SORT, es llamada desde el programa que lleva el mayor peso en la realización de una tarea determinada, y la subrutina MAX, es llamada desde la subrutina SORT.

Cuando la subrutina MAX termina su ejecución, la dirección que debe cargarse en el PC es la que corresponde a la instrucción Move [POINTERA], TEMP en la subrutina SORT. Cuando la subrutina SORT termina su ejecución la dirección que se cargará en el PC será la correspondiente a next

instruction en el programa "principal"

8.13-STACK (PILA)

- ❑ El stack es una zona de la memoria principal, que es direccionada de forma especial e independiente del contador de programa (PC).
- ❑ Los datos que son mandados al *stack*, se guardan en posiciones consecutivas (lista simple). Las nuevas entradas, es decir, los nuevos datos que se envían al stack, sólo pueden ser guardados al final de la lista.
- ❑ El modo de trabajo de esta zona de la memoria es **LIFO** (*Last Input-First Output*), por lo tanto, el último valor añadido a la pila es el primero que será extraído (leído).
- ❑ La dirección donde el valor que se manda a la pila va a ser guardado, o la dirección de la pila donde se va a leer un valor, es siempre conocida gracias al **STACK POINTER** (SP). El SP es un registro (contador) de la CPU cuyo contenido es la dirección del último dato de la pila.
- ❑ El SP puede apuntar a las posiciones de la pila, de forma ascendente o de forma descendente, mientras se van “introduciendo” datos a la pila. La mayor parte de los μP lo hacen de forma descendente, es decir, cada vez que se envía un valor a la pila, el SP es decrementado y el dato escrito, y cuando se “sacan” datos de la pila, el dato es leído y el SP es incrementado.
- ❑ La operación de añadir elementos a la pila se llama **PUSH** y la de extraerlos se llama **PULL** o **POP**. Muchos μP poseen instrucciones específicas de trabajo con la pila donde se incluyen estas ordenes.

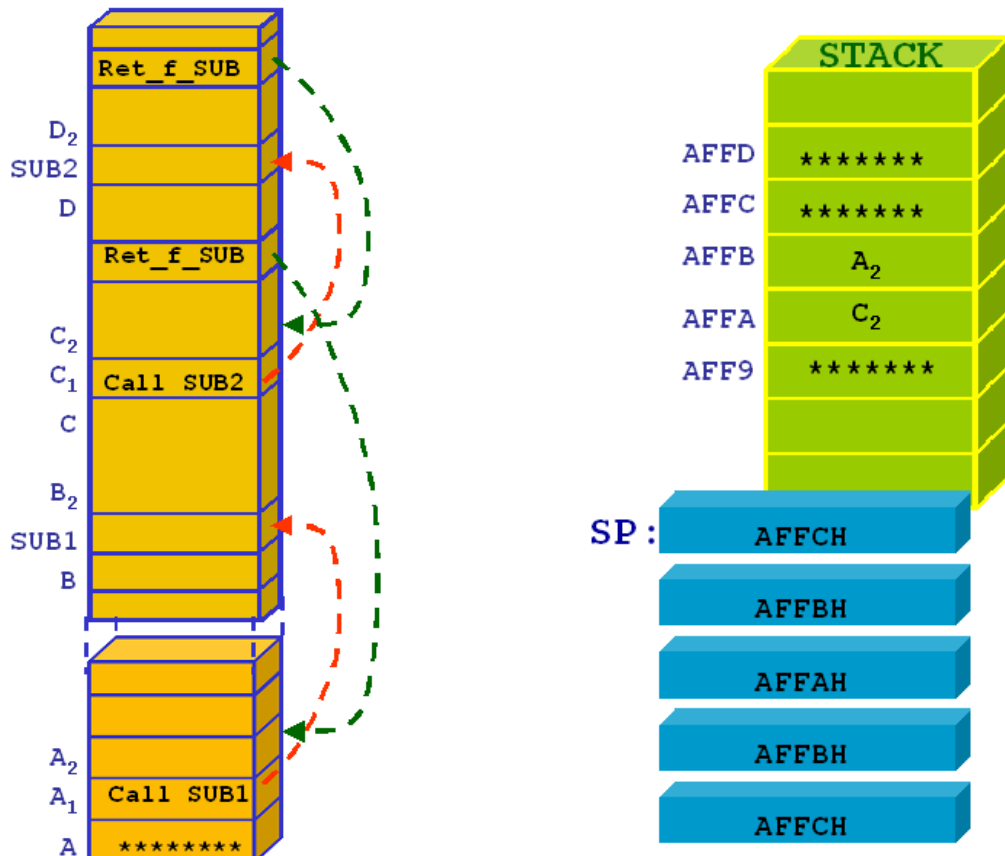


En el ejemplo de la figura superior, se define el comienzo de la pila en la dirección AFFFH. La acción de añadir un elemento a la pila, hace que el SP se decremente. Después de haber mandado tres elementos en la pila, el stack pointer, apunta a la dirección AFFDH. Si se mandara un elemento mas a la pila (Push XXXXX), el SP tomaría el valor AFFCH y emplazaría el nuevo elemento XXXXX, en la nueva dirección apuntada. Si la operación fuera la de extracción de un elemento de la pila (Pull CCCCC), se obtendría el elemento de la dirección apuntada AFFDH, y posteriormente se incrementa el SP al valor AFFEH.

USO DE LA PILA PARA LA UNIÓN DE SUBROUTINAS

- Cuando se llama a una subrutina (instrucción *call* ejecutándose), el PC contiene la dirección de la siguiente instrucción a ejecutar en el programa que hace la llamada, es decir, la dirección de retorno, a donde debe volver el PC cuando finalice la ejecución de la subrutina. La ejecución de la instrucción tipo *call* consiste en guardar el contenido del PC en el *stack* (**PUSH**) y cargar la dirección de la primera instrucción de la subrutina en el PC.

Cuando la ejecución de la subrutina ha sido completada, se realiza la vuelta al programa llamador mediante una instrucción de *Return_From_Subroutine*. Esta instrucción extrae (**PULL**) el valor contenido en la dirección de la pila donde está apuntando el SP y lo carga en el PC.



En la figura superior el programa llamador llama a la ejecución de la SUB1 (*Call SUB1*). Inmediatamente el SP decrementa su contenido (*AFFBH*) para apuntar a la dirección de la pila donde se guarda la dirección de vuelta (A₂) de la SUB1, es decir, donde tiene que continuar el programa llamador una vez se halla finalizado la ejecución de la SUB1. El programa comienza a ejecutar la subrutina 1 cuyo principio está en la dirección SUB1. Dentro de esta subrutina, existe otra llamada a otra subrutina (*Call SUB2*), el SP decrementa su valor (*AFFAH*), se guarda en la pila la dirección de vuelta de la subrutina 2 (C₂) y se carga en el PC (**Program Counter**) la dirección de inicio de la nueva subrutina (SUB2). Cuando se termina la ejecución de esta subrutina (*Ret_f_SUB*) en el PC se carga la dirección de vuelta (C₂), el SP incrementa su valor (*AFFBH*) y se termina de ejecutar lo que faltaba de la subrutina 1. Cuando ésta finaliza (*Ret_f_SUB*), en el PC se carga la dirección de vuelta de la misma (A₂), el SP incrementa su valor (*AFFCH*) y se continua con el programa llamador.

El trabajo que realiza la pila en cuanto a guardar las direcciones de vuelta de las posibles subrutinas que vayan apareciendo en la ejecución de un programa, y la devolución de estas direcciones al contador de programa, es totalmente automático.

8.14-PASO DE PARÁMETROS A UNA SUBROUTINA.

- Los valores que un programa llamador da a una subrutina para que ésta trabaje con ellos, es decir, los parámetros, pueden ser pasados (dados al programa subrutina) mediante el uso de registros o posiciones fijas de memoria, también poniéndolos inmediatamente después de la instrucción de llamada a la subrutina, y por último a través de la pila.

Ej.

POSICIONES DE MEMORIA.

	Move	#n, NCOUNT
	Move	#ADDRESSA, POINTERA
	Move	#ADDRESSB, POINTERB
	Call	SUBM
	next instruction	
SUBM	.	
	Move	[POINTERA], [POINTERB]
	Increment	POINTERA
	Increment	POINTERB
	Decrement	NCOUNT
	Branch_if >0	SUBM
	Ret_from_Subr	

En este programa que trata el ejemplo que se viene analizando, el número de entradas o posiciones de memoria a tratar, se pasa a través de la posición de memoria NCOUNT, y las direcciones de comienzo de la lista A (#ADDRESSA) y de la lista B (#ADDRESSB), se pasan mediante unos registros de la CPU que actúan como punteros.

DESPUÉS DE LA INSTRUCCIÓN DE LLAMADA.

	Call	SBMOV
	#n	
	ADDRESSA	
	ADDRESSB	
	next instruction	
SBMOV		
	Move	[SP], POINTERC
	Move	[POINTERC], NCOUNT
	Increment	POINTERC
	Move	[POINTERC], POINTERA
	Increment	POINTERC
	Move	[POINTERC], POINTERB
	Increment	POINTERC
LOOP	Move	POINTERC, [SP]
	Move	[POINTERA], [POINTERB]
	Increment	POINTERA
	Increment	POINTERB
	Decrement	NCOUNT
	Branch_if >0	LOOP
	Ret_from_subr	

Para este caso, cuando se ejecuta la instrucción de llamada a la subrutina, en la pila se va a guardar el valor del PC, el cual estará apuntando a la posición de memoria que contiene el dato #n (posiciones de memoria a tratar).

En el PC se carga la dirección correspondiente a SBMOV, y comienza el proceso de ejecución de la subrutina. La primera instrucción *Move [SP], POINTERC*, lleva el contenido de la posición de la pila donde está apuntando, el SP, a un registro llamado POINTERC.

En ese momento, dicho contenido, será el valor del PC, guardado cuando se hizo el salto a la ejecución de la subrutina. En la siguiente instrucción, *Move [POINTERC], NCOUNT* el valor #n, es llevado a la posición de memoria NCOUNT. En la siguiente instrucción, *Increment POINTERC*, el valor que se está incrementando es el del PC, con lo cual ahora éste, apuntará a la siguiente posición de memoria, es decir la que contiene el valor ADDRESSA. Este valor es llevado a POINTERA mediante la siguiente instrucción. El procedimiento, es análogo para llevar ADDRESSB a POINTERB.

Por último, se observa que el contenido de POINTERC, se vuelve a incrementar. Esto es para que el PC apunte a la dirección de memoria de programa que contiene next instrucción. Este valor de POINTERC, es el que se guarda en la pila, en la dirección apuntada por SP, que será el que se tome para restaurar el PC a la vuelta de la subrutina, cuando se ejecute la instrucción Ret_from_subr.

A TRAVÉS DEL STACK.

		
	Decrement	SP
	Move	#n, [SP]
	Decrement	SP
	Move	#ADDRESSA, [SP]
	Decrement	SP
	Move	#ADDRESSB, [SP]
	Call	SBMOV
	Add	#3, SP
	next instruction	
SBMOV		
	Move	3(SP), NCOUNT
	Move	2(SP), POINTERA
	Move	1(SP), POINTERB
LOOP	Move	[POINTERA], [POINTERB]
	Increment	POINTERA
	Increment	POINTERB
	Decrement	NCOUNT
	Branch_if>0	LOOP
	Return_from_subroutine	

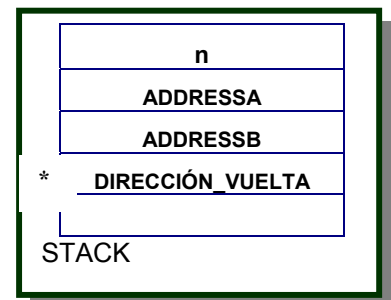
En este último caso, el programa llamador pone los parámetros que van a ser pasados, en la pila. Con este procedimiento el paso de los parámetros se hace independiente de la zona de memoria de datos, y no hay que acceder a direcciones concretas para buscarlos, con lo cual flexibiliza la inclusión de subrutinas en los programas.

La forma de trabajo es la siguiente: Se decrementa el SP para apuntar a una posición vacía (o cuyo contenido es irrelevante) de la pila, para llevar a esa

posición el valor de #n. Se vuelve a decrementar el SP, para llevar a la pila el valor de #ADDRESSA, y se decrementa de nuevo para llevar ahora el valor #ADDRESSB.

En el comienzo de la ejecución de la subrutina, se lleva el valor de #n a NCOUNT. Para conseguir el valor de la posición donde está #n, vemos que tenemos que sumar 3 al contenido del SP (El valor del SP no es alterado). Esto es así, porque cuando se llamó a la subrutina, el SP se volvió a decrementar para llevar a la pila el valor de la dirección de vuelta contenida en el PC.

El * indica donde apunta el SP cuando la subrutina SBMOV es llamada

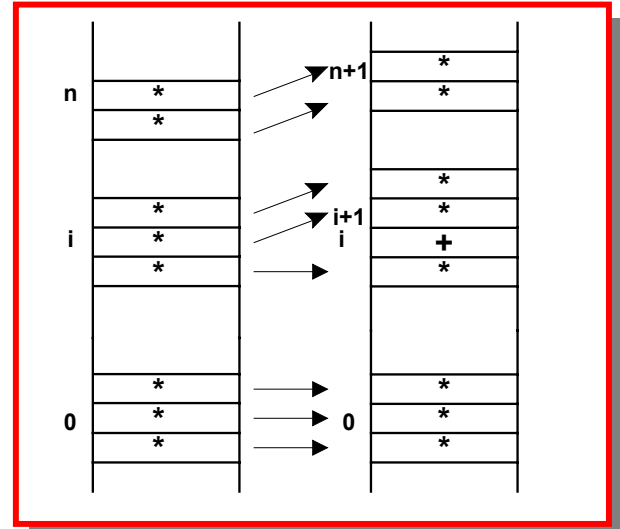


Cuando se vuelve de la subrutina lo primero que se hace es sumar 3 al SP, con esto la pila volverá a estar en el estado en que estaba cuando se comenzó a introducir los parámetros, es decir, el SP apuntando por encima de la dirección de la pila que contiene #n.

8.15-LISTA ENLAZADA. (LINKED LIST).

- En las listas simples, es decir, lista de elementos que guardan una determinada relación entre ellos, (las vistas hasta ahora) se guardan en la memoria principal de tal manera, que elementos sucesivos, ocupan posiciones contiguas de memoria. Este esquema es fácil de usar, pero plantea un problema cuando los elementos de la lista deben guardar cierto orden:

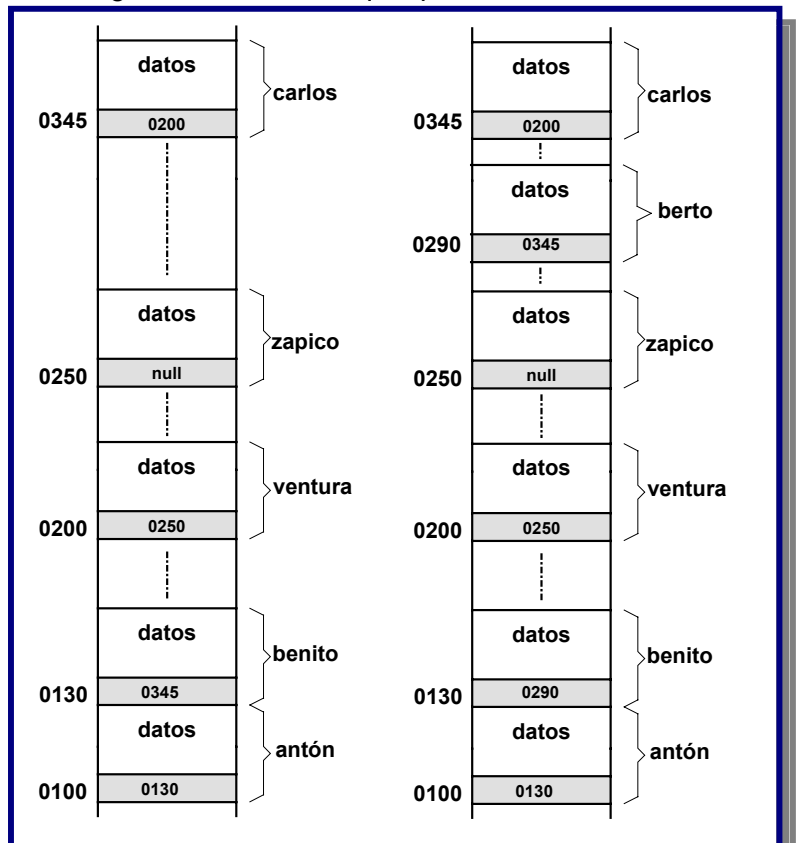
Supongamos que los elementos de una lista deben ser mantenidos en un cierto orden, por ejemplo, en orden alfabético. Añadir un elemento al final de la lista es simple, pero añadirlo dentro de la lista (para seguir manteniendo el orden) es más complicado. Si queremos introducir en la lista el elemento “+”, el cual para mantener el orden de la lista, debe ir en la posición i, se deberán desplazar todos los elementos desde la posición i hasta el final, un lugar, para dejar el espacio correspondiente al del elemento que se quiere introducir.



LISTA ENLAZADA:

- En una lista enlazada, no es necesario guardar los elementos sucesivos en posiciones contiguas de memoria, pueden ser guardados en cualquier lugar de la memoria principal.
- Para mantener el orden correspondiente, cada elemento de la lista, incluye un número (*LINK*), que indica la localización del siguiente elemento que pertenece a la lista. Este número puede ser la dirección del siguiente elemento, de esta manera cada componente perteneciente a la lista enlazada, consiste en un *Link* (enlace) y la información correspondiente de la que consta el elemento. De esta manera, no es necesario que elementos sucesivos de la lista, estén en posiciones contiguas de la memoria.

En el ejemplo que se da en la figura, se pretende introducir un nuevo elemento en la lista, este elemento es *berto*. Se introduce en cualquier espacio libre de la memoria, y lo único que se debe cambiar es el link de *benito* para que en vez de apuntar a *carlos*, apunte a *berto*. El link de *berto* deberá entonces apuntar a *carlos*.

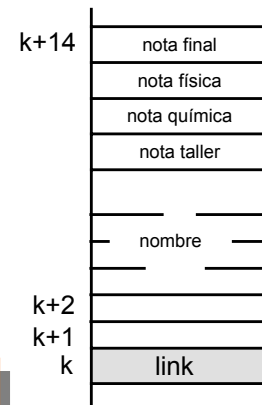


Para borrar un elemento de la lista el proceso es análogo.

- ❑ El último elemento de la lista no apunta a ninguna posición. En lugar de esto, su enlace consiste en un código que indica que él es el elemento final de la lista.
- ❑ Las entradas correspondientes a un elemento de la lista, deben ser guardadas en posiciones contiguas, de esta manera pueden ser identificadas en relación a la entrada de enlace del elemento.
- ❑ La referencia a elementos individuales no es tan fácil como en una lista simple. Para acceder a un elemento de la lista enlazada, se debe comenzar por la cabecera de la lista, es decir, el primer elemento de la misma, y de forma secuencial acceder a los links hasta llegar al elemento el requerido.

EJ:

Supongamos que queremos encontrar la nota final más alta en una clase. Consideramos que cada término de una lista enlazada, es la ficha de un alumno, con diversas notas de otras tantas materias y una nota final. Cada ficha de un alumno ocupa 15 palabras consecutivas en la memoria, organizadas como muestra la figura dada.



	Move	#LISTADDRESS, LINKPTR
	Clear	ACCUM
LOOP	Compare	ACCUM, 14(LINKPTR)
	Branch_if_≥0	NEXT
	Move	14(LINKPTR), ACCUM
NEXT	Compare	#0, LINKPTR
	Branch_if_=0	DONE
	Move	[LINKPTR], LINKPTR
	Branch	LOOP
DONE	next instruction	

El proceso consiste en llevar la dirección del primer elemento de la lista (`#LISTADDRESS`), dada por la dirección de su *link*, a un puntero. Se prepara el acumulador, que es el registro que se

va a usar para ir guardando la nota más elevada encontrada hasta ese momento. En cada paso por el bucle de comparación, se va a comparar el valor almacenado en el acumulador, con el valor contenido en la posición de memoria que corresponde a la nota final de cada elemento. La dirección de la posición de memoria que contiene la nota final, va a corresponder siempre, en cada elemento, con la dirección que corresponde al *link*, *sumado 14*.

Para ir procesando los elementos de la lista, hacia el final del programa se ve como en el puntero se carga el valor del *link*, que como ya se ha dicho, es la dirección que corresponde al siguiente elemento de la lista.

Para determinar cuando se finaliza el proceso, cada link que se procesa se compara con un *null* (#0), que es el carácter que indica que el elemento al cual pertenece, es el último de la lista.

Programa alternativo:

El proceso que sigue este programa para la obtención de la nota más alta, es bastante parecido al explicado en el programa anterior.

	Move	#LISTADDRESS, LINKPOINTR
LOOP	Move	14(LINKPOINTR), ACCUM
NEXT	Move	[LINKPOINTR], LINKPOINTR
	Compare	#0, LINKPOINTR
	Branch_if_=0	DONE
	Compare	ACCUM, 14(LINKPOINTR)
	Branch_if_≥0	NEXT
	Branch	LOOP
DONE	next instruction	

8.16-CADENA DE CARACTERES. (CHARACTER STRINGS)

- ❑ Una cadena de caracteres, es una colección de caracteres definida como una entidad con cierto sentido. Puede ser tan corta como una letra, o un nombre, etc. y tan larga como se quiera.
- ❑ Las cadenas de caracteres, pueden ser manipuladas de muchas maneras. Entre las operaciones elementales con cadenas de caracteres, están las de comparación de dos cadenas, cambio de una parte de la cadena, concatenación, ordenación, etc.
- ❑ El modo de manipulación de las cadenas de caracteres, es mediante el trabajo con el código ASCII que corresponde a cada carácter que forma la cadena. Como se recordará, cada carácter es codificado con 7 bits. En la siguiente tabla se da la correspondencia entre código y carácter.

b3 b2 b1 b0		b6 b5 b4						
		000	001	010	011	100	101	110
0000	NUL	DLE	SPACE	0	@	P	'	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	`	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DEL

NUL NUL/IDLE
 SOH START OF HEADER
 STX START OF TEXT
 ETX END OF TEXT
 EOT END OF TRANSMISION
 ENQ ENQUIRY
 ACK ACKNOWLEDGEMENT
 BEL AUDIBLE SIGNAL
 BS BACK SPACE
 HT HORIZONTAL TAB
 LF LINE FEED
 VT VERTICAL TAB
 FF FORM FEED
 CR CARRIAGE RETURN
 SO SHIFT OUT

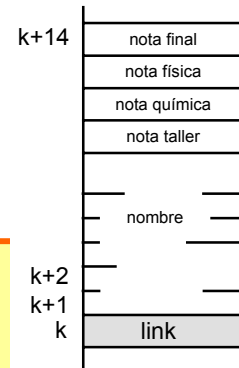
SI SHIFT IN
 DLE DATA LINK ESCAPE
 DC1-DC4 DEVICE CONTROL
 NAK NEGATIVE ACKNOWLEDGEMENT
 SYN SYNCHRONOUS IDLE
 ETB END OF TRANSMITTED BLOCK
 CAN CANCEL (ERROR IN DATA)
 EM END OF MEDIUM
 SUB SPECIAL SEQUENCE
 ESC ESCAPE
 FS FILE SEPARATOR
 GS GROUP SEPARATOR
 RS RECORD SAPARATOR
 US UNIT SAPARATOR
 DEL DELETE/IDLE

Ej.

Supongamos que deseamos encontrar la nota más alta de un estudiante determinado, en una lista enlazada, siendo la estructura de cada elemento de la lista igual a la dada en la figura.

Un posible programa, que hace uso de la comparación de cadenas, para encontrar el nombre buscado, es el que se da a continuación.

El comienzo de la lista enlazada viene dado por la posición de memoria #LISTADDRESS. Esta dirección es la que contiene el *link* del primer elemento de la lista, según el esquema dado para cada elemento.



	Move	#LISTADDRESS, LINKPOINTR
	Branch	NAMEINIT
NEXTITEM	Move	[LINKPOINTR], LINKPOINTR
NAMEINIT	Compare	#0, LINKPOINTR
	Branch_if_0	NOTFOUND
	Move	#STUDENT, POINTER1
	Move	LINKPOINTR, POINTER2
	Increment	POINTER2
MATCH	Compare	[POINTER1]autoinc, [POINTER2]autoinc
	Branch_if_n0	NEXTITEM
	Compare	#CR, [POINTER1]
	Branch_if_n0	MATCH
	instrucciones para presentar el nombre y las notas	
NOTFOUND	instrucciones que muestran el mensaje de que el nombre del estudiante no está en la lista	

La cadena de caracteres que identifica al alumno que se quiere buscar, comienza en la posición de memoria llamada #STUDENT.

El último elemento de la lista enlazada, tendrá como *link* el carácter NUL (#0000000). El final del nombre del alumno a buscar, viene dado por un carácter #CR (#0001101).

La operación básica del programa consiste por un lado, en tomar el comienzo de cada elemento de una lista enlazada utilizando el puntero LINKPOINTR, y comprobar que no es el último de la lista. Por otro lado, utilizando el puntero POINTER2, ir recorriendo cada uno de los caracteres que componen el nombre del alumno en el elemento de la lista, y compararlos uno a uno con el nombre a buscar, el cual se va direccionando mediante el puntero, POINTER1.

En la primera ocurrencia en la cual la coincidencia de caracteres no se cumple, se produce un salto a NEXTITEM. La instrucción que se ejecuta en esta posición de memoria es *Move [LINKPOINTR], LINKPOINTR*. Esta instrucción lo que hace es que LINKPOINTR, apunte al siguiente elemento de la lista, pues como se ha dicho, el contenido de las posiciones donde éste apunta, es el link hacia el siguiente elemento, o el carácter final de lista.

Tanto si el nombre del alumno se encuentra en la lista, como si no, el programa debe dar paso a las instrucciones pertinentes que indiquen una u otra incidencia.

¿Qué sucede si el nombre a buscar es el de Manuel, y en la lista enlazada existe el nombre Manuela?

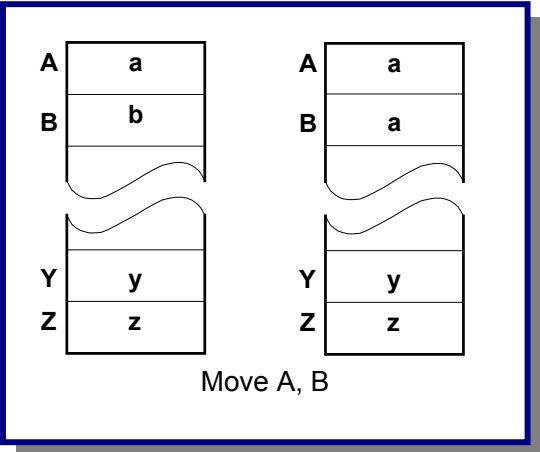
8.17-MODOS DE DIRECCIONAMIENTO BÁSICOS.

- Los modos de direccionamiento representan las formas en las que se puede acceder a los operandos que necesitan las instrucciones. En una misma instrucción pueden darse dos modos de direccionamiento diferente, una para el operando fuente, y otra para el operando destino.

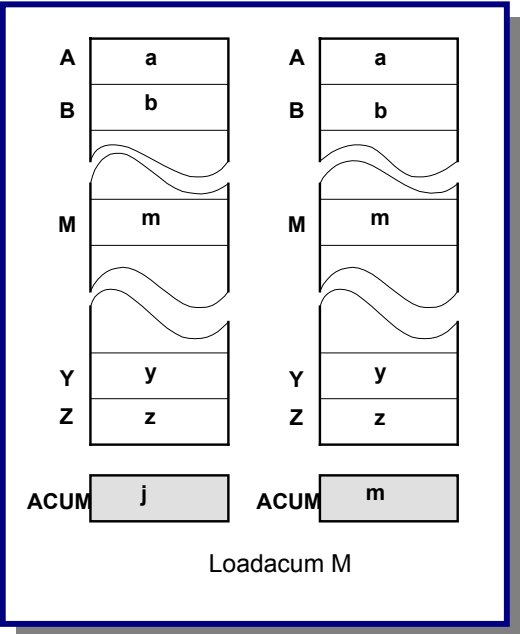
NOMBRE	OPERACIÓN	EJEMPLO
ABSOLUTO	La dirección del operando está dada explícitamente en la instrucción	Move A, B B $\leftarrow$ [A]
REGISTRO (IMPLÍCITO)	La localización del operando es un registro de la CPU especificado en la instrucción.	Load_acum M ACUM $\leftarrow$ [M] Store_acum M M $\leftarrow$ [ACUM]
INMEDIATO	El valor del operando se da explícitamente en la instrucción.	Move #n, A A $\leftarrow$ #n
INDEXADO	La dirección del operando es la suma de los contenidos de un registro índice y un valor índice X. Ambos son dados en la instrucción.	Clear 180 (R) $A_{\text{eff}} = 180 + [R]$ (A_{eff} = dirección efectiva)
INDIRECTO	La dirección efectiva del operando es el contenido de un registro o una posición de memoria cuya dirección se da en la instrucción.	Move [A], [B]
RELATIVO	La dirección del operando es especificada con relación al contenido del PC. Un valor X es dado en la instrucción. Sumado este valor al PC se obtiene, la dirección del operando.	Clear 180 (PC) $A_{\text{eff}} = 180 + [PC]$
AUTOINCREMENTO	El contenido del reg. o pos. de mem. se incrementa después de la operación especificada por la instrucción.	Move [A] _{autoinc} , [B] _{autoinc}
AUTODECREMENTO	El contenido del reg. o pos. de mem. se decrementa y después se realiza la operación especificada por la instrucción.	Move [A] _{autodec} , [B] _{autodec}

En la página siguiente se muestra de forma gráfica algunos modos de direccionamiento.

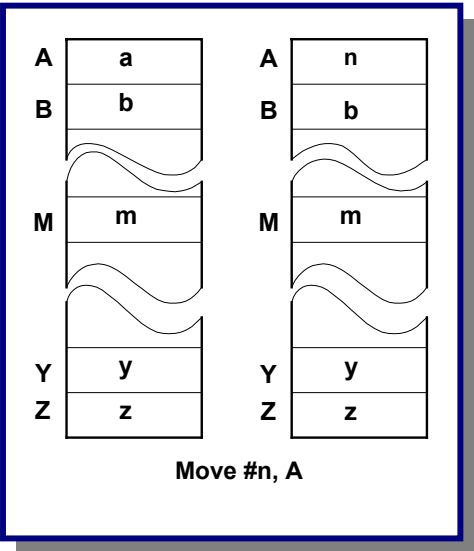
ABSOLUTO



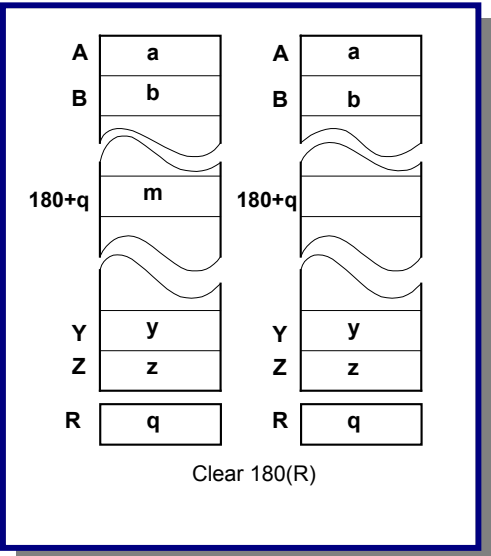
REGISTRO



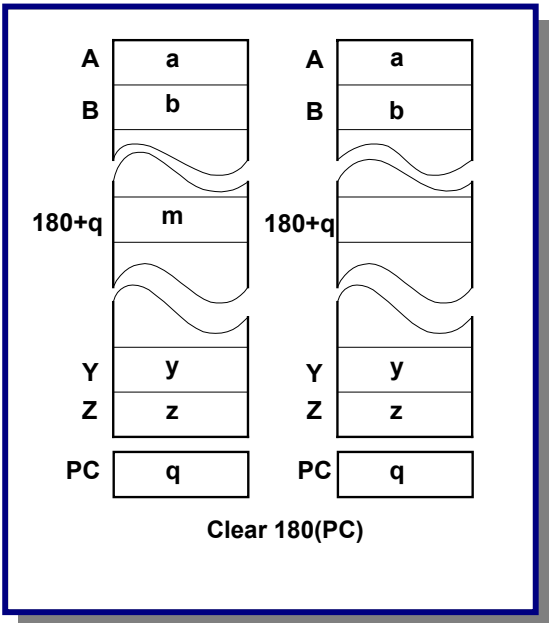
INMEDIATO



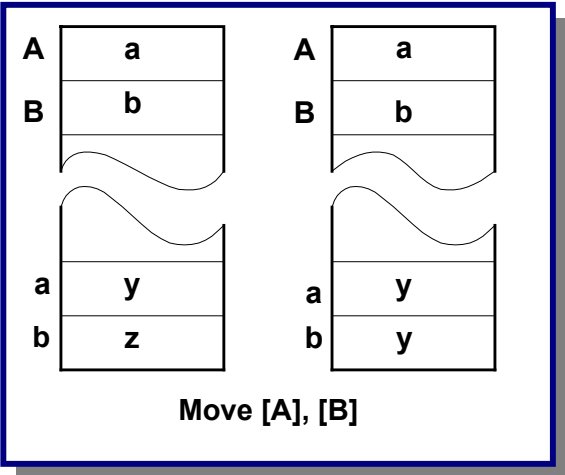
INDEXADO



RELATIVO



INDIRECTO



EJEMPLO:

*esto es un ejemplo de la estructura de parte de un programa que llama a la subrutina

* SUBCOPY , la cual copia 50 elementos de la lista A a la lista B. Suponemos que

*lista A comienza en la posición 2000 y lista B en la 2500.

*POINTERA, POINTERB y POINTERC son registros de la CPU.

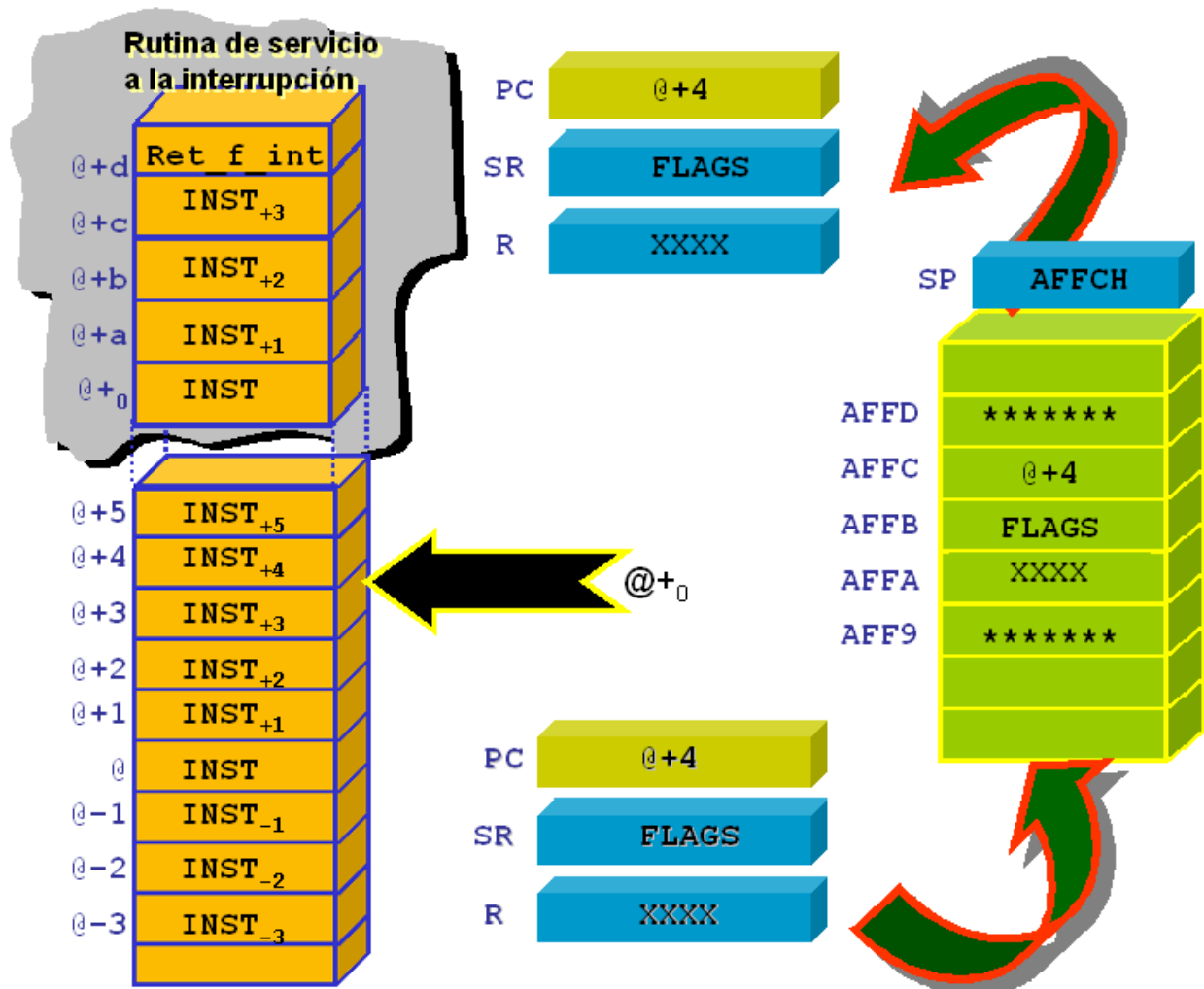
ADDRESSA	EQU	2000		
ADDRESSB	EQU	2500		;uso de directivas
	.			
	.			
	.			
	BSR	SUBCOPY		;salta a la subrutina
	DATA	50, ADDRESSA, ADDRESSB		;directiva
	.			
	.			

*esta es la subrutina copiar.

NCOUNT	EQU	3000		;directiva
	ORG	1000		;directiva
SUBCOPY	MOVE	SP, POINTERC		
	MOVE	[POINTERC], NCOUNT		
	INC	POINTERC		
	MOVE	[POINTERC], POINTERA		
	INC	POINTERC		
	MOVE	[POINTERC], POINTERB		
	INC	POINTERC		
	MOVE	POINTERC, SP		
LOOP	MOVE	[POINTERA], [POINTERB]		
	INC	POINTERA		
	INC	POINTERB		
	DEC	NCOUNT		
	BGT	LOOP		;salta (condicional) a LOOP, ;si al decrementar (DEC) ;NCOUNT, el contenido de ;éste es mayor que cero. ;(BGT)
	RTS			;Vuelta de la subrutina.
	END			;directiva

8.18-INTERRUPCIONES.

- Una interrupción es una petición que se realiza a la CPU para que ésta, deje la tarea que está realizando, y preste su atención al mecanismo que genera la interrupción. La interrupción puede ser generada por la interfaz de un dispositivo que esté capacitado para enviar o recibir datos, puede ser generada también por un circuito que controle el correcto funcionamiento de un sistema, y puede también ser generada por instrucciones.



Para el caso de un dispositivo externo que a través de su interfaz pide a la CPU su atención, mediante una interrupción, la secuencia de acontecimientos que se producen se representa en la figura:

La CPU, se encuentra ejecutando un programa y durante este proceso, cuando está ejecutando la instrucción situada en la dirección @+3, un dispositivo solicita ser atendido, provocando una interrupción. La CPU consulta ciertos datos dados por el programador, para saber si esa interrupción que llega en ese momento, es digna de atención (por su nivel), o por el contrario, se debe ignorar y seguir ejecutando la tarea que estaba realizando.

Si la interrupción debe ser atendida, antes de ejecutar la instrucción inmediatamente siguiente a la que estaba ejecutando cuando se produjo la interrupción, la CPU almacena el **PC** (*Program Counter*), el **SR** (*Status Register*) y a veces algunos registros mas, en la pila (stack). El valor del PC guardado, tiene la dirección de la instrucción que se debería haber ejecutado si no hubiera habido una interrupción.

Una vez almacenados estos registros, la CPU comprueba qué dispositivo necesita de su atención. Cada uno de los dispositivos que puede solicitar en cualquier momento la atención de la CPU mediante una interrupción, tiene asociada una posición particular de la memoria, del sistema, llamada *VECTOR*. En el caso de la interrupción de la figura, el vector asociado con la interrupción es @+ 0. Entonces, la CPU se va al vector que corresponde a ese dispositivo (@+ 0) y de esa posición de memoria, obtiene la dirección donde debe comenzar a ejecutar una subrutina específica (*Rutina de Atención a la Interrupción*) para la atención del dispositivo en cuestión.

Cuando la CPU termina de atender al dispositivo mediante la ejecución de la rutina específicamente diseñada para él, vuelve a lo que estaba haciendo cuando llegó la interrupción que acaba de ser atendida. Una instrucción especial de *vuelta de interrupción*, hace que en el PC se cargue el valor guardado de forma momentánea en la pila, además del SR y de los registros que hubiera guardado cuando aceptó atender la interrupción, de esta manera, el entorno vuelve a ser exactamente el mismo que había cuando fue interrumpido el proceso.

- De lo explicado se pueden sacar ciertas conclusiones:

La interrupción se inicia por un suceso externo.

Puede tener lugar en cualquier parte dentro del programa interrumpido y en cualquier momento. Es pues un suceso asíncrono con el programa principal.

La rutina de atención no tiene porque tener nada en común con el programa interrumpido.

- Como se puede observar, el funcionamiento general del μP es diferente en el caso de subrutinas y de interrupciones. La instrucción de final de una y otra también son diferentes. (**RTS** –Return From Subroutine–, **RTE** –Return From Exception–). Recordemos que una subrutina se inicia por una instrucción específica dentro del programa llamador (se ejecutan por lo tanto de forma síncrona con el programa llamador), y que constituye parte esencial de dicho programa.

SOFTWARE DE INTERRUPCIONES.

Veamos como puede realizarse una tarea mediante el uso de interrupciones.

```

MAIN.....
.....
.....
Call      SHOW
.....
.....
SHOW Move    SRPANT, ACCUM
AND     #1, ACCUM
Branch_if_0 SHOW
Move    CHAR, DRPANT
Rt_from_s

```

Este otro programa, hace uso de las interrupciones. Cuando el dispositivo esté preparado para recibir el carácter, se lo hace saber a la CPU mediante una interrupción.

Ej. Se quiere enviar un carácter ASCII a un monitor.

En este programa, el método que se utiliza es el de chequeo del registro de estado de la interfaz. El programa entra en un bucle hasta que puede enviar el carácter.

```

MAIN .....
.....
Move    #SHOW, INTVECTOR
Enable_int
.....
.....
SHOW Disable_int
Move    CHAR, DRPANT
Rt_from_i

```

Como se puede observar, el programa principal o programa con el mayor peso en la aplicación, se encarga de guardar en el vector que corresponde en este caso al monitor, la dirección donde comienza la rutina de atención a la interrupción que la misma interfaz del monitor provoque. Acto seguido, habilita las interrupciones, es decir, permite que a partir de ese momento, un dispositivo puede provocar una interrupción.

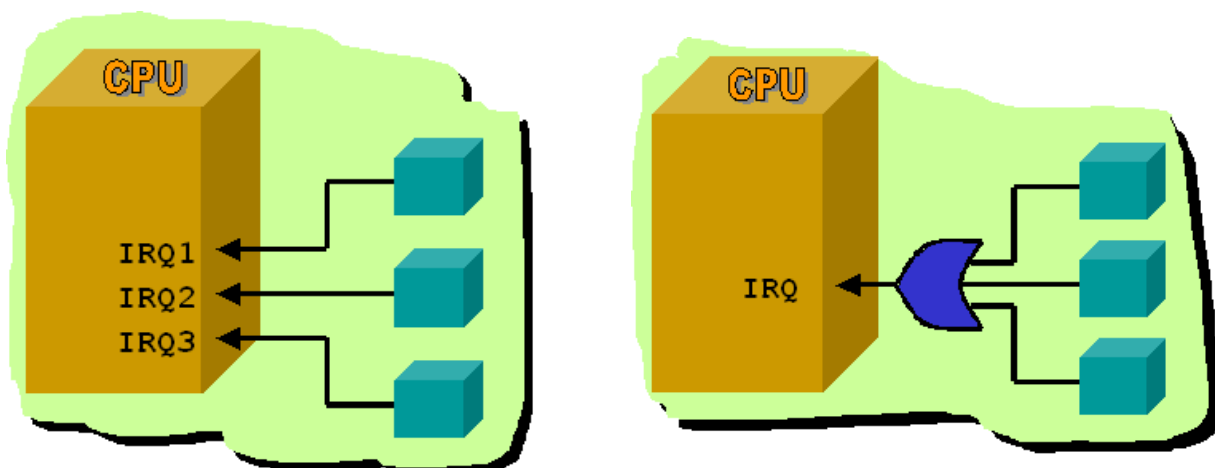
Al comienzo de la subrutina de servicio o atención a la interrupción, se deben inhabilitar (si la CPU no lo hace de forma automática), las interrupciones. El motivo es que si no se hace, y la interrupción sigue activa al ejecutarse la primera instrucción de atención a la interrupción, otra interrupción ocurriría, generando la ejecución de la misma atención una y otra vez.

MÁSCARA DE INTERRUPCIÓN.

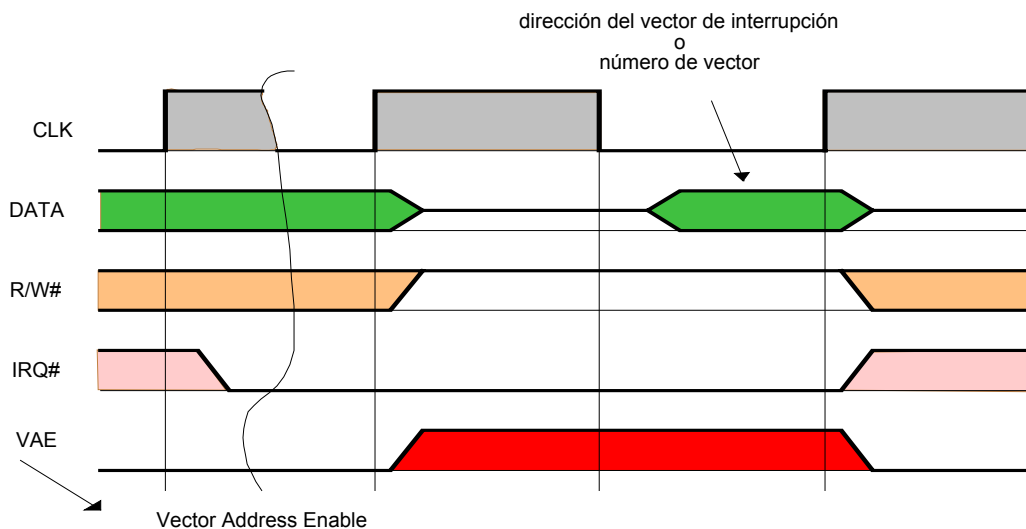
- ❑ Es un mecanismo que permite al programador prohibir o autorizar que una interrupción sea tomada en cuenta o sea ignorada. Cuando se solicita una interrupción, la CPU comprueba la máscara para saber si dicha interrupción debe ser tratada o no.
- ❑ Desde el punto de vista del programador, este mecanismo consiste en uno o varios bits pertenecientes al SR. Según el valor de éstos, ciertas interrupciones serán tratadas y otras no. Existen instrucciones específicas para la manipulación de los bits de máscara.
- ❑ La máscara de interrupción puede ser también activada de forma automática por el μP al ejecutar la rutina de atención a una interrupción, para evitar que una interrupción de menor nivel (que no debería ser tratada) interrumpa la ejecución de la rutina en curso.
- ❑ En algunos microprocesadores, existen líneas para interrupciones que no pueden ser inhabilitadas por el programador. Estas líneas suelen llevar el nombre de **NMI** (*Non Maskable Interrupts*).
- ❑ Con el fin de que cuando ocurran este tipo de interrupciones no se entre en el círculo descrito anteriormente de interrupción interrumpida una y otra vez, se suelen hacer las NMI sensibles al flanco en vez de al nivel.

HARDWARE DE INTERRUPTACIONES.

- ❑ La interfaz del dispositivo externo debe incluir un circuito que genere una señal de petición de interrupción
- ❑ La CPU puede tener una o más líneas que pueden estar destinadas a las peticiones de interrupción. En cada línea puede haber varios dispositivos de I/O conectados.



- ❑ Cuando varios dispositivos están conectados en una línea de petición de interrupción, la rutina de atención necesita saber cual de los dispositivos solicita el servicio. Esto se puede realizar chequeando los registros de estado de cada dispositivo, uno por uno, hasta encontrar aquel que solicitó la interrupción (*POLLING SCHEME*). Este método puede resultar lento si existen muchos periféricos y el que solicitó la interrupción se chequea en los últimos lugares.
- ❑ Otro método viene dado por un mecanismo hardware que permite que un dispositivo, cuya petición de interrupción ha sido aceptada, se identifique él mismo. Esta manera de identificación puede venir dada en forma de código preestablecido, el cual puede ser una dirección (vector) que apunta a una localización de la memoria principal, o puede ser también un número (número de vector) que está relacionado con una posición de la memoria. Este sistema se conoce como *INTERRUPCIONES VECTORIZADAS*.



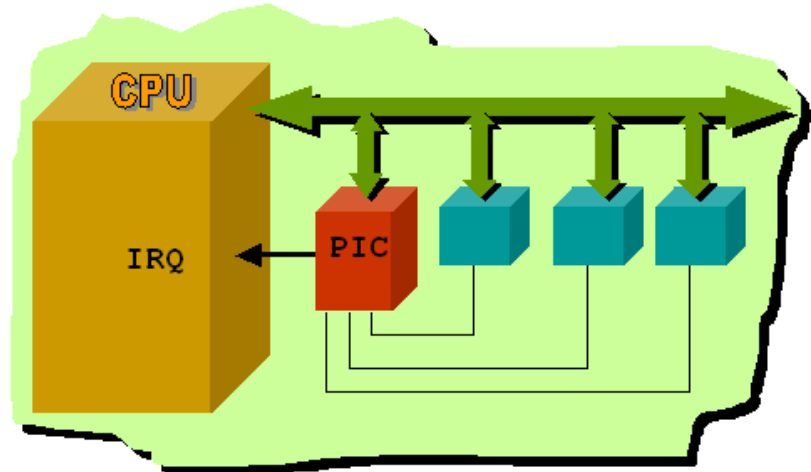
El dispositivo activa la línea de **IRQ**. Cuando el μP acepta la petición de interrupción, activa el **VAE** como una indicación de que ha reconocido la petición de interrupción y que está listo para recibir el vector de interrupción. El dispositivo responde enviando la dirección de dicho vector. Después de recibir esta dirección, el μP salva el PC, SR y otros registros que deban ser salvados, y lee la dirección de comienzo de la rutina de atención a la interrupción. Esta dirección la obtiene de la posición de memoria cuya dirección fue entregada por el dispositivo en forma de vector.

- ❑ Cuando existen varios dispositivos en una línea que solicitan interrupción a la vez, la señal VAE es transferida de un dispositivo al siguiente, siempre y cuando dichos dispositivos estén preparados para esta forma de trabajo. El primero que recibe esta señal de aquellos que solicitaron atención, transmite su vector de interrupción y bloquea la propagación de la señal a otros dispositivos. Una vez que su rutina ha sido tratada, desbloquea la señal VAE para que pase al siguiente dispositivo. (*DAISY-CHAIN*).
- ❑ Algunas veces es necesario que una rutina de atención a una interrupción sea a su vez interrumpida. Este puede ser el caso de un dispositivo cuya operación sea temporalmente crítica por la pérdida de datos si no es atendido. La forma de arbitrar esto, es asignar a diferentes dispositivos, diferentes prioridades. De esta manera, una interrupción interrumpe la ejecución de la rutina de atención a otra interrupción solamente, si el nivel de prioridad de aquella es superior al de esta última.
- ❑ La distribución de prioridades según lo expuesto anteriormente, puede ser llevada a cabo de diferentes maneras. Una forma es mediante varias líneas de interrupción, cada una con un nivel de prioridad determinado. En cada línea se podrán conectar los dispositivos según la prioridad que el diseñador considere oportuna.

Otra forma es que la prioridad venga codificada en varias líneas proporcionadas para este propósito. En este caso las peticiones de interrupción deben ser primeramente codificadas para que la CPU determine su nivel de prioridad.

CONTROLADOR DE INTERRUPCIONES (PIC- PROGRAMMABLE INTERRUPT CONTROLLER).

- ❑ Tiene como misión la de expandir el número de entradas de interrupción de la CPU, y gestionar las solicitudes de interrupción que se realicen a la misma.
- ❑ Cada familia de μP posee sus propios circuitos PIC.
- ❑ Todas las líneas de interrupción de los periféricos pasan por el PIC.
- ❑ El PIC indica a la CPU cual es el periférico que debe ser atendido. Esta información se envía en forma de código que representa el número de interrupción.
- ❑ Si se da el caso de varias interrupciones simultáneas, el PIC también se encarga de la gestión de la prioridad.



TRAPS. (INTERRUPCIONES SOFTWARE)

- ❑ En este caso la interrupción se produce cuando se ejecuta la instrucción, por lo tanto es síncrona con la ejecución del programa.
- ❑ El proceso que sigue a este tipo de interrupciones es el mismo que cuando la interrupción es generada por un dispositivo de I/O.
- ❑ La dirección de comienzo de la subrutina de atención a la interrupción está contenida en el vector de interrupción que corresponde a cada interrupción software.
- ❑ Estas interrupciones también están sometidas al nivel de interrupción para ser aceptadas o no.

8.19-EXCEPCIONES.

- ❑ Son interrupciones provocadas por sucesos inesperados o anormales en el funcionamiento del μP ; Códigos de operación que no existen, accesos a posiciones fuera del mapa de memoria, intento de división entre 0... El proceso que sigue a una excepción es el mismo que el explicado para las interrupciones, aunque pueden variar factores como el que se salven en la pila unos registros u otros, el que se atienda antes la petición, el que existan líneas en la CPU dedicadas a estas excepciones, etc.

En la nomenclatura de Motorola®, las interrupciones hardware, software y excepciones se engloban dentro de la palabra *excepción*.

8.20-EJEMPLOS (6809 PROGRAMACIÓN)

Movimiento de una lista

```

LOOP    LDX      #0
        LDA      LISTA,X
        STA      LISTB,X
        LEAX     1,X
        CMPX     N
        BLO      LOOP
        ...

```

```

        CLRA
        LDX      #LISTA
        LDY      #LISTB
LOOP    LDB      A,X
        STB      A,Y
        INCA
        CMPA     N
        BLO      LOOP

```

```

        LDA      #50
        LDX      #LISTA
        LDY      #LISTB
LOOP    LDB      ,X+
        STB      ,Y+
        DECA
        BHI      LOOP
        ...

```

```

        LDA      #50
        LDX      #LISTA
        LDY      #LISTB
        BSR      MOVE
        ...
        .
        .
        .
MOVE    LDB      ,X+
        STB      ,Y+
        DECA
        BHI      MOVE
        RTS

```

Lectura de un carácter del teclado

```

WAIT    LDA      STATUS
        ANDA     #1
        BEQ      WAIT
        LDA      INBUF
        ...

```

Para mostrar una línea por pantalla.

```

CR      EQU      $0D      código ASCII para cr
        LDX      #MSG
WAIT    LDA      STATUS
        ANDA     #2
        BEQ      WAIT
        LDA      ,X+
        STA      OUTBUF
        CMPA     #CR
        BNE      WAIT
        ...

```

Rutina para buscar el número más alto en una lista.

	LDY	#LIST
	LDD	#500
	BR	MAX
	...	
	.	
MAX	LEAX	D,Y
	STX	LSTEND
	TFR	Y,X
	LDA	,X
LOOP	CMPA	,X+
	BGE	NEXT
	LEAY	-1,X
	LDA	-1,X
NEXT	CMPX	LSTEND
	BLO	LOOP
	RTS	

(OBSERVAR EL PASO DE PARAMETROS MEDIANTE REGISTROS DEL μ P)

Copiar una lista de 50 entradas de una zona de memoria a otra zona.

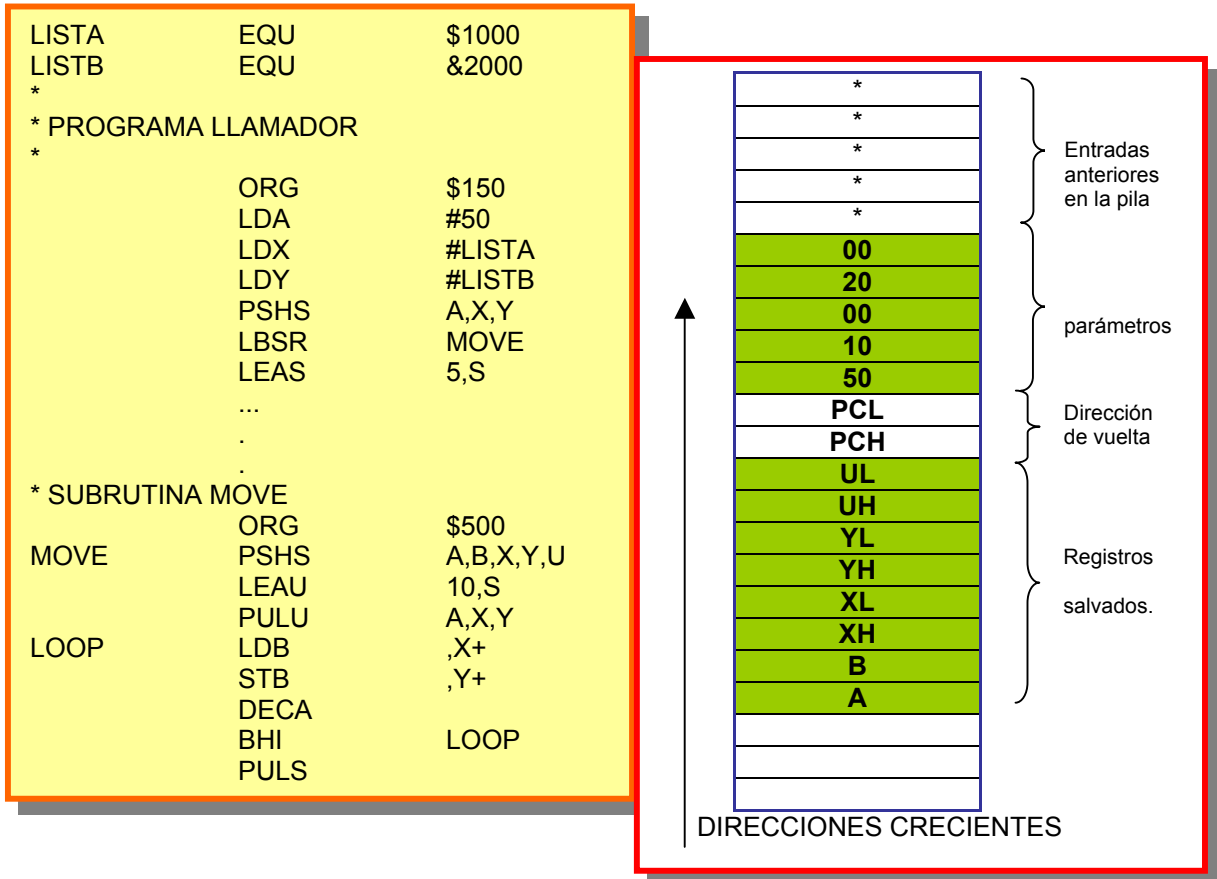
(OBSERVAR EL PASO DE PARÁMETROS DESPUÉS DE LA INSTRUCCIÓN DE LLAMADA).

	BSR	MOVE
	FCB	50
	FDB	LISTA, LISTB
	...	
	.	
	.	
MOVE	LDU	,S
	PULU	A,X,Y
	STU	,S
LOOP	LDB	,X+
	STB	,Y+
	DECA	
	BHI	LOOP
	RTS	

	BSR	MOVE
	FCB	50
	FDB	LISTA,
LISTB		
	...	
	.	
	.	
MOVE	LDU	,S
	LDA	,U+
	LDX	,U++
	LDY	,U++
	STU	,S
LOOP	LDB	,X+
	STB	,Y+
	DECA	
	BHI	LOOP
	RTS	

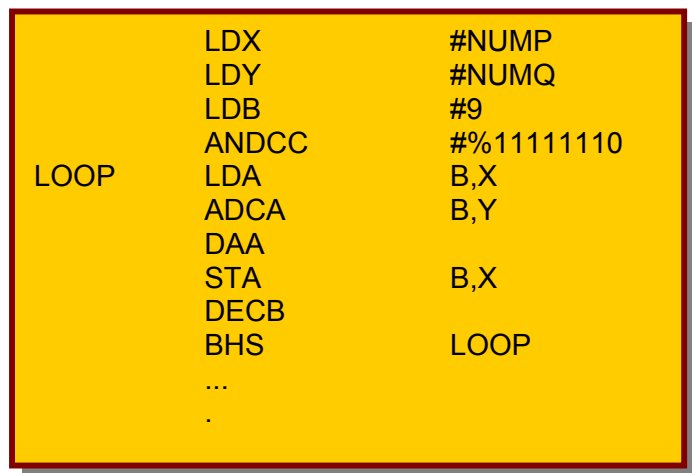
	BSR	MOVE
	FCB	50
	FDB	LISTA,LISTB
	...	
	.	
	.	
MOVE	PSHS	A,B,X,Y,U
	LDU	8,S
	PULU	A,X,Y
	STU	8,S
LOOP	LDB	,X+
	STB	,Y+
	DECA	
	BHI	LOOP
	PULS	A,B,X,Y,U,PC

PASO DE PARÁMETROS EN LA PILA.



Contenido de la pila al ejecutar la subrutina.
(después de PSHS)

Suma de operandos BCD de 20 dígitos



Determinar la paridad de un elemento de 8 bits

```

      LDA      DATA
      CLRB
LOOP  LSRB
      ADC      #0
      TSTA
      BNE     LOOP
      LSRB
      BCS     IMPAR
* procesar paridad par
      .
      .
      .
IMPAR .
      .

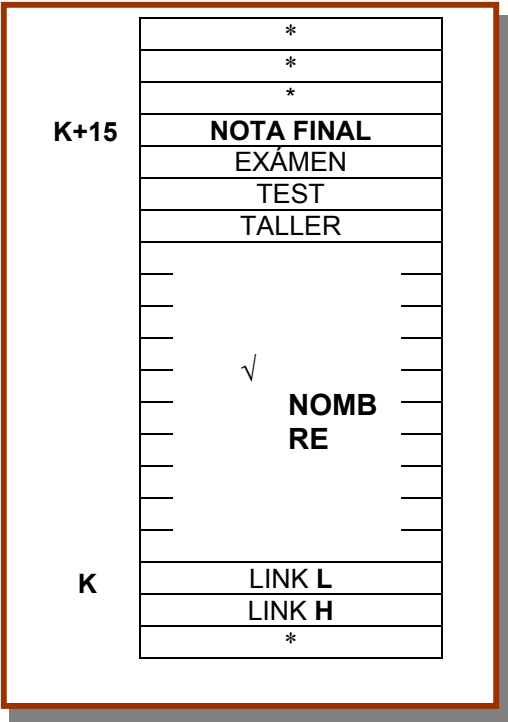
```

Encontrar la nota final más alta en una lista enlazada

```

      LDX      #LSTADR
LOOP  LDA      15,X
NEXT  LDX      ,X
      BEQ      DONE
      CMPA    15,X
      BGE     NEXT
      BRA     LOOP
DONE  ...
      .

```





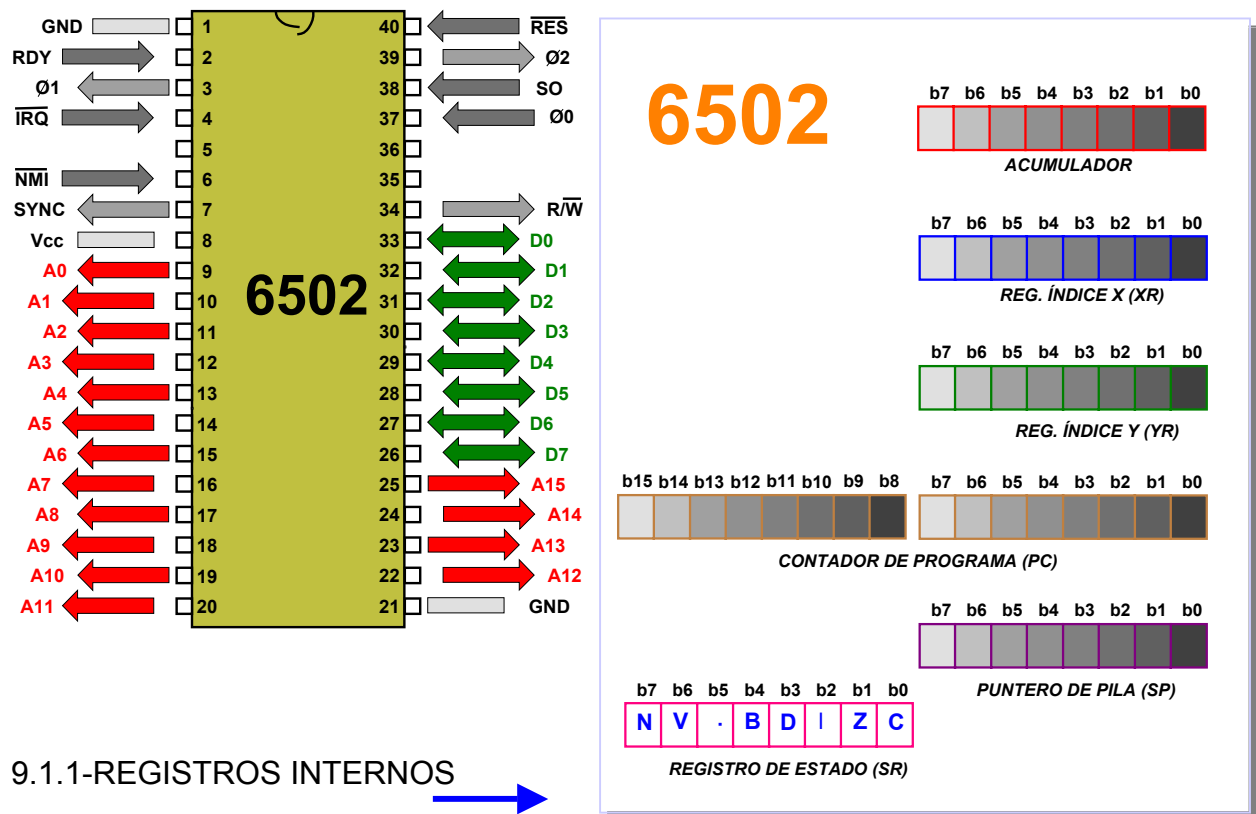
microprocesador

9

9.-µP DE 8 BITS. GENERALIDADES

- La denominación de un µprocesador por su número de bits, obedece a la forma de indicar el número de bits que posee su bus de datos para trabajo externo, independientemente de la capacidad de los registros internos.

9.1-6502



9.1.1-REGISTROS INTERNOS

9.1.2-MODOS DE DIRECCIONAMIENTO (6502).

INMEDIATO:

El valor del operando es el valor que se da después de la instrucción.

LDA #OOH

ABSOLUTO:

El valor dado en la instrucción es la posición de memoria donde se encuentra el operando.

LDA 21ABH

POR PÁGINA CERO:

Como el anterior pero dirigido a la página cero.

LDA 61H (0061H)

POR PÁGINA CERO, ÍNDICE X O Y:

El operando está contenido en una posición de la página cero que se obtiene al sumar el contenido del registro índice X o Y con la dirección dada en la instrucción.

LDA 34H,X

LDA 51H,Y

ABSOLUTO, ÍNDICE X O Y:

La dirección efectiva se obtiene sumando el contenido del registro índice X o Y a la dirección dada en la instrucción.

LDA 457BH, X

LDA 45BDH, Y

POR ACUMULADOR:

La instrucción solamente implica al acumulador

ROR

IMPLICADO:

El operando va implícito en la propia instrucción.

CLC

RELATIVO:

El valor dado en la instrucción se suma o resta al contador de programa para provocar un salto a la dirección del operando.

BEQ 14

INDIRECTO:

La dirección dada se carga en el PC para provocar un salto.

JMP 7FBBH

INDEXADO INDIRECTO X:

En la instrucción se da un valor, y una referencia al registro índice X. Este valor, sumado con el contenido del registro índice X produce una dirección en la página cero. El contenido de esta dirección (LB) y el de la siguiente (HB) forman la dirección efectiva.

LDA [10H, X]

INDIRECTO INDEXADO Y:

En la instrucción se da un valor, y una referencia al registro índice Y. Este valor, apunta a una posición de la página cero. El contenido de esta posición y el de la siguiente (colocados en orden inverso) forman una dirección que sumada al contenido de Y dará la dirección efectiva.

LDA [10H], Y

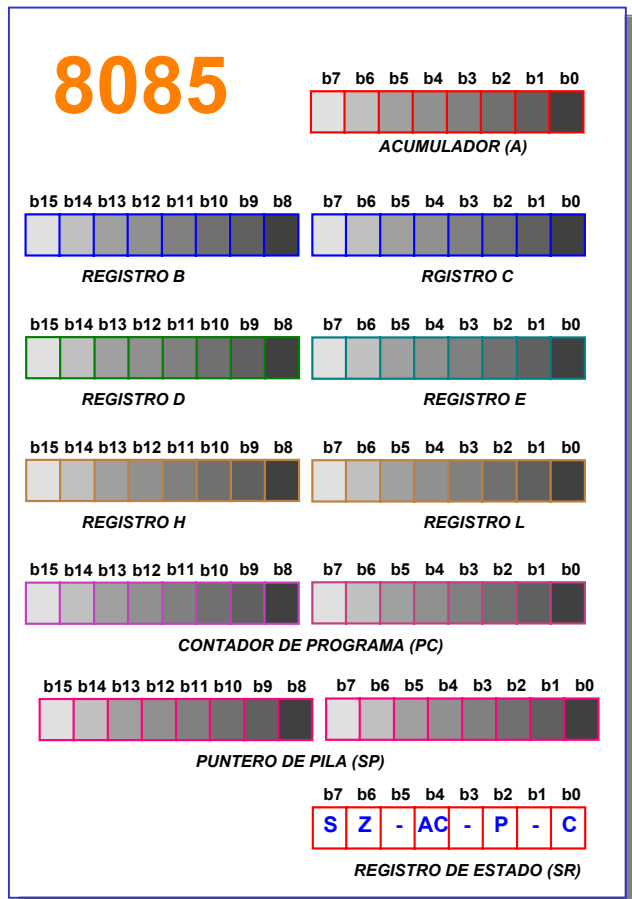
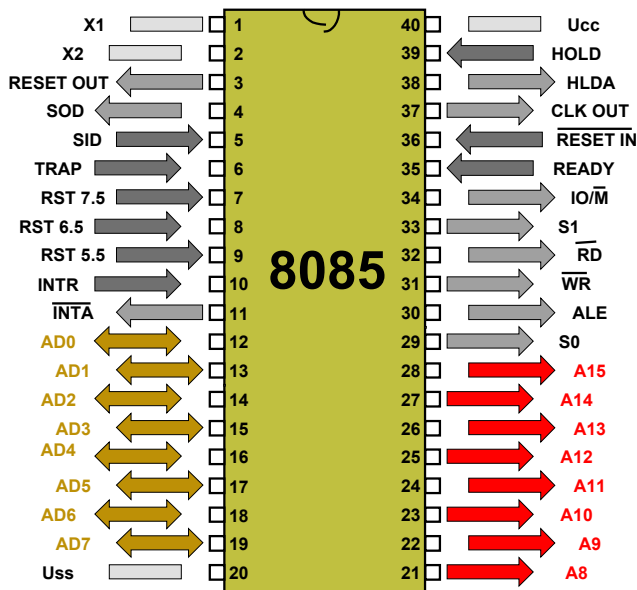
9.1.3-REGISTRO DE ESTADO (6502)



- **N:**
FLAG DE SIGNO. Representa el signo de la operación efectuada. 1⇒ negativo; 0⇒ positivo.
- **V:**
FLAG DE OVERFLOW. Esta bandera sólo tiene sentido en operaciones con signo e indica que el resultado de una operación no puede representarse mediante el número de bits que quedan dedicados a la magnitud. Cuando se trabaja con signo, el MSB se dedica a representar el signo.
- **B:**
BREAK. Indica si la interrupción ha sido provocada por software (BRK) poniéndose a 1 o si ha sido provocada por hardware (IRQ) poniéndose a 0.

- ❑ **D:**
FLAG DE TRATAMIENTO ARITMÉTICO. Puesto a 1 la ALU realiza operaciones en decimal o BCD. Puesto a 0 las realiza en binario o hexadecimal.
- ❑ **I:**
FLAG DE ENMASCARAMIENTO DE INTERRUPCIONES. Puesto a 1 supone la no-aceptación de las interrupciones enmascarables (IRQ).
- ❑ **Z:**
FLAG DE CERO. Es puesto a 1 cuando el resultado de una operación es cero.
- ❑ **C:**
FLAG DE ACARREO. Es puesto a 1 cuando hay acarreo en el MSB.

9.2-8085



9.2.1-REGISTROS INTERNOS



9.2.2-MODOS DE DIRECCIONAMIENTO (8085)

INMEDIATO:

El valor del operando es el valor que se da después de la instrucción.

ADDI 32

DIRECTO:

El valor dado en la instrucción es la posición de memoria donde se encuentra el operando.

STA 21AB

POR REGISTRO:

La instrucción especifica el registro o pareja de registros donde está el operando.

ADD C

POR PAREJA DE REGISTROS INDIRECTO:

La instrucción determina la pareja de registros que contienen la dirección de memoria donde se encuentra el operando.

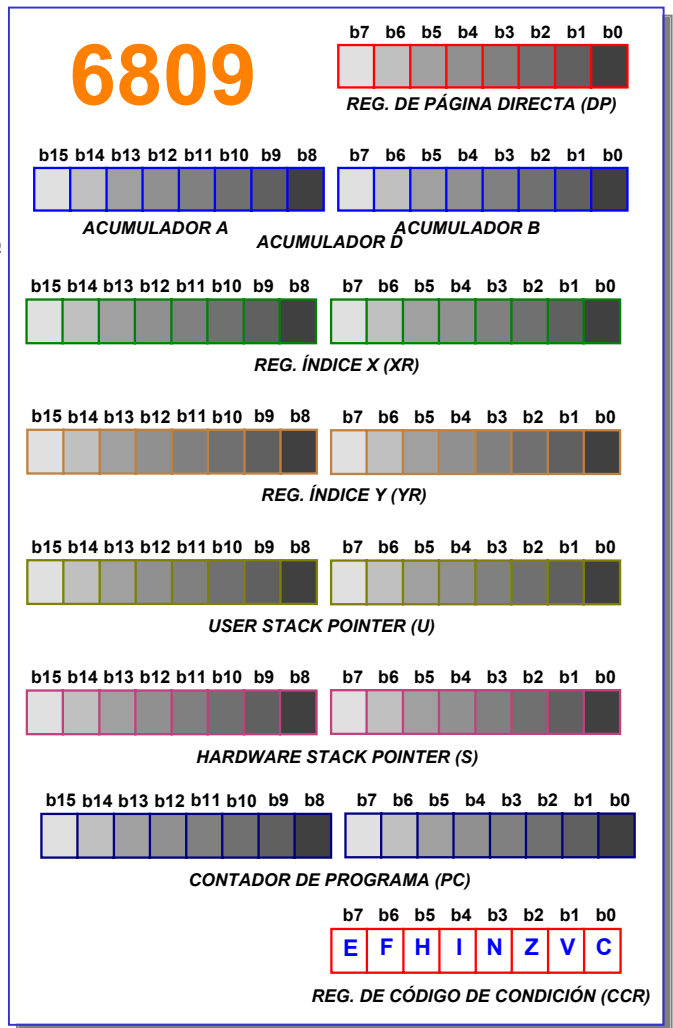
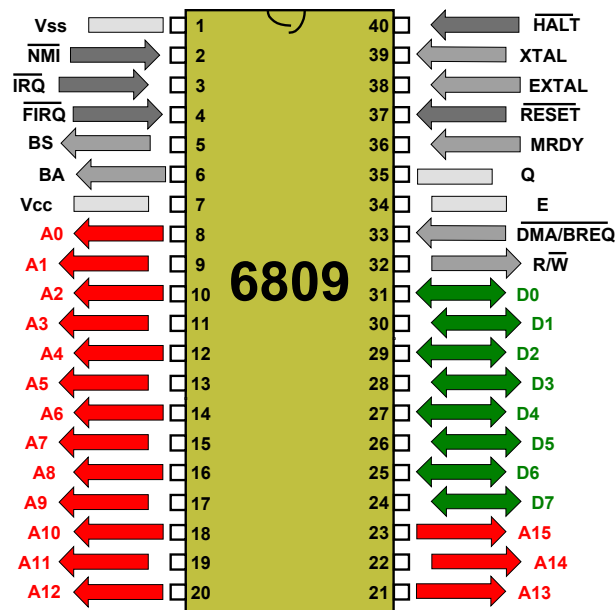
LDAX D

9.2.3-REGISTRO DE ESTADO (8085)



- ❑ **S:**
FLAG DE SIGNO. Representa el signo de la operación efectuada. 1⇒ negativo; 0⇒ positivo. En las operaciones lógicas toma el valor del BIT de más peso del resultado depositado en el acumulador.
- ❑ **Z:**
FLAG CERO. Se pone a 1 cuando el resultado de una operación ha sido cero.
- ❑ **AC:**
FLAG DE ARRASTRE AUXILIAR. Puesto a 1, indica que ha habido acarreo en el cuarto BIT. Se emplea para trabajar en BCD.
- ❑ **P:**
FLAG DE PARIDAD. Se pone a 1 cuando el número de unos en el acumulador es un número par.
- ❑ **C:**
FLAG DE ACARREO. Puesto a 1, indica que ha habido acarreo en el octavo BIT, tanto en una suma como en una resta.

9.3-6809



9.3.1-REGISTROS INTERNOS



(El acumulador D trata con los 16 bits)

9.3.2-MODOS DE DIRECCIONAMIENTO (6809)

ABSOLUTO LARGO (EXTENDIDO):

La dirección de 16 bits del operando es dada de forma explícita con los dos bytes que siguen al código de operación.

LDA \$1F4C

ABSOLUTO CORTO:

Los 8 bits menos significativos de una dirección de 16 bits son dados en el byte que sigue al código de operación. Los 8 bits más significativos están en el registro de página directa.

LDA \$4C

POR REGISTRO:

La operación especificada por la instrucción implica un operando en uno de los registros del microprocesador.

CLRA

INMEDIATO:

El valor del operando es dado explícitamente en uno o dos bytes que siguen al código de operación.

LDX #00

INDEXADO:

La dirección del operando es la suma del contenido de un registro de 16 bits (X, Y, U, S) y un valor índice que es dado por uno o dos bytes que siguen al código de operación o en uno de los acumuladores.

CLR 52, X (la posición será la X + 52)

AUTOINCREMENTO:

La dirección del operando es el contenido de un registro de 16 bits (X, Y, U, S). Después de que el operando es accedido, el valor de este registro es incrementado en 1 o 2 según la instrucción.

LDY , X+

AUTODECREMENTO:

La dirección del operando es el contenido de un registro de 16 bits (X, Y, U, S). Antes de que el operando sea accedido, el valor de este registro es decrementado en 1 o 2 según la instrucción.

STA , -X

RELATIVO:

La dirección del operando es el valor del contador de programa más un número de 8 o 16 bits con signo computado de forma interna por el ensamblador. El programador simplemente da la dirección del operando deseado o un nombre que lo representa.

BRA LOOP

INDIRECTO:

Este modo puede ser usado conjuntamente con los modos extendido, indexado, autoincremento, autodecremento y relativo. En cada caso, la dirección efectiva determinada en el modo primario de acceso, es usada para acceder a una posición de memoria que contiene la dirección del operando buscado.

CLR [,X]

9.3.3-REGISTRO DE CÓDIGO DE CONDICIÓN. CCR (6809).

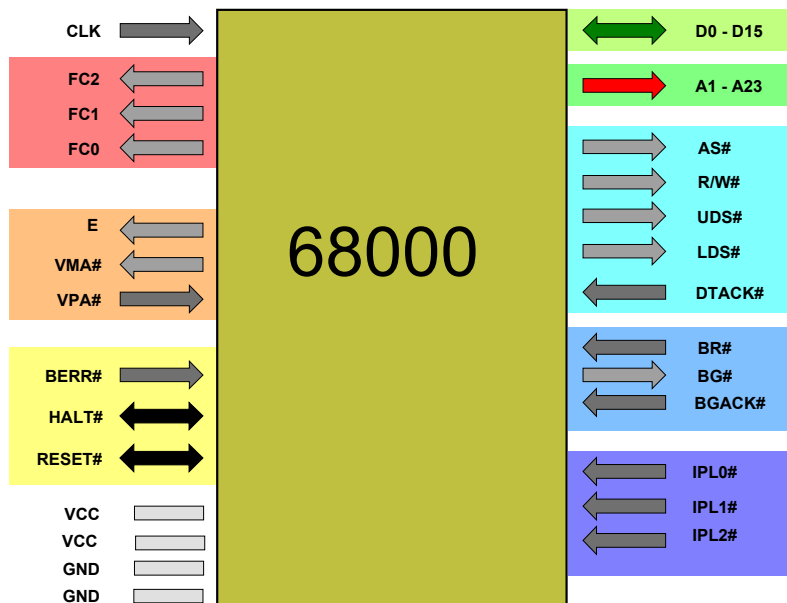
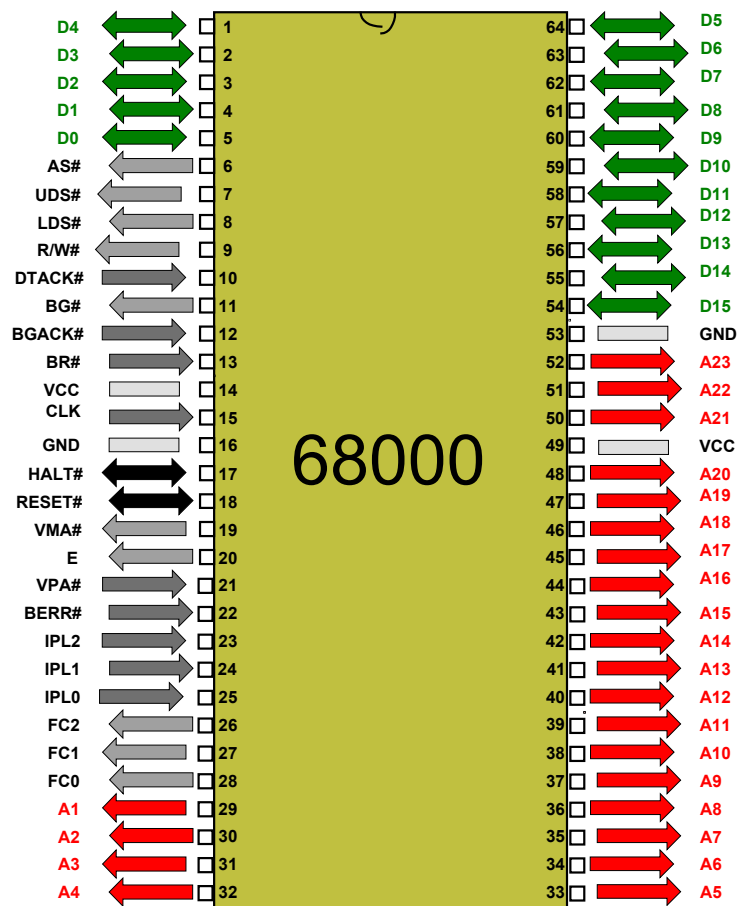


- **E:**
FLAG DE ENTERO. Es puesto a 0 cuando es aceptada una interrupción FIRQ. Para otras interrupciones es puesto a 1.
- **F:**
FLAG DE MÁSCARA DE FIRQ. Si este BIT está puesto a 1 la interrupción por la línea FIRQ no será reconocida.
- **H:**
FLAG DE ARRASTRE INTERMEDIO. Es puesto a uno cuando existe acarreo en el cuarto BIT.
- **I:**
FLAG DE MÁSCARA DE IRQ. Si este BIT está puesto a 1 la interrupción por la línea IRQ no será reconocida.
- **N:**
FLAG DE NEGATIVO. Es puesto a 1 si el resultado de una operación es negativo.

- ❑ **Z:**
FLAG DE CERO. Es puesto a 1 si el resultado de una operación es cero.
- ❑ **V:**
FLAG DE OVERFLOW Es puesto a 1 cuando el resultado de una operación aritmética es mayor en magnitud de lo que puede almacenar el registro destino especificado en la instrucción.
- ❑ **C:**
FLAG DE ARRASTRE. Es puesto a 1 cuando existe acarreo en el octavo BIT. (MSB).

9.4-μP. 16 BITS. 68000

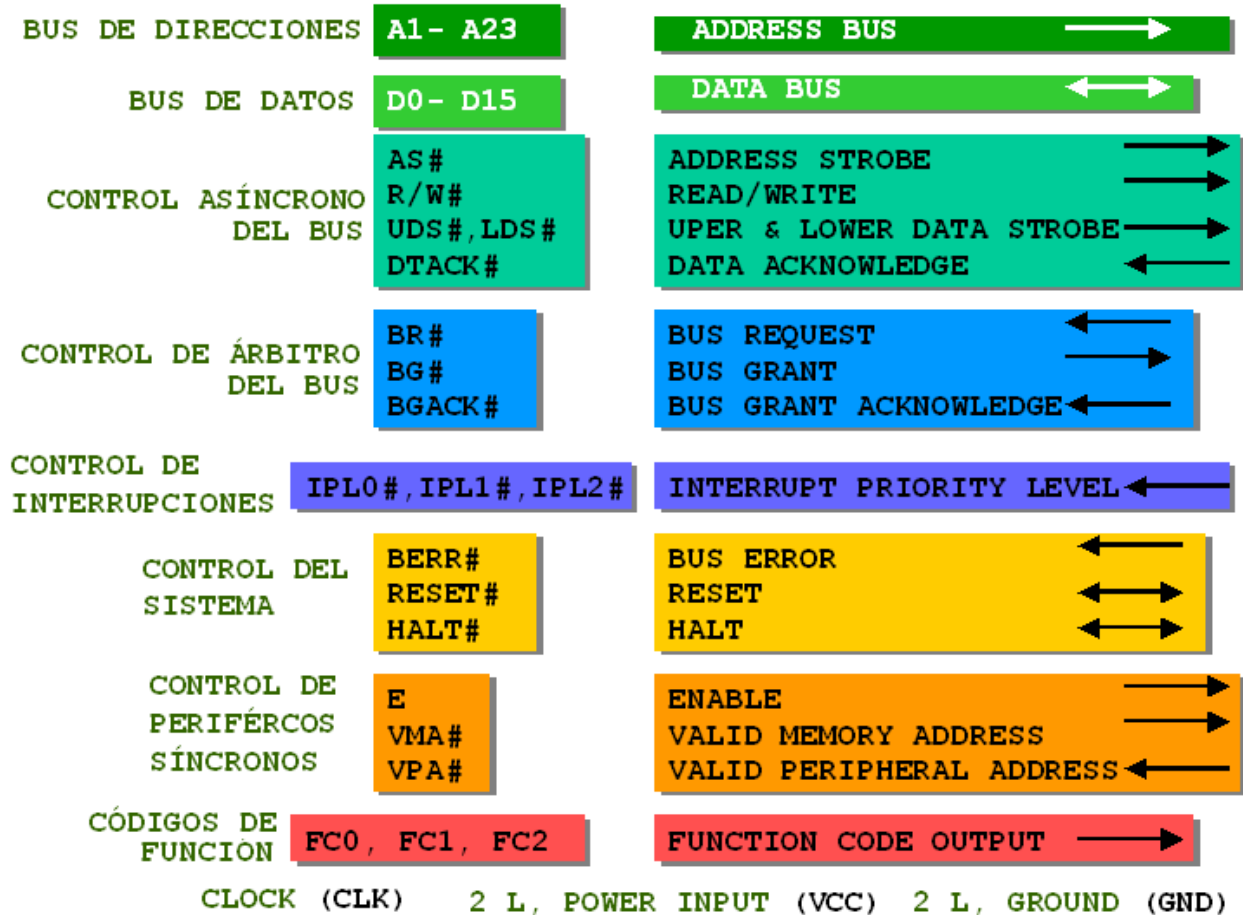
Distribución de terminales



Representación funcional

- CÓDIGOS DE FUNCIÓN
- CONTROL DE PERIFÉRICOS SÍNCRONOS
- CONTROL DEL SISTEMA
- BUS DE DATOS
- BUS DE DIRECCIONES
- CONTROL ASÍNCRONO DEL BUS
- CONTROL DE ÁRBITRO DEL BUS
- CONTROL DE INTERRUPCIONES

Denominación de los terminales:

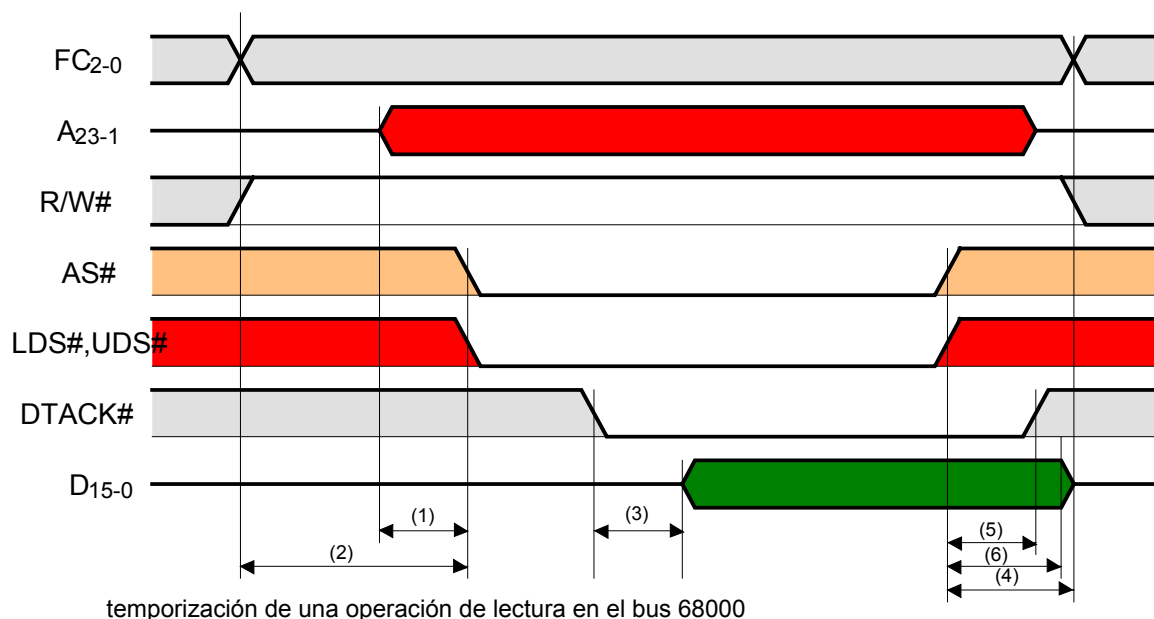


9.4.1-TERMINALES DEL 68000

A1-A23, D0-D15, AS#, R/W#, UDS#, LDS#, DTACK#.

- El 68000 trabaja con **BYTES** (8 bits), **WORDS** (16 bits) y **LONG WORDS** (32bits). Cuando tiene que transferir una palabra larga, lo hace como dos palabras consecutivas; Cuando la operación especifica *Byte* e implica una posición par del mapa de memoria, es decir, A0 = 0, solamente la señal **UDS#** es activada y la transferencia es realizada por las líneas D8 a D15. Si la operación implica una posición impar (A0 = 1), solamente la señal **LDS#** se activará, realizándose la transferencia por las líneas D0 a D7.
- Cuando la operación especifica *Word* o *Long Word*, las dos señales (LDS# y UDS#) son activadas, por lo tanto habrá transferencia por las 16 líneas del **DB**.
- El número de líneas del AB (Address Bus) es de 23 (A23-A1); Sin embargo, el µP genera direcciones de 24 bits. Las 23 líneas corresponden a los 23 bits más significativos del **AB**. La línea correspondiente a A0 se suple con las líneas UDS# y LDS#, activando una u otra o las dos, según la posición del mapa memoria desde donde o hacia donde se realiza la transferencia.
- La línea **AS#** indica, cuando está activa, que en el AB hay una dirección válida y estable. Esta línea permanece activa durante la transferencia del bus y se desactiva a la vez que las señales de *data strobe* (UDS# y LDS#) una vez que el µP ha recibido la activación de la señal **DTACK#** por parte del dispositivo direccionado o por la lógica de selección del mismo.

CICLO BUS DE LECTURA



- (1) t_{as} Tiempo desde que se activan las líneas A_{23-1} hasta que se activa $AS\#$. ($\geq 20\text{ns}$)
- (2) t_{fs} Tiempo desde que se activan las líneas FC_{0-2} hasta que se activa $AS\#$. ($\geq 50\text{ ns}$)
- (3) t_{dd} Tiempo que tarda en poner el dispositivo los datos en el DB desde que activa la línea $DTACK\#$. ($\leq 65\text{ ns}$)
- (4) t_{ah} Tiempo que mantiene el μP activas las líneas de función y dirección después de que ha desactivado las líneas de estrobe. ($\geq 20\text{ ns}$).
- (5) t_{ackh} Tiempo en el cual el dispositivo direccionado debe desactivar $DTACK\#$ después de que las señales de estrobe han sido desactivadas. ($\leq 190\text{ ns}$)
- (6) t_{dhr} Tiempo que deben ser mantenidos los datos en el bus después de desactivar las líneas de estrobe. ($\geq 0\text{ ns}$)

- El ciclo bus está definido por el tiempo y la evolución de las señales implicadas en el proceso de transferencia de información entre el μ procesador y un dispositivo conectado al bus.

Para este caso de un ciclo de lectura, al comienzo del ciclo bus, el μP activa las líneas de función, FC_2 - FC_1 - FC_0 para indicar el modo de operación y el tipo de información que se va a transferir. Al mismo tiempo, pone la línea $R/W\#$ a nivel alto indicando que la operación que va a realizar es una lectura. Más tarde, el μP pone la dirección de la posición del mapa de memoria deseada en el AB y después de un retardo activa las líneas de estrobe. Este retardo en la activación de las líneas de estrobe, es necesario para facilitar que la dirección puesta en las líneas del AB se estabilice y pueda ser considerada información válida.

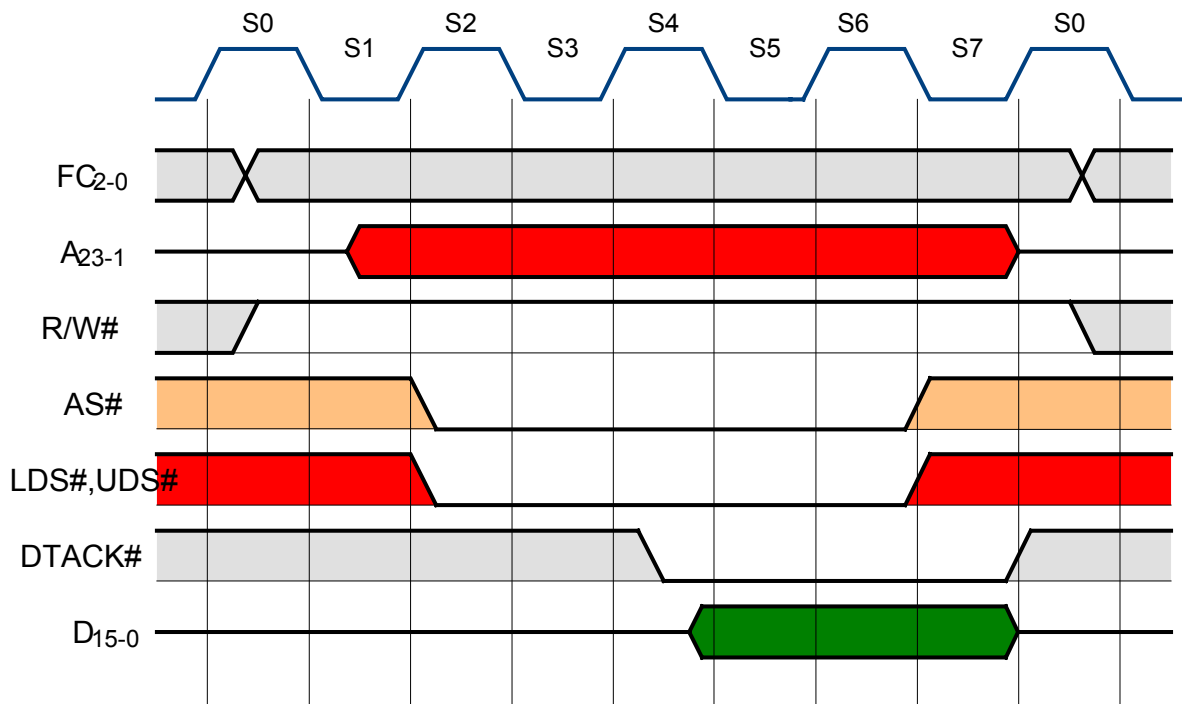
La dirección es decodificada, y el dispositivo seleccionado o los circuitos encargados de realizar la selección del mismo según los valores de las señales de dirección (lógica de selección), activa la línea de $DTACK\#$. El dispositivo pone la información solicitada, en el DB.

El μ procesador pone la información en su latch y desactiva las líneas de estrobe. $DTACK\#$ es desactivada y la información en el DB es retirada.

Las señales de FC y del AB se preparan para la realización de un nuevo ciclo.

- Aunque puede parecer que no existe ninguna restricción temporal en la activación o desactivación de algunas señales, esto no es así. Este ciclo bus se realiza durante ocho estados de la señal de reloj, y en cada uno de estos estados está definida la realización de una determinada acción por parte de alguna de las señales implicadas en el ciclo bus.
- El 68000 es un dispositivo síncrono, cuya operación es controlada por una señal de reloj **CLOCK**, y que realiza los accesos al bus de forma asíncrona.

La relación que existe entre las señales en el bus y el reloj del μP es la mostrada en la figura siguiente.



- Los fenómenos que suceden en cada uno de los estados son los siguientes:

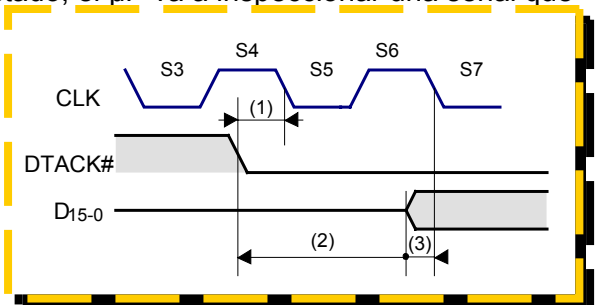
Estado 0 (S0): Comienza el ciclo de lectura. El μ procesador sitúa en las líneas de función (FC0, FC1, FC2) un código válido y activa la línea de R/W# para la realización de una lectura del bus.

Estado 1(S1): A la entrada de este estado, pone el valor contenido en buffer de direcciones interno en el AB. Es decir, sitúa una dirección en el bus de direcciones.

Estado 2 (S2): En el flanco de subida de este estado, el μP activa las líneas de estrobe, es decir, AS#, LDS# y UDS#.

Estado 3 (S3): Durante este estado no se altera ninguna señal. Es un estado de retardo.

Estado 4 (S4): En el flanco de bajada de este estado, el μP va a inspeccionar una señal que define la terminación de ciclo, (DTACK# o BERR#) o VPA#. Si esta última ha sido activada, el ciclo se convierte en un ciclo de periférico. Por otro lado, si ninguna de las dos señales, DTACK# o BERR# está activa, el μ procesador va a insertar estados de espera en forma de periodos enteros de reloj, hasta que cualquiera de las dos señales sea activada.



Para que el estado activo de la señal de DTACK# o BERR# sea reconocido dentro de S4, debe haberse producido un tiempo (1) anterior a la inspección de las líneas, pues si no es así, el estado activo no será reconocido y serán insertados estados de espera.

Estado 5 (S5): Durante este estado, igual que en S3, tampoco se altera ninguna señal. Es otro estado de retardo

Estado 6 (S6): En el flanco de bajada la información presente en el bus de datos es llevada al buffer de datos del μ P. Para que los datos sean cargados correctamente en el buffer, éstos deben estar presentes en el DB un tiempo (3) anterior a su carga al final de S6. Esto quiere decir que los datos tienen que ser puestos en el DB por el dispositivo leído en un tiempo no superior a (2).

Estado 7 (S7): A la entrada de este estado, el μ procesador, desactiva las líneas de estrobe (AS#, UDS# y LDS#). En el flanco de subida, el bus de direcciones es puesto en estado de alta impedancia, con lo cual se desactiva DTACK# o BERR#.

- La línea de BERR# será activada por algún dispositivo encargado de vigilar (*Watchdog*) que todo en el bus vaya correctamente. Si un dispositivo direccionado, no entrega la señal de DTACK# dentro del tiempo establecido por los estados de reloj, el μ P inserta estados de espera. Para limitar el tiempo en el cual el sistema está en espera, el watchdog, después de un tiempo prefijado y si antes no se ha activado DTACK#, activará la señal BERR# que da paso a una interrupción.

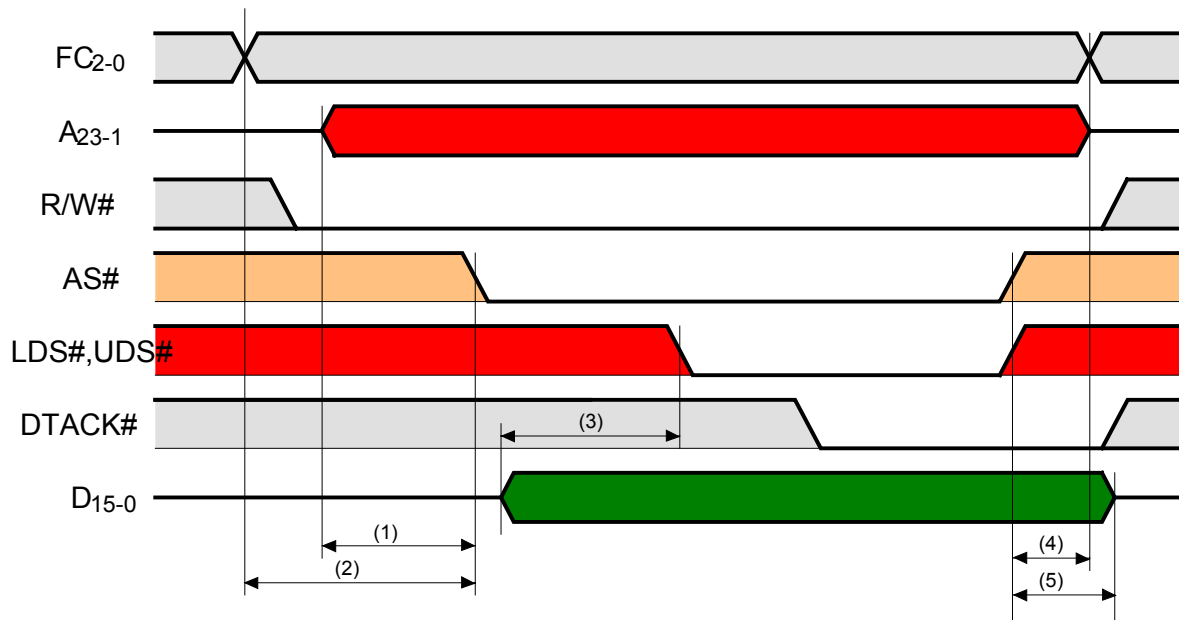
μ P: “quiero leer información de este dispositivo y de esta posición determinada (AS#, LDS# UDS#)”. DISPOSITIVO: “de acuerdo, (DTACK#) toma la información (DB)”.

Esta forma de trabajo del bus en el cual se produce una especie de diálogo entre el μ P y las interfaces de los dispositivos representa un esquema asíncrono de acceso al bus, con los matices explicados anteriormente.

Las siguientes figuras representan otros tipos de ciclo bus: El Ciclo Bus de Escritura también desarrollado en ocho estados de reloj y el de Lectura-Modificación-Escritura que se desarrolla en 20 estados.

- Este último ciclo (Read-Modify-Write) es usado únicamente por la instrucción TAS (test and set). Consiste en una operación de lectura, modificación del dato en la ALU y una escritura del dato modificado en la misma dirección. La línea de AS# permanece activa durante todo el ciclo, haciendo que éste sea indivisible. Esta operación es útil en sistemas multiprocesadores donde la memoria principal es compartida entre varios μ P.

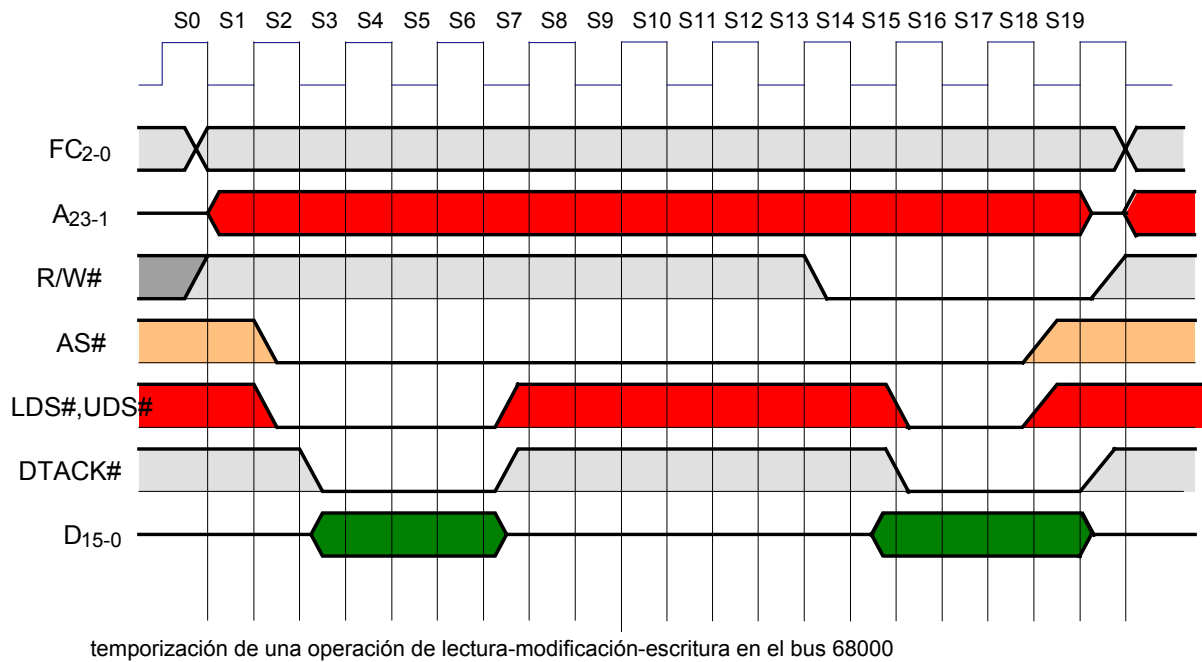
CICLO BUS DE ESCRITURA.



(1) ≥ 20 ns (2) ≥ 50 ns (3) ≥ 20 ns (4) ≥ 20 ns (5) ≥ 20 ns

CICLO BUS DE LECTURA-MODIFICACIÓN-ESCRITURA.

□ La figura representa también la relación temporal con los estados de reloj.



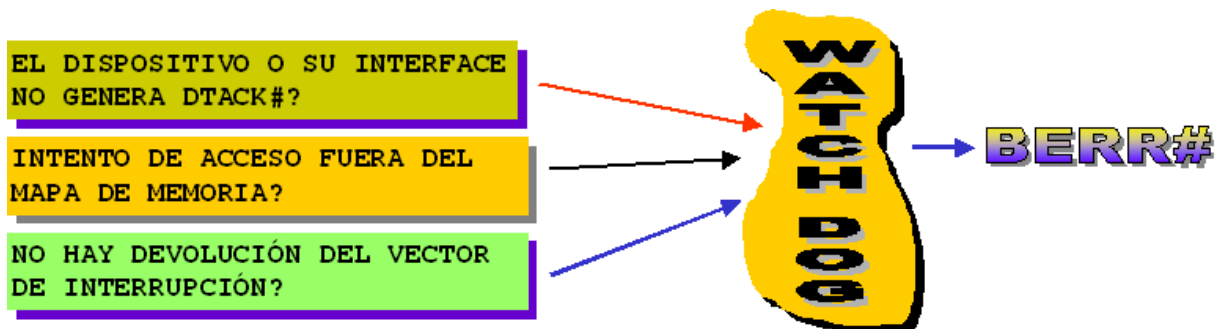
9.4.2-LÍNEAS DE FUNCIÓN. F0-F2.

FC2-FC0	ACCIÓN
000	USO FUTURO
001	DATOS DE PROGRAMA EN MODO USUARIO
010	INSTRUCCIONES DE PROGRAMA EN MODO USUARIO
011	USO FUTURO
100	USO FUTURO
101	DATOS DE PROGRAMA EN MODO SUPERVISOR
110	INSTRUCCIONES DE PROGRAMA EN MODO SUPERVISOR
111	RECONOCIMIENTO DE INTERRUPCIÓN

- La información codificada en estas tres líneas muestra el modo de trabajo del micro (*usuario o supervisor*) y el tipo de espacio de direcciones que está siendo accedido. Estos códigos son válidos siempre y cuando la línea AS# sea activa.

BUS ERROR. BERR#.

- Cuando un dispositivo que ha sido direccionado, no activa la línea de DTACK#, bien debido a un fallo en el mismo dispositivo, o bien a un fallo en la lógica de selección, hace que el μP introduzca ciclos de espera de forma indefinida. Para evitar esta situación, se conecta un dispositivo (*Watchdog*) al bus que detecta cuando hay un fallo de generación de DTACK#, activando la línea BERR#. La activación de esta línea lleva a una excepción que sacará al μP del estado de error.
- En el diseño del dispositivo mencionado anteriormente, hay que tener en cuenta otro tipo de situaciones que deben activar la línea de BERR#. Estas situaciones pueden ser el intento de acceso a posiciones de memoria fuera del mapa, la no-devolución del nº de vector de interrupción cuando sea requerido, y cualquier otro error dependiente de una aplicación.



RESET#.

- Cuando el programa ejecuta la instrucción *RESET*, esta línea es de salida, poniéndose a nivel bajo. El tiempo durante el cual esta línea está activa es de 124 ciclos de reloj. Con esta función, se trata de inicializar todos los elementos conectados al sistema, en concreto aquellos que tengan su terminal RESET# conectados con el del μP . Esta instrucción no afecta al estado interno de éste.
- Cuando se utiliza como entrada, este terminal inicializa el equipo, incluido el estado interno del μP . Para desencadenar este tipo de inicialización, deben activarse al mismo tiempo las líneas de RESET# y HALT#. La activación simultánea de estas líneas origina una excepción con número de vectores 0 y 1.
- La activación de estos terminales se realiza por dos circunstancias:

1º- Cuando se alimenta el sistema. En este caso la activación la realiza de forma automática el circuito diseñado para tal fin.

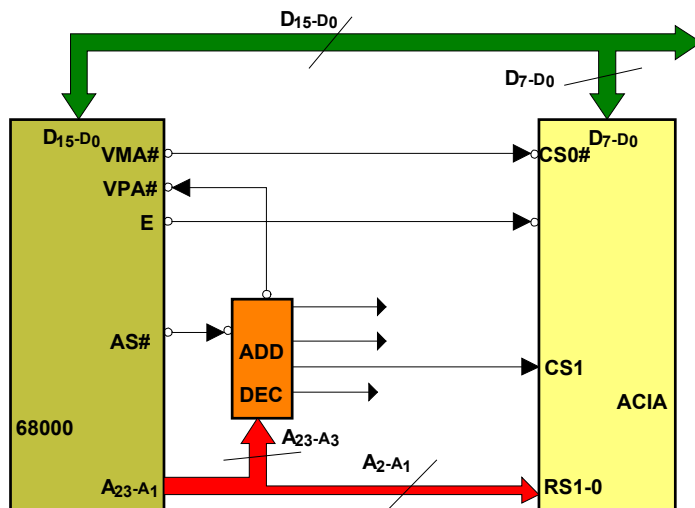
2º- Cuando el sistema deja de funcionar o funciona de forma descontrolada. En este caso es el usuario quien activa los terminales mediante un pulsador que se implementa en el propio sistema.

HALT#.

- ❑ Cuando es activada de forma externa, hace que el μP pare toda actividad del bus al finalizar el ciclo bus en curso; todas las líneas de control pasan a su estado inactivo y todas las líneas triestado pasan a su estado de alta impedancia.
- ❑ Se activa de forma interna, cuando el μ procesador está ejecutando instrucciones del tipo de atención a una excepción originada por la activación de BERR#, y en al mismo tiempo se produce otro BERR#. (En este caso el sistema queda fuera de control.)

E, VMA#, VPA#.

- ❑ Estas señales sirven para poder conectar al bus, dispositivos síncronos de I/O de la familia 6800.



- ❑ El μP entrega de forma continua una señal de reloj por el terminal **E** (Enable) cuya frecuencia es 1/10 la del reloj conectado al terminal CLK (6 periodos de CLK en bajo y 4 periodos de CLK en alto). Esta señal es la usada para el sincronismo de los dispositivos I/O.

- ❑ El decodificador de direcciones o la lógica de selección, reconoce todas las direcciones que corresponden a dispositivos síncronos. Cuando

se direcciona uno de estos dispositivos y se valida la dirección con AS#, la lógica de selección, además de seleccionar el dispositivo requerido, activa la línea **VPA#** (Valid Peripheral Address), indicando al μP que este dispositivo necesita una operación síncrona, por lo tanto, no debe esperar una señal DTACK# por parte del mismo. Como respuesta, el μP activa la línea **VMA#** (Valid Memory Address), indicando con ello al dispositivo periférico direccionado, que la dirección en el AB es válida y que el procesador está sincronizado con la señal **E**.

IPL0#, IPL1#, IPL2#

- ❑ Estas entradas, indican en forma codificada desde 0 hasta 7, el nivel de prioridad de una interrupción solicitada por un dispositivo. Cuando en estas líneas aparece codificado el nivel 7 (111) quiere decir que el dispositivo que solicita la interrupción debe ser atendido. Si en estas líneas aparece codificado el 0 (000) quiere decir que ningún dispositivo solicita interrupción.

BR#, BG#, BGACK#.

- Estas señales sirven para determinar qué dispositivo se hace dueño del bus (*Bus Master*).

BR#: La activación de esta señal indica al μ procesador, que otro dispositivo solicita el control de los buses.

BG#: El μ P activa esta señal para indicar a los posibles peticionarios *bus masters*, que cuando finalice el ciclo bus en curso, cederá el control de los buses.

BGACK#: La activación de esta señal indica al μ P que otro dispositivo es el *bus master*. Esta señal no debe activarse hasta que se cumplan las siguientes condiciones:

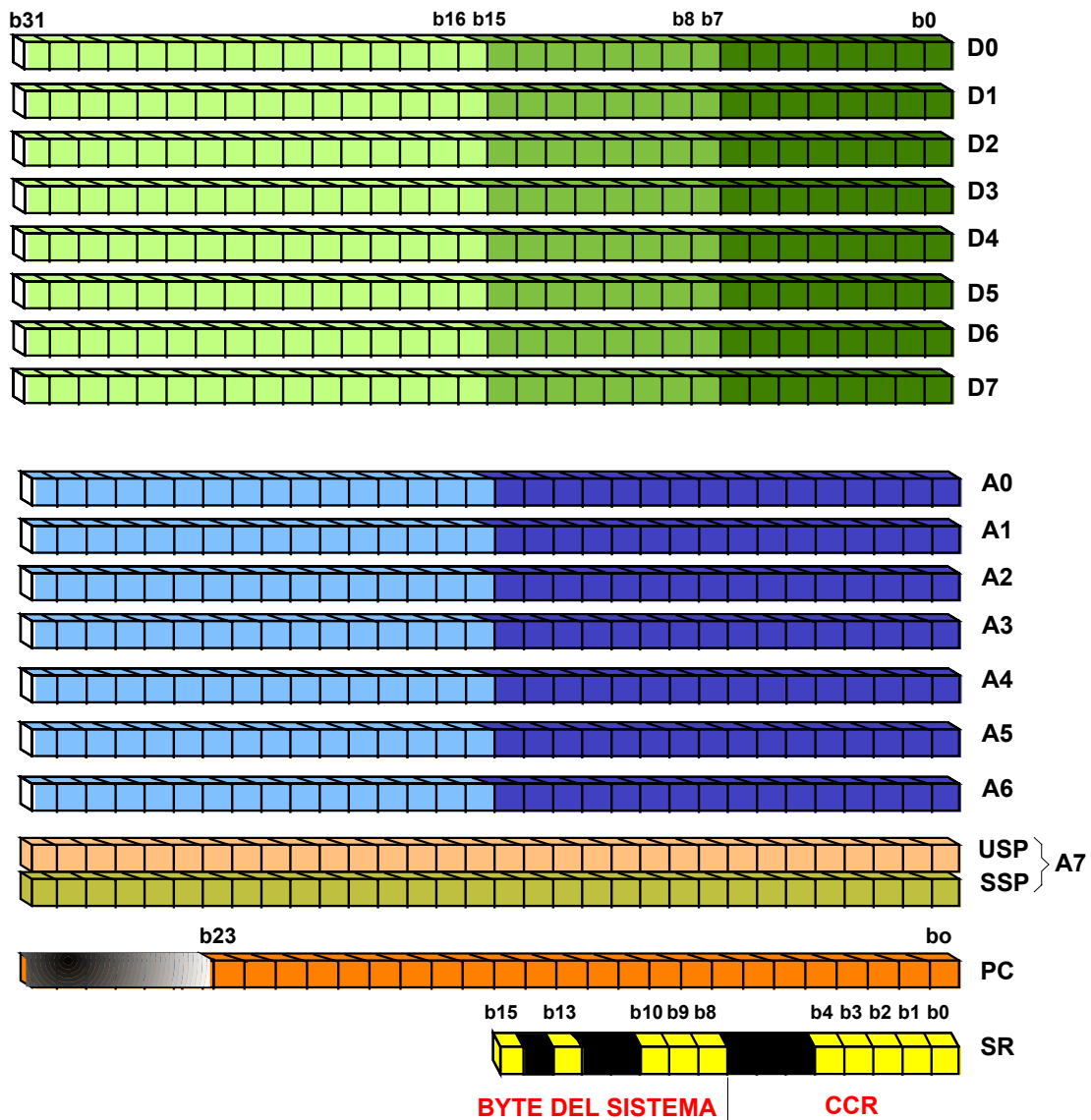
BG# ha sido recibida.

AS#, es inactiva. Esto indica que el μ P no está usando el bus.

DTACK# es inactiva, lo que indica que ni la memoria ni un periférico están haciendo uso del bus.

BGACK# lógicamente es inactivo con lo cual ningún otro dispositivo está solicitando hacerse con el control de los buses.

9.4.2-68000. REGISTROS

**REGISTROS DE DATOS: (Dn). D0-D7.**

- Pueden operar con datos de tres tamaños:

BYTE (B), desde el b₀ al b₇.

WORD (W), desde b₀ hasta b₁₅.

LONG WORD (LW). b₀ hasta b₃₁.

- El tamaño de operando con el que debe trabajar una instrucción se especifica poniendo en la propia instrucción un punto seguido de B para trabajar con tamaño byte, punto y W para el trabajo con Word y punto y L para el trabajo con Long Word. Si no se indica el tamaño del dato, el ensamblador por defecto toma el tamaño Word.

Ej.

Move.B.....; Move.W.....; Move.L.....

- ❑ Cuando se trabaja con operandos de tamaño *Byte* o *Word*, los bytes de mayor orden no se ven afectados.

Por otro lado, una instrucción sí puede especificar que se haga extensión del BIT de signo. La extensión de signo consiste en copiar el MSB (b7 en el dato tamaño *Byte* y b15 en el de tamaño *Word*) en los lugares que quedan hacia la izquierda, hasta el BIT 31.

REGISTROS DE DIRECCIONES: (An) A0-A7.

- ❑ Trabajan con 16 o 32 bits de longitud. Los registros A0 a A6, guardan información de direccionamiento. Esta información puede ser direcciones de memoria, direcciones de I/O o valores índices para ser usados en direccionamientos indexados. Cuando son utilizados para direccionar una posición del mapa, el byte más alto es ignorado.
- ❑ El registro A7 sirve de puntero de pila (Stack Pointer). Su funcionamiento está asociado con los dos modos de operación (*modo supervisor* y *modo usuario*). En el modo supervisor, todas las instrucciones máquina pueden ser ejecutadas y el SR puede ser accedido en todos sus bits. En este modo el A7 se refiere al *SUPERVISOR STACK POINTER* (SSP). En modo usuario, algunas instrucciones no pueden ser ejecutadas, ni todos los bits del SR pueden ser accedidos. El A7 se refiere en este modo al *USER STACK POINTER* (USP). En ningún caso es posible trabajar sobre los dos registros de forma simultánea, por eso aunque son dos, sólo tienen una referencia.
- ❑ Cuando un registro de direcciones es el operando destino, todo él se ve afectado, de tal manera, que si la operación es con un operando de tamaño **.W**, se realiza extensión de signo en el registro.

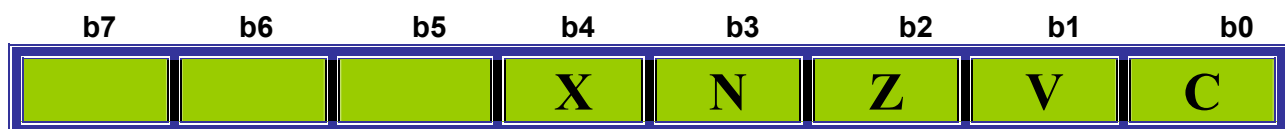
CONTADOR DE PROGRAMA: (PC)

- ❑ Es un contador de 32 bits pero solo los 24 bits menos significativos son usados para poner la dirección en el *address bus*.

STATUS REGISTER: (SR)

- ❑ El byte de menor orden (b₀ a b₇), se llama *Condition Code Register, CCR (REGISTRO DE CÓDIGOS DE CONDICIÓN)*. El byte de mayor orden (b₈ a b₁₅), se llama *Byte del Sistema*.

REGISTRO DE CÓDIGOS DE CONDICIÓN CCR. (68000 SR)

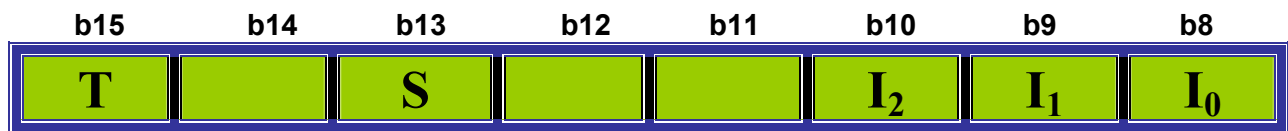


(PUEDE SER MANIPULADO EN LOS MODOS USUARIO Y SUPERVISOR)

- ❑ **X:**
EXTEND. Es usado en la implementación de operaciones aritméticas de precisión múltiple (≥ 64 bits). Toma el valor del flag C salvo en algunas instrucciones.
- ❑ **N:**
NEGATIVE. Es una copia del MSB (b7, b15 o b31 depende del tamaño del operando) resultante de todas operaciones aritméticas y lógicas. Cuando se trabaja con signo, indica resultado negativo.

- **Z:**
ZERO. Es puesto a 1 si el resultado de una operación o de una transferencia (*MOVE*) es cero.
- **V:**
OVERFLOW. Es puesto a 1 cuando el resultado de una operación aritmética con signo, es mayor en magnitud de lo que se puede almacenar con el número de bits dedicados a representar dicha magnitud. Si se trabaja sin signo, este flag no tiene significado. También es afectado por determinadas instrucciones.
- **C:**
CARRY. Es puesto a 1 cuando existe acarreo en el MSB; El resultado supera el número de bits de los operandos, ambos del mismo tamaño (8, 16 ó 32 bits). También es afectado por determinadas instrucciones.

BYTE DE SISTEMA (68000 SR).



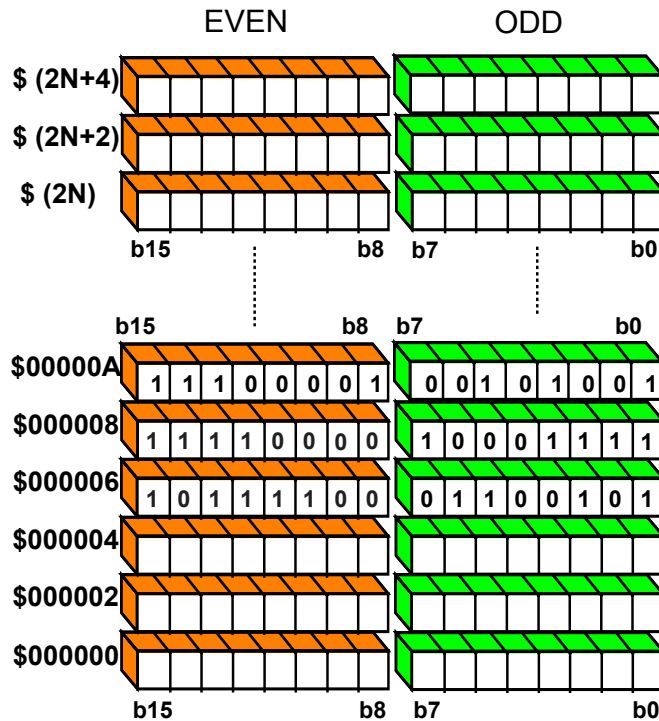
(SOLO PUEDE SER MANIPULADO EN MODO SUPERVISOR)

- **T:**
TRACE MODE. Si está puesto a 1 el programa se ejecuta en modo traza. En este modo, se produce una excepción después de cada instrucción. La excepción puede ser aprovechada para ejecutar una rutina que muestre la evolución de los registros o memoria con cada instrucción, de esta manera se puede hacer una depuración del programa.
- **S:**
SUPERVISOR STATE. Si este BIT está puesto a 1 el modo de trabajo del μP es el de supervisor; en caso contrario es el de usuario. Hay que tener en cuenta por lo tanto que si el μ procesador está ejecutando un programa en modo usuario, solamente una excepción puede cambiar el modo de privilegio.
- **I₂, I₁, I₀:**
INTERRUPT MASK. Una interrupción será atendida si su nivel de interrupción dado por el código presente en las líneas IPL0#-IPL2#, es superior al código binario dado por estos tres bits. Si el nivel de la interrupción es 7, esta se atiende, independientemente del valor de la máscara.

9.4.3-ORGANIZACIÓN DE LA MEMORIA.

- El 68000 es un μP de direccionamiento a byte y cuyo rango va desde 0 a $2^{24}-1$, es decir, 16MB. Como su bus de datos es de 16 bits, es conveniente pensar que la memoria está compuesta por palabras de 16 bits donde cada palabra tiene una dirección par (even).
- Un dato de tamaño BYTE puede ser guardado en cualquier posición direccionable del mapa de memoria, ya sea ésta par (even) o impar (odd).
- Si el tamaño es WORD, éste ocupa dos posiciones consecutivas de la memoria y sólo puede ser guardado en una dirección par de la memoria.

La dirección de la posición que contiene la palabra, es la de aquella que contiene el byte de mayor orden. El byte de menor peso se almacena en la dirección más alta, que será siempre una dirección impar.



- Cuando el operando es de 32 bits (LONG WORD), ocupa cuatro posiciones consecutivas de la memoria. En este caso también, este dato sólo puede ser guardado en una posición cuya dirección sea par. La dirección de la posición que contiene la *palabra larga*, es la de la posición que contiene el byte de mayor orden (b_{23} a b_{31}) y el byte de menor peso (b_0 a b_7) se guarda en la dirección más alta de las cuatro que contienen la LW.
- Cuando se trabaja con la pila (STACK) que es una zona determinada de la memoria principal, los datos serán siempre guardados en posiciones pares, incluso si se opera con tamaño byte.

Ej.

Tomando los valores representados en la figura, los operandos que tenemos desde la dirección \$000006 hasta la \$00000B son: [valor almacenado(dirección)]

Tamaño byte: \$BC (\$000006), \$65 (\$000007), \$F0 (\$000008), \$8F (\$000009), \$E1 (\$00000A), \$29 (\$00000B).

WORD: \$BC65 (\$000006), \$F08F (\$000008), \$E129 (\$00000A)

LONG WORD: \$BC65F08F (\$000006), \$F08FE129 (\$000008)

9.4.4-MODOS DE DIRECCIONAMIENTO.

DIRECTO.

- ❑ **DIRECTO A REGISTRO DE DATOS, *Dn*:** El operando (S, D) está contenido en un registro de datos.

MOVE.W D2,D3

- ❑ **DIRECTO A REGISTRO DE DIRECCIONES, *An*:** El operando (S, D) está contenido en un registro de direcciones.

SUBA.L D0,A2

ABSOLUTO.

- ❑ **ABSOLUTO LARGO, *Abs.L*:** La dirección efectiva del operando (S, D) se da en la instrucción. Debe expresarse mediante long word. (El byte más significativo se ignora).

MULS \$25022,D2

- ❑ **ABSOLUTO CORTO, *Abs.W*:** Como el anterior, pero el argumento es una dirección de tamaño word. La dirección efectiva se obtiene haciendo extensión de signo de la palabra operando.

EORI.B #\$FF,\$8000

RELATIVO (A PC).

- ❑ **RELATIVO CON DESPLAZAMIENTO, *d(PC)*:** La dirección del operando (S) se obtiene sumando el contenido del contador de programa y un desplazamiento de 16 bits con signo indicado en la instrucción.

LEA \$200(PC), A4

- ❑ **RELATIVO CON ÍNDICE Y DESPLAZAMIENTO, *d(PC,Xi)*:** La dirección efectiva del operando fuente se obtiene sumando al contador de programa el contenido del registro empleado en la instrucción y un desplazamiento de 8 bits con signo.

MOVE \$B(PC,D1), D5

INDIRECTO.

- ❑ **INDIRECTO POR REGISTRO, *(An)*:** La dirección de memoria donde se encuentran los operandos (S, D) está en un registro de dirección especificado en la instrucción.

ASL.W (A3)

- ❑ **INDIRECTO CON POSTINCREMENTO, *(An)+*:** Igual que el anterior pero después de la obtención del argumento el registro se incrementa en 1 (si el dato es byte), 2 (si es word) o 4 (si es long word). El A7 incrementa en 2 aunque se trabaje con byte.

CMPM (A0)+,(A4)+

CONTINÚA->

- ❑ **INDIRECTO CON PREDECREMENTO, $-(An)$:** Como el indirecto por registro pero en este caso, primero se decrementa el contenido del registro en 1, 2 o 4 (según la longitud del dato) y después se obtiene el operando. El A7 decrementa en 2 aunque se trabaje con byte.

OR.B D3,-(A0)

- ❑ **INDIRECTO CON DESPLAZAMIENTO, $d(An)$:** La dirección efectiva del dato se determina sumando el contenido del registro de dirección con un desplazamiento de 16 bits con signo.

CHK \$24(A4),D3

- ❑ **INDIRECTO CON ÍNDICE Y DESPLAZAMIENTO, $d(An, Xi)$:** La dirección efectiva se calcula sumando, el contenido del registro de dirección más el contenido de un registro de datos (W, LW) o de dirección índice, más un desplazamiento de 8 bits con signo.

SUB.W 2(A0,D1.W),D2

INMEDIATO.

- ❑ **INMEDIATO, $Immed$:** El operando fuente es el dato a utilizar. Forma parte de la instrucción y está situado en memoria siguiendo al código de operación. El símbolo '#' precede al operando.

ANDI.B #\$9A,D5

- ❑ **INMEDIATO RÁPIDO, $Immed$:** Como el anterior, pero el dato es un valor comprendido entre 1 y 8 (3 bits, donde 000 denota el 8) para instrucciones de suma y resta, o entre -128 y 127 (8 bits en complemento a 2) para la instrucción de movimiento de datos.

ADDQ.W #6,A0

IMPLICITO.

- ❑ **REGISTRO IMPLICITO:** El operando va implícito en el código de operación de la instrucción. El *op. code* varía dependiendo del registro al que se hace referencia, SR, USP, SSP, CCR O PC.

MOVE \$17122,CCR

9.4.5-SET DE INSTRUCCIONES. (NOTAS)

(Ver set de instrucciones)

Como complemento al set de instrucciones, se presentan a continuación algunas explicaciones sobre determinadas instrucciones. Las instrucciones no explicadas en este apartado, en principio no deben presentar problema de comprensión alguno.

INSTRUCCIONES DE RAMIFICACIÓN (BRANCH INSTRUCTIONS).

- Estas instrucciones pueden usar un offset de 8 o 16 bits; Este offset es un número dado en complemento a 2, y especifica la distancia en bytes entre el valor actual del PC y la dirección de la instrucción hacia donde se hace el salto. (*BRA, JMP, BSR, JSR, RTS, RTR*)

JMP A_{eff} , **BRA** label **BSR** label **JSR** A_{eff} **RTS** **RTR**

- En las ramificaciones condicionales, habrá ramificación siempre que se cumpla la condición especificada. La comprobación de la condición se realiza testeando los flags del CCR según las operaciones indicadas en la tabla.

Bcc (Branch Conditionally).

Si cc se cumple hay un salto a la dirección indicada en el operando de la instrucción.

BGT label

DBcc (Decrement and Branch Conditionally).

Se testea la condición especificada por el sufijo "cc". Si la condición es satisfecha, la instrucción que sigue a *DBcc* se ejecuta normalmente. Si esta condición no es satisfecha el valor contenido en el registro D_n se decrementa en una unidad. Si el valor resultante en $D_n \geq 0$ hay un desplazamiento dado por 16 bits con signo, al lugar indicado en la instrucción. Si el valor resultante en D_n es < 0 se ejecuta la instrucción que sigue a *DBcc*, como en el primer caso.

DBGD D_n , label

SUFJO cc	NOMBRE	TEST INTERNO
T	True	Always true
F	Fals	Always false
HI	High	$C \text{ OR } Z = 0$
LS	Lower or Same	$C \text{ OR } Z = 1$
CC	Carry Clear	$C = 0$
CS	Carry Set	$C = 1$
NE	Not Equal	$Z = 0$
EQ	Equal	$Z = 1$
VC	Overflow Clear	$V = 0$
VS	Overflow Set	$V = 1$
PL	Plus	$N = 0$
MI	Minus	$N = 1$
GE	Greater or Equal	$N \text{ X-OR } V = 0$
LT	Less Than	$N \text{ X-OR } V = 1$
GT	Greater Than	$Z \text{ OR } (N \text{ X-OR } V) = 0$
LE	Less or Equal	$Z \text{ OR } (N \text{ X-OR } V) = 1$

T y F no pueden ser usados en la instrucción Bcc

INSTRUCCIONES DE COMPARACIÓN.

- La comparación siempre se realiza mediante la resta del operando destino y el operando fuente (destination – source), según los modos de direccionamiento. El resultado de esta resta no se lleva a ningún sitio. El propósito de estas instrucciones es el de actuar sobre los flags del CCR para que según el valor que tomen, reflejen la relación entre los operandos comparados. (*CMP, CMPA, CMPI, CPM*)

CMP A_{eff} , D_n **CMPA** A_{eff} , An **CMPI** #Imm, A_{eff} **CPM** (An) + , (An)+

INSTRUCCIONES GENERADORAS DE EXCEPCIONES:

- Cuando estas instrucciones son ejecutadas, generan una excepción, es decir, se termina momentáneamente la ejecución del programa en curso para llamar a una rutina que preste su atención a la instrucción ejecutada. (*CHK*, *TRAPV*, *TRAP*).

CONDICIONALES:

CHK, Compara el operando fuente con el contenido de un registro de datos que se especifica como destino. Si el contenido del registro es menor que cero, o mayor que el valor dado en la dirección efectiva, se genera excepción con vector nº 6.

CHK A_{eff} , Dn

TRAPV, Si cuando se ejecuta esta instrucción el flag V del CCR está puesto a 1, se genera una excepción con vector nº 7. (la rutina de atención a esta excepción tratará de avisar de la situación de overflow).

TRAPV

INCONDICIONAL:

TRAP, Cuando se ejecuta esta instrucción, se genera una excepción cuyo nº de vector se obtiene mediante la suma del número de 4 bits dado en la instrucción de modo inmediato(#) y el valor 32. Los vectores de excepción reservados para esta instrucción son por lo tanto los vectores que van desde el número 32 al 47.

TRAP #Inm

INSTRUCCIONES PRIVILEGIADAS.

- Estas instrucciones solamente pueden ser ejecutadas cuando el μ procesador se encuentra en modo Supervisor. Si se intenta su ejecución en modo usuario, originan una excepción con vector nº 8. (*ANDI*, *EORI*, *MOVE*, *ORI*, *RESET*, *RTE*, *STOP*)

ANDI, *EORI*, *ORI* y *MOVE* son privilegiadas cuando el destino es el SR.

ANDI #Imm, SR, EORI #Imm, SR MOVE A_{eff} , SR ORI #Imm, SR

MOVE Cuando transfiere el contenido del USP desde o hacia un An, estando en el modo supervisor.

MOVE USP, An MOVE An, USP

RESET.

- Cuando se ejecuta esta instrucción, activa el terminal RESET# del μ P. Esta activación permite reinicializar todos los dispositivos que tengan su línea reset conectada a la del μ procesador, pero sin afectar a éste, más que para incrementar el PC en dos, y apuntar de esta manera a la siguiente instrucción.

RESET

RTE.

- Esta instrucción es la que lleva desde la rutina de atención a una excepción (exceptuando RESET), al programa que estaba en curso cuando se produjo la excepción. Va siempre al final de la subrutina. Esta instrucción recupera de forma automática de la pila, el PC y el SR.

RTE

STOP.

- ❑ Carga el SR con el valor inmediato dado en la instrucción y detiene la ejecución del μP . Sólo puede salir de este estado mediante una excepción de *traza*, de *interrupción* o de *reset*.

STOP #Imm

INSTRUCCIONES DE TESTEO DE BITS.

- ❑ Estas instrucciones operan sobre bits individuales. La localización del BIT sobre el que se opera viene dado en forma inmediata (bit#) o en un registro de datos (Dn). (*BCHG*, *BCLR*, *BSET*, *BTST*).
- ❑ La operación que se realiza es la siguiente: El complemento del BIT designado se pasa como flag Z, entonces la operación designada por la instrucción se ejecuta sobre el valor original del BIT.

BCHG complementa el valor del BIT indicado.

BCHG Dn, A_{eff} **BCHG #Imm, A_{eff}**

BCLR pone el BIT indicado a 0

BCLR Dn, A_{eff} **BCLR #Imm, A_{eff}**

BSET pone el BIT indicado a 1

BSET Dn, A_{eff} **BSET #Imm, A_{eff}**

BTST no realiza operación, más que la de pasar el complemento del BIT al flag Z.

BTST Dn, A_{eff} **BTST #Imm, A_{eff}**

INSTRUCCIONES DE DIVISIÓN.

- ❑ Existen dos instrucciones para la división. *DIVS* (división con signo) y *DIVU* (división sin signo). Los operandos sin signo, son tratados como positivos, usando todos los bits para representar la magnitud.

El dividendo (destino), es un operando Long Word y está en un registro de datos (Dn). El divisor (fuente) es un operando formato Word. El resultado de la operación consiste en un cociente con o sin signo que ocupa los 16 bits menos significativos del registro de datos (Dn), y un resto (también con o sin signo) que ocupa los 16 bits más significativos. El signo del resto es el mismo que el del dividendo.

RESTO	COCIENTE	Dn

DIVS.W A_{eff}, Dn DIVU.W A_{eff}, Dn

OTRAS INSTRUCCIONES

MOVEM.

- ❑ Transfiere el contenido de uno o más registros hacia, o desde posiciones consecutivas de memoria. Cuando se trabaja con words (16 bits) al transferir datos desde la memoria a registros (Dn o An), se aplica extensión de signo al valor del registro.
- ❑ El orden en que se acceden los registros es siempre D0 a D7 seguido de A0 a A7, independientemente de cómo aparecen en la lista.

MOVEM lista de registros, A_{eff}MOVEM A_{eff}, lista de registros

Ej.

MOVEM.L (A7)+,D0-D2/A1/A3

En esta instrucción están implicados como destino 5 registros; D0, D1, D2, A1 y A3. El contenido de 5 posiciones “consecutivas” de memoria conteniendo Long Words, empezando en la apuntada por A7, se transferirá a los registros.

MOVEP.

- Esta instrucción sirve para poder transferir datos de forma compacta, entre registros de 8 bits de dispositivos periféricos y los registros de datos del μ procesador. Operandos de 16 o 32 bits pueden ser transferidos byte por byte. El byte de mayor orden del registro se transfiere primero y el de menor orden, el último. Las direcciones del mapa de memoria se incrementan en 2 después de cada transferencia.
- Los registros del periférico en cuestión, deben estar situados, bien todos en direcciones pares o bien todos en direcciones impares y consecutivas en los dos casos. Este requerimiento se realiza en el diseño hardware del sistema.

MOVEP Dn, d(An) MOVEP d(An), Dn

Ej.

MOVEP.L D3,\$200(A5)

3B56C174	D3
1F0000	A5

1F0208H		
1F0206H	74	
1F0204H	C1	
1F0202H	56	
1F0200H	3B	
1F01FE		

Si la primera dirección hubiera sido impar, todos los datos estarían en la columna de la derecha.

NOP.

- Esta instrucción, lo único que hace es consumir 4 ciclos de reloj, sin realizar ninguna operación. (ejecución de retardos, eliminación de instrucciones sin alterar el resto del programa).

NOP

NEG.

- Realiza la resta entre 0 y el operando destino (0 – d), es decir, realiza un cambio aritmético de signo, guardando el resultado en el destino. Se puede decir que devuelve el complemento a dos del operando.

NEG A_{eff}

LINK, UNLK.

- Estas instrucciones facilitan el paso de parámetros a subrutinas, a través de la pila.

LINK

Las operaciones que realiza son las siguientes: En el primer paso, carga en la pila el contenido del registro de direcciones especificado en la instrucción (el SP se decrementa en 4 unidades como corresponde a los 4 bytes que abarca el contenido de un An). En el

siguiente paso, el contenido del SP se pasa al registro de direcciones especificado y finalmente, suma el desplazamiento de 16 bits con signo, dado como operando, al SP. Como se ve, la operación consiste en reservar una zona de la memoria en la pila, mediante el desplazamiento SP hacia otra dirección.

LINK An, #desplazamiento.

UNLK

Deshace todos los cambios que realiza LINK. Carga el SP con el contenido del registro de direcciones especificado, después carga en el registro de direcciones la Long Word desde la posición apuntada por el SP.

UNLK An

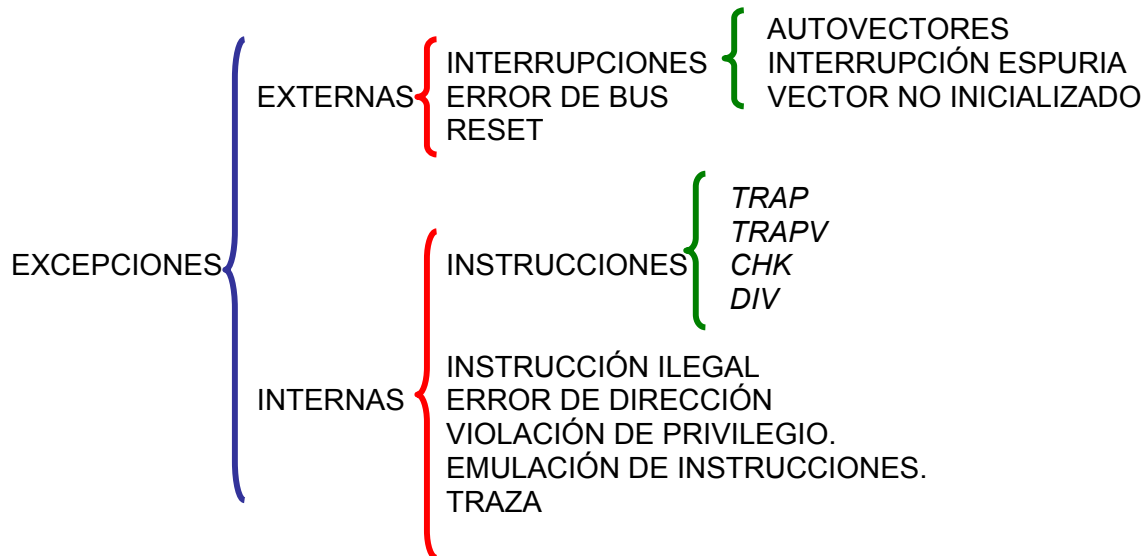
EJEMPLOS.

	BSR	SUBM
	DC.W	49
	DC.L	LISTA,LISTB
	Siguiete instrucción	
	.	
	.	
SUBM	MOVEM.L	D1/A0-A2,-(A7)
	MOVEA.L	16(A7),A0
	MOVE.W	(A0)+,D1
	MOVEM.L	(A0)+,A1-A2
	MOVEA.L	A0,16(A7)
LAZO	MOVE.L	(A1)+,(A2)+
	DBRA	D1,LAZO
	MOVEM.L	(A7)+,D1/A0-A2
	RTS	

	ORG	\$1000
LISTA	DS.L	50
	ORG	\$2000
LISTB	DS.L	50
	ORG	\$4000
	MOVE.W	#49,-(A7)
	PEA	LISTA
	PEA	LISTB
	BSR	SUBM
	ADDI.L	#10,A7
	Siguiete inst	
	.	
	.	
SUBM	MOVEM.L	D1/A1-A2,-(A7)
	MOVE.W	24(A7),D1
	MOVEA.L	20(A7),A1
	MOVEA.L	16(A7),A2
LAZO	MOVE.L	(A1)+,(A2)+
	DBRA	D1,LAZO
	MOVEM.L	(A7)+,D1/A1-A2
	RTS	

9.4.6-EXCEPCIONES (NOTAS).

- ❑ Una excepción es un fenómeno que da lugar a la interrupción de la ejecución normal de un programa, para dar paso a la ejecución de una rutina específica (rutina de atención a la excepción)
- ❑ Las causas que originan una excepción pueden ser de origen externo o de origen interno.



- ❑ El proceso que se desencadena tras una excepción es el siguiente:
 - 1º - Se copia internamente el SR, se activa el modo supervisor (S=1) y se desactiva el modo traza (si estaba activado) para evitar que la rutina de atención se ejecute en modo paso a paso. Para el caso de las excepciones de RESET e Interrupciones, también se modifica la máscara de prioridad de interrupción.
 - 2º - Se determina el número de vector de excepción. Para las interrupciones, el μ procesador obtiene el número de vector mediante un Ciclo Bus de Reconocimiento de Interrupción. Para todas las demás excepciones, la lógica de control interna proporciona el número de vector.
 - 3º - (Para todas las excepciones, salvo la de RESET). Se salva en la pila de supervisor (SSP+00) el PC (L), (SSP-02) el PC (H) y (SSP-04) la copia interna obtenida en el primer paso, del SR. El valor que se guarda del PC, es el que apunta a la siguiente instrucción no ejecutada, salvo en las excepciones de *error de bus* y *error de dirección*, en las cuales, el valor del PC es impredecible. Estas excepciones, llevan también a la pila información adicional (códigos de función, si se trata de una instrucción o no, estado de la línea de R/W#, dirección accedida, código de operación) que sirve para facilitar la investigación de las causas de error.
 - 4º - Se inicia la ejecución de la rutina de atención desde la posición contenida en el PC; este valor ha sido tomado de forma automática de la tabla de vectores correspondiente a la excepción en curso.
- ❑ Todas las excepciones, menos la de RESET, tienen asignado un único número de excepción (vector) que va desde 2 hasta 255 (254 vectores). RESET tiene asociados dos vectores con los números 0 y 1. Esto hace un total de 256 vectores de excepción. Asociado a cada vector hay cuatro bytes de memoria, en los que está contenida la dirección de comienzo de la rutina de atención correspondiente.

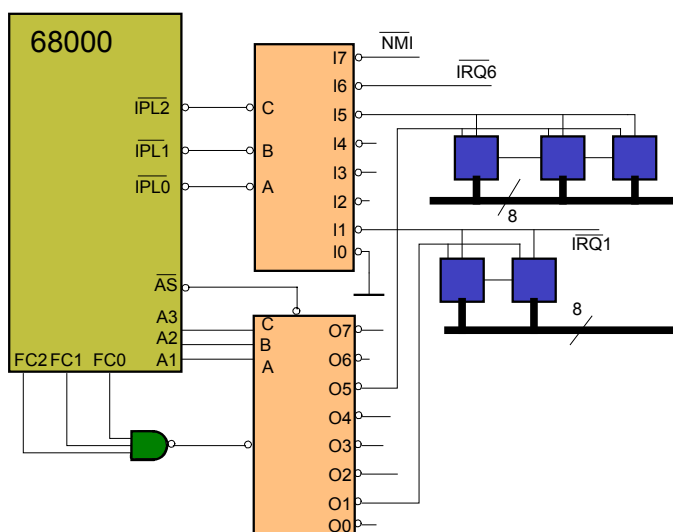
- El número de bytes de memoria ocupados por los vectores es $256 \times 4 = 1024$ bytes, es decir 1024 posiciones de memoria. El margen de direcciones va desde la dirección 000000H hasta la 0003FFH.
- El número de vector, multiplicado por cuatro nos da la dirección de la tabla de vectores donde se encuentra la dirección de atención correspondiente. El μ procesador obtiene la dirección correspondiente a un vector determinado, desplazando hacia la izquierda dos lugares el valor del número de vector y añadiendo ceros a la derecha y a la izquierda hasta completar los 24 bits que determinan la dirección.
- En la rutina de atención el programador deberá tener en cuenta factores como salvar aquellos registros de los que se va a hacer uso en el tratamiento de la rutina para recuperarlos a la salida de la misma, con los valores originales.

INTERRUPCIÓN.

- Al finalizar la ejecución de cada instrucción el μ P chequea el código en las líneas IPL ($IPL0\#$, $IPL1\#$, $IPL2\#$). Si el valor que encuentra es cero, indica que no hay una petición de interrupción. Si el valor codificado es distinto de cero, lo compara con la prioridad dada al programa que se está ejecutando. Esta prioridad del programa, viene dada por el código definido en la *máscara de nivel* del Byte de Sistema (I_0 , I_1 , I_2). Cuando el nivel de la solicitud de interrupción es superior al del programa, aquella se atenderá, en caso contrario se ignora, a excepción de una solicitud de nivel 7 (única que es sensible al flanco), que se atiende siempre. Cuando se acepta la interrupción, la máscara de nivel tomará el mismo valor que el nivel de la interrupción aceptada.
- Mediante un codificador de prioridad de 8 a 3 líneas, se establece el sistema de prioridades para diferentes dispositivos conectados a sus entradas. Cada una de estas entradas corresponde a un nivel, por lo tanto, todos los dispositivos conectados a la misma entrada, tienen el mismo nivel de prioridad. La diferencia entre estos dispositivos va a estar en el número de vector que van a entregar a la CPU cuando ésta lo solicite. El codificador va a entregar al μ procesador un código correspondiente a la línea de entrada excitada.

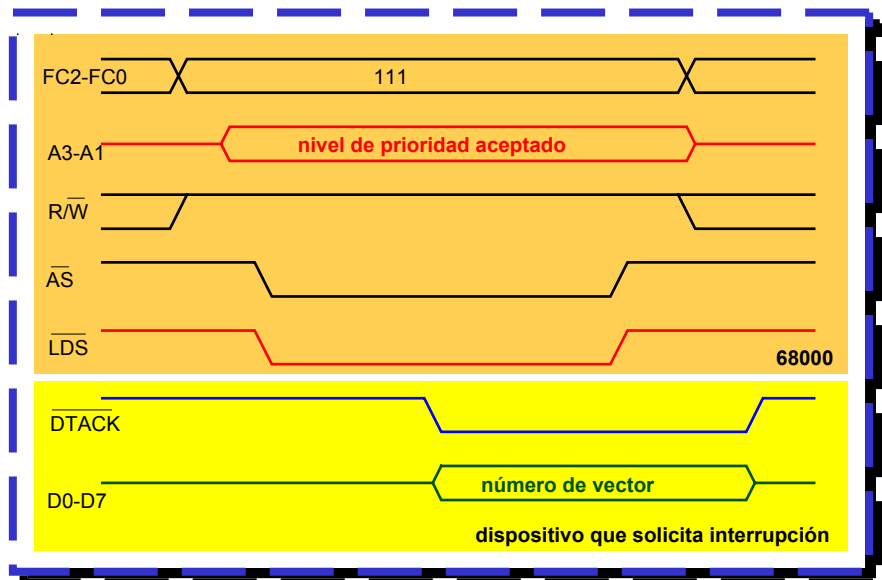
SECUENCIA DE LA INTERRUPCIÓN.

- Un dispositivo solicita ser atendido. La interfaz de este dispositivo genera una señal de petición de interrupción.

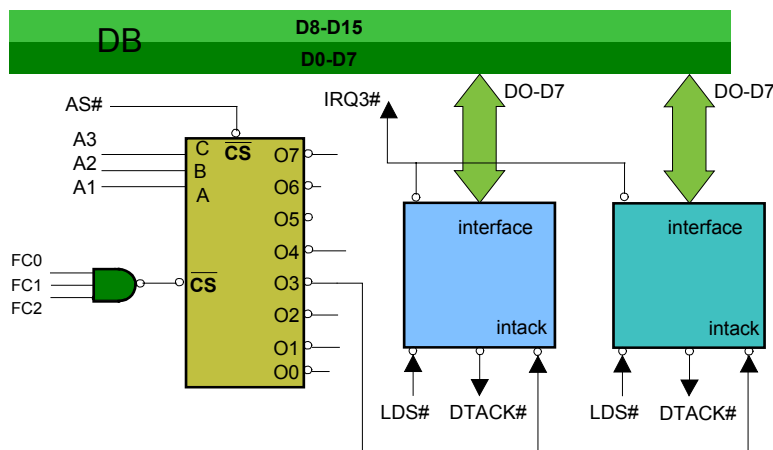


- Si la interrupción es aceptada, por ser de un nivel superior (o igual en el caso de interrupción de nivel 7) al definido por la máscara, el μ P pone en las líneas A3, A2, y A1 del AB el código correspondiente al nivel de la interrupción aceptada. Las demás líneas del AB pasan a nivel alto.
- Al mismo tiempo los terminales FC0, FC1 Y FC2 se ponen a 1, indicando que el ciclo bus que sigue es un *ciclo de reconocimiento de interrupción*.

- Mediante la lógica adecuada, y atendiendo a las anteriores condiciones y señales, se genera una señal (que llamaremos *intack*), que es aplicada a una entrada del dispositivo que originó la interrupción. Al recibir esta señal, el dispositivo debe mandar por la parte baja del DB (D0-D7), el número de vector que tiene asignado.



- El proceso de la lectura del número de vector es similar a un ciclo de lectura en memoria. Para ello, el μ procesador direcciona el dispositivo que debe ser leído. El proceso de direccionamiento se realiza de la forma descrita en el punto anterior. También se activa la línea R/W# para un proceso de lectura, y se activan las líneas de AS# y LDS# (parte baja del AB). El dispositivo seleccionado pone el número de vector en la parte baja del bus de datos y activa la línea de DTACK#.
- Una vez que la CPU ha recibido el número de vector, coloca en el PC la dirección que corresponde (n° de vector multiplicado por 4) y lee de la tabla de vectores, la dirección de comienzo de la rutina de atención al periférico solicitante.



- En sistemas complejos, donde se deben conectar más de un dispositivo a la misma línea de interrupción (misma prioridad), la señal de *intack* se suministra a todas las interfaces de los dispositivos conectados a la misma línea de interrupción. Aquel que solicitó la interrupción, será el que manda el vector.

Para solucionar el problema de que varios dispositivos con la misma prioridad soliciten atención a la vez y por lo tanto se les demande su vector de interrupción, se puede utilizar el sistema de *Cadena de Margaritas (Daisy Chain Connection)*.

- En la conexión Daisy Chain la señal de *intack* se transfiere de forma secuencial de una interfaz a la siguiente. El primer dispositivo más cercano eléctricamente al μ P, recibe primero la señal. Si este dispositivo es de uno de los que generó interrupción, bloquea la señal y envía su n° de vector. Si no generó interrupción, simplemente deja pasar la señal al siguiente dispositivo. Los dispositivos deben estar preparados para utilizar este esquema, de tal manera que, cuando aquel que solicitó la interrupción haya sido atendido, se ponga en un estado tal que no vuelva a bloquear la señal que el μ P genere para atender otra petición de interrupción de las pendientes.

- ❑ Las pastillas periféricas de la familia 68000 poseen un registro donde se puede escribir el número de vector (número de vector programable), asignado a un determinado periférico. En caso de que los periféricos no hayan sido inicializados o programados con un número de vector concreto, si solicitan interrupción, y ésta es aceptada, entregan por defecto el **Vector nº 15**.
- ❑ Los números de vector comprendidos entre **64** y **255**, pueden ser definidos por el programador.
- ❑ Si como consecuencia de una interrupción aceptada, y una vez realizado todo el proceso de direccionamiento y activación de las señales de estrobe, la CPU está esperando la señal de **DTACK#**, o **VPA#** para validar un número de vector, y dicha señal no la recibe dentro de unos ciclos de reloj determinados, el circuito supervisor de bus (WATCHDOG), activará una señal **BERR#**. El μ P diferencia el procesamiento de este error del de error de bus, tratándolo como una excepción llamada **SPURIOUS INTERRUPT**, **Vector nº 24**. (esta interrupción se puede producir generada por un ruido eléctrico).
- ❑ Para poder conectar interfaces diseñados para otras familias de μ P, los cuales no tienen en su arquitectura la facilidad de guardar y transmitir un nº de vector de interrupción, el 68000 asigna para el caso de que soliciten atención mediante interrupción, los números de vector **25 al 31 (Autovectores)**. En este caso cuando la CPU demanda un número de vector, la interfaz devuelve **VPA#** (Valid Peripheral Address), esto dice al μ P que se trata de un dispositivo sin la capacidad de enviar un número de vector, y por lo tanto debe usar el sistema de *autovectores*. La CPU no espera la señal **DTACK#** y procesa la rutina de atención generando internamente un número de vector (**25 a 31**) según el nivel de la interrupción solicitada.

RESET

Como salida:

- ❑ Cuando se ejecuta la instrucción RESET, el μ P pone el terminal de RESET a 0 durante 124 ciclos de reloj, con el fin de realizar una inicialización hardware de los elementos conectados al sistema. No afecta a los registros internos del μ P.

Como entrada:

- ❑ Se trata como una excepción con números de vectores **0** y **1**. El vector **0** (bytes \$000000, \$000001, \$000002 y \$000003) contiene el valor inicial que el programador da al SSP. El vector **1** (bytes \$000004, \$000005, \$000006 y \$000007) contiene la dirección de comienzo de la ejecución del programa principal, o rutina que da paso a la ejecución del mismo. Puede ser también la dirección de comienzo de una rutina que carga el Sistema Operativo en la memoria principal, o rutina (BOOTSTRAP) que carga el programa de aplicación desde un medio de almacenamiento masivo.
- ❑ La rutina de atención al reset no tiene que guardar ningún registro en la pila pues el proceso de reset inicializa el contenido de los mismos. Tampoco debe terminar con una instrucción de **RTE**.
- ❑ El vector de reset y la rutina de inicialización deben estar en memoria al encender el equipo, es decir, deberán estar en la memoria ROM, zona baja del mapa de memoria.
- ❑ El arranque tiene lugar durante 6 ciclos de bus. Durante estos ciclos de bus, la máscara de interrupción se pone con el valor 111, se activa **S** y se desactiva **T**, además de las lecturas indicadas en el cuadro:

1º: **INACTIVO**
 2º: Lee \$000000
 3º: Lee \$000002
 4º: Lee \$000004
 5º: Lee \$000006
 6º: Lee PC

- ❑ Para desencadenar el proceso de reset externo deben activarse simultáneamente los terminales **RESET#** Y **HALT#** al menos durante 100 ms.

BUS ERROR

- ❑ El sistema 68000 debe ser implementado con un circuito (diseñado por el usuario) que supervise (WATCHDOG) la actividad del bus y active el terminal **BERR#** cuando se den casos de intento de acceso al mapa de memoria donde no existe dispositivo, intento de escritura en ROM, error en un periférico. Por ejemplo, si la señal **DTACK#** se retrasa respecto de **AS#** más allá de un límite prefijado debe activar **BERR#**.
- ❑ Esta excepción da lugar a que se interrumpa el ciclo bus en curso, se salve el PC, SR y también información adicional como la primera palabra de la instrucción que estaba siendo procesada cuando se produjo el error, tipo de acceso (lectura o escritura), códigos de función, etc. y posteriormente se ejecute la rutina de atención a la excepción, cuya dirección de comienzo se cargará desde el **Vector nº 2**.

Si se produce un Error de Bus durante el procesamiento de otro Error de Bus, Error de Dirección o de Reset, el μ procesador para su actividad entrando en un estado de “parálisis total” (*halt*). Solamente un Reset externo puede sacarlo de ese estado.

ADDRESS ERROR.

Esta excepción se produce cuando se intenta acceder a una WORD, LONG WORD o Instrucción en una posición impar de memoria. El proceso de tratamiento de esta excepción es similar al de la excepción por Error de Bus, incluida la información que se lleva a la pila de supervisor. **Vector nº 3**.

Si se produce un Error de Dirección durante el procesamiento de otro Error de Dirección, Error de Bus o de Reset, el μ procesador entra en estado *halt*.

PRIVILEGE VIOLATION.

- ❑ Esta excepción es causada por el intento de ejecución de instrucciones reservadas al modo supervisor cuando el μ P está en modo usuario. **Vector nº 8**.
Las instrucciones privilegiadas son: ANDI al SR, EORI al SR, MOVE USP, ORI al SR, RESET, RTE y STOP.

ILLEGAL INSTRUCTION.

- ❑ El OP Code (Código de Operación), no corresponde con ninguno de los OP Code del 68000. **Vector nº 4**.

UNIMPLEMENTED INSTRUCTIONS.

- ❑ Si un OP Code que no está implementado en el 68000, comienza por 1010 (**vector nº 10**) o por 1111 (**vector nº 11**), da lugar a una excepción. Esto puede servir para la implementación de instrucciones particulares en modo software.

TRACING

- ❑ Se produce cuando el BIT T del SR está en estado 1.

- En este modo, al finalizar la ejecución de cada instrucción se produce una excepción con **vector nº 9**. Esta excepción es útil para la depuración de programas.

INSTRUCCIONES

Estas excepciones son causadas como parte integrante de la instrucción que se ejecuta.

TRAP #n ($0 \leq n \leq 15$)

- Generan un número de vector que está comprendido entre **32 y 47** dependiendo del valor inmediato de n en la instrucción.

TRAPV (*Trap on Overflow*). Esta instrucción, da lugar a una excepción si al ejecutarse, existe una situación de overflow (flag **V** está a 1). **Vector nº 7**.

CHK (*Check Register Against Boundaries*). Efectúa la comparación entre el contenido de un registro de datos, Dn y un operando. Si el Dn contiene un valor menor que 0 o mayor que el del operando se genera la excepción con **vector nº 6**.

DIVS y DIVU si el divisor es 0 dan origen a la excepción con **vector nº 5**.

9.4.7-ASIGNACIÓN DE LOS VECTORES DE EXCEPCIÓN.

Nº DE VECT.	DIRECCIÓN	ASIGNACIÓN
0	\$000000	RESET. VALOR INICIAL DEL SSP
1	\$000004	RESET: VALOR INICIAL DEL PC
2	\$000008	ERROR DE BUS
3	\$00000C	ERROR DE DIRECCIÓN
4	\$000010	INSTRUCCIÓN ILEGAL
5	\$000014	DIVISIÓN POR CERO
6	\$000018	INSTRUCCIÓN CHK
7	\$00001C	INSTRUCCIÓN TRAPV
8	\$000020	VIOLACIÓN DE PRIVILEGIO
9	\$000024	TRAZA
10	\$000028	EMULACIÓN 1010
11	\$00002C	EMULACIÓN 1111
12 a 14	\$000030 a \$000038	RESERVADOS PARA POSTERIORES USOS
15	\$00003C	VECTOR DE INTERRUPCIÓN NO INICIALIZADO
16 a 23	\$000040 a \$00005C	RESERVADOS PARA POSTERIORES USOS
24	\$000060	INTERRUPCIÓN ESPURIA
25	\$000064	AUTOVECTOR DE INTERRUPCIÓN NIVEL 1
26	\$000068	AUTOVECTOR DE INTERRUPCIÓN NIVEL 2
27	\$00006C	AUTOVECTOR DE INTERRUPCIÓN NIVEL 3
28	\$000070	AUTOVECTOR DE INTERRUPCIÓN NIVEL 4
29	\$000074	AUTOVECTOR DE INTERRUPCIÓN NIVEL 5
30	\$000078	AUTOVECTOR DE INTERRUPCIÓN NIVEL 6
31	\$00007C	AUTOVECTOR DE INTERRUPCIÓN NIVEL 7
32 a 47	\$000080 a \$0000BC	VECTORES DE INSTRUCCIONES TRAP (INTERRUPCIONES SOFTWARE)
48 a 63	\$0000C0 a \$0000FF	RESERVADOS PARA POSTERIORES USOS
64 a 255	\$000100 a \$0003FC	VECTORES DE INTERRUPCIÓN DE USUARIO.

9.4.8-PRIORIDAD DE LAS EXCEPCIONES:

GRUPO	ORDEN	EXCEPCIÓN	
0	1º	RESET	Aborta la ejecución de la instrucción en curso y procesa la excepción dentro de los dos ciclos de reloj siguientes.
	2º	ERROR DE DIRECCIÓN	
	3º	ERROR DE BUS	
1	4º	MODO TRAZA	Termina de ejecutar la instrucción en curso y después procesa la excepción.
	5º	INTERRUPCIÓN EXTERNA	
	6º	INSTRUCCIÓN ILEGAL	
	7º	VIOLACIÓN DE PRIVILEGIO	
2	8º	POR INSTRUCCIONES	La ejecución de la instrucción lleva al procesamiento de la excepción.

9.5-SET DE INSTRUCCIONES

(ver apéndice)



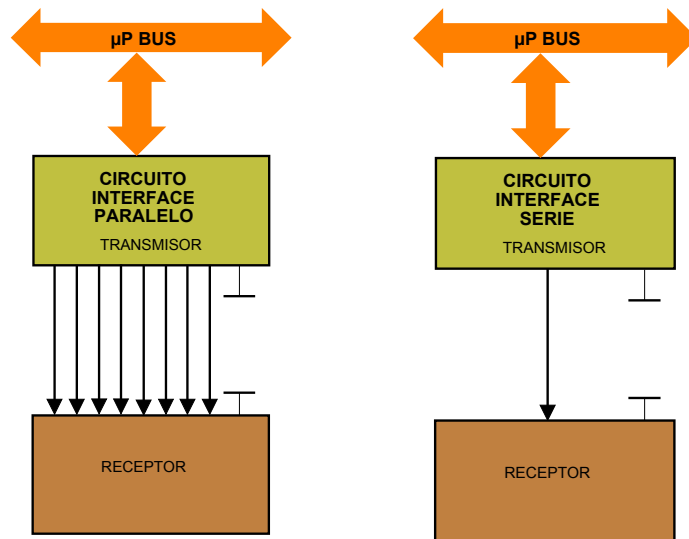
Input / Output

10

10.-INPUT/OUTPUT (I/O)

La transmisión de información entre el sistema μP , y los dispositivos conectados a él, se puede realizar de dos formas diferentes.

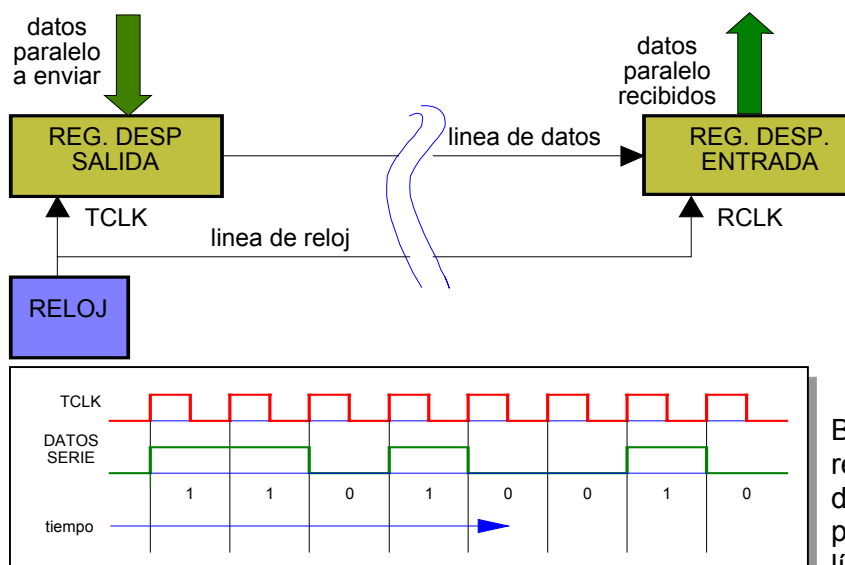
- Una manera es transmitir cada dato o palabra, de una sólo vez, es decir, todos los bits que forman la palabra, se transmiten de forma simultanea por un camino compuesto por un conjunto de líneas eléctricas, que transcurren de forma paralela. Cada BIT que forma la palabra, es emplazado en su correspondiente línea. Esta forma de transmisión, se llama **transmisión paralelo**.



- Otra forma, consiste en utilizar una sola línea y transmitir por ella de forma secuencial, cada uno de los bits que forman la palabra, según un orden establecido. A esta forma se la llama **transmisión serie**. Un circuito interfaz se encarga de convertir los datos en formato paralelo procedentes del bus de la computadora en formato serie, en el cual cada BIT es transmitido uno después de otro.
- El método de transmisión serie, tiene la ventaja respecto al paralelo, de un ahorro de hardware, especialmente cuando el emisor y receptor están separados por distancias relativamente grandes. Por otro lado la transmisión paralelo es más rápida. La velocidad de transmisión viene dada en bits por segundo o Baudios (1 Baudio = 1 BIT/s).

10.1-TRANSMISIÓN SERIE.

10.1.1-IDEA BÁSICA.

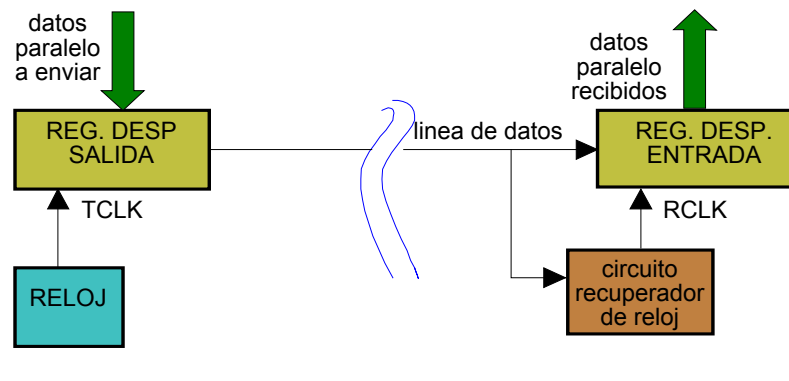


Un registro de desplazamiento (carga-paralelo, lectura-serie) en el emisor se encarga de convertir el flujo de datos de formato paralelo a formato serie. Un proceso opuesto ocurre en el receptor.

Bajo el control de la señal de reloj del emisor (TCLK) los datos son desplazados BIT por BIT y transmitidos por la línea de datos.

En el receptor, los bits son “introducidos” uno por uno hacia el registro de desplazamiento (carga-serie, lectura-paralelo) usando el reloj en el receptor (RCLK).

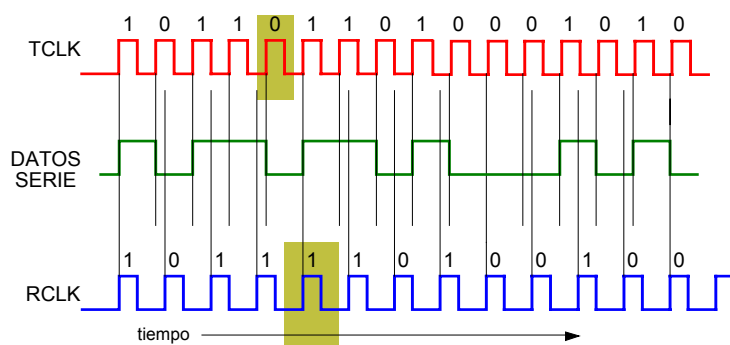
- ❑ Para que la transferencia de datos sea correcta, los relojes del emisor y receptor no solo deben tener la misma frecuencia, sino la misma relación de fase respecto a los datos; es decir, la posición relativa de los flancos de subida del RCLK y las transiciones de los datos recibidos, debe ser la misma que para el TCLK y los datos transmitidos.
- ❑ La transmisión serie tal y como se plantea en la idea básica presenta dos problemas; (1º) necesidad de hilo para el reloj, (2º) debido a las posibles diferencias entre los tiempos de propagación entre las dos líneas, este sistema no garantiza la relación de fase que se necesita entre el reloj y los datos en el receptor.
- ❑ Varios sistemas se han desarrollado para transmitir la señal de reloj junto con los datos por la misma línea. Estos sistemas se ajustan a la forma esquemática representada. Un sistema recuperador de la señal de reloj se encarga de obtener del flujo de datos, información sobre la señal de reloj, para generar la señal RCLK.



La manera en la cual la información de reloj se incorpora al flujo de datos origina dos formas de transmisión serie; *síncrona* y *asíncrona*.

10.1.2-TRANSMISIÓN SÍNCRONA.

- ❑ En este tipo de transmisión, el oscilador del receptor se diseña de tal manera, que su frecuencia de oscilación sea ajustable dentro de un pequeño margen en torno a su valor nominal. La misión del circuito recuperador de reloj es vigilar continuamente la relación de fase entre la señal recibida y RCLK, de tal manera que siempre que se necesite, este circuito ajuste la frecuencia del oscilador local para mantener la requerida relación de fase. El circuito que implementa esta idea es conocido como *phase-lock loop* (PLL).
- ❑ El funcionamiento correcto del circuito recuperador de reloj, va a depender de que existan transiciones (0-s a 1-s y 1-s a 0-s), en el flujo de información. Si se envía una cadena de bits en la cual existen en parte de ella una serie seguida de todos 0-s o todos 1-s, el circuito recuperador, puede separarse en un alto grado de la frecuencia correcta, causando errores. Para evitar esto, la información que se transmite, se codifica en uno de los códigos de línea existentes, que asegure que lo anterior no llegue a suceder.



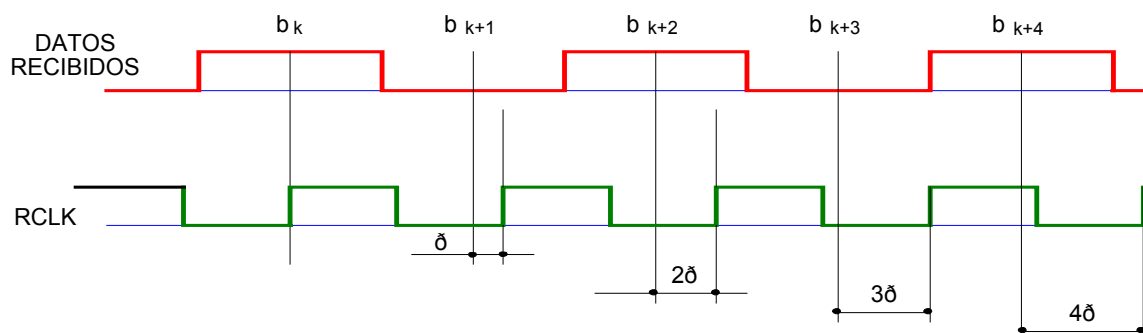
En la figura se muestra el efecto que un desvío en la frecuencia de reloj del receptor tiene sobre la información que se va a recibir. En el 5º impulso de reloj, se envía un 0, pero el receptor, debido al desvío de frecuencia de su reloj respecto a la del emisor, en el 5º impulso va a recibir un 1.

- ❑ Los datos o caracteres, no son enviados de forma aislada, sino formando bloques de gran longitud. Al comienzo de cada bloque se transmiten uno o dos caracteres de sincronismo, que alertan al circuito recuperador de reloj, preparándolo y dando tiempo para que pase su régimen transitorio de adecuación al régimen estacionario de trabajo, para la recepción sincronizada a la frecuencia correcta. El hecho de que la longitud de la información que se transmite sea grande, incrementa mas la posibilidad de error debido al desvío de la frecuencia del receptor (de aquí la necesidad de una exactitud perfecta en las frecuencias de emisor y receptor)
- ❑ En el receptor, deben ser separados unos caracteres de otros ya que no existe separación entre ellos en el flujo de información que se transmite. Para ello, se debe conocer cual es el primer BIT de cada dato, lo cual se logra sabiendo cual es el primer BIT después del carácter o caracteres de sincronismo. Llevando la cuenta correspondiente, y sabiendo cuantos bits componen la palabra, se determina donde comienza y termina cada una de ellas. Como carácter de sincronismo suele utilizarse el carácter ASCII, SYN (16H).

10.1.3-TRANSMISIÓN ASÍNCRONA.

- ❑ En este tipo de transmisión, la información es transmitida palabra por palabra, de forma aislada, y no existe un tiempo fijo entre la transmisión de una palabra y otra. Sí existe un tiempo fijo entre la transmisión de un BIT y otro dentro de una palabra.

La relación entre el reloj y los datos en el receptor se muestra en la figura.



La idea en la que se basa este tipo de transmisión es la siguiente:

Supongamos que los bits (b_k , b_{k+1} , b_{k+2} , b_{k+3} , b_{k+4}) son cargados en el registro de desplazamiento de receptor, en los flancos de subida consecutivos de la señal RCLK. Los datos serán recibidos correctamente si cada flanco de RCLK ocurre dentro del periodo de duración de cada BIT.

Supongamos también, que RCLK es mas lento (menos frecuencia) que TCLK. Si consideramos que para el BIT b_k , el flanco de subida de RCLK, ocurre en la mitad del periodo del BIT, en b_{k+1} el flanco de subida de RCLK, ocurrirá después (un tiempo δ), del punto medio del periodo de este BIT. Aún así, este BIT se recibirá correctamente siempre y cuando $\delta < T/2$ (siendo T el periodo de cada BIT). En algún BIT b_{k+n} , la acumulación de $n\delta$ será superior a $T/2$, con lo cual la recepción comenzará a ser errónea. Esto también se puede aplicar en el caso de que RCLK sea más rápido que TCLK.

- ❑ Para evitar los errores descritos anteriormente, lo que se hace es transmitir la información en pequeños grupos de bits, de tamaño fijo.
- ❑ TCLK y RCLK son diseñados para que sus frecuencias sean tan próximas como técnicamente sea posible. Además, un circuito detector de arranque en el receptor, ajusta la posición del flanco de subida del RCLK para situarlo lo más cerca posible de la mitad del

periodo del primer BIT de cada grupo. Para realizar esto, un oscilador del circuito de arranque, está diseñado para que su frecuencia de oscilación sea 16 o 32 veces superior a la del reloj del receptor.

Cuando este circuito detecta el flanco de inicio de transmisión, es decir, el start BIT, el oscilador se pone en marcha, se hace un conteo de 8 periodos (o 16 según la relación de frecuencias), y al terminar este conteo comienza el reloj del receptor. De esta manera los flancos del RCLK son coincidentes, por lo menos en el inicio de recepción, con la mitad del periodo de BIT.

Con pocos bits por grupo, y un bajo valor de δ , se garantiza una correcta recepción de todos los bits.

- Un esquema que implementa esta idea es el protocolo de transmisión *Start-Stop* cuyas características son las siguientes:

- * Cuando no se está transmitiendo, la línea de transmisión se mantiene en estado alto.

- * Los datos se envían en grupos de 5 a 8 bits.

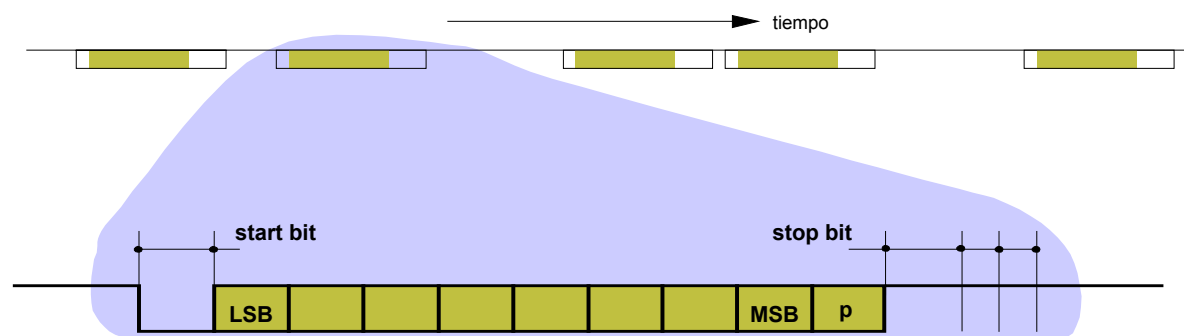
- * Cada grupo, va precedido de un BIT 0 llamado *start BIT* (BIT de arranque) y seguido de un BIT 1 llamado *stop BIT* (BIT de parada). No se puede transmitir una nueva palabra si como mínimo no ha transcurrido un tiempo igual al periodo de BIT del stop BIT. En algunos casos el tiempo mínimo que debe transcurrir para poder transmitir otra palabra debe ser de 1,5 o 2 veces el periodo de BIT, es decir, se utilizan 1,5 o 2 bits de parada.

- * Antes del *stop BIT* puede ser transmitido (opcional) un BIT de paridad (p).

- * Los bits de datos son transmitidos comenzando por el LSB.

- * El flanco de bajada del *start BIT* es usado por el circuito receptor para estimar la posición media del periodo del primer BIT de datos.

- * Después de recibir los bits del grupo, el receptor busca el *BIT de stop*. Si encuentra un 0 se dice que ha ocurrido un **framing error**. Este tipo de error puede ocurrir por un *BIT de start* erróneo, causado posiblemente por ruido eléctrico en la línea de transmisión. También puede ocurrir porque la diferencia en frecuencia entre el TCLK y RCLK es demasiado grande.



- ❑ **b3- RF (Receiver Full):** Cuando un carácter ha sido recibido y está disponible en Rx, este bit se pone a 1. Cuando el µP lee el registro, RF pasa a 0.
- ❑ **b4- TE (Transmitter Empty):** Vale 1 cuando el registro Tx está vacío.
- ❑ **b5, b6- DSR y DCD:** Representan el estado de las líneas DSR# (*Data Set Ready*) y DCD# (*Data Carrier Detect*). Son utilizadas con las señales RS-232-C.
- ❑ **b7- IRQ:** Se pone a uno cuando la línea IRQ# está activa. (Solicitud de INT.)

CONTROL REGISTER.

La velocidad de la transmisión viene determinada por la velocidad (CLOCK) con la que se hace la conversión serie- paralelo o paralelo- serie.

El reloj externo se conecta a los terminales TxC para el transmisor y RxC, para el receptor (pueden tener diferentes velocidades). La frecuencia de estos relojes debe ser 16 veces mayor que la velocidad deseada.

El reloj interno se consigue conectando al generador de baudios un cristal entre los terminales XTAL1 y XTAL2. Para generar las velocidades normalizadas, el cristal que se coloca es de 1.8432 MHz.

b7	Stop Bits
0	1 bit
1	2 bits

b6	b5	Word length
0	0	8 bits
0	1	7 bits
1	0	6 bits
1	1	5 bits

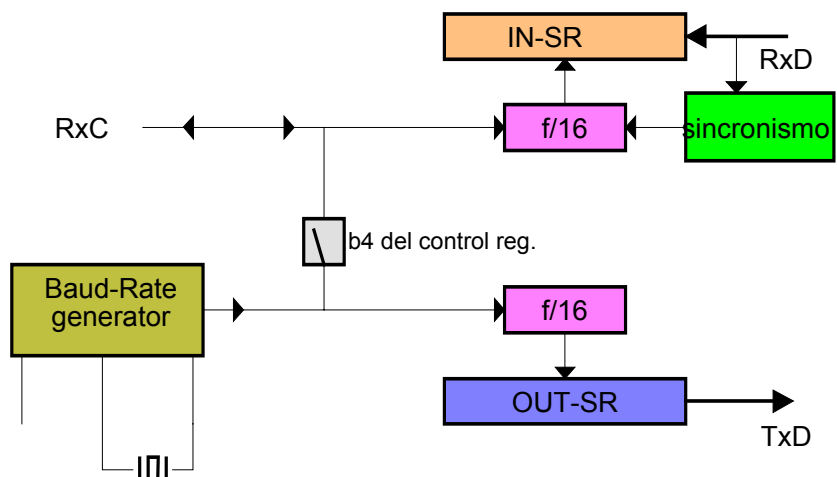
b3	b2	b1	b0	Baud Rate Generator
0	0	0	0	RELOJ EXTERNO
0	0	0	1	50 BAUDIOS
0	0	1	0	75 B
0	0	1	1	109.92 B
0	1	0	0	134.58 B
0	1	0	1	150 B
0	1	1	0	300 B
0	1	1	1	600 B
1	0	0	0	1200 B
1	0	0	1	1800 B
1	0	1	0	2400 B
1	0	1	1	3600 B
1	1	0	0	4800 B
1	1	0	1	7200 B
1	1	1	0	9600 B
1	1	1	1	19200 B

❑ **b4- Rx CLOCK:**

b4=1. El reloj del transmisor es el mismo que el del receptor, y su velocidad viene dada también por los bits del baud rate generator.

b4=0. Se utiliza un reloj externo (RxC).

El terminal RxC (Receive Clock) puede ser de entrada o de salida, según el modo de funcionamiento seleccionado. Si b4=1, la velocidad de transmisión y recepción son iguales. En este caso RxC da a la salida la frecuencia de trabajo x 16. Sirve para sincronizar la transmisión /recepción serie con otros dispositivos, compartiendo el mismo reloj.



COMMAND REGISTER

- ❑ **b0- DTR (Data Terminal Ready):** Si está a cero pone la línea DTR# (perteneciente a las señales RS-232-C), inactiva indicando que transmisor y receptor están inhabilitados. Si está a 1, activa la línea DTR# y habilita el transmisor y receptor.
- ❑ **b1- Rx int (Receiver Interrupts):** Si está en 0 permite interrupciones hacia la CPU con la recepción de cada carácter.
- ❑ **b2 b3- Tx control (Transmitter Control):** Estas líneas controlan interrupciones. También controlan otra señal RS-232-C llamada RTS# (*request to send*). Con la combinación 01 (Ver cuadro) cuando Tx-R está vacío, porque el carácter se ha enviado desde éste al OUT-SR, se activa IRQ#. Cuando el μ P envía un nuevo carácter a Tx-R se desactiva la petición de interrupción.
- ❑ **b4- Echo ctr (Echo Control):** Si está puesto a 1, todos los datos recibidos son copiados en el registro transmisor con un retardo correspondiente a 1/2 del periodo de BIT.
- ❑ **b5 b6 b7- PC (Parity Control):** Si el BIT 5 está en 0, la ACIA no genera BIT de paridad para los caracteres que transmite, ni lo espera en los caracteres que recibe. (Ver cuadro)

Tx Control			
b3	b2	INT	RTS#
0	0	DIS.	HIGH
0	1	EN.	LOW
1	0	DIS.	LOW
1	1	DIS.	LOW

b7	b6	b5	Parity Control
X	X	0	NO PARITY
0	0	1	ODD
0	1	1	EVEN
1	0	1	Tx SETS PARITY BIT Rx IGNORES PARITY
1	1	1	Tx RESETS PARITY BIT Rx IGNORES PARITY

Cuando se van a usar interrupciones para operaciones de entrada o salida, una instrucción de programa debe habilitar al transmisor y receptor para el trabajo con las mismas. Para ello, se debe escribir en *Tx control* del Command Register, el patrón 01 y en *Rx int.* del mismo registro, el valor 0. De esta manera, siempre que la ACIA recibe un carácter, activa la línea IRQ# y pone a 1 el BIT RF y el BIT IRQ del Registro de Estado. Cuando el μ P lee el carácter, la ACIA desactiva la línea IRQ#, y pone a cero los bits RF y IRQ. De la misma forma, cuando el buffer transmisor está vacío, la ACIA activa la línea IRQ# y pone a 1 los bits TE y IRQ. Cuando el μ P envía un carácter para ser transmitido, se desactiva la línea IRQ# y los bits TE y IRQ vuelven a 0. Este estado se mantiene hasta que el carácter es transferido desde el buffer TX, hasta el OUT-SR.

SELECCIÓN DE LOS REGISTROS:

- ❑ Los diferentes registros son direccionados mediante las líneas RS1 y RS2, según las combinaciones dadas en el cuadro.

RS1	RS0	READ	WRITE
0	0	RX	TX
0	1	STATUS	CHIP RESET. (WRITE % 00000010 IN COMMAND REGISTER)
1	0	COMMAND	COMAND
1	1	CONTROL	CONTROL

Si se activa la línea de RESET#, en el command register se escribe el valor %00000010. También se ponen a 0 todos los bits del registro de control.

SEÑALES RS-232-C

- ❑ **RTS#** (*Request To Send*). Se utiliza para indicar al módem que la ACIA está lista para iniciar una transmisión.
- ❑ **CTS#** (*clear to send*). Mediante esta señal, el módem le indica a la ACIA que está preparado para la transmisión.
- ❑ **DTR#** (*Data Terminal Ready*). Indica al módem el estado de la ACIA. Inhabilitada si DTR# = 1 y habilitada DTR# = 0.
- ❑ **DSR#** (*Data Set Ready*). Indica a la ACIA que el módem está preparado para el intercambio de datos, que no está en modo test, voz, o marcado, que ha finalizado todo handshake inicial y tono de respuesta y retrasos han finalizado.
- ❑ **DCD#** (*Data Carrier Detect*). El módem indica a la ACIA la recepción de la portadora, es decir, se ha establecido el canal de comunicación con el equipo remoto.

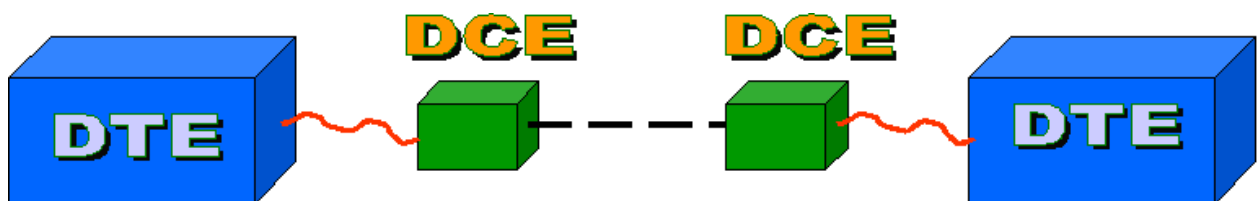
Ejemplo de configuración de la ACIA.

RX	EQU	\$D60000	reg. receptor
TX	EQU	\$D60000	reg. emisor
STATUS	EQU	\$D60002	reg. De estado
CMND	EQU	\$D60004	reg. De comando
CNTRL	EQU	\$D60006	reg. De control
LINE	EQU	\$E00000	dirección datos de entrada
CR	EQU	\$0D	código ASCII para CR
	ORG	\$1000	
	MOVE.B	#\$1E,CNTRL	formato de la transmisión
	MOVE.B	#\$B,CMND	
	MOVEA	#LINE,A0	inicializa pointer
NEXT	BSR	RCVCHR	
	MOVE.B	DO,(A0)+	
	BSR	TRMCHR	
	CMP.B	#CR,D0	busca CR
	BNE	NEXT	
	BSR	INTEPRT	
*rutina para recibir un carácter			
RCVCHR	BTST.B	#3,STATUS	
	BEQ	RCVCHR	
	MOVE.B	RX,D0	
	RTS		
*rutina para transmitir un carácter			
TRMCHR	BTST.B	#4,STATUS	
	BEQ	TRMCHR	
	MOVE.B	D0,TX	
	RTS		
INTEPRT			
			
	END		

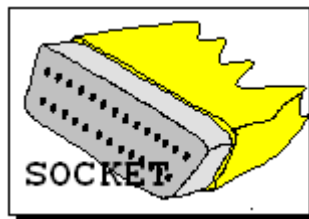
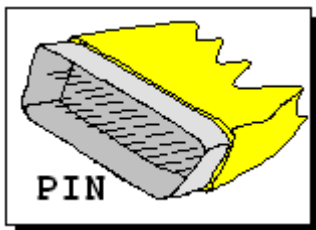
PROGRAMA EJEMPLO PARA EL 68000 QUE LEE UNA LÍNEA DE UN TECLADO Y LA MANDA A LA PANTALLA.

10.3-RS- 232-C (V.24, V.28)

- ❑ Es un estándar comercial definido bajo las siglas RS-232-C por la EIA (*Electronic Industries Association*) en USA y por las siglas V.24, completada con V.28 por la CCITT (*International Consultative Committee for Telephony and Telegraphy*) en Europa, para el intercambio de señales entre equipos, mediante una comunicación serie. Define las características eléctricas que deben cumplir las señales, las funciones que deben realizar y las especificaciones mecánicas que deben cumplir los conectores pertenecientes a esta norma.
- ❑ Este standard divide los equipos en dos categorías:
 1. **DTE** (*DATA TERMINAL EQUIPMENT*). Comprende todo equipo que procese datos, como computadoras, video-terminales, impresoras...
 2. **DCE** (*DATA CIRCUIT-TERMINATING EQUIPMENT, o también, DATA COMMUNICATIONS EQUIPMENT*). Comprende todos los dispositivos, como el módem, cuya función primordial es proveer un camino en la comunicación entre dos dispositivos DTE.



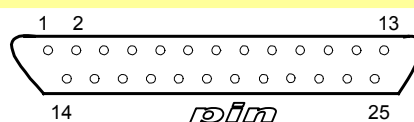
- ❑ Los conectores tipo D (D-9 o D-25, según el número de contactos) son macho (Pin), para los DTE y hembra (Socket) para los DCE.



DESCRIPCIÓN FUNCIONAL DE LAS SEÑALES RS-232-C.(conector D-25)

DESIGN (EIA).	PIN	FUENTE	FUNCIÓN	COMENTARIOS
AA	1		PROTECTIVE GROUND	Equipo eléctrico y señal dc
AB	7		SIGNAL GROUND	Referencia de masa
BA	2	DTE	TRANSMITTED DATA (TxD)	Datos que se transmiten
BB	3	DCE	RECEIVED DATA (RxD)	Datos que se reciben
CA	4	DTE	REQUEST TO SEND (RTS#)	DTE indica al módem que está listo para transmitir datos (canal primario)
CB	5	DCE	CLEAR TO SEND (CTS#)	DCE activa la línea hacia el módem remoto y envía esta señal al DTE local indicándole que la línea esta lista para la transmisión. (primario)
CC	6	DCE	DATA SET READY (DSR#)	Indica al DTE que DCE no está en modo test, voz, o marcado, que todo handshake inicial, tono de respuesta y retrasos han terminado, es decir, DCE está preparado para trabajar.
CD	20	DTE	DATA TERMINAL READY (DTR#)	El DTE indica al módem local que está preparado.
CE	22	DCE	RING INDICATOR	DCE ha detectado un tono de llamada en la línea telefónica.
CF	8	DCE	DATA CARRIER DETECTED (DCD#)	Cuando el módem local detecta la señal RTS, activa los circuitos de enlace con la línea, enviando una señal portadora al módem remoto, este la detecta y activa DCD avisando a su DTE.
CG	21	DCE	SIGNAL QUALITY DETECTOR	Indica que la señal está libre de error
CH-CI	23	DTE/DCE	DATA SIGNAL RATE SELECTOR	Selecciona entre dos posibles velocidades
DA	24	DTE	TRANSMITTER CLOCK	Proporciona en la señal información de reloj al DCE
DB	15	DCE	TRANSMITTER CLOCK	Proporciona en la señal información de reloj al DTE
DD	17	DCE	RECEIVER CLOCK	Proporciona en la señal información de reloj al DTE
SBA	14	DTE	SECONDARY TRANSMITTED DATA	Canal secundario de baja velocidad
SBB	16	DCE	SECONDARY RECEIVED DATA	Canal secundario de baja velocidad
SCA	19	DTE	SECONDARY REQUEST TO SEND	DTE está listo para transmitir datos (canal secundario)
SCB	13	DCE	SECONDARY CLEAR TO SEND	DCE está listo para transmitir datos (canal secundario)
SCF	12	DCE	SECONDARY DATA CARRIER DETECTED	

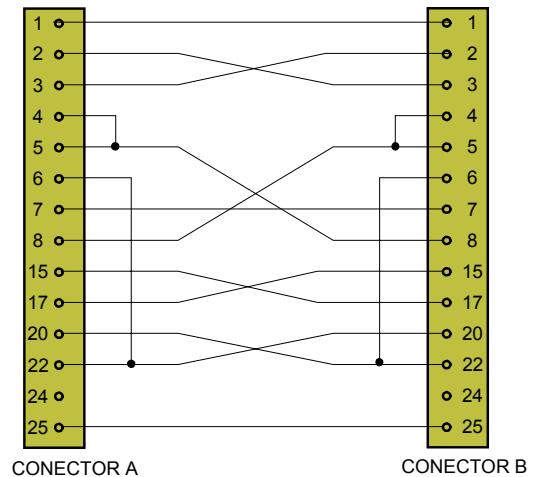
LOS TERMINALES SOMBREADOS SON LOS DE LAS SEÑALES PRINCIPALES.



LA CORRESPONDENCIA CON LOS TERMINALES DEL CONECTOR D9 SON LAS SIGUIENTES: (D25-D9) 2-3, 3-2, 4-7, 5-8, 6-6, 7-5, 8-1, 20-4.

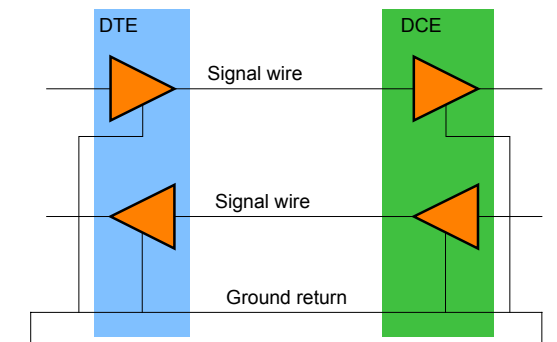
- ❑ La forma en la que los modems transmiten la información, es modulando y demodulando una señal portadora, con dicha información. Antes de que la transmisión pueda tener lugar, cada MODEM debe activar su portadora. Cuando ambas portadoras están activadas, se dice que existe un canal de comunicación *full-duplex* (en los dos sentidos simultáneamente), entre los dos modems. El papel que desempeñan las señales RS-232-C es permitir al DTE de cada extremo, controlar la operación de su correspondiente DCE asociado.
- ❑ En algunos enlaces se establecen dos canales para cada dirección de la transmisión, usando dos diferentes portadoras. El primero es el canal primario usado para la transmisión de datos. El segundo, el cual generalmente es de menor velocidad, se llama canal secundario y se usa para transferir información de control.

- Cuando la distancia a cubrir en la transmisión no es muy grande, se puede eliminar el uso del módem, local y remoto (null-modem), realizando la conexión dada en la figura. En este caso, los dos finales son DTE según la terminología de la norma, por lo tanto, las conexiones son tal que cada uno ve al otro como un DCE.

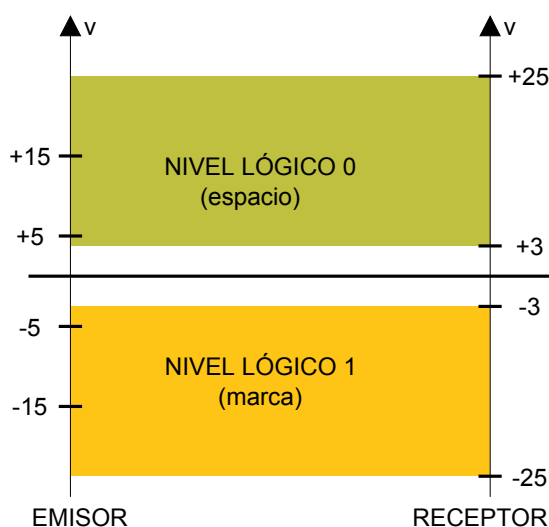


CARACTERÍSTICAS ELÉCTRICAS.

- El modo de transmisión es *no balanceada* o de *terminación simple* (*single-ended transmission*). En este modo, los terminales de la masa de la señal del DTE y DCE, están unidos (pin 7), y un hilo (el hilo activo) se usa para llevar la señal. Esta constitución es aceptable para bajas velocidades (**maximun data rate 20kb/s**) y cortas distancias (**maximun cable length 15 m.**). Cuando no se necesita una alta precisión se pueden usar cables de hasta 100 m.



NIVELES DE VOLTAJE.

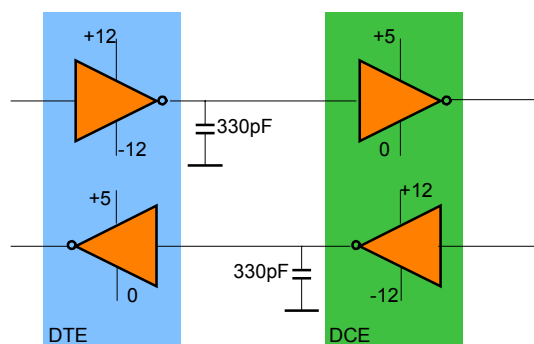


- Los niveles lógicos (0 y 1), también son conocidos como “espacio” para designar el 0 y “marca” para el 1.
- El receptor interpreta cualquier voltaje por debajo de -3 volts. como un 1, y voltajes por encima de $+3$ volts. como 0. El emisor entrega un valor lógico 0 con una tensión por encima de 5 V, y un valor lógico 1 con una tensión inferior a -5 V, por lo tanto el margen de ruido es de 2 volts.
- Para las señales de control, se utilizan los mismos niveles de voltaje, pero en este caso, un voltaje positivo indica **on** o activo, y un voltaje negativo indica **off** o inactivo.

Las tensiones del emisor vienen dadas en base a una carga equivalente comprendida

entre $3k\Omega$ y $7k\Omega$.

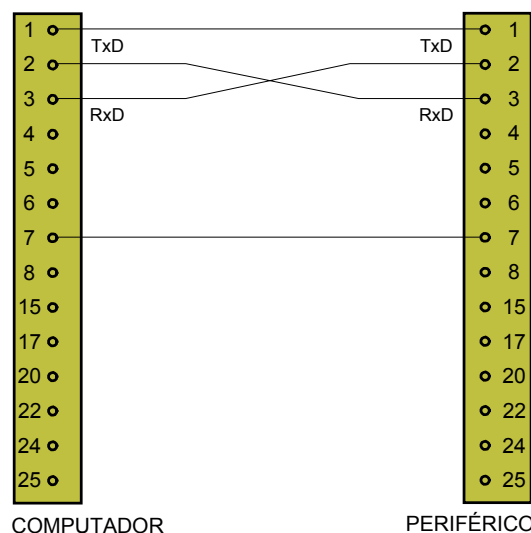
- Las tensiones con las que las UART y USART trabajan, no satisfacen la norma RS-232-C. Para convertir las tensiones de estos circuitos a los niveles de tensión requeridos, se utilizan los excitadores de línea, que son pastillas integradas, con la circuitería interna necesaria para convertir los niveles de TTL a las tensiones RS-232-C que manejen la línea. Por otro lado, existen los receptores de línea que lo que hacen es convertir las tensiones RS-232C de la línea a los niveles de tensión con los que trabajan las UART y USART. En la figura se representa la conexión de estos drivers. Los condensadores son puestos en la línea para limitar el Slew Rate al máximo permitido por la norma.



- Corriente máxima de salida del transmisor en cortocircuito: **0.5 A.**
- Máximo slew rate: **30 V/μs**
- Resistencia máxima de salida sin alimentación: **300Ω.**
- Resistencia de entrada del receptor: entre **3KΩ y 7 KΩ.**
- Cuando existan cables largos en la comunicación, la masa de la señal debe ser conectada a tierra (cable verde y blanco en el conector de alimentación), solamente en un único punto, para evitar que se creen lazos de tierra.

TRANSMISIÓN A TRES HILOS.

- Es posible realizar una transmisión serie sólomente haciendo uso de tres hilos, entre un microcomputador y un periférico. En este caso se debe tener en cuenta cuando debe comenzar la transmisión y cuando debe detenerse. Para ello, el periférico envía por su línea TxD (pin 2 de la interfaz del periférico) unos caracteres de control que indican al computador cuando debe transmitir y cuando debe dejar de hacerlo. Estos caracteres son X-ON (DC1 o 11H en ASCII) para que transmita y X-OFF (DC3 o 13H en ASCII) para que deje de hacerlo. El protocolo para llevar a cabo este tipo de transmisión se llama **X-ON / X-OFF**.



- La anterior forma de realizar el protocolo, requiere que el periférico pueda mandar datos al computador.
- Otra forma de realizar el protocolo es haciendo uso de las conexiones pertinentes, en las líneas RTS#, CTS#, DSR#, DCD#, DTR#, además de las tres líneas para la transmisión de los datos.

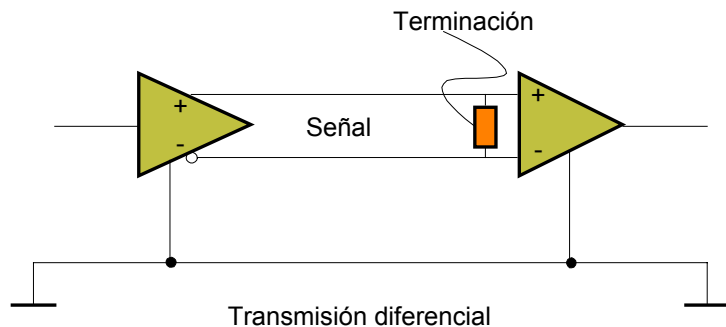
10.4-RS- 449 [RS-422 (V.11), RS-423 (V.10)]

- La limitación más importante que tiene la conexión RS-232-C viene del uso de la transmisión single-ended, la cual es bastante sensible al ruido. Esta sensibilidad es tanto mas pronunciada cuanto mayor es la frecuencia de trabajo.

Una técnica que minimiza los efectos del ruido es la transmisión *diferencial balanceada*.

- En la transmisión diferencial, para llevar la señal, se usan dos cables. La masa proporciona un voltaje de referencia para los circuitos del transmisor y receptor. La señal se transmite en forma de una diferencia de voltaje entre los dos cables, es decir, la referencia de tensión no es fija (masa) sino que es flotante, por lo tanto, el voltaje medio entre las señales de los cables y la masa (voltaje en modo común), no tiene efecto sobre el receptor.

- Se trata de que la línea de transmisión esté balanceada, haciendo que los dos cables sean idénticos tanto mecánicamente como eléctricamente, de esta manera, los voltajes de ruido inducidos en ellos serán iguales. Como el receptor solamente es sensible a la diferencia de voltaje entre los dos cables, los voltajes de ruido serán anulados.

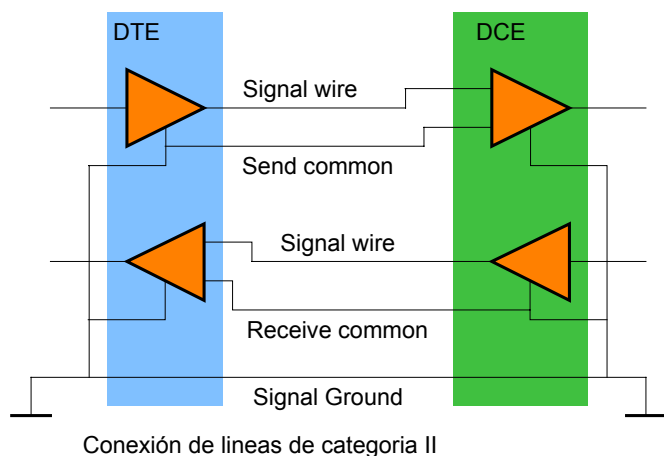


- Otros conductores, transportando corriente y que estén próximos al lazo formado por los dos cables empleados para la transmisión diferencial, inducirán corrientes que se verán como una diferencia de voltaje a la entrada del receptor. Para reducir estos voltajes, lo que se hace es reducir el lazo al mínimo mediante el trenzado de los dos cables, además, apantallarlos para aislarlos más si cabe, del entorno donde se encuentren. Cables trenzados y apantallados (STP)

RS-449, define dos canales: Canal primario y canal secundario.

- El canal primario usa un conector de 37 pines. Diez de las señales se llaman señales de categoría I y son transmitidas usando el esquema de transmisión diferencial balanceada, por lo tanto usan dos pines cada una. El resto de las señales se llaman señales de categoría II, las cuales usan un esquema de transmisión llamado *diferencial no balanceado* donde el emisor usa el esquema *single-ended*, y el receptor el *diferencial*.

La señal de masa en el DTE se conecta a través de la línea Send Common a las entradas negativas de todos los receptores de categoría II en el DCE y viceversa.



La transmisión no balanceada es bastante menos efectiva que la balanceada, por lo tanto las señales incluidas en la categoría II son aquellas que cambian de estado de forma poco frecuente, y no necesitan receptores rápidos.

- En el canal secundario, se utiliza un conector adicional de nueve pines para transportar las señales de control del de este canal. Todas las señales de este canal son de categoría II.

Características.

- Para la transmisión diferencial balanceada, la velocidad de transferencia puede llegar hasta 10Mb/s a una distancia nominal de 1200 m. Para la transmisión no balanceada, la velocidad baja a 100Kb/s. (estos límites pueden ser ampliados dependiendo del tipo de cable usado y del entorno donde se desarrolla la transmisión). Las características eléctricas de los drivers y receptores se especifican en los standard RS- 422 para las señales de categoría I y en la RS- 423 para las señales de categoría II.

SEÑALES DEL CANAL PRIMARIO EN RS-449

DESIGN.	PIN	FUENTE	FUNCIÓN	CATEGORIA	EQUIVALENTE RS-232-C
SG	19		SIGNAL GROUND		AB
SC	37	DTE	SEND COMMON		
RC	20	DCE	RECEIVE COMMON		
SD	4, 22	DTE	SEND DATA	I	BA
RD	6, 24	DCE	RECEIVE DATA	I	BB
IS	28	DTE	IN SERVICE	II	
IC	15	DCE	INCOMING CALL	II	CE
TR	12, 30	DTE	TERMINAL READY	I	CD
DM	11, 29	DCE	DATA MODE	I	CC
RS	7, 25	DTE	REQUEST TO SEND	I	CA
CS	9, 27	DCE	CLEAR TO SEND	I	CB
RR	13, 31	DCE	RECEIVER READY	I	CF
SQ	33	DCE	SIGNAL QUALITY	II	CG
NS	34	DTE	NEW SIGNAL	II	
SF*	16	DTE	SELECT FREQUENCY	II	
SR*	16	DTE	SIGNALING RATE SELECTOR	II	CH
SI	2	DCE	SIGNALING RATE INDICATOR	II	CI
TT	17, 35	DTE	TERMINAL TIMING	I	DA
ST	5, 23	DCE	SEND TIMING	I	DB
RT	8, 26	DCE	RECEIVE TIMING	I	DD
LL	10	DTE	LOCAL LOOPBACK	II	
RL	14	DTE	REMOTE LOOPBACK	II	
TM	18	DCE	TEST MODE	II	
SS	32	DTE	SELECT STANDBY	II	
SB	36	DCE	STANDBY INDICATOR	II	

NOTAS. El pin 1 se conecta al conector y al apantallado del cable y a la masa solamente en un extremo.

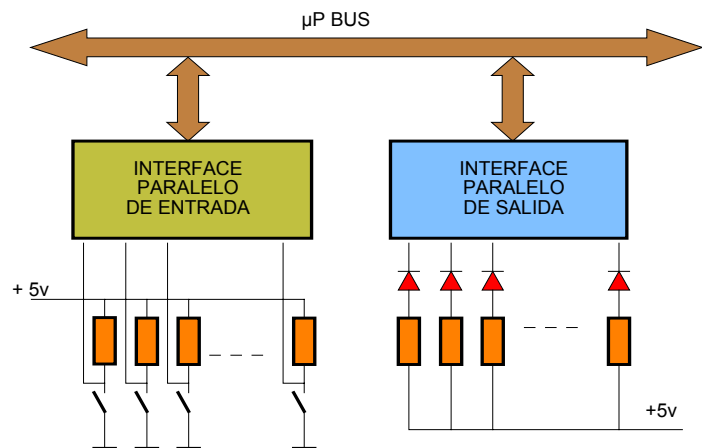
* Las señales SF y SR comparten el mismo pin. Sólomente una de las dos se usa en una instalación determinada.

10.5-TRANSMISIÓN PARALELO.

- En su forma más simple una interfaz paralelo lo único que hace es proveer un buffer entre el bus del μP y el dispositivo externo.

Recordemos que un buffer en general, se refiere al componente de un sistema que aísla un subsistema de otro. Se necesita cuando dos subsistemas tienen diferentes características eléctricas o de temporización.

Actualmente existe un gran número de circuitos VLSI que se ocupan bajo el control del μP , de la transmisión de datos en paralelo. La forma en la que trabajan estos circuitos, los puertos que poseen la función de cada puerto etc. es variada. Algunas características generales son:

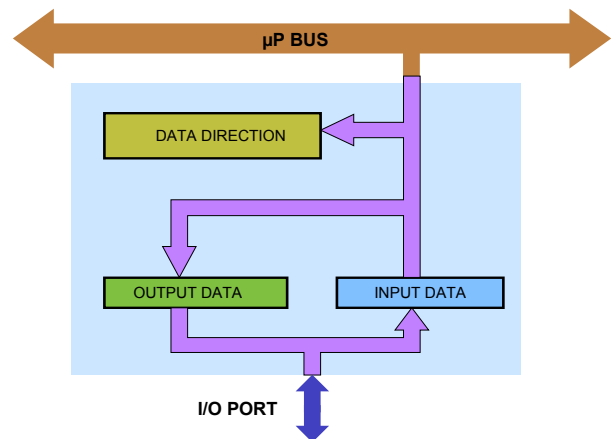


10.5.1-ENTRADAS POR LATCH.

- La información en el puerto de entrada puede estar disponible durante un tiempo relativamente largo o un tiempo relativamente corto. En este último caso puede ser que la información se pierda debido a que el μP "llegue tarde" a la lectura de los datos. Para evitar esta situación, la información es almacenada (*latched*) en un buffer de entrada para poder ser leída por el μP cuando sea necesario.

10.5.2-I/O COMBINADAS.

- El puerto posee circuitería necesaria para las funciones de entrada y salida en las mismas líneas, estas son activadas para el trabajo en una u otra dirección (I/O) mediante el control del programa.
- El registro de dirección de datos tiene cada BIT asociado a una línea del puerto para especificar la dirección de transferencia en esa línea. Un cero en un BIT determinado indica que la línea asociada a ese BIT es de entrada. Un 1 indica que la línea es de salida.

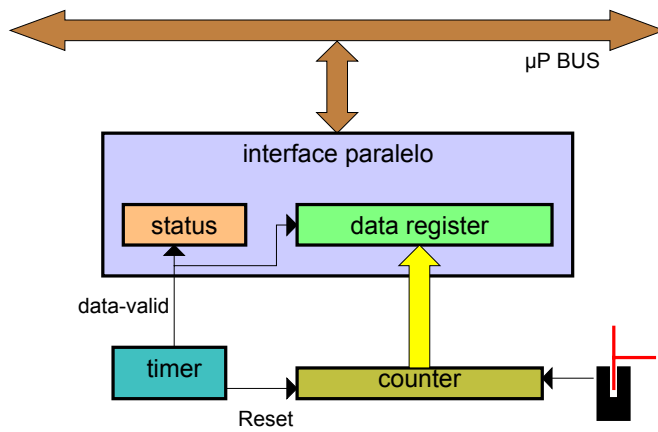


10.5.3-HANDSHAKE. (PROTOCOLO)

ENTRADA.

Se va a explicar el dialogo de señales entre elementos externos y la interfaz paralelo, a través de un ejemplo.

Supongamos que queremos medir la velocidad de giro de un motor. Para ello nos valemos del montaje de la siguiente figura. El principio de funcionamiento del circuito es el siguiente:

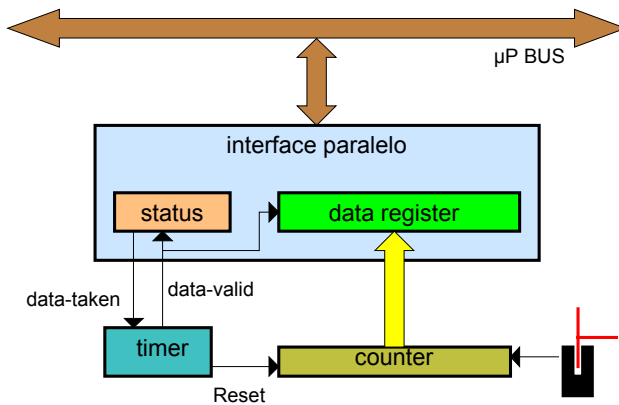


1º.- En el comienzo de un ciclo, el timer inicializa el contador.

2º.- Después de cada segundo el timer manda una señal "data-valid". Con esta señal, carga el contenido del contador en el registro de datos, y además pone un BIT de estado (status BIT) a 1, indicando que el contenido del registro de datos contiene información para ser leída por el μ P. El timer resetea al contador y comienza un nuevo ciclo.

3º.- Una rutina I/O espera a que el BIT del registro de estado se ponga a 1, en ese momento, lee el registro de datos, pone un 0 en el BIT del registro de estado y procesa el contenido del registro de datos.

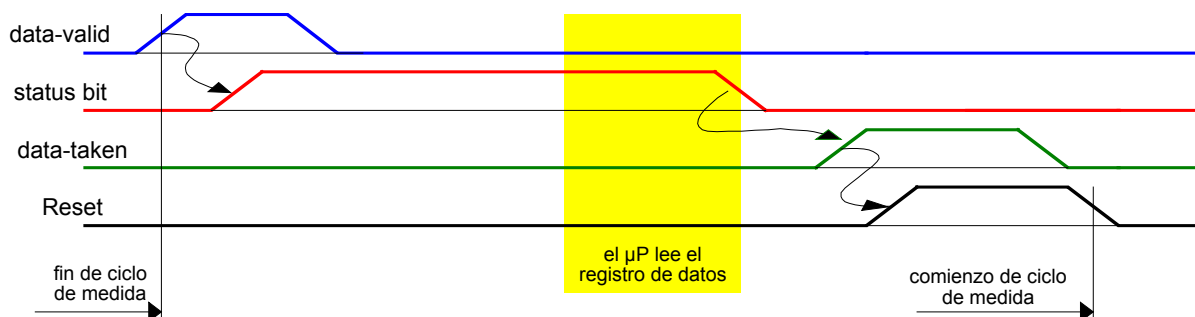
- En el ejemplo, el μ P lee información que sucedió en el exterior del sistema un tiempo anterior a la acción de lectura. Se supone que cuando una muestra está lista, el μ P ha leído la muestra anterior, cosa que no tiene porqué ser cierta (este hecho va a depender de la velocidad de los cálculos de procesamiento, de las tareas que tiene que realizar el micro, etc.). También, según lo definido hasta ahora en el circuito, no existe manera alguna de que el μ P indique al circuito externo, cuando está dispuesto a recibir más información. Es decir, hasta ahora solo habla el dispositivo externo.



Se supone que cuando una muestra está lista, el μ P ha leído la muestra anterior, cosa que no tiene porqué ser cierta (este hecho va a depender de la velocidad de los cálculos de procesamiento, de las tareas que tiene que realizar el micro, etc.). También, según lo definido hasta ahora en el circuito, no existe manera alguna de que el μ P indique al circuito externo, cuando está dispuesto a recibir más información. Es decir, hasta ahora solo habla el dispositivo externo.

- Para solucionar la falta de "diálogo" entre los dos subsistemas, se puede añadir otra línea (*data-taken*) entre el timer y la interfaz, mediante la cual el μ P indique a aquel que la información anterior ha sido leída y que está preparado para recibir nuevos datos. De esta manera, no se mandaran nuevos datos hasta que los anteriores hayan sido leídos por el microcomputador (no serán sobrescritos los datos anteriores). El ciclo va a estar condicionado por la velocidad o tareas del μ P.

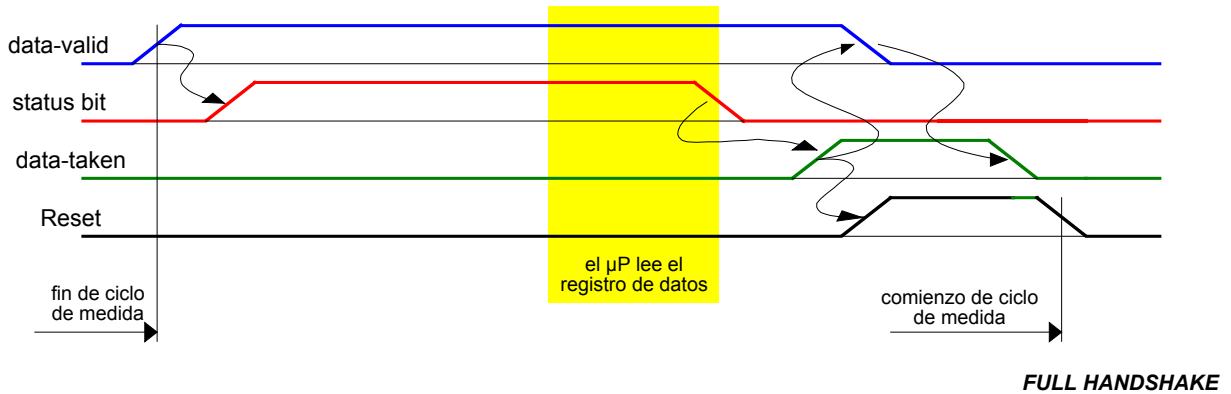
La secuencia de sucesos, expresada en forma gráfica, que va a tener lugar en este nuevo circuito puede ser la siguiente:



PULSE MODE HANDSHAKE OR PARTIAL HANDSHAKE.

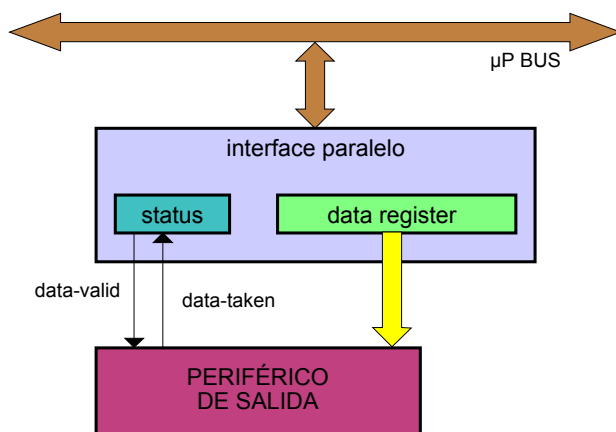
El timer envía la señal *data-valid* a la interfaz, y acto seguido, después de un tiempo determinado vuelve a poner esta línea en su estado inactivo. En este caso el timer no puede saber si su aviso ha sido reconocido o no, simplemente se limita a iniciar el proceso mediante un impulso de duración fija. Esta forma de dialogo se llama *Pulse Mode Handshake* o *Partial Handshake*.

Otra posible forma de realizar la anterior tarea puede ser la siguiente:



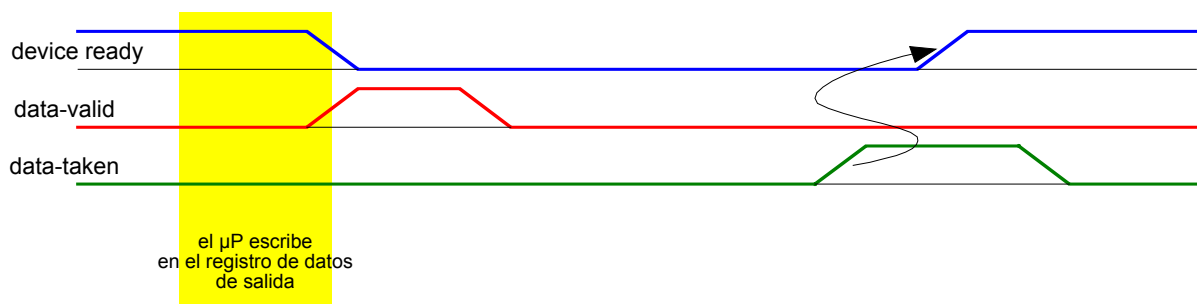
En este caso, *data-valid* no se desactiva hasta que el μP ha leído los datos. La duración de este impulso va a depender de la velocidad o tareas del micro. Es una forma de indicar que la señal de *data-valid* ha sido reconocida por la interfaz. Este tipo de handshake se llama *Full Handshake*.

SALIDA.

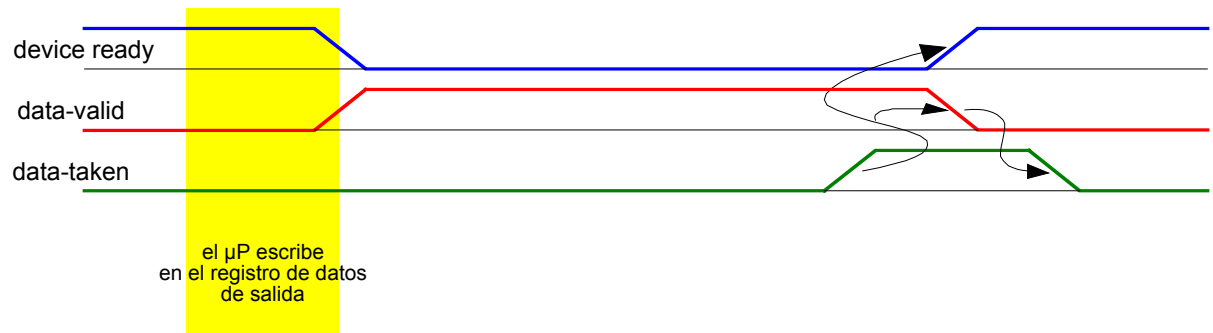


El *status BIT* en este caso se sustituye por un BIT llamado *device ready*, que indica el estado del dispositivo; Cuando éste está dispuesto a recibir datos, el BIT se pone a 1. Cuando el μP escribe datos en el registro de datos, la interfaz pone a 0 el BIT de estado y envía un impulso en la línea de *data-valid*. Cuando el dispositivo externo finaliza su operación, envía una señal de *data-taken*, haciendo que el BIT de estado se ponga otra vez a 1.

Una posible secuencia de operaciones dada en forma grafica es la que se muestra en la siguiente figura.

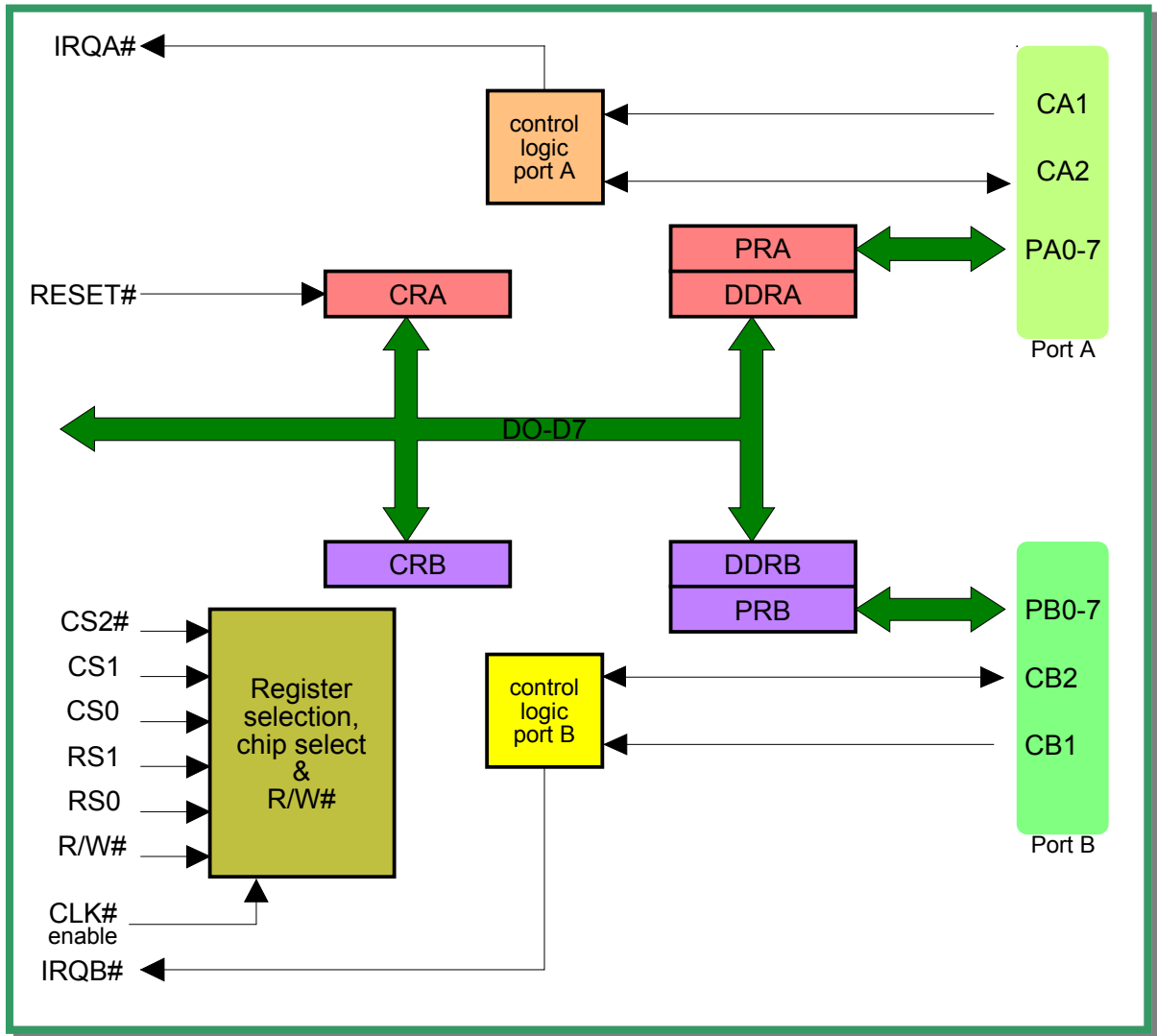


Otra posible secuencia será:



FULL HANDSHAKE

10.6-INTERFACE PARALELO. PIA (PERIPHERAL INTERFACE ADAPTER)



6821 PIA (PERIPHERAL INTERFACE ADAPTER)

- ❑ La PIA se conecta con el bus del μP por medio de las líneas $CS0$, $CS1$, $CS2\#$, $RS0$, $RS1$, $R/W\#$, $D0-D7$, $RESET\#$, $IRQA\#$, $IRQB\#$ Y $CLK\#$.
- ❑ Posee dos puertos, cada uno de los cuales, tiene 8 líneas de datos (PA_{0-7} y PB_{0-7}), y dos líneas de control ($CA1$ y $CA2$ para el puerto A, y $CB1$ y $CB2$ para el puerto B).
- ❑ Cada puerto tiene asociados tres registros direccionables:

DATA DIRECTION REGISTER A (**DDRA**) -PORT A
 DATA DIRECTION REGISTER B (**DDRB**) -PORT B
 PERIPHERAL REGISTER A (**PRA**) -PORT A
 PERIPHERAL REGISTER B (**PRB**) -PORT B
 CONTROL REGISTER A (**CRA**) -PORT A
 CONTROL REGISTER B (**CRB**) -PORT B

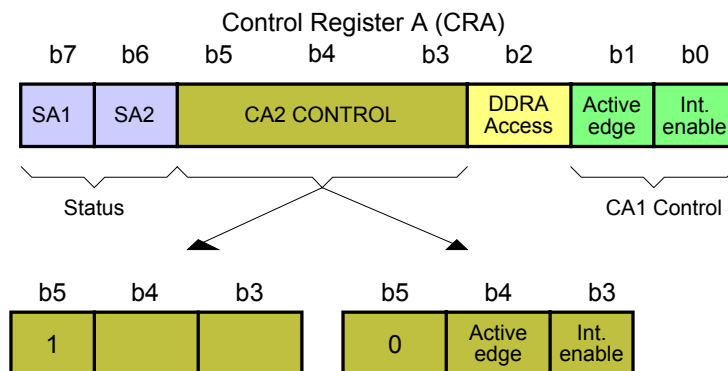
- ❑ Los dos **PR-s** (Peripheral Registers), se usan para transferir los datos desde y hacia el dispositivo de I/O. Una operación de lectura de estos registros, transfiere los datos hacia el bus del μP . Una operación de escritura a la misma posición carga el registro de datos correspondiente con un byte procedente del bus.

- ❑ No hay *latches* de entrada en los puertos de periféricos. Esto quiere decir que cuando **PRA** o **PRB** son seleccionados para ser leídos, los datos que van a ser transferidos hacia el bus del μP , serán los que en ese momento se encuentren en las líneas del puerto de datos.
- ❑ Para las operaciones de salida si hay un *buffer*.
- ❑ Los **DDR** permiten programar la dirección del flujo de datos de forma individual para cada línea. Un 1 en el BIT correspondiente, programa como salida la línea del puerto asociada a ese BIT. Un 0 lo hace como entrada.
- ❑ La selección de cada uno de los seis registros se realiza mediante las dos líneas de selección RS0 y RS1, tal como se muestra en la tabla. Estas líneas deberían conectarse a las líneas menos significativas del *address bus*.

RS1	RS0	POSICIÓN SELECCIONADA
0	0	PRA (SI $CRA_2 = 1$) DDRA (SI $CRA_2 = 0$)
0	1	CRA
1	0	PRB (SI $CRB_2 = 1$) DDRB (SI $CRB_2 = 0$)
1	1	CRB

CONTROL REGISTER A (CRA).

(EL PRINCIPIO DE FUNCIONAMIENTO PARA CRB ES ANÁLOGO AL DE CRA).



❑ SA1 (b7).

-Se pone a 1 con el flanco activo de CA1.

-Se pone a 0 cuando PRA es leído.

❑ SA2 (b6).

-Se pone a 1 con el flanco activo de CA2.

-Se pone a 0 cuando PRA es leído.

❑ DDRA Access (b2).

-Si está a 1, RS1-RS0 = 0-0 se refiere a PRA.

-Si está a 0, 0-0 se refiere a DDRA.

CA1 CONTROL

❑ Active edge. (b1)

-0 indica que el flanco activo en CA1 es el negativo.

-1 indica que el flanco activo en CA1 es el positivo.

❑ Int. enable. (b0)

-0 inhabilita la interrupción.

-1 indica que una transición activa en la línea CA1 activará **IRQA#** y pondrá **SA1** a 1. La petición de interrupción va a permanecer activa hasta que el μP lea el registro de datos del puerto a (PRA).

b1	ACT. EDGE
0	BAJADA
1	SUBIDA

b0	INT.
0	DISABLE
1	ENABLE

CA2 CONTROL.

b5 = 0 indica que CA2 funciona como entrada.

□ Active edge. (b4).

- 0 indica que el flanco activo en CA2 es el negativo.
- 1 indica que el flanco activo en CA2 es el positivo.

b4	ACT. EDGE
0	BAJADA
1	SUBIDA

□ Int. enable. (b3).

- 0 inhabilita la interrupción.
- 1 indica que cuando hay una transición activa en CA2 pondrá SA2 a 1 y va a activar IRQA#. SA2 y IRQA# van a permanecer activos hasta que el μP lea el registro de datos (PRA).

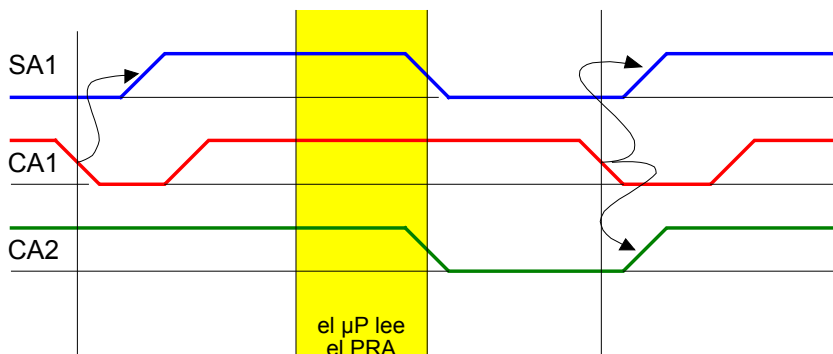
b3	INT.
0	DISABLE
1	ENABLE

b5 = 1 indica que CA2 funciona como salida.

- En este caso la función de b3 y b4 es la mostrada en la tabla.

b4	b3	
0	0	Full handshake para dispositivo de entrada
0	1	Low pulse cuando PRA se lee
1	0	CA2 $\rightarrow$ 0
1	1	CA2 $\rightarrow$ 1

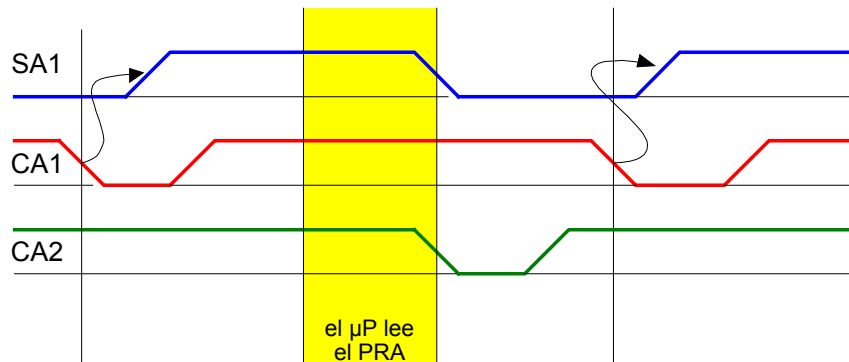
- SA2 no tiene significado y es igual a 0.
- La línea CA2 como salida, junto con la línea CA1 (CA1 es siempre línea de entrada) pueden desarrollar el handshake necesario para controlar las transferencias de entrada en el puerto A. En este caso b5, b4 y b3 de CRA, tienen que ser respectivamente 1, 0, y 0:
- CA2 está normalmente en su estado inactivo. Un dispositivo de entrada pone datos en las líneas de entrada del puerto A y activa CA1, (que puede ser la línea de data-valid)
- Como resultado el BIT SA1 se pone a 1.



- Después de que el μP lee los datos de entrada, la PIA activa CA2 (esta línea es la equivalente a data-taken). CA2 va a permanecer en este estado activo, hasta que se reciba otra transición en CA1, indicativo de que

hay mas datos disponibles en la entrada. (Los cronogramas de la figura, realizan un full handshake según el manual del fabricante).

- La línea CA2 puede ser usada para enviar un impulso a un dispositivo I/O. Si ponemos los bits b5, b4 y b3 del CRA a 1, 0 y 1 respectivamente, un impulso bajo de duración igual a un periodo de reloj, se transmite por la línea CA2 cada vez que el μP lee el PRA. Una aplicación para este modo puede ser la de implementar un partial handshake.



periodo de reloj, se transmite por la línea CA2 cada vez que el μP lee el PRA. Una aplicación para este modo puede ser la de implementar un partial handshake.

DIFERENCIAS DE FUNCIONAMIENTO ENTRE CRA Y CRB.

- El *full handshake* y el *pulse mode handshake* se aplican para la operación de salida en el caso del puerto B (CB1, CB2). Es decir, en el caso del puerto B, CB2 hace la función de *data-valid*, la cual se activa cuando el μP escribe nuevos datos en el PRB. El dispositivo de salida responde con *data-taken* en CB1.

RESET

- Cuando se activa esta línea, CRA, CRB, DDRA y DDRB se llenan con 0-s. Como resultado, todas las líneas bidireccionales de la PIA se configuran como entradas.

PROGRAMA EJEMPLO:

```

DDRB EQU    $EA0005
PRB   EQU    $EA0005
CRB   EQU    $EA0007
      CLR.B   CRB
      MOVE.B  #$FF,DDRB
      MOVE.B  #$24,CRB
      MOVEA   #RESULTS,A0
LAZO  MOVE.B  (A0)+,D0
      BEQ     FIN
      BSR     SEND
      BRA     LAZO
FIN   .
      .
SEND  BTST.B  #$80,CRB
      BEQ     SEND
      MOVE.B  D0,PRB
      RTS

```

PROGRAMA PARA EL μP 68000 QUE ENVÍA UN MENSAJE A UNA IMPRESORA POR MEDIO DEL PUERTO B DE UNA PIA.

CONSIDERACIONES HARDWARE.

- ❑ Cuando se utilice *handshake signalling*, un dispositivo de entrada debe ser conectado al puerto A y un dispositivo de salida, al puerto B.

PULSE MODE HANDSHAKE.

- ❑ El programa que controla la operación de entrada, lee el BIT de estado SA1 de forma repetida hasta que éste se hace 1. En ese instante, el programa lee los datos de entrada del PRA.

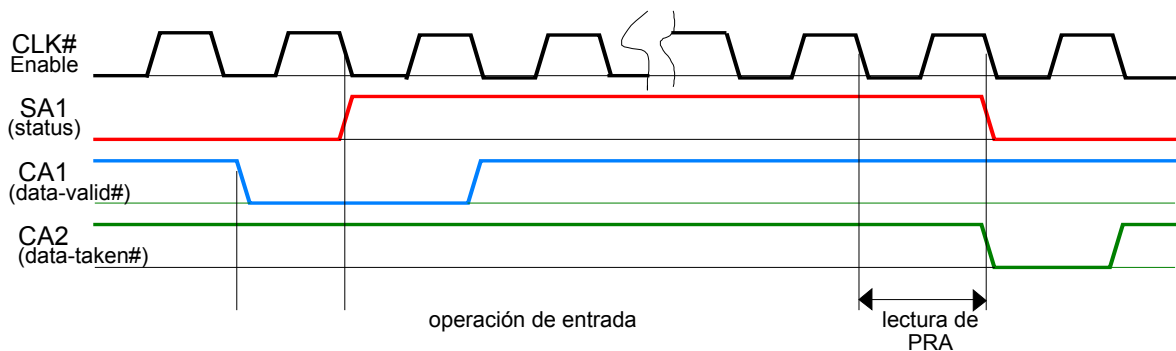
-La activación de SA1 sucede al final del periodo de reloj en el cual el estado activo de CA1 es reconocido.

-Después el μP lee el CRA para chequear el BIT SA1.

-Acto seguido, al verificar que SA1 es 1, el μP lee PRA.

-Al terminar la operación de lectura, la PIA pone a 0 el BIT de estado y manda un impulso de duración un periodo de reloj por CA2, indicando al dispositivo de entrada que el μP ha leído los datos.

-El tiempo en el cual el flanco activo en CA1 es recibido no guarda ninguna relación con los flancos del enable (CLK). CA1 es asíncrono respecto al enable. Por esta razón la duración de CA1 en su estado activo debe ser superior a un periodo de reloj, para que de esta manera pueda ser reconocida.



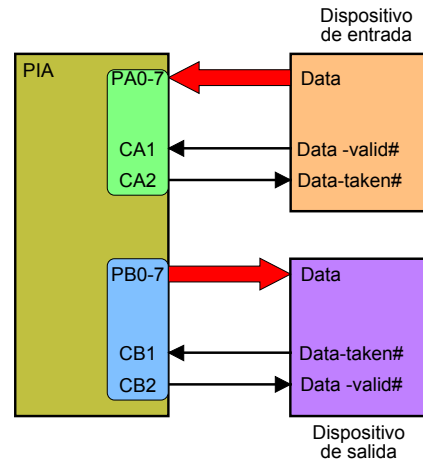
- ❑ En la operación de salida, el μP chequea el BIT de estado SB1, entonces escribe los datos en el PRB y pone a 0 el BIT de estado, SB1. En este momento los datos aparecen en las líneas de salida del PB.

-La PIA espera un tiempo de medio periodo de CLK, para que los datos esten "asentados" y activa CB2, indicando que los datos están disponibles en el puerto.

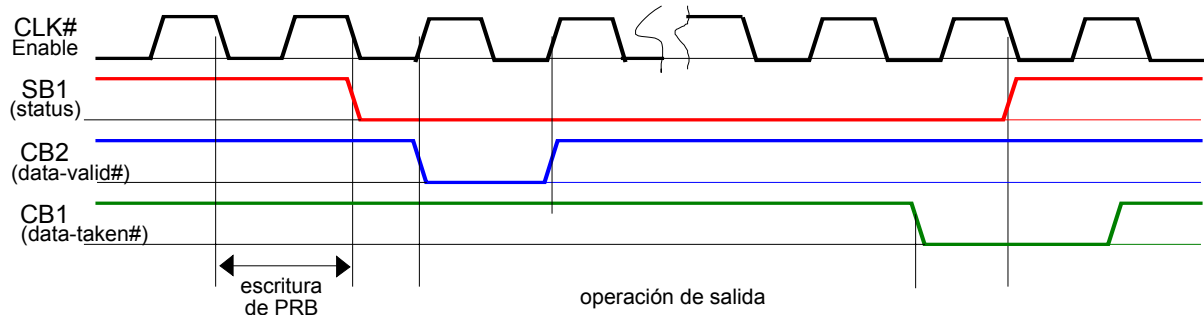
-Cuando el dispositivo de salida ha leído el puerto, manda una señal por CB1. Esta señal hace que el BIT de estado, SB1 vuelva a 1 al final del periodo de reloj en el que sucede data-taken, indicando que el dispositivo está listo para otra operación de salida.

-El tiempo en el cual el flanco activo en CB1 es recibido no guarda ninguna relación con los flancos del enable (CLK). CB1 es asíncrono respecto al enable.

-Para garantizar que CB1 es reconocida, el impulso mandado por el dispositivo en CB1 debe ser de más duración que un periodo del CLK.

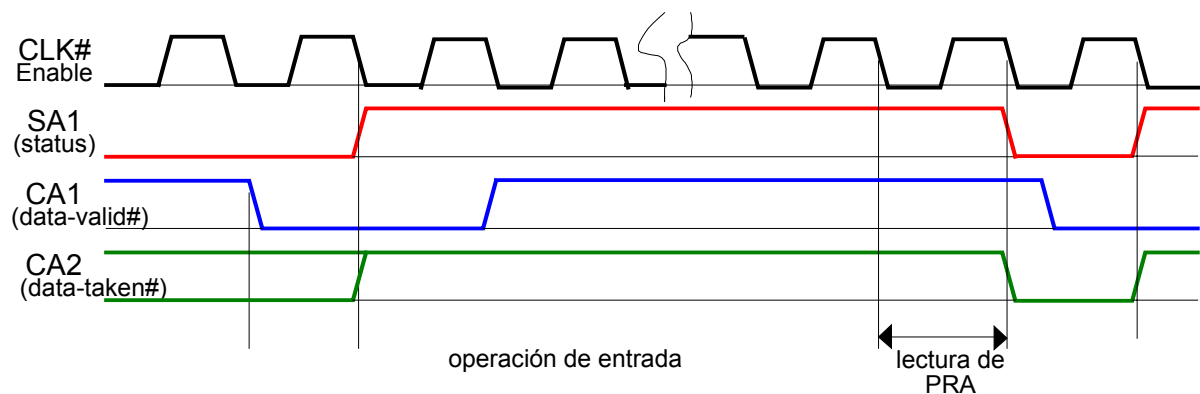


La duración de los impulsos en las líneas CA2 y CB2 tienen una duración fija de un periodo de reloj. (*pulse mode handshake.*)



FULL HANDSHAKE.

□ Operación de entrada:



-El dispositivo de entrada hace una petición a través de la línea CA1. Esta petición indica que el dispositivo ha colocado datos en las líneas del puerto de entrada.

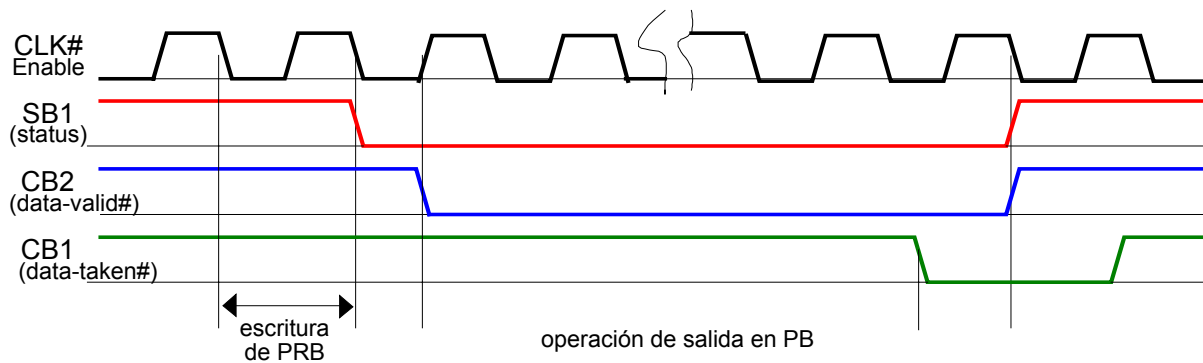
-La señal en CA1, hace que se active el BIT de estado SA1. En este mismo instante, la PIA desactiva CA2 independientemente del estado previo que tuviera esta señal.

-El μP chequea el registro de estado y lee los datos de entrada. Una vez leídos, la PIA desactiva el BIT de estado SA1 y activa CA2. Esta activación de CA2, informa al dispositivo de que los datos han sido leídos por el μP .

-El dispositivo de entrada debe poner CA1 a su estado inactivo antes de que CA2 pase a activo.

-CA2 se mantiene activa hasta que el dispositivo de entrada señala una nueva transferencia de entrada activando CA1. La PIA responde poniendo SA1 a 1 y desactivando CA2...

□ Operación de salida:



-Después de chequear el BIT de estado SB1, el μP escribe los datos de salida en el registro PRB. Al final de la escritura, la PIA pone a 0 SB1.

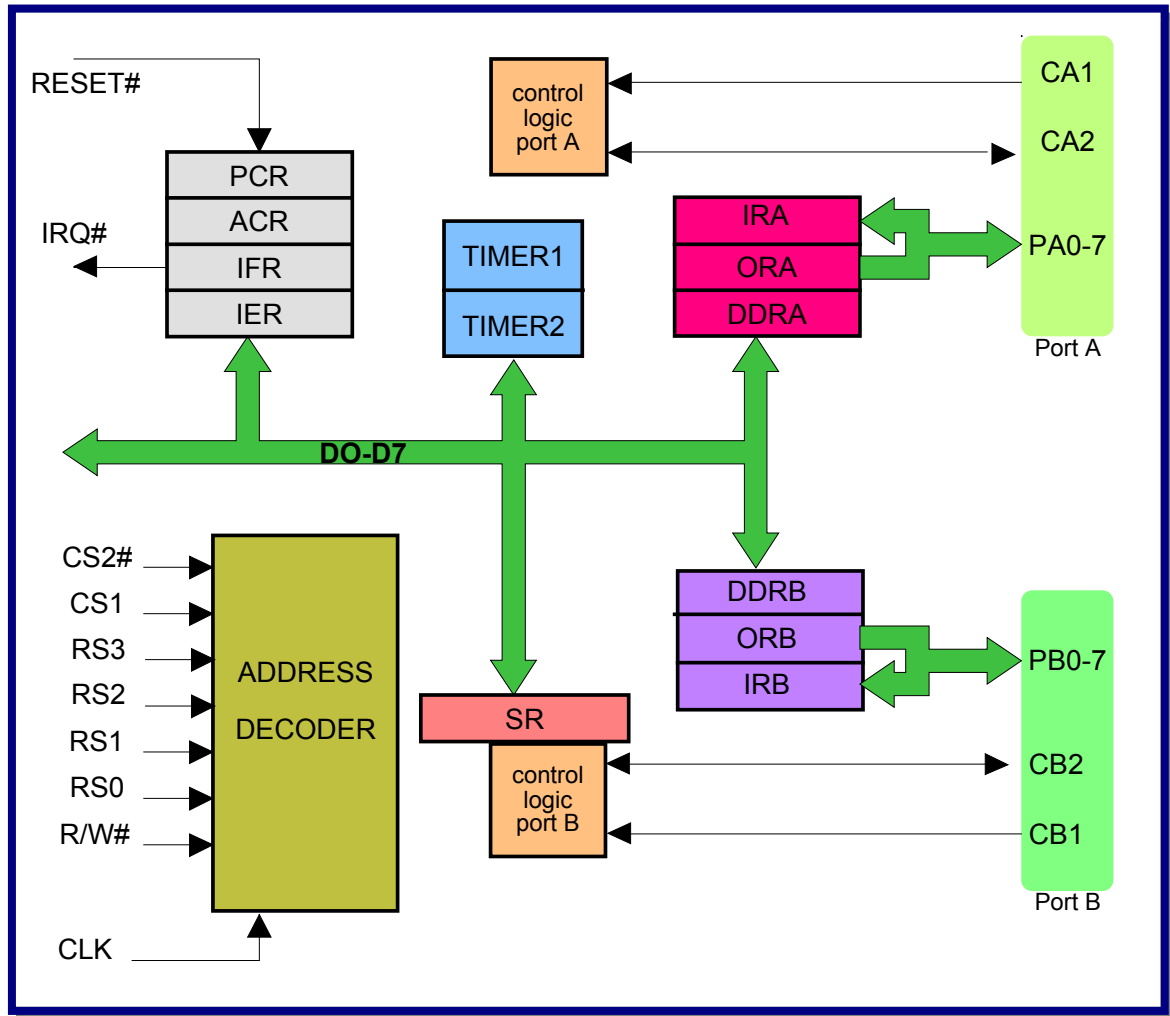
-Un semiperiodo de reloj después (flanco de subida), la PIA activa CB2.

-Cuando los datos han sido aceptados por el dispositivo de salida, éste activa la línea CB1. Como respuesta, la PIA pone CB2 y el BIT de estado, en estado alto.

-El dispositivo de salida debe poner la línea CB1 a su estado inactivo, antes de que la PIA active CB2 para indicar el comienzo de una nueva transferencia.

-El impulso resultante en CB1, debe ser de mayor duración que un periodo de reloj.

10.7-INTERFACE PARALELO. VIA (VERSATILE INTERFACE ADAPTER)



6522 VIA (VERSATILE INTERFACE ADAPTER)

- ❑ Las líneas del lado izquierdo en la figura, conectan la VIA con el bus del μP .
- ❑ Cada puerto tiene asociado
 - un registro de entrada:

IRA (*INPUT REGISTER A*) para PA y **IRB** (*INPUT REGISTER B*) para PB.
 - un registro de salida:

ORA (*OUTPUT REGISTER A*) para PA y **ORB** (*OUTPUT REGISTER B*) para PB.
 - un registro de dirección de los datos:

DDRA (*DATA DIRECTION REGISTER A*) para PA y **DDRB** (...) para PB.
- ❑ Común a ambos puertos, tiene los siguientes registros de información de estado y control:

PCR (*PERIPHERAL CONTROL REGISTER*), **ACR** (*AUXILIARY CONTROL REGISTER*), **IFR** (*INTERRUPT FLAG REGISTER*) y **IER** (*INTERRUPT ENABLE REGISTER*).
- ❑ El puerto B tiene además asociado un registro de desplazamiento **SR** (*SHIFT REGISTER*)

- Tiene además 6 registros que están asociados con los dos *timers*.
- La selección de los registros se realiza mediante las líneas RS0 a RS3:

RS3	RS2	RS1	RS0	READ	WRITE	REGISTER
0	0	0	0	IRB	ORB	
0	0	0	1	IRA	ORA	
0	0	1	0	DDRB		
0	0	1	1	DDRA		
0	1	0	0	T1C-L		T1 COUNTER
0	1	0	1	T1C-H		
0	1	1	0	T1L-L		T1 LATCH
0	1	1	1	T1L-H		
1	0	0	0	T2C-L		T2 COUNTER
1	0	0	1	T2C-H		
1	0	1	0	SR		
1	0	1	1	ACR		
1	1	0	0	PCR		
1	1	0	1	IFR		
1	1	1	0	IER		
1	1	1	1	IRA	ORA	

- La dirección de entrada y salida de datos es la misma para cada puerto. La selección de uno u otro se realiza mediante la línea de R/W#. Los registros de entrada se seleccionan mediante la operación de lectura y los registros de salida mediante la operación de escritura.
- Cada puerto posee 8 líneas. Cada una de estas líneas puede ser programada de forma individual como entrada o como salida. Para ello, se debe poner el BIT asociado a cada una de ellas en el DDR, a 0 (entrada) o a 1 (salida).
- Además, cada puerto posee también dos líneas, CA1 y CA2 el puerto A y CB1, CB2 el puerto B. La principal misión de estas

líneas es la de ejecutar el handshake en la transferencia de datos.

- Para la combinación 1111, el efecto es el mismo que para la combinación 0001, pero en el caso de 1111, no existe protocolo CA1, CA2

FORMATO DE LOS REGISTROS DE ESTADO Y CONTROL.

- El IFR contiene la información de estado de cualquier operación de I/O, no solamente de las interrupciones.

En la siguiente tabla se describen los modos de operación de las líneas de control **CA1** y **CA2** del puerto **A**. Un modo de operación en particular, se selecciona con los campos **MA1** y **MA2** del **PCR**. Los campos **MB1** Y **MB2** determinan el modo de operación de las líneas de control del puerto **B**.

Para cada código de MA1 y MA2, en la tabla de la página siguiente se da el modo de operación de la línea de control (CA1, CA2) afectada y las condiciones bajo las cuales el BIT correspondiente en el registro de estado (IFR), es puesto a 1 o a 0.

b7	b6	b5	b4	b3	b2	b1	b0
MB2			MB1		MA2		MA1

Peripheral Control Register (PCR)

b7	b6	b5	b4	b3	b2	b1	b0
TC			SRC			LB	LA

Auxiliary Control Register (ACR)

b7	b6	b5	b4	b3	b2	b1	b0
IRQ	T1	T2	SB1	SB2	SR	SA1	SA2

Interrupt Flag Register (IFR)

b7	b6	b5	b4	b3	b2	b1	b0
S/C#	T1	T2	CB1	CB2	SR	CA1	CA2

Interrupt Enable Register (IER)

REGISTRO PCR: MODOS DE OPERACIÓN DE LAS LÍNEAS DE CONTROL DEL PUERTO A¹.

CAMPO DE CONTROL		LÍNEA DE CONTROL		BANDERA DE ESTADO		
NOMB	VALOR	NOMB	MODO DE OPERACIÓN	NOM	SET	CLEARED
MA1	0	CA1	ENTRADA; ACTIVO FLANCO DE BAJADA	SA1	TRANSICIÓN H-L EN CA1	LECTURA IRA, ESCRITURA ORA O ESCRITURA DE 1 EN IFR ₁
	1		ENTRADA; ACTIVO FLANCO DE SUBIDA		TRANSICIÓN L-H EN CA1	
MA2	000	CA2	ENTRADA; ACTIVO FLANCO DE BAJADA	SA2	TRANSICIÓN H-L EN CA2	LECT. IRA, ESCR. EN ORA, O ESC. DE 1 EN IFR ₀
	001		ENTRADA; ACTIVO FLANCO DE BAJADA		TRANSICIÓN H-L EN CA2	
	010		ENTRADA; ACTIVO FLANCO DE SUBIDA		TRANSICIÓN L-H EN CA2	LECT. IRA, ESCR. EN ORA, O ESC. DE 1 EN IFR ₀
	011		ENTRADA; ACTIVO FLANCO DE SUBIDA		TRANSICIÓN L-H EN CA2	
	100		SALIDA; ACTIVO NIVEL BAJO. FULL HANDSHAKE ²			ESCRITURA DE 1 EN IFR ₀
	101		SALIDA; PULSO BAJO ³ . POSIBLE PARTIAL HANDSHAKE ⁴			
	110		SALIDA; PUESTA A BAJA			
	111		SALIDA; PUESTA A ALTA			

¹ EL PUERTO B TRABAJA DE LA MISMA FORMA EXCEPTUANDO LAS SIGUIENTES NOTAS.

² CA2 PASA A BAJO SIGUIENDO UNA OPERACIÓN DE LECTURA DE IRA, Y PASA A ALTO EN EL SIGUIENTE FLANCO ACTIVO EN CA1. CB2 TRABAJA DE LA MISMA FORMA PERO PARA UNA OPERACIÓN DE ESCRITURA EN ORB.

³ UN IMPULSO BAJO SE TRANSMITE POR CA2 DURANTE UN PERIODO DE RELOJ SIGUIENDO UNA OPERACIÓN DE LECTURA EN IRA. CB2 TRABAJA DE LA MISMA FORMA PERO PARA UNA OPERACIÓN DE ESCRITURA EN ORB.

⁴ CA1 (CB1) Y CA2 (CB2) OPERAN DE FORMA INDEPENDIENTE EN ESTE MODO, PERO JUNTOS PUEDEN SER USADOS PARA IMPLEMENTAR UN PARTIAL HANDSHAKE.

REGISTRO ACR.

CAMPO DE CONTROL		DESCRIPCIÓN
LA	0	SELECCIONA ENTRADAS "UNLATCHED" DURANTE UNA OPERACIÓN DE LECTURA EN PA.
	1	SELECCIONA ENTRADAS "LATCHED" DURANTE UNA OPERACIÓN DE LECTURA EN PA.
LB	0	SELECCIONA ENTRADAS "UNLATCHED" DURANTE UNA OPERACIÓN DE LECTURA EN PB.
	1	SELECCIONA ENTRADAS "LATCHED" DURANTE UNA OPERACIÓN DE LECTURA EN PB.
SRC		SHIFT REGISTER CONTROL (VER MAS ADELANTE)
TC		TIMER CONTROL (VER MAS ADELANTE)

- Un flanco activo en CA1 hace que la bandera SA1 se ponga a 1. También, si la entrada mediante latch está habilitada ($LA=1$), la información en las líneas PA_{7-0} se carga en el registro de entrada, IRA. Por lo tanto, LA tiene que ser puesto a 1 si queremos capturar la información de las líneas del puerto en el momento que CA1 es activado.

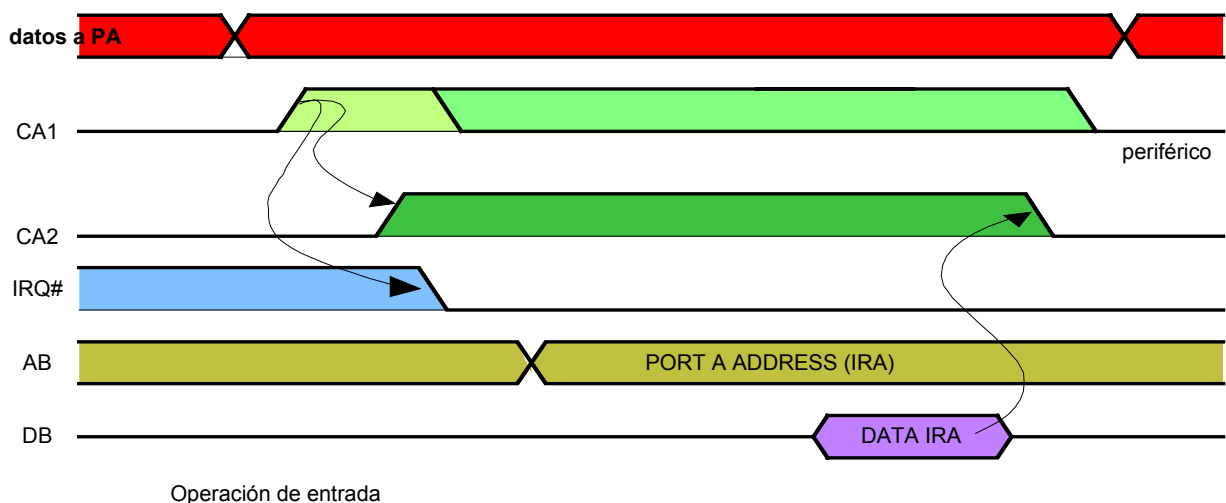
Cuando se realiza una lectura de IRA, la información que se va a leer, es la que se cargó en IRA cuando se activó CA1, no la información que esté presente en los terminales del puerto en ese momento de lectura de IRA.

- Si $LA=0$ el único efecto que tiene una transición activa en CA1 es poner a 1 el flag de estado correspondiente, SA1. Si se realiza una lectura de IRA, la información que se va a leer es la que en el momento de la lectura, está en los terminales del puerto A. En cualquier caso, el flag SA1 permanece en 1 hasta que es puesto a 0 por una instrucción del programa que lee IRA o escribe en ORA. Este flag también puede ser puesto a 0 directamente escribiendo un 1 en el BIT 1 del registro IFR.

Las líneas de control CB1 y CB2 trabajan de la misma forma que CA1 y CA2 respectivamente, exceptuando los dos modos de handshake. Cuando el campo MB2 es puesto a 100 o 101, la línea CB2 se activa siguiendo una operación de escritura en ORB, en lugar de una de lectura en IRB. Por lo tanto, cuando se quiera usar alguno de los modos de handshake para controlar la transferencia de información entre la VIA y un dispositivo de I/O, el dispositivo de entrada se debe conectar al PA, mientras que el dispositivo de salida se debe conectar a PB.

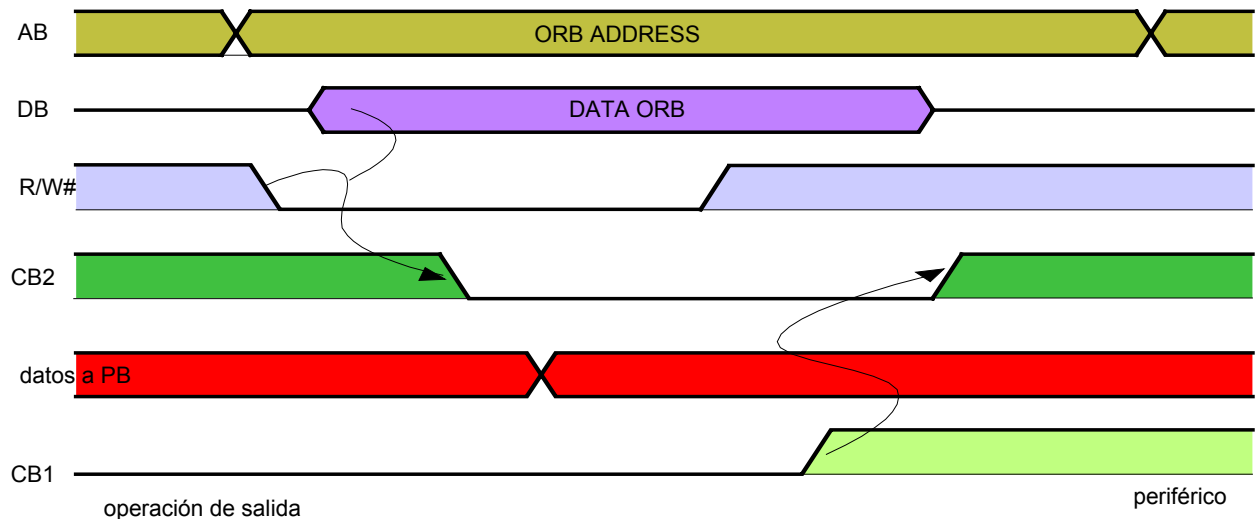
Cuando se realiza la conexión del puerto A como entrada y el puerto B como salida, la secuencia de entrada es la siguiente:

1. El dispositivo externo manda datos al puerto A y activa la línea CA1.
2. La VIA activa CA2 al recibir la señal CA1.
3. La CPU detecta la activación de CA1 y lee el puerto A cuando convenga. Al leer el puerto, la VIA desactiva CA2.
4. Si CA1 del IER está en 1, se producirá interrupción (IRQ#) hacia la CPU.



La secuencia en una operación de salida es:

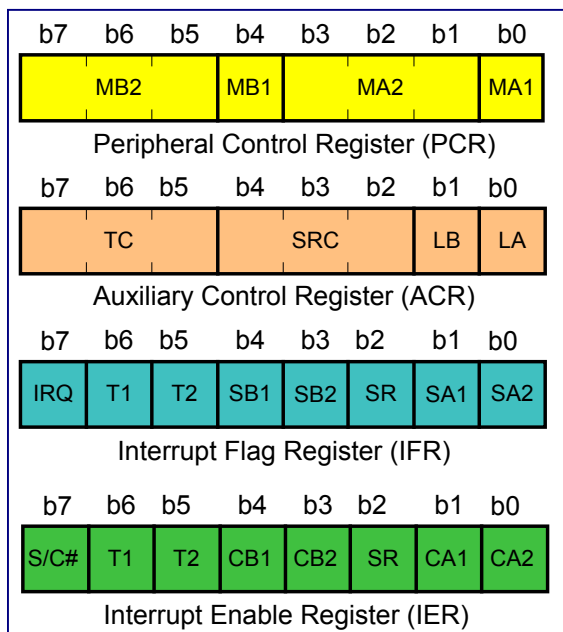
- 1. La CPU escribe en ORB el dato que se va a transferir. De forma automática la línea CB2 se pone a cero.
- 2. La puesta a cero de CB2, alerta al periférico sobre la presencia de datos en el puerto B.
- 3. El periférico toma los datos de PB y activa CB1. Al recibir la VIA CB1, desactiva CB2.



- La activación de la línea de RESET# hace que los registros de estado, control y DDR se llenen de 0-s. Como resultado todas las líneas bidireccionales, se configuran como entradas, evitando la activación de cualquier dispositivo conectado a las salidas.

INTERRUPCIONES, TIMERS Y SR.

REGISTRO IFR



BIT	NOMB	DESCRIPCIÓN
0	SA2	VER DOS PÁG. .ANTES
1	SA1	VER DOS PÁG. .ANTES
2	SR	FLAG DE ESTADO PARA SR
3	SB2	VER DOS PÁG. .ANTES
4	SB1	VER DOS PÁG. .ANTES
5	T2	FLAG DE ESTADO PARA T2
6	T1	FLAG DE ESTADO PARA T1
7	IRQ	FLAG DE INTERRUPCIÓN

- El modo de operación de cada puerto de la VIA se controla mediante la información depositada en PCR y ACR.

- El registro (IFR) tiene 8 bits de estado o flags (b_0 - b_7): SA2 (b_0) y SA1 (b_1) para el puerto A, SR (b_2) para el registro de desplazamiento, SB2 (b_3) y SB1 (b_4) para el puerto B, T2 (b_5) y T1 (b_6) para los timers.

El BIT b_7 (IRQ) refleja el estado de la línea IRQ#. Siempre que IRQ# sea activa, el BIT b_7 es 1.

- La VIA puede manejar 7 posibles fuentes de interrupción que coinciden con los 7 bits menos significativos de IER. Estas fuentes son los dos timers, las dos líneas de control de cada puerto, y el Registro de desplazamiento.
- Para que una interrupción generada por una de estas fuentes se transmita al μP a través de la línea IRQ, primero, tiene que estar habilitada en el IER.
- Hay una correspondencia BIT a BIT entre IFR y IER. Una interrupción esta habilitada cuando el BIT correspondiente en el IER está puesto a 1. El IER actúa por lo tanto como máscara de interrupciones.

P EJ. Si se pone b_4 de IER (CB1) a 1, cualquier suceso que ocasione que el BIT asociado a CB1 en el IFR (SB1 en este caso) se ponga a 1, ocasionará también, que el flag IRQ se ponga a 1 y la línea IRQ# se active. La petición de interrupción va ha permanecer activa hasta que el BIT SB1 sea puesto a 0.

- Los bits del IFR no pueden ser puestos a 1 mediante una instrucción de escritura, pero si que pueden ser puestos a 0 (todos menos b_7) escribiendo un byte en el IFR que contenga 1 en los bits que se quieran poner a 0 en el registro.

USO DE IER PARA CONTROL DE LAS INTERRUPCIONES.

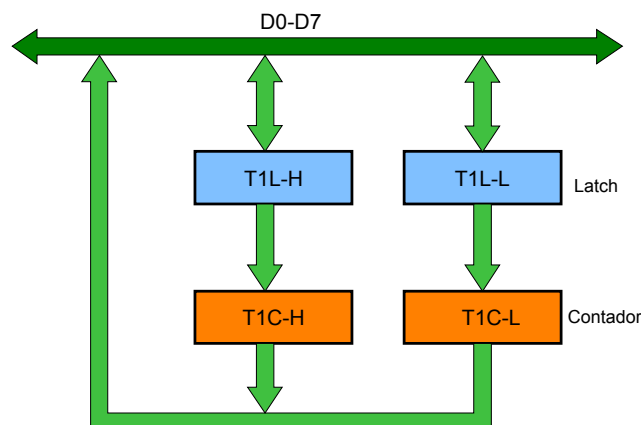
- La información a escribir en IER es interpretada de una forma especial. El b_7 (S/C# - SET/CLEAR#-) juega un papel muy importante en esta forma especial de trabajo. Cuando el patrón a escribir en IER tiene S/C# =1, los bits correspondientes del IER con los 1-s del patrón serán puestos a 1. Los demás bits del IER, no se verán afectados. De igual forma, cuando el patrón a escribir tiene S/C# =0, los bits correspondientes del IER con los 1-s del patrón serán puestos a 0. Cuando se realiza una lectura del IER, siempre se lee b_7 con 0.

P EJ. Para habilitar CB1 y CA2 debemos escribir en IER el patrón 10010001. Para inhabilitar esas mismas interrupciones debemos escribir en IER el patrón 00010001.

TIMERS.

- Los temporizadores pueden ser usados para producir retardos programados o controlar la operación del registro de desplazamiento, SR.

T1.



- El T1 consta de un latch de 16 bits, T1L, y de un contador descendente también de 16 bits, T1C.
- No hay camino directo para transferir datos desde el bus del μP al contador. El μP no puede escribir directamente en el contador, pero si puede leer en cualquier momento el contenido de este. Por otro lado, sí puede leer y escribir directamente en el latch.

- Para cargar un valor inicial en el contador, el μP escribe primero en el byte de menos peso, **T1C-L**. Sin embargo, los datos no van directamente al contador, se almacenan en el latch, T1L-L. De la misma forma, cuando el μP escribe en el **T1C-H**, los datos son cargados en T1L-H. Tan pronto como esta última transferencia se realiza, los 16 bits del latch son copiados automáticamente en el contador, y la operación del timer comienza (se dispara el timer).
- El valor inicial de la cuenta no cambia pues se mantiene en el latch. La variación de la cuenta ocurre sólomente en el contador descendente.
- El μP puede acceder al byte alto y bajo del latch en operaciones de escritura y lectura. Escribir en T1L-L, tiene el mismo efecto que escribir en TIC-L. Sin embargo, escribir en T1L-H carga nueva información en el latch, pero no la transfiere del latch al contador ni tampoco dispara el timer.

RS3-RS0	READ	WRITE	REGISTER	
0000	x		IRB	
		x	ORB	
0001	x		IRA	
		x	ORA	
0010	x	x	DDRB	
0011	x	x	DDRA	
0100	x		T1C-L	Transfiere el contenido de TIC-L al bus del μP Carga datos desde el bus del μP al T1L-L.
		x		
0101	x		T1C-H	Transfiere el contenido de T1C-H al bus del μP . Carga datos del bus del μP al T1L-H, después copia T1L en T1C y dispara el timer.
		x		
0110	x	x	T1L-L	Byte bajo del latch de T1
0111	x	x	T1L-H	Byte alto del latch de T1
1000	x		T2C-L	Transfiere el contenido de T2C-L al bus del μP . Carga datos del bus del μP al T2L-L.
		x		
1001	x		T2C-H	Transfiere el contenido de T2C-H al bus del μP . Carga datos del bus del μP al T2L-H, Transfiere T2L-L en T2C-L y dispara el timer.
		x		
1010	x	x	SR	
1011	x	x	ACR	
1100	x	x	PCR	
1101	x	x	IFR	
1110	x	x	IER	
1111	x		IRA	
no pt		x	ORA	

- El T1 tiene dos modos de operación: Cuenta única y cuenta continua.

En estos modos, el timer puede ser programado para producir una señal de salida en la línea PB₇ del puerto B.

- La selección de los modos de funcionamiento se realiza a través de los dos bits más significativos del ACR (b6 y b7).

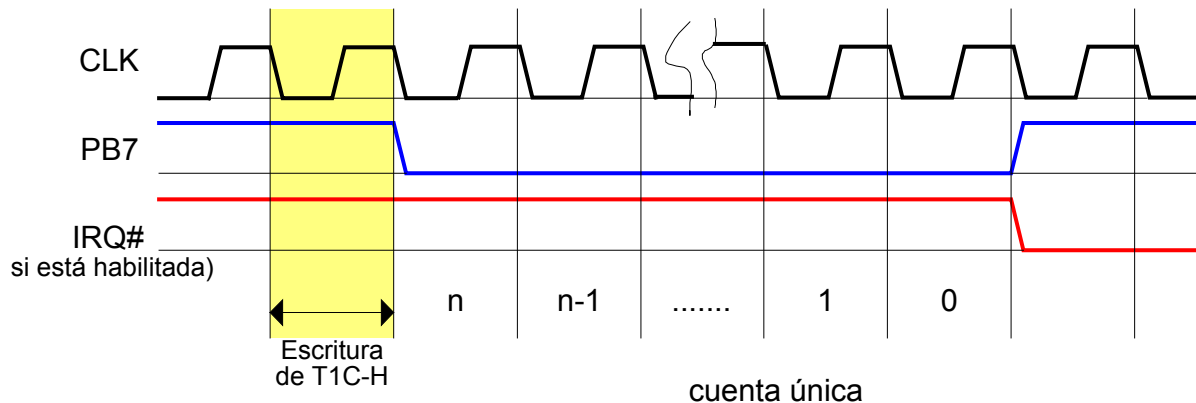
ACR

b7	b6	MODO	SALIDA PB7
0	0	C. ÚNICA	DISABLE
0	1	CONTINUA	DISABLE
1	0	C. ÚNICA	IMPULSO BAJO
1	1	CONTINUA	ONDA CUADRADA

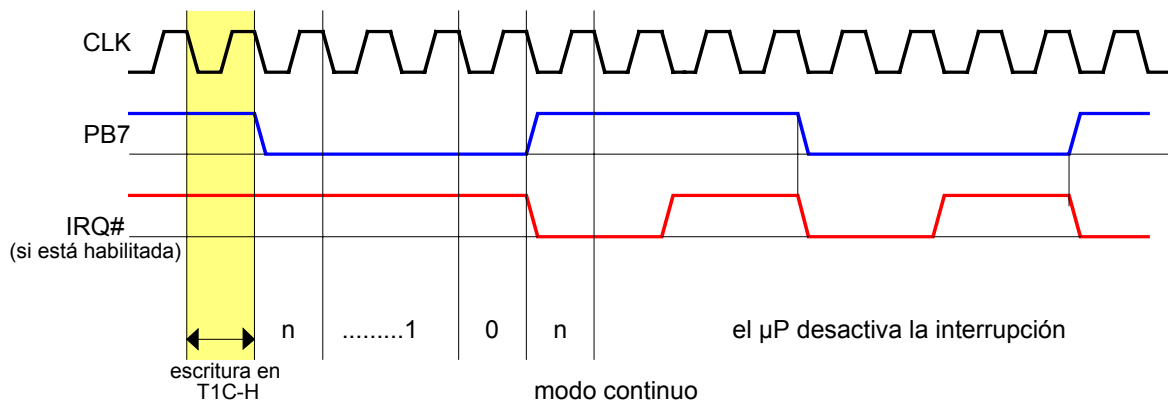
- El BIT 6 selecciona el modo continuo (b6=1) o modo de cuenta única (b6=0).
- Cuando el BIT 7 está en 1 se produce una señal de salida en PB₇, (en este caso la línea debe estar programada como línea de salida en el DDRB). Cuando se selecciona esta opción y antes de que se dispare el timer, la línea PB₇ pasa a un estado alto. Si el BIT 7 está a 0, las cuentas no tienen ningún efecto en la salida (PB₇).

- Si el BIT 7 de ACR o el BIT 7 de DDRB esta puesto a 0, la línea PB7 es independiente del timer y se comporta como cualquier otra línea del puerto B.

en las figuras siguientes se da la señal de salida en ambos modos de operación:



La línea PB7 pasa a nivel alto al poner ACR_7 a 1. Pasa a nivel bajo al final del periodo de reloj en el cual el μP escribe en el T1C-H para disparar el timer. Esta línea vuelve a estado alto al final del periodo de reloj en el cual el contador alcanza la cuenta 0000H. La duración del estado bajo en PB7 es de $n+1$ periodos de reloj, donde n es el valor de 16 bits que fue cargado en el contador. Al iniciarse la cuenta (escritura de T1C-H), el flag de T1 (BIT 6 de IFR), se pone a cero si es que estaba en 1. Al llegar al valor 0000H el flag de T1 se pone a 1. Una vez alcanzado el primer paso por 0000H, el temporizador continúa contando hacia atrás a partir del valor FFFFH. Los sucesivos pasos por cero no dan lugar a los procesos anteriores.

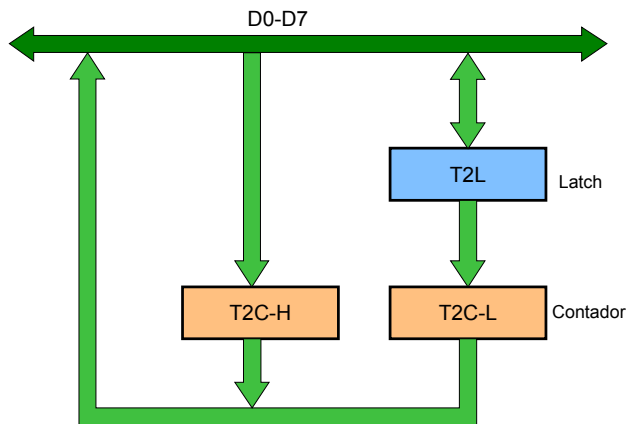


En modo continuo, PB7 cambia cada vez que el contador alcanza la cuenta 0000H. También se genera una petición de interrupción (si ésta está activada). La petición de interrupción es desactivada cuando el μP lee el contador o dispara un nuevo periodo de temporización. Cuando el contador llega a 0000H, continúa la cuenta desde el valor inicial que sigue almacenado en el latch.

- El estado del timer puede ser observado por el μP en cualquier momento, inspeccionando el BIT 6 del IFR. Este BIT se pone a 1 cuando el contador alcanza 0000H, y se pone a 0 cuando el μP lee el T1C-L o comienza un nuevo periodo de conteo al depositar un nuevo valor en T1C-H. También se puede poner a 0 escribiéndolo directamente en el IFR.
- Cuando el temporizador opera en modo continuo, el flag de T1 (BIT 6 de IFR) se pone a 1 cada vez que la cuenta pasa por 0000H, por lo tanto, inspeccionando de forma continua el IFR, el μP puede realizar ciertas tareas a intervalos regulares (Hay que tener en cuenta que

se debe hacer el reset de este BIT en cada periodo, posiblemente en la rutina de atención a la IRQ#). Esta misma labor puede realizarse mediante interrupciones (BIT 6 de IER a 1) en vez de inspeccionar el IFR a través de un lazo de espera.

T2



- ❑ Consta de un contador descendente de 16 bits y de un latch de 8 bits.
- ❑ El modo de operación puede ser también de cuenta única o continua. La selección se realiza mediante el BIT 5 del ACR.
- ❑ En el modo de cuenta única el funcionamiento es similar al T1. La operación del timer comienza de forma automática al escribir el byte de mayor peso en T2C-H.

- ❑ En el modo continuo, la operación del timer se controla a través de la línea 6 del puerto B (PB6), la cual debe ser programada como entrada. (DDRB₆=0).

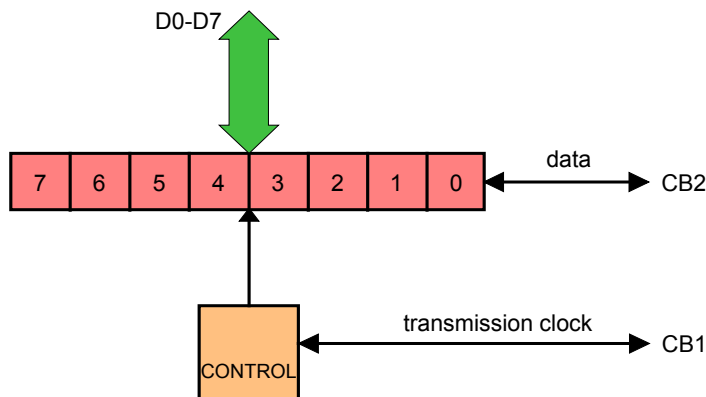
b5	MODO	
0	C. ÚNICA	SIMILAR A T2
1	CONTINUA	CADA FLANCO DESCENDENTE EN PB6 DECREMENTA EL COUNTER. CUANDO COUNTER LLEGA A 0, (IFR ₅ ← 1)

- ❑ La función que realiza ahora el T2 no es de temporizador. Simplemente cuenta los impulsos de entrada presentes en la línea PB6 y activa la bandera de estado cuando han sido recibidos un número determinado de los mismos.

El μP carga los bytes de menor peso y mayor peso de la cuenta deseada en T2 de la misma forma que en T1. Después de cargar T2C-H, cada flanco descendente que se recibe en PB6, decrementa en 1 el contenido del contador. Cuando el contador llega a 0000H, la bandera de T2 en IFR se activa y se genera una petición de interrupción (si IER₅ = 1). El contador sigue decrementándose por impulsos de entrada, como en el caso de T1. El flag T2 se borra al escribir en T2C-H o al leer T2L-L.

SR.(SHIFT REGISTER)

- ❑ El registro de desplazamiento puede ser usado para recibir o transmitir datos en serie a través de la línea CB2.



- ❑ El reloj de transmisión usado para desplazar los datos dentro o fuera del registro puede provenir de tres fuentes diferentes: (1) de la señal de reloj del bus del μP , (2) del T2 y (3) de una señal externa aplicada en CB1.

- ❑ En los casos en los cuales el reloj se genera en la VIA (los dos primeros), esta señal aparece en

CB1 para ser usada por el dispositivo externo que recibe o envía los datos en serie.

- ❑ Cuando se utiliza T2 como reloj de transmisión, éste funciona en modo continuo, recargando el contador con el valor de T2L-L (8 bits). Cada vez que la cuenta pasa por cero, se produce un desplazamiento.
- ❑ El control del registro de desplazamiento se realiza mediante los bits 2, 3 y 4 del ACR.
- ❑ La operación de desplazamiento se inicia al leer o escribir en el SR.
- ❑ La bandera de estado del SR (BIT 2 del IFR) se activa (exceptuando en el modo 100) después de haberse completado 8 desplazamientos y la operación termina. Al mismo tiempo, se genera una petición de interrupción, si ésta está habilitada ($IER_2 = 1$).
- ❑ La bandera de estado, y la petición de interrupción se inhabilitan cuando el μP comienza otra operación de desplazamiento. También pueden ser puestos a cero, escribiendo directamente en el IFR.
- ❑ En el modo 4 (100) la información es desplazada hacia fuera usando el byte de menor peso de T2 como reloj.

b4 b3 b2	DIRECCIÓN	CLK	Nº de desplazamientos
000	DISABLED	-	0
001	IN	T2	8 ($IFR_2 \leftarrow 1$)
010	IN	CLK	8 ($IFR_2 \leftarrow 1$)
011	IN	CB1	8 ($IFR_2 \leftarrow 1$)
100	OUT	T2	CONTINUO
101	OUT	T2	8 ($IFR_2 \leftarrow 1$)
110	OUT	CLK	8 ($IFR_2 \leftarrow 1$)
111	OUT	CB1	8 ($IFR_2 \leftarrow 1$)

En este modo, una vez comenzado el desplazamiento, la operación continua de forma rotativa, es decir, el BIT desplazado fuera a través del LSB del SR, es cargado de forma automática en el MSB. El patrón de 8 bits se transmite de forma repetida a través de CB2.

ACR

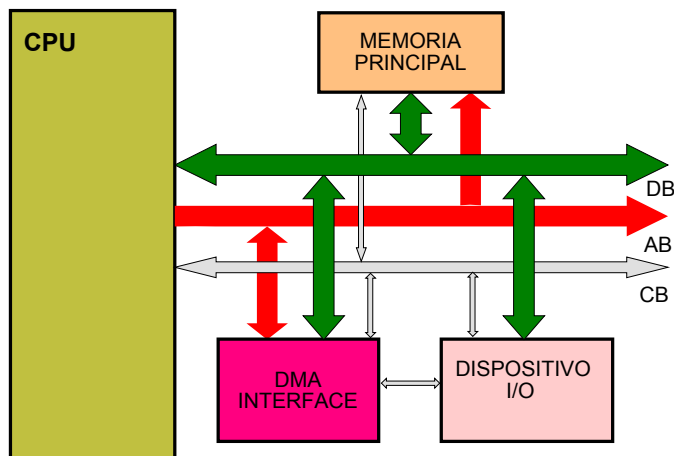
El proceso anterior continua hasta que el μP cambia el modo en ACR.

CONSIDERACIONES HARDWARE.

- ❑ Los modos handshake utilizados para controlar las transferencias son prácticamente idénticos a los de la PIA.
- ❑ El puerto B es capaz de entregar una corriente superior a la del puerto A, por lo que cuando se tengan que utilizar los dos puertos, uno como entrada y otro como salida, se tomará el puerto A como entrada y el puerto B como salida. Además, existen también algunas diferencias funcionales entre estos dos puertos que hacen que uno sea mas adecuado que el otro para una aplicación concreta.
- ❑ Se debe tener precaución a la hora de conectar cargas a los puertos. Si a un puerto se le somete a una carga excesiva porque la impedancia efectiva a él conectada es muy pequeña, con relación a la impedancia interna del puerto (Z_i/Z_o), puede darse el caso de que la transmisión por parte de un puerto de un valor lógico 1, sea visto como un cero en el circuito a él conectado.

10.8-DMA. (DIRECT MEMORY ACCESS)

En los casos anteriores de transmisión serie y paralelo, cualquier operación de entrada o salida, requería la intervención directa del μ procesador. En una operación de entrada por ejemplo, el μ P lee la palabra del buffer en el interfaz de entrada, y la deposita en un registro o posición de memoria. Cada una de las palabras debe ser tratada de esta manera, y bien sea mediante chequeo del registro de estado de la interfaz de entrada, o bien mediante interrupciones provocadas por este, estas operaciones de transferencia de datos requieren numerosos accesos a través de los buses del sistema.



- Un método que acelera el proceso de transferencia de información en ambos sentidos, entre la memoria principal del sistema y dispositivos I/O es el llamado Acceso Directo a Memoria, y es ventajoso siempre que se vayan a transferir bloques de información. Aplicando este método, la intervención del μ P se reduce a la tarea de iniciar el proceso. Una vez iniciado éste, la transferencia de los bloques de información, se realiza entre el dispositivo externo y posiciones consecutivas de memoria, pero

sin que cada palabra tenga que pasar por el microprocesador. De hecho, el microprocesador debe ser aislado para que no interfiera durante la operación.

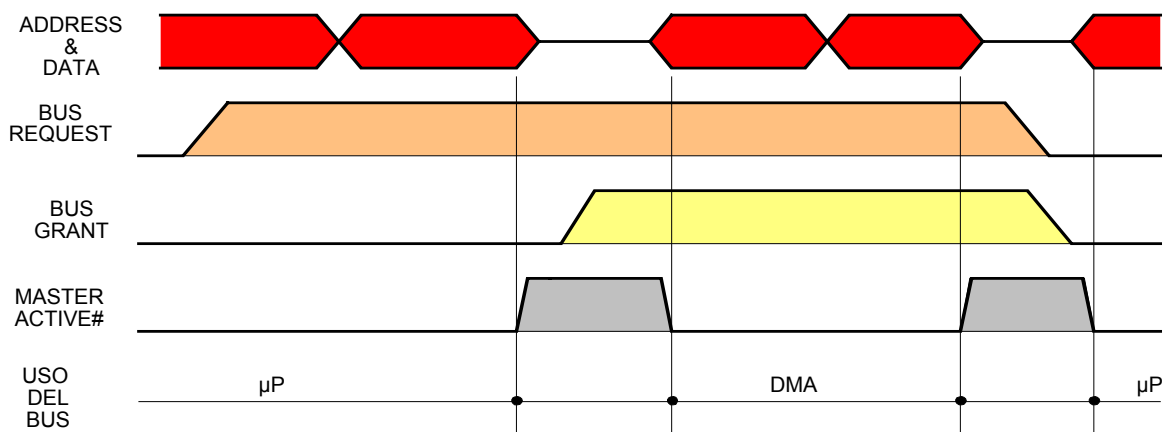
- El inicio de una tarea de DMA, requiere especificar los siguientes parámetros al dispositivo encargado de realizar la función, el cual recibe el nombre de *Direct Memory Access Controller* (DMAC):
 - Dirección inicial de la información en la memoria principal y cantidad de información a transferir. En otros casos es la dirección inicial y final del bloque de memoria sobre el cual tiene que operar el DMAC.
 - Sentido de transferencia de los datos; desde el dispositivo a la memoria o viceversa.
 - En ciertos casos, localización de los datos en el dispositivo externo.
- El μ P deposita estos datos en registros de la interfaz de DMA (DMAC), previstos para este fin, y ordena el comienzo de la transferencia.
- Durante la transferencia, la CPU puede esperar a que la operación concluya o realizar otras tareas. En este caso, hay que tener en cuenta que toda tarea que lleve consigo una transferencia de datos que implique a la memoria principal, al μ P o al DMAC, tiene que llevarse a cabo sobre un camino común a todos ellos, que es el BUS.
- Para que la interfaz DMA, transfiera datos desde o hacia la memoria principal, primero tiene que hacerse con el control del bus (*BUS MASTER*). Sólomente el *bus master* puede emplazar una dirección en el AB y manejar las líneas de control para indicar una operación de lectura o escritura.
- La ventaja principal del uso del DMA está en la rapidez con que realiza este tipo de tareas comparado con el tiempo que se requiere cuando es la propia CPU la que mueve cada dato.

- Generalmente, una transferencia DMA necesita numerosos ciclos de bus para completarse, y en cada uno de estos ciclos es transferida una palabra. La manera y secuencia de utilización de esos ciclos de bus que se necesitan para realizar la transferencia de información, depende de la velocidad de los dispositivos envueltos en el proceso. Si la velocidad del dispositivo de I/O no es muy elevada se pueden aprovechar ciclos de bus para que el μP busque (*FETCH*) y ejecute algunas instrucciones.

PROCESO GENÉRICO DE UNA TRANSFERENCIA DEL BUS

- La transferencia del dominio del bus del μP al DMAC, tiene lugar intercambiando señales de control entre los dos dispositivos. Existen generalmente dos líneas para este propósito, estas son *Bus Request* (BR) y *Bus Grant* (BG).

El proceso de transferencia puede ser como sigue:



El DMAC activa la línea *Bus Request* para solicitar el gobierno del bus. En respuesta, el μP detiene la ejecución de instrucciones y posteriormente se desconecta del bus, poniendo todos sus drivers de salida en estado de alta impedancia. Después, activa la línea *Bus Grant* para indicar al DMAC las acciones tomadas, es decir, que el ya está aislado del bus.

- Cuando el DMAC no necesita utilizar más el bus, se lo comunica al μP desactivando la línea BR. El μP entonces, vuelve a hacerse con el control del bus.

Una línea *Master Active#* en el ejemplo, gobernada por el Bus Master, indica a las interfaces de los dispositivos que ignoren las líneas de dirección mientras esta línea no esté activa. En estos periodos, las líneas de dirección tienen durante un cierto tiempo valores impredecibles, con lo que podrían dar lugar a la selección errónea de otros dispositivos.

- Una vez que el DMAC se ha hecho con el control del bus, puede transferir palabra por palabra o transferir bloques de palabras a la vez. Existen diferentes alternativas para optimizar la velocidad en la ejecución de tareas cuando el bus debe ser compartido:

a.- Una alternativa, es hacer que el microprocesador se detenga (entre en un estado HALT) durante el tiempo en el cual el DMAC es el bus master. En este modo, se aprovecha al máximo la capacidad de transferencia de información del DMA.

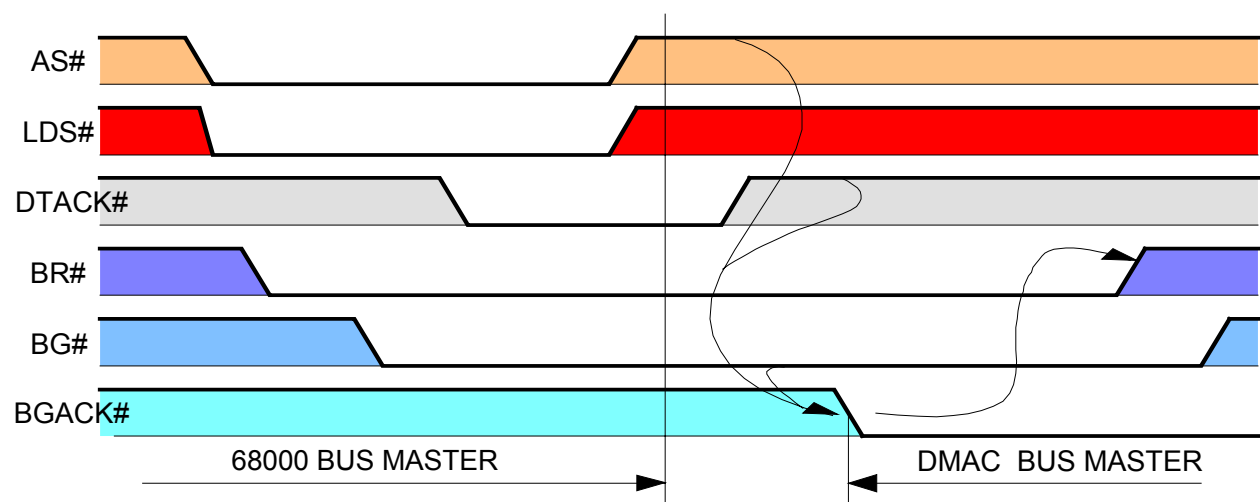
b.- Cuando el DMAC transfiere una palabra, y después devuelve el dominio del bus al μP se dice que trabaja en modo de robo de ciclo (*Cycle-Stealing Mode*). En este caso existe un mal uso del tiempo debido a los periodos en los cuales las líneas del bus son inestables

(línea de *Master Active#* inactiva, en el ejemplo dado anteriormente) y por lo tanto no son utilizadas por ningún elemento. Una posible utilización de este modo podría ser en la siguiente situación:

Supongamos un driver de disco que puede transferir 100 KWords/s, conectado a un μP que puede transferir 2MWords/s. En este caso, la interfaz DMA necesita transferir una palabra cada 20 ciclos bus. Si suponemos que para la transferencia del dominio del bus de un dispositivo al otro se necesitan 2 ciclos bus y un ciclo más para la transferencia de la palabra, esto nos deja 17 ciclos sobre 20 para que el μP se dedique a otras tareas.

c.- Otra alternativa es la de compartir el bus. El DMAC se va a hacer con el control del bus en aquellos periodos en los cuales el microprocesador no hace uso de ellos por estar realizando un ciclo de trabajo interno.

TRANSFERENCIA DEL BUS EN EL μP 68000.



- En el 68000 existen tres terminales de señal destinados a la transferencia del control del bus. Estos terminales son: Bus Request# (BR#), Bus Grant# (BG#) y Bus Grant Acknowledge# (BGACK#). En la figura, un DMAC demanda el control del bus activando la línea BR# del μP . Esta demanda puede venir en cualquier momento. En la figura, coincide con proceso de una transferencia de información (AS# activada).

El μP en respuesta, activa la línea de BG#, informando al DMAC que cuando termine el ciclo que está en curso, puede hacerse con el bus. El DMAC, se asegura que los dispositivos han terminado el ciclo que estaba en curso, testando las líneas AS# y DTACK#. Cuando observa que las dos líneas han pasado a un estado inactivo, activa la línea BGACK# y desactiva la línea de petición del bus (BR#). A partir de este momento, el DMAC es el dueño del bus, indicándoselo al μP mediante la línea BGACK# en estado activo. Cuando no necesite ser el bus master, desactiva esta última línea.

Hay que señalar que en estas transferencias, la señal AS# realiza la misma función que la línea que en el ejemplo genérico de transferencia, se denominó Master active#.

REGISTROS TÍPICOS EN UN DMAC.

En la figura se muestran cuatro registros típicos de un DMAC. La forma en la que el μP accede a estos registros es la misma que la usada para acceder a los registros de cualquier dispositivo I/O.

MEMORY ADDRESS REG.

Guarda la dirección del comienzo del bloque de memoria hacia donde, o desde donde los datos, van a ser transferidos.

BYTE COUNT REG.

Guarda el número de palabras (Bytes) que deben ser transferidas.

DMA CONTROL REG.

Este registro permite al programador definir ciertos aspectos de la transferencia de información DMA.

b0 (R/W#): Indica el tipo de transferencia que se va a realizar. La transferencia de datos desde la memoria al dispositivo I/O, se considera una operación de lectura (b0= 1). La transferencia desde el dispositivo a la memoria es una operación de escritura (b0= 0).

b1-b2 (MODE): Mediante estos bits se va a definir la forma en la cual la transferencia va a tener lugar, dependiendo de las características del μP . Puede establecerse que la transferencia sea en modo de *robo de ciclo*, en modo de parada del micro, etc. También puede definirse mediante este campo, la manera en la cual se va a realizar el proceso de bus master.

b3 (UP/DOWN#): Define la secuencia de acceso a las posiciones de memoria implicadas en la transferencia. Dependiendo del estado de este BIT, la memoria será accedida de una u otra forma a partir de una dirección de comienzo determinada.

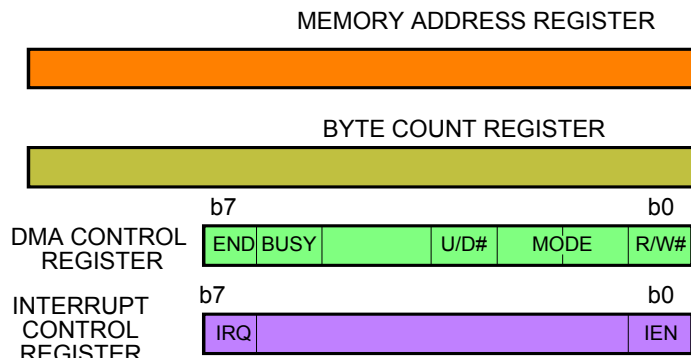
b6 (BUSY): Se pone a 0 cuando el registro contador de bytes llega a 0, es decir, cuando se ha procesado toda la información que se determinó mediante el registro contador de bytes. Hay que tener en cuenta que el control de los buses puede pasar varias veces del DMAC al μP y viceversa en una operación de DMA. Cuando el μP es el *bus master*, puede consultar en cualquier momento si el DMA está disponible para iniciar otra transferencia.

b7 (END): Cuando se pone a 1 indica que la anterior petición de tarea para el DMA ha terminado.

INTERRUPT CONTROL REG.

b0 (IEN- *INTERRUPT ENABLE BIT*): Puesto a 1 permite que la interfaz genere una petición de interrupción después de completar una tarea DMA (END =1).

b7 (IRQ): Se mantiene en estado 1 mientras la salida de la petición de interrupción es activa. Esto facilita el proceso de polling.



Dependiendo de los dispositivos a los que va destinado, el número de registros puede ser superior. Por ejemplo, un driver de disco necesita que se le especifique la posición del disco donde están los datos, posicionamiento de la cabeza de R/W, etc. En todo caso, cualquier DMAC posee más registros que los vistos, que le hacen flexible para un número relativamente elevado de diferentes aplicaciones.

- Los DMAC poseen varios canales para poder trabajar con diferentes dispositivos de I/O. Cada uno de los canales, tiene su *BYTE COUNT REGISTER*, *MEMORY ADDRESS REGISTER*... etc., pero la mayor parte del control es común para los diferentes canales.

En estos casos, para solucionar conflictos entre peticiones simultaneas en varios canales, existen generalmente dos métodos:

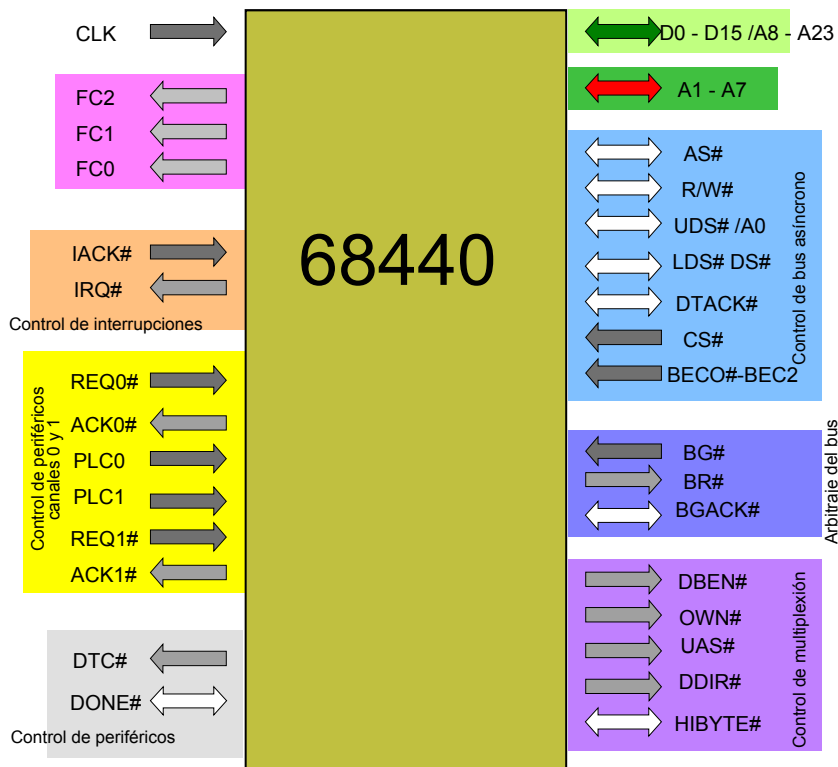
PRIORIDAD FIJA: En este caso, el nivel de prioridad viene determinado por hardware en el DMAC. Este esquema es válido cuando los dispositivos conectados a los canales, tienen diferencias muy grandes en cuanto a demandas. Por ejemplo una petición para una transferencia DMA realizada por un dispositivo de alta velocidad –HDD- se le debe dar prioridad alta pues puede perder datos si no se atiende rápido.

Este método no es eficaz cuando los dispositivos tienen más o menos los mismos requerimientos de servicio.

PRIORIDAD EN ANILLO: Con este método, inicialmente un dispositivo tiene la máxima prioridad y otro la mínima, y entre estos dos están otros también con una prioridad definida. A partir de aquí, aquel que sea atendido pasa a ocupar el último lugar en la escala de prioridades; El que ocupaba el segundo lugar, pasa a ocupar el primero en prioridades. Cualquier canal que recibe servicio pasa a ser el último en la lista.

MC 68440, DDMA (DUAL-CHANEL DIRECT MEMORY ACCESS CONTROLLER)

MODOS DE OPERACIÓN:



DISTRIBUCIÓN FUNCIONAL

- IDLE.
Estado inactivo después de un RESET, o finalización de las transferencias.

- MPU.
En este estado el DDMA, es controlado por el μP para escribir o leer sus registros. El DDMA se selecciona activando CS# por la lógica de selección del sistema

- DMA.
Estado en el cual, el DDMA es el bus master.

TERMINALES:

PIN	NOMBRE	I	O	COMENTARIOS
D0-D15/ A8-A23	BUS DE DATOS/ 2 BYTES AB			LOS DATOS Y LOS 2 BYTES DEL BUS DE DIRECCIONES SE MULTIPLEXAN.
A1-A7	BYTE BAJO DEL AB			JUNTO CON UDS# /A0, BYTE BAJO DEL AB
FC0-FC2	CÓDIGOS DE FUNCIÓN.			
CS#	CHIP SELECT			CHIP SELECT EN MODO MPU
AS#	ADDRESS STROBE	IDLE, MPU	DMA	COMO I, INDICA CUANDO EL DDMA PUEDE CONTROLAR LOS BUSES
R/W#	READ/WRITE	MPU	DMA	
UDS#/A0	UPPER DATA STROBE			BUS DE 16 BITS/BUS DE 8 BITS
LDS#/DS#	LOWER DS/ DATA STROBE			BUS DE 16 BITS/BUS DE 8 BITS
DTACK#	DATA TRANSFER ACK.			
BEC0#- BEC2#	BUS EXCEPTION CONTROL			ENTRADA CODIFICADA DE TERMINACIÓN ANORMAL DEL CICLO EN CURSO
BR#	BUS REQUEST			PETICIÓN DE CONTROL DEL BUS
BG#	BUS GRANT			RECONOCIMIENTO DE BR# Y CESIÓN DEL BUS
BGACK#	BG ACKNOWLEDGE			RECONOCIMIENTO DE CONTROL DEL BUS
OWN#	OWN			ACTIVA CUANDO EL DDMA CONTROLA EL BUS
UAS#	UPPER ADDRESS STROBE			D0-D15/A8-A23 CONTIENEN DIRECCIÓN
DBEN#	DATA BUFFER ENABLE			D0-D15/A8-A23 CONTIENEN DATO
DDIR#	DATA DIRECTION			CONTROL DE DIRECCIÓN DE LOS DATOS
HIBYTE#	HIGH BYTE			CONVERSIÓN DE DATOS: 8/16; 16/8
IRQ#	INTERRUPT REQUEST			
IACK#	INT ACKNOWLEDGE			
REQn#	REQUEST			SOLICITUD DE TRANSFERENCIA CANAL n
ACKn#	ACKNOWLEDGE			CONFIRMACIÓN DE SOLICITUD RECIBIDA
PCLn#	PERIPHERAL CONTL LINE			FUNCIÓN PROGRAMABLE
DTC#	DATA TRANSFER COMPLET			TERMINACIÓN NORMAL DE TRANSFERENCIA
DONE#	DONE			EL DATO QUE SE TRANSFIERE ES EL ÚLTIMO DEL BLOQUE
CLK	CLOCK			

REGISTROS.

- ❑ El DDMA contiene 35 registros accesibles para el programador. Uno de los registros es de control general (*GCR*), y los otros 34 están divididos en dos grupos de 17; cada uno de estos grupos está asociado a un canal del chip.
- ❑ Los registros son direccionados mediante las líneas A1-A7. Estas líneas, junto con las A8-A23 y UDS#/A0, forman parte del AB del DDMA que sirve para direccionar la memoria o un periférico en las transferencias.
- ❑ Generalmente, el AB no va a ser utilizado para el direccionamiento del periférico. La forma típica de direccionar este, es mediante las líneas de control de canal, y control de periféricos mientras que el AB direcciona la memoria. Este tipo de configuración se llama *direccionamiento implícito del periférico*. (Otro tipo de configuración es la llamada por *direccionamiento explícito del periférico*, en la cual, el AB direcciona de forma alternada el periférico y la memoria).

GCR	GENERAL CONTROL REGISTER
BFCR	BASE FUNCTION CODE REGISTER
DFCR	DEVICE FUNCTION CODE REGISTER
CPR	CHANEL PRIORITY REGISTER
MFCR	MEMORY FUNCTION CODE REGISTER
EIVR	ERROR INTERRUPT VECTOR REGISTER
NIVR	NORMAL INTERRUPT VECTOR REGISTER
BAR	BASE ADDRESS REGISTER
BTCR	BASE TRANSFER COUNT REGISTER
DAR	DEVICE ADDRESS REGISTER
MAR	MEMORY ADDRESS REGISTER
MTCR	MEMORY TRANSFER COUNT REGISTER
SCR	SEQUENCE CONTROL REGISTER
CCR	CHANNEL CONTROL REGISTER
DCR	DEVICE CONTROL REGISTER
OCR	OPERATION CONTROL REGISTER
CSR	CHANNEL STATUS REGISTER
CER	CHANNEL ERROR REGISTER

\$FE		GCR
\$3E		
\$3C		
\$3A		
\$38		BFCR
\$36		
\$34		
\$32		
\$30		DFCR
\$2E		
\$2C		CPR
\$2A		
\$28		MFCR
\$26		EIVR
\$24		NIVR
\$22		
\$20		
\$1E		
\$1C	BAR	
\$1A	BTCR	
\$18		
\$16		
\$14	DAR	
\$12		
\$10		
\$0E		
\$0C	MAR	
\$0A	MTCR	
\$08		
\$06	SCR	CCR
\$04	DCR	OCR
\$02		
\$00	CSR	CER

MODOS DE TRANSFERENCIA.

RÁFAGA (VELOCIDAD MÁXIMA): El DDMA mantiene el control del bus, durante todo el proceso de transferencia de un bloque completo.

RÁFAGA (VELOCIDAD LIMITADA): Existe un reparto de tiempo entre el μP y el DDMA establecido previamente. Cuando los ciclos asignados a la operación DMA terminan, se devuelve el control del bus al μP aunque no se haya terminado la transferencia del bloque. El DDMA vuelve a tomar el control cuando al micro se le terminen los ciclos a él asignados. Este proceso dura mientras el bloque no haya sido transferido por completo.

RÁFAGA (SOLICITUD EXTERNA): El periférico mediante la señal REQ# activada, indica al DDMA que la operación de transferencia aún no ha terminado y que por lo tanto debe seguir siendo el bus master.

INDIVIDUAL: El periférico genera un impulso cada vez que necesite transferir un dato.

CONTINUO: Mediante este modo se pueden realizar transferencias de bloque no consecutivos.

RECARGA: Si durante la transferencia, el periférico detecta un error en los datos, activa la línea PCL con lo cual el DDMA reinicia la transferencia del bloque donde se ha detectado el error.



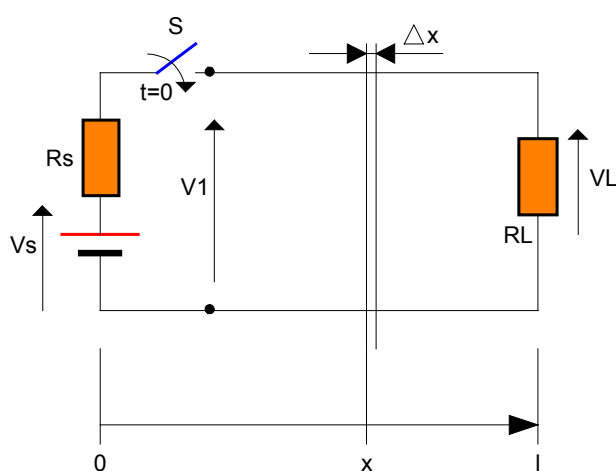
Líneas de transmisión

11

11.-LÍNEAS DE TRANSMISIÓN.

Una línea de transmisión se puede considerar como un camino metálico que sirve para transferir energía eléctrica de un punto a otro. La transferencia de energía se puede realizar mediante la propagación eléctrica de DC o de altas frecuencias. El comportamiento de una línea de transmisión, es fácilmente predecible cuando la transferencia de energía por ella, se realiza a baja frecuencia. Cuando la transferencia es a alta frecuencia, su comportamiento es bastante más complejo y peculiar. Como regla general, la teoría de líneas de transmisión se aplica cuando el tiempo de subida de la señal es más pequeño que el doble del tiempo de propagación de esa señal por la línea.

- Para entender los fenómenos en una línea de transmisión, vamos a examinar el flujo de corriente y voltaje en un circuito como el de la figura, cuando se cierra el interruptor, es decir, cuando se aplica al circuito un escalón de voltaje.

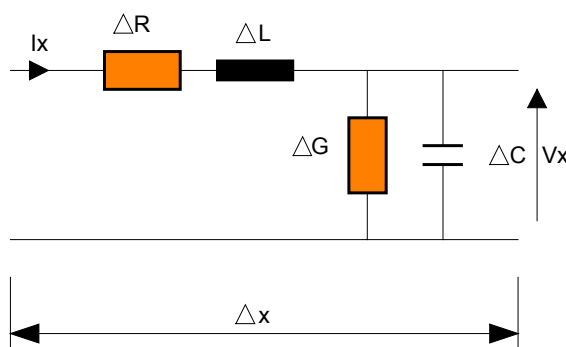


Cuando cerramos el interruptor s en $t=0$, los efectos de esta acción no ocurren de forma instantánea en toda la línea. La causa, que en este caso es el cierre del interruptor, va a provocar un efecto que se va a propagar desde la fuente hasta la carga a través de los hilos conductores, con una velocidad finita.

Las líneas de transmisión que unen la fuente con la carga, pueden ser vistas como un número infinito de secciones infinitesimales representadas por Δx . Las dos líneas que forman cada sección, pueden ser representadas por un circuito equivalente que integre las respuestas de la línea a

excitaciones eléctricas, en cada una de las secciones.

- En este circuito equivalente se representan los efectos capacitivos (ΔC) e inductivos (ΔL) de la línea, así como el efecto de la resistencia de los conductores (ΔR), y efecto de pérdidas asociadas con el medio dieléctrico (ΔG). En el mismo tenemos:



ΔR . Resistencia por unidad de longitud. [Ω/m].

ΔL . Inductancia por unidad de longitud. [H/m].

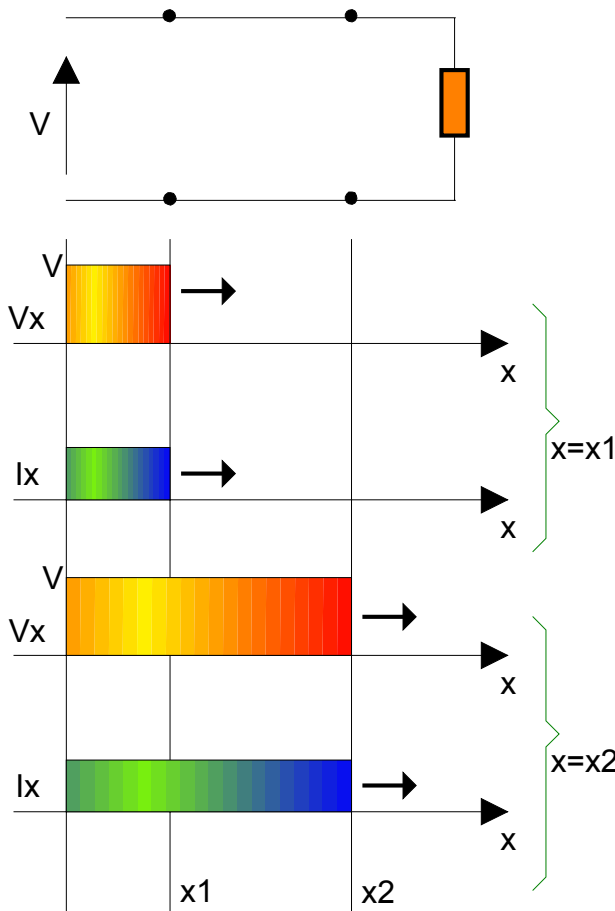
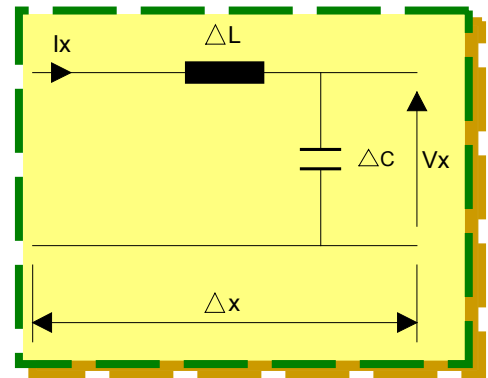
ΔG . Conductancia por unidad de longitud. [Si/m].

ΔC . Capacidad por unidad de longitud. [F/m].

CARACTERÍSTICAS DEL RG-58 (COAXIAL).

$\Delta R = 36 \Omega/km$.
 $\Delta L = 2532 nH/m$
 $\Delta C = 101 pF/m$
 $\Delta G \approx 0$

- El circuito puede reducirse a un modelo más simplificado, en el cual, se ignoran los efectos de la resistencia de los hilos y el efecto de pérdidas asociadas con el dieléctrico.
- Cuando cerramos **s**, se aplica a la línea un escalón de voltaje de magnitud V_1 . Como resultado, una corriente comienza a circular por la primera sección infinitesimal de la línea. A medida que la corriente va cargando la capacidad de esa primera sección, el incremento de la tensión en el condensador hace que la corriente circule hacia la siguiente sección para cargar la capacidad de ésta y así sucesivamente hasta recorrer las infinitas secciones.



Este efecto se puede ver como dos olas; una “ola de voltaje” y otra “ola de corriente” asociada, que se propagan por la línea.

- La velocidad de propagación de esta ola viene dada por la expresión

$$S = (LC)^{-1/2}$$

donde, L y C representan la inductancia y capacidad de la línea, por unidad de longitud.

- Cuando el frente de esta ola, se mueve una distancia dx , la capacidad adicional que se carga hasta la tensión V_1 es

$$C \, dx.$$

la carga necesaria es

$$dQ = V_1 C \, dx.$$

la corriente asociada será:

$$\begin{aligned} I_1 &= dQ/dt = V_1 C \, dx/dt = V_1 C S = V_1 C (LC)^{-1/2} \\ &= V_1 C^{(1-1/2)} L^{-1/2} = V_1 C^{1/2} L^{-1/2} = V_1 (C/L)^{1/2}. \\ I_1 &= V_1 (C/L)^{1/2}. \end{aligned}$$

La relación entre V_1 e I_1 en cada sección viene dada por

$$V_1 / I_1 = (L/C)^{1/2}$$

Esta relación tiene dimensión resistiva. La línea de transmisión aparece para la onda de voltaje y corriente como una carga resistiva de valor R_0 .

- Este valor $R_o = (L/C)^{1/2}$ es conocido como **IMPEDANCIA CARACTERÍSTICA DE LA LÍNEA.**

Si el análisis se realiza para el circuito sin simplificar, el valor de la impedancia característica de la línea, está definida por la fórmula

$$R_o = [(R + j\omega L)/(G + j\omega C)]^{1/2}.$$

Si la línea se construye para que $R \ll \omega L$ y $G \ll \omega C$ la impedancia característica de la línea para frecuencias altas, es constante e independiente de la frecuencia y longitud de la línea: $R_o \approx (L/C)^{1/2}$

Observar que para frecuencias muy bajas, la expresión de la impedancia característica queda:

$$R_o \approx [R/G]^{1/2}$$

- Las características de una línea de transmisión vienen determinadas por la R de los cables, L y C. Estos dos últimos valores, vienen determinados por las características físicas de la línea de transmisión (forma, tamaño, espaciado), y por la constante dieléctrica, κ del medio aislante.

Para dos hilos paralelos, y teniendo como medio dieléctrico entre ellos el aire, una reducción de distancia entre los hilos, acarrea un aumento de C y una disminución de L y por lo tanto una disminución de R_o .

Si se introduce un dieléctrico entre los hilos, aumenta C y se mantiene constante L, por lo tanto también disminuye R_o . (un dieléctrico reduce la velocidad de propagación $S = (LC)^{-1/2}$)

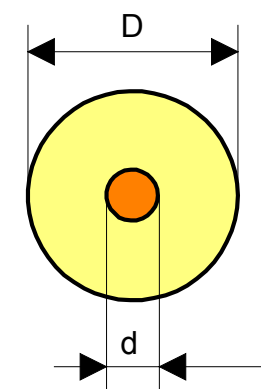
La impedancia de la línea para el caso de dos hilos paralelos, con dieléctrico el aire viene dado por

$$R_o \approx 276 \log (4h/d)$$

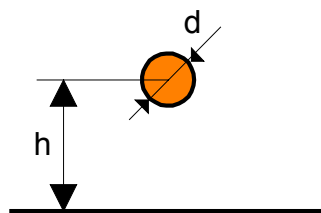
Y para el cable coaxial

$$R_o = 138 / (\kappa)^{1/2} \log (D/d)$$

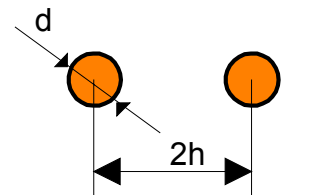
κ = cte del dieléctrico.



CABLE COAXIAL



HILO SOBRE TIERRA



HILOS PARALELOS

La capacidad por metro para las líneas de transmisión de la figura es:

CABLE COAXIAL. $C \approx 2\pi\kappa / \ln (D/d)$

HILO SOBRE TIERRA. $C \approx 2\pi\kappa / \ln (4h/d)$

HILOS PARALELOS. $C \approx \pi\kappa / \ln (4h/d)$

constantes dieléctricas	(κ)		(κ)
Vacío	1	baquelita	4,9
Aire seco	1,00059	mica	5,0
Teflón	2,1	caucho de neopreno	6,7
Papel parafinado	2,5	poliestireno	2,56

11.1-CARACTERÍSTICAS.

VELOCIDAD DE PROPAGACIÓN.

- Las ondas electromagnéticas viajan a la velocidad de la luz al propagarse por el vacío, y un poco más lentas al propagarse por el aire. En las líneas de transmisión donde el material metálico utilizado generalmente es el cobre, y los dieléctricos son muy diversos dependiendo del tipo de línea, la velocidad de la onda electromagnética es mucho más baja.
- El valor de la velocidad de propagación viene determinada principalmente por la constante dieléctrica del medio aislante que separa los conductores. En la práctica, si el espacio alrededor de los conductores está totalmente ocupado con un aislante que tenga características uniformes, la velocidad de propagación es independiente de la geometría del conductor y vale

$$S = c/(\kappa)^{1/2}$$

siendo c la velocidad de la luz en el espacio libre (29,9793cm/ns)

También se utiliza la fórmula:

$$S = (LC)^{-1/2}$$

donde L y C representan la inductancia y capacidad de la línea, por unidad de longitud.

- Podemos poner la longitud de onda en función de las anteriores expresiones. Tenemos que para una onda electromagnética en el vacío la longitud de onda viene dada por $\lambda = c/f$. Para otro medio vendrá dada por

$$\lambda = S / f = c/[(\kappa)^{1/2} f]$$

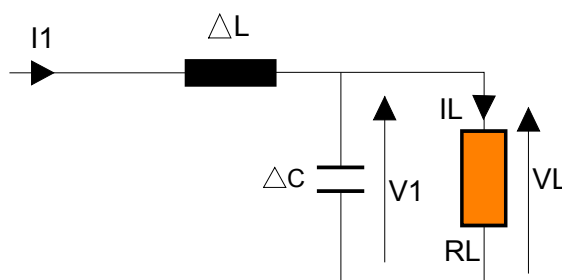
A frecuencias bajas, o lo que es lo mismo, longitudes de onda grandes, el voltaje a lo largo de toda la línea permanece constante. Si la frecuencia es muy alta (longitud de onda pequeña), en la línea pueden estar presentes al mismo tiempo varias longitudes de onda con lo que el voltaje a lo largo de la línea puede variar considerablemente. Para tener percepción de este fenómeno, muchas veces la longitud de la línea se da en longitudes de onda. Por ejemplo, una línea que mida 10 m, y trabaje a una frecuencia tal que la longitud de onda sea de 5 cm, se puede decir que es una línea de 200 longitudes de onda.

REFLEXIONES.

Consideremos que sucede cuando un frente de onda alcanza el final de la línea.

- Cuando la capacidad de la última sección de la línea se carga, la corriente comienza a fluir hacia la carga RL .

Si **$RL = Ro$** :



Toda la corriente I_1 , finalmente fluirá a través de R_L , creando un voltaje en la carga $V_L = V_1$, voltaje al cual la capacidad ha sido cargada.

En este caso no suceden mas cambios en la corriente o en la tensión. La carga absorbe toda la potencia incidente (línea acoplada). El sistema ha alcanzado el estado estable.

$$V_L = V_1 = V_s[R_L/(R_s+R_L)].$$

Siendo R_s la resistencia de salida de la fuente.

Si **$R_L \neq R_o$** ,

el estado estable no se alcanza en el momento en que la onda alcanza la carga, porque la relación entre V e I asociada con la onda incidente no puede mantenerse en la carga.

$$V_1/I_1 = R_o \neq R_L$$

El escalón de voltaje en la carga R_L , es decir, V_L , será diferente de V_1 . Esta diferencia $V_L - V_1$ es un escalón que debe propagarse ahora en sentido de la fuente para cargar la capacidad de la línea con el nuevo voltaje, V_L .

Una parte de la corriente I_1 , la I_L , fluye a través de la carga R_L originando $V_L = I_L \cdot R_L$. Otra parte de la corriente I_1 , la I_2 , será la corriente asociada a la onda que vuelve a la fuente.

$$I_1 = I_L + I_2; \quad I_2 = I_1 - I_L.$$

- Los valores de V e I de esta onda que vuelve, tienen que guardar la relación impuesta por la impedancia de la línea de transmisión, es decir,

$$(V_L - V_1)/(I_1 - I_L) = R_o$$

$$\text{como } I_1 = V_1/R_o, \quad I_L = V_L/R_L \text{ resulta } V_L = 2R_L \cdot V_1/(R_L + R_o)$$

el escalón de la onda reflejada valdrá

$$V_2 = V_L - V_1 = [(R_L - R_o)/(R_L + R_o)]V_1 = \rho_L V_1;$$

$$\mathbf{V_2 = \rho_L V_1}$$

donde $\rho_L = (R_L - R_o)/(R_L + R_o)$ se llama *coeficiente de reflexión*.

- Cuando la onda reflejada alcanza la fuente, puede ser reflejada de nuevo. En este caso el coeficiente de reflexión será

$$\rho_s = (R_s - R_o)/(R_s + R_o).$$

MARGEN DE ρ :

$$\rho_L = (R_L - R_o)/(R_L + R_o). \quad \left\{ \begin{array}{ll} R_L = 0 \text{ (carga en cortocircuito)} & \rho_L = -1; \mathbf{V_2 = -V_1} \\ & \text{La suma de } V_2 \text{ y } V_1 \text{ es mínima} \\ R_L = R_o & \rho_L = \mathbf{0}; \mathbf{V_2 = 0} \\ R_L = \infty \text{ (carga en circuito abierto)} & \rho_L = \mathbf{1}; \mathbf{V_2 = V_1} \\ & \text{La suma de } V_2 \text{ y } V_1 \text{ es máxima} \end{array} \right.$$

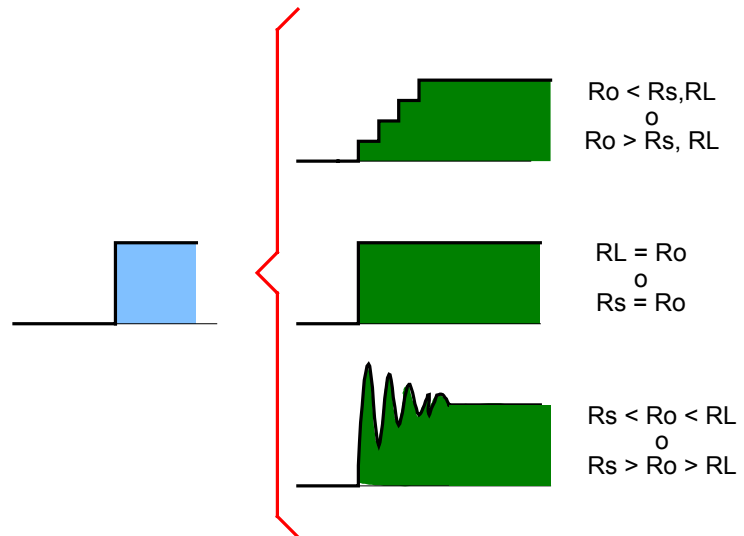
- Como el coeficiente de reflexión en ambos extremos es menor que 1 el movimiento de idas y vueltas de la onda se extinguirá y el voltaje en cualquier punto de la línea alcanzará el estado estable.

$$V = [R_L / (R_s + R_L)] V_s \quad (\text{en el estado estable.})$$

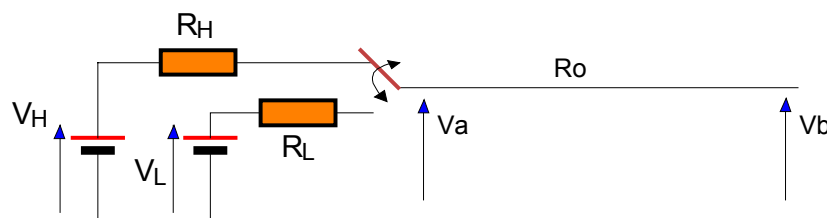
El valor de la tensión V_1 , en el momento del cierre del interruptor viene dado por

$$V_1 = [R_o / (R_s + R_o)] V_s$$

- Para una máxima transferencia de potencia desde la fuente a la carga (es decir, sin energía reflejada), una línea de transmisión debe terminarse con una carga puramente resistiva de valor igual a la impedancia característica de esa línea. En la figura se representan las diferentes tensiones medidas en la carga como respuesta a una entrada escalón, para diferentes valores de las resistencias de la carga y de la fuente. En la parte superior, la tensión final en la carga se logra de forma escalonada. En la figura del centro, impedancias adaptadas, el estado estable de la señal se logra sin oscilaciones ni escalones (para $R_s = R_o$ el estado estable se alcanza después de una reflexión), y en la parte inferior, se observa que el estado estable se logra después de una oscilación. En ninguno de los tres casos, el estado estable se logra de forma instantánea, si no que ha tenido que transcurrir un tiempo hasta que la onda alcanza el final de la línea.

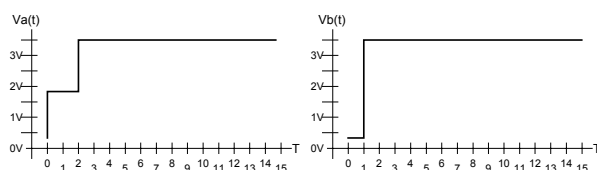


En las siguientes figuras se muestra un ejemplo de un dispositivo bipolar con sus impedancias típicas en estado alto y bajo, manejando una línea de 10cm e intervalo de propagación $T = 1,7\text{ns}$, abierta al final.

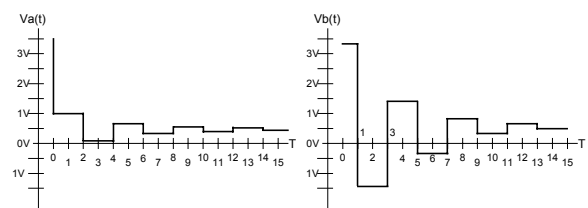


$V_H = 3,3\text{V}$, $V_L = 0,3\text{V}$, $R_H = 30\Omega$, $R_L = 5\Omega$, $R_o = 30\Omega$.

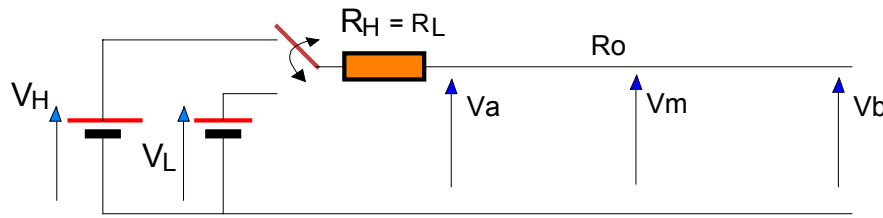
El conmutador simula el paso de estados H-L y L-H



paso de L-H

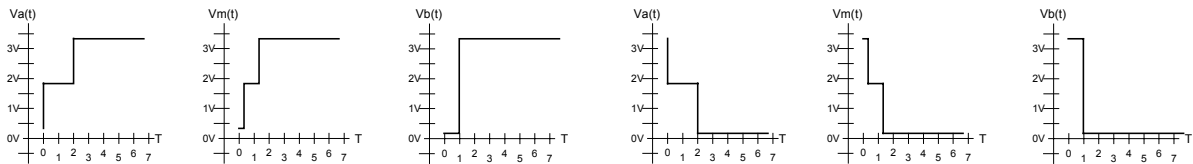


paso de H-L



En los casos siguientes, la resistencia es la misma para ambos estados.

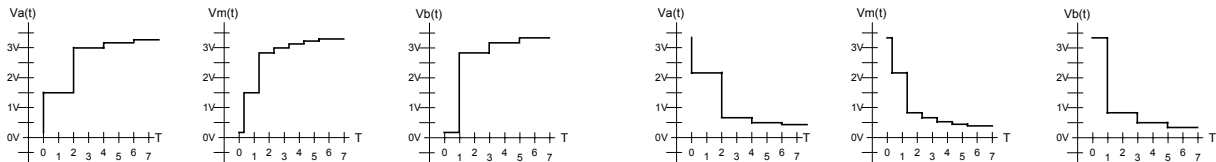
$V_H = 3,3V$, $V_L = 0,3V$, $R_H = R_L = 30\Omega$, $R_o = 30\Omega$.



L-H

H-L

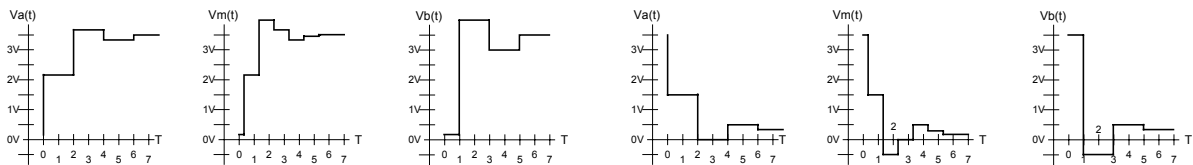
$V_H = 3,3V$, $V_L = 0,3V$, $R_H = R_L = 30\Omega$, $R_o = 20\Omega$.



L-H

H-L

$V_H = 3,3V$, $V_L = 0,3V$, $R_H = R_L = 30\Omega$, $R_o = 50\Omega$.



L-H

H-L

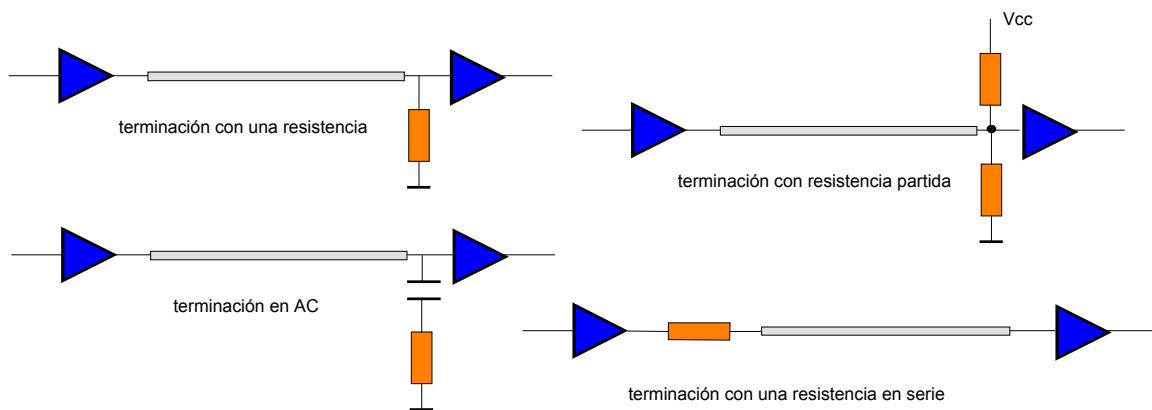
PÉRDIDAS EN UNA LÍNEA DE TRANSMISIÓN

- El hecho de que fluya una corriente por la línea de transmisión, lleva consigo un calentamiento de la misma y por lo tanto una pérdida de la potencia útil. Esta pérdida de potencia se cuantifica como $I^2 R$ y representa las *perdidas por calentamiento*. Para minimizar estas pérdidas, la línea de transmisión debe reducirse en su longitud o ampliar el diámetro de la línea, teniendo en cuenta en este último caso que la modificación del diámetro conlleva la variación de la impedancia característica y por lo tanto la corriente por la línea.

Efecto pelicular: Cuando fluye una corriente por un conductor circular aislado, el flujo magnético creado tiene forma de círculos concéntricos. La densidad de flujo creada, es mayor hacia el centro del conductor que hacia la superficie, esto hace que la inductancia cerca del centro del conductor sea mayor que cerca de la superficie y por lo tanto a frecuencias altas (RF) la corriente fluye en su mayor parte a lo largo de la superficie del conductor. Esto equivale a una reducción de la sección útil transversal del conductor y por lo tanto equivale a un aumento de la resistencia del mismo. Por encima de 100MHz, prácticamente toda la corriente circula por la superficie del conductor.

- ❑ Pérdidas por radiación. Si los conductores están separados una distancia que representa una fracción considerable en relación con una longitud de onda, los campos electrostáticos y electromagnéticos que rodean el conductor hacen que la línea actúe como antena y radie energía hacia conductores cercanos. La forma de evitar esta pérdida es blindando convenientemente la línea.
- ❑ Pérdidas por calentamiento del dieléctrico causado por la diferencia de potencial entre los conductores de la línea de transmisión.
- ❑ Pérdidas por acoplamientos o conexiones en la línea. Las conexiones mecánicas crean “discontinuidades”, que tienden a calentarse y a radiar energía.

TERMINACIONES TÍPICAS.



Terminación con una resistencia. El valor de la resistencia es el mismo que la resistencia típica de la línea. Tiene la desventaja de un incremento en la corriente (potencia).

Terminación con resistencia partida. La resistencia equivalente Thevenin de las dos resistencias debe ser igual a la resistencia característica de la línea. Al estar conectado a V_{cc} tiene a veces gran consumo de corriente.

Terminación en AC. El valor de la resistencia (R), es el mismo que el de la resistencia característica de la línea. El valor del condensador se suele tomar según la fórmula $RC > 4T$ donde T es el tiempo de propagación de la línea. Reduce el consumo de potencia al eliminar la componente DC, aunque no tanto en altas frecuencias.

Terminación con resistencia serie. Es la técnica más utilizada. Con esta R la impedancia de salida del driver de la línea se ve aumentada, con lo que el coeficiente de reflexión disminuye.

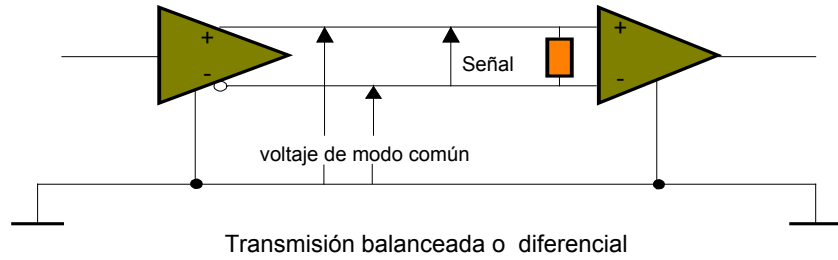
TRANSMISIÓN.

- ❑ Las formas de transmisión pueden clasificarse según la configuración de las líneas en *balanceadas* y *no balanceadas* o de *terminación simple*.

Transmisión balanceada o diferencial.

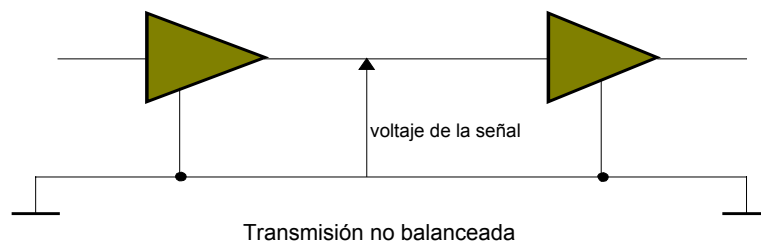
- ❑ En este tipo de transmisión, tenemos una señal que tiene una magnitud determinada con respecto a la referencia común o masa y otra señal que tiene una magnitud opuesta a la anterior, medida sobre la misma referencia, es decir, tenemos dos conductores con una señal en cada uno de ellos que viaja en sentido opuesto respecto a la masa. En este caso, la señal que se propaga desde la fuente al destino, se obtiene como la diferencia de tensiones entre los dos conductores.

- En este tipo de transmisión, el voltaje de ruido que se induce en los conductores, también llamado voltaje de modo común, al ser el mismo en ambos hilos y obtenerse la señal como diferencia entre las tensiones en los conductores, al realizar esa diferencia, se anula en la carga.



Transmisión no balanceada o de terminación simple.

- Un conductor se encuentra a potencial de masa, y el otro es el portador de la señal. El conductor que va a masa, puede ser también referencia para otros conductores que portan señal, pero cualquier señal se tomará desde su conductor y la referencia común de masa.



Las líneas de transmisión balanceadas, pueden conectarse a las líneas no balanceadas y viceversa por medio de un transformador especial llamado *balun* (balanced-unbalanced).

ANCHO DE BANDA.

El ancho de banda representa el rango de frecuencias en un canal de comunicación. Por ejemplo, las señales de la voz están comprendidas aproximadamente entre 300 y 3300 Hz., por lo tanto el ancho de banda de un canal de voz es aproximadamente de 3000 Hz. (3300 – 300)

VELOCIDAD MÁXIMA DE TRANSFERENCIA DE DATOS.

El teorema de Nyquist establece que la máxima velocidad de transferencia de datos en un canal libre de ruidos, con un ancho de banda H y un número de niveles para los datos V, es

$$2 H \log_2 V \text{ bps}$$

Entonces para un canal con un ancho de banda de 3KHz, con codificación digital binaria, la máxima velocidad de transferencia será de 6000 bps.

11.2-DATOS DE ALGUNAS LÍNEAS:

Conductor simple alejado de tierra.

inductancia nominal	resistencia característica	capacidad nominal	velocidad de propagación
2 $\mu\text{H/m}$	600 Ω	6 pF/m	$\approx 83\%$

pista en PCB.

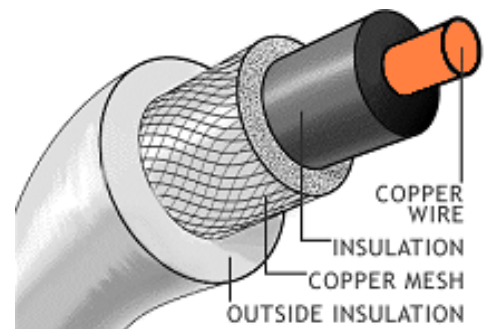
inductancia nominal	resistencia característica	capacidad nominal	velocidad de propagación
0,5 a 1 $\mu\text{H/m}$	70 a 100 Ω	50 a 150 pF/m	$\approx 66\%$

línea de un bus.

inductancia nominal	resistencia característica	capacidad nominal	velocidad de propagación
0,5 a 1 $\mu\text{H/m}$	20 a 40 Ω	1000 a 3000 pF/m	≈ 33 a 16 %

11.2.1-CABLE COAXIAL

- ❑ Los cables de RF más comunes son de tipo coaxial con apantallamiento simple, y pueden operar hasta los 500 MHz. Para operación en frecuencias más altas, frecuentemente se utiliza coaxial de doble apantallamiento. El apantallamiento doble tiene una efectividad del 99%. Si el apantallamiento se realiza con una fina capa de conductor como el aluminio, la efectividad llega, teóricamente al 100%.
- ❑ El cable coaxial básico está formado por un conductor central, rodeado por un dieléctrico que da soporte a una malla de cobre exterior concéntrica, cubierta a su vez por un aislante. El conductor interior suele hacerse mediante el trenzado de hilos de cobre o bien mediante un cilindro sólido de cobre.
- ❑ El dieléctrico es un material no conductor, cuya composición es fundamental para las buenas características de transmisión.
- ❑ La malla exterior de cobre, tiene como propósitos servir de contacto entre la masa de todos los equipos conectados, y al mismo tiempo atenuar la entrada y salida de interferencias eléctricas (energía de RF) producidas a frecuencias de trabajo relativamente altas. De hecho, los cables coaxiales son prácticamente inmunes a las radiaciones externas y ellos mismos irradian muy poca energía. La capa exterior de aislante, evita que la malla pueda entrar en contacto con otros elementos.



Características del cable coaxial tipo RG-59

Peso Kg/Km	resistencia cond. Central	resistencia malla	resistencia característica	capacidad nominal	velocidad de propagación																						
47	80 Ω/Km	17 Ω/Km (malla Cu de 100 hilos)	75Ω	68 pF/m	66%																						
Atenuación:																											
<table><tr><th>MHz</th><th>10</th><th>20</th><th>40</th><th>60</th><th>80</th><th>100</th><th>150</th><th>200</th><th>250</th><th>300</th></tr><tr><th>dB/100m</th><td>4.0</td><td>5.5</td><td>7.5</td><td>9.5</td><td>11.0</td><td>12.0</td><td>14.5</td><td>17.0</td><td>19.0</td><td>21.0</td></tr></table>						MHz	10	20	40	60	80	100	150	200	250	300	dB/100m	4.0	5.5	7.5	9.5	11.0	12.0	14.5	17.0	19.0	21.0
MHz	10	20	40	60	80	100	150	200	250	300																	
dB/100m	4.0	5.5	7.5	9.5	11.0	12.0	14.5	17.0	19.0	21.0																	

Características del cable coaxial tipo RG-58

Peso Kg/Km	resistencia cond. Central	resistencia malla	resistencia característica	capacidad nominal	velocidad de propagación																						
37	36 Ω/Km	15.5 Ω/Km (malla Cu de 100 hilos)	50Ω	101 pF/m	66%																						
Atenuación:																											
<table><tr><th>MHz</th><th>10</th><th>20</th><th>40</th><th>60</th><th>80</th><th>100</th><th>150</th><th>200</th><th>250</th><th>300</th></tr><tr><th>dB/100m</th><td>5.0</td><td>7.0</td><td>10.0</td><td>12.0</td><td>14.0</td><td>16.0</td><td>20.0</td><td>23.0</td><td>26.0</td><td>28.0</td></tr></table>						MHz	10	20	40	60	80	100	150	200	250	300	dB/100m	5.0	7.0	10.0	12.0	14.0	16.0	20.0	23.0	26.0	28.0
MHz	10	20	40	60	80	100	150	200	250	300																	
dB/100m	5.0	7.0	10.0	12.0	14.0	16.0	20.0	23.0	26.0	28.0																	

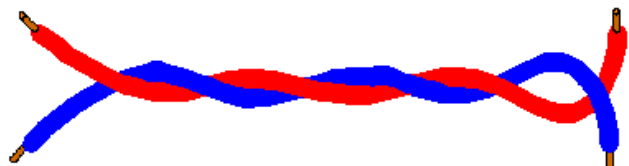
Características del cable coaxial tipo RG-58

Peso Kg/Km	resistencia cond. Central	resistencia malla	resistencia característica	capacidad nominal	velocidad de propagación																				
150	6 Ω/Km	4.7 Ω/Km (capa + malla Cu de 192 hilos)	50Ω	101 pF/m	66%																				
Atenuación:																									
<table><tr><th>MHz</th><th>100</th><th>200</th><th>300</th><th>400</th><th>500</th><th>600</th><th>700</th><th>800</th><th>900</th></tr><tr><th>dB/100m</th><td>5.3</td><td>7.5</td><td>9.3</td><td>10.8</td><td>12.2</td><td>13.5</td><td>14.5</td><td>15.8</td><td>17.0</td></tr></table>						MHz	100	200	300	400	500	600	700	800	900	dB/100m	5.3	7.5	9.3	10.8	12.2	13.5	14.5	15.8	17.0
MHz	100	200	300	400	500	600	700	800	900																
dB/100m	5.3	7.5	9.3	10.8	12.2	13.5	14.5	15.8	17.0																

11.2.2-PAR TRENZADO.

UTP (Unshielded Twisted Pair).

- Son pares de conductores, cada uno de ellos rodeado por su propio aislante, y que van trenzados tal y como se muestra en la figura.



- Cada uno de los conductores puede estar formado por un conjunto de hilos finos y su correspondiente aislante, o mediante un único cilindro macizo y su aislante.
- El par trenzado es una forma bastante eficaz y a la vez simple de reducir la captación de interferencia capacitiva (campo eléctrico) y magnética. El hecho de trenzar el cable hace que exista una distribución más homogénea de las capacidades unitarias del mismo. Esto a la vez garantiza que el acoplamiento en modo común va a estar equilibrado, permitiendo un alto rechazo a este modo (ruido).

Cuando en un campo magnético variable se introduce una espira de material conductor, en ésta se inducen corrientes eléctricas. La magnitud de la corriente inducida es proporcional al flujo magnético que atraviesa la espira, que a su vez es proporcional al área de la espira. El hecho de que exista la trenza, reduce el área de la espira a prácticamente cero. Cada giro invierte la dirección de la inducción, por lo que si se asume un campo externo uniforme, dos giros sucesivos anulan la interacción de los cables con el campo.

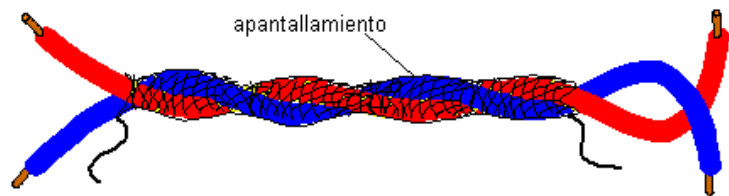
La captación de interferencia, se reduce a pequeñas zonas en los extremos del par, a la debida por la falta de uniformidad del campo externo y a las posibles irregularidades del trenzado. Cuando el área de terminación del par está dentro del campo, el número de giros por unidad de longitud no es importante, aunque lógicamente siempre se debe reducir este área de terminación sin trenzar, al mínimo.

Twisted Pair.

inductancia nominal	resistencia característica	capacidad nominal	velocidad de propagación
0,5 a 1 $\mu\text{H/m}$	80 a 120 Ω	50 a 100 pF/m	$\approx 66\%$

STP (Shielded Twisted Pair).

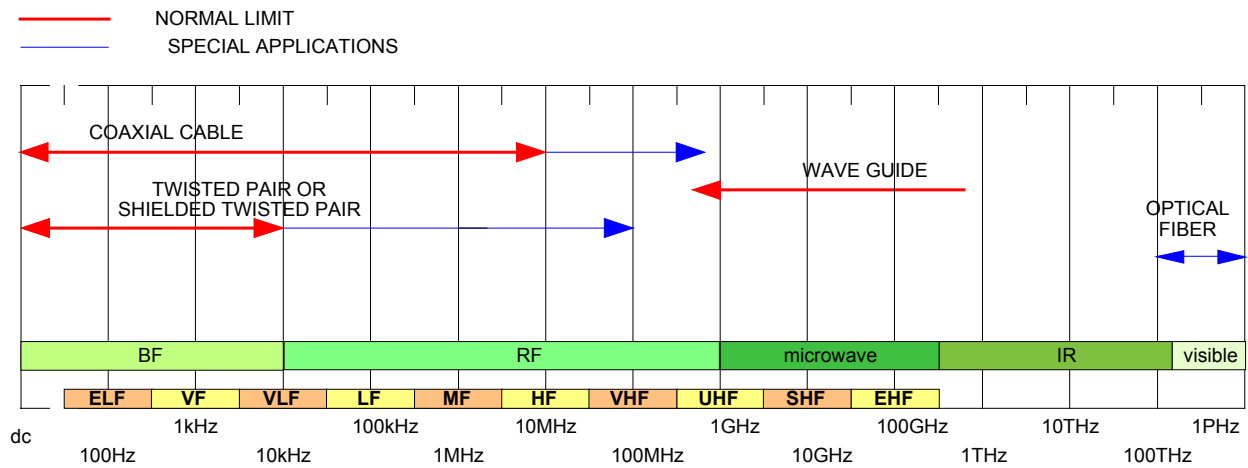
Este tipo de cable posee un apantallamiento metálico que rodea al par de hilos trenzados. Esta pantalla, además de reducir las interferencias externas o que la misma línea pueda radiar, sirve para la puesta a masa de la línea, lo que permite una transmisión de tipo diferencial.



FTP (Foiled Twisted Pair).

Este tipo de línea consiste en un conjunto de pares trenzados albergados en una misma canalización (mazo).

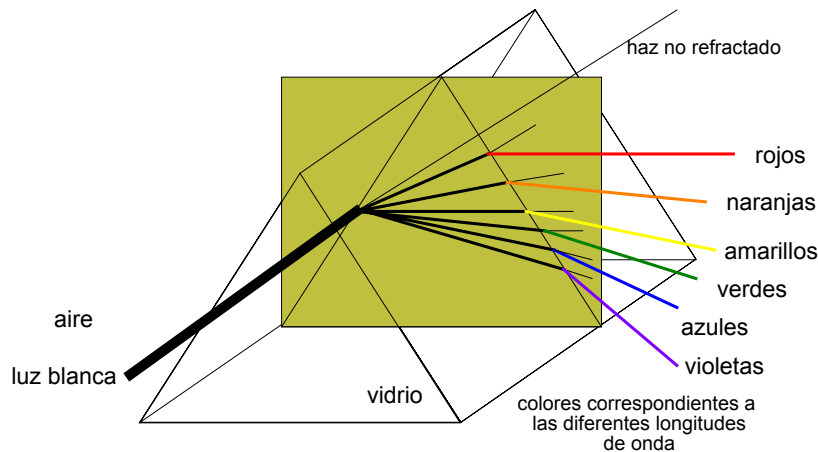
11.2.3-RANGO DE APLICACIÓN DE LAS DIFERENTES LINEAS:



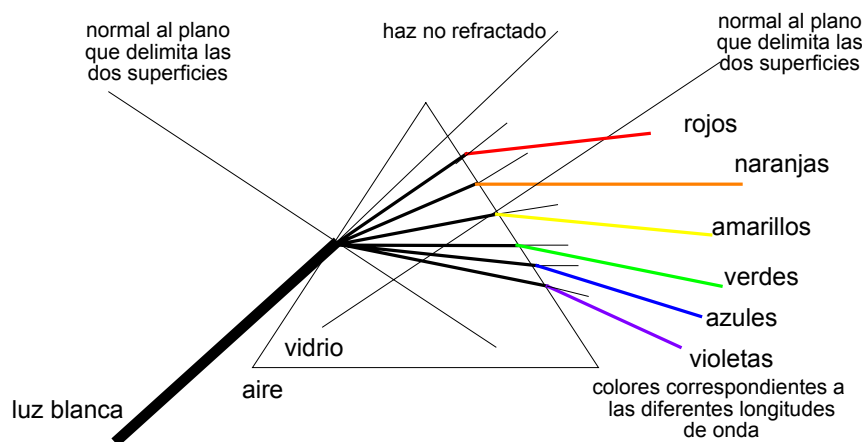
11.3-TRANSMISIÓN ÓPTICA.

REPASO DE ÓPTICA

- La luz en el espacio viaja con una velocidad de 29,9792458 cm/ns. Esta velocidad no es la misma cuando el medio es más denso que el espacio, ni tampoco es la misma para unas frecuencias y otras. Cuando la luz pasa de un medio con una determinada densidad a otro medio con densidad mayor, el haz de luz cambia de dirección es decir, se *refracta*, tendiendo su nueva dirección a acercarse a la normal en el punto de impacto de la luz, del plano que forman los dos medios.



- Por ejemplo, la luz solar que contiene todas las longitudes de onda, cuando pasa de un medio (aire) a otro medio más denso (vidrio), se refracta. Unas frecuencias se refractan mas que otras, dando origen al espectro de colores. El fenómeno de refracción también ocurre cuando la luz pasa de un medio más denso (vidrio) a otro menos denso (aire). En este caso, el haz refractado tiende a alejarse de la normal.



- El cambio de dirección que sufre el haz, es función del *índice de refracción* (n) dado por la fórmula

$$n = c/v$$

c = velocidad de la luz en el espacio libre

v = velocidad de la luz en el material específico

Índice de refracción de algunas sustancias

sustancia	n	sustancia	n
Sólidos a 20° C		Líquidos a 20° C	
Diamante (C)	2,419	Agua	1,333
Fluorita (CaF ₂)	1,434	Alcohol etílico	1,361
Sílice (SiO ₂)	1,458	Benceno	1,501
Vidrio óptico	1,52		
Cristal	1,66	Gases a 0° C, 1 atm	
Poliestireno	1,49	Aire	1,000293
Fibra de vidrio	1,5 – 1,9	Dióxido de carbono	1,00045
Silicio (Si)	3,4		
Galio-Arseniuro	3,6		

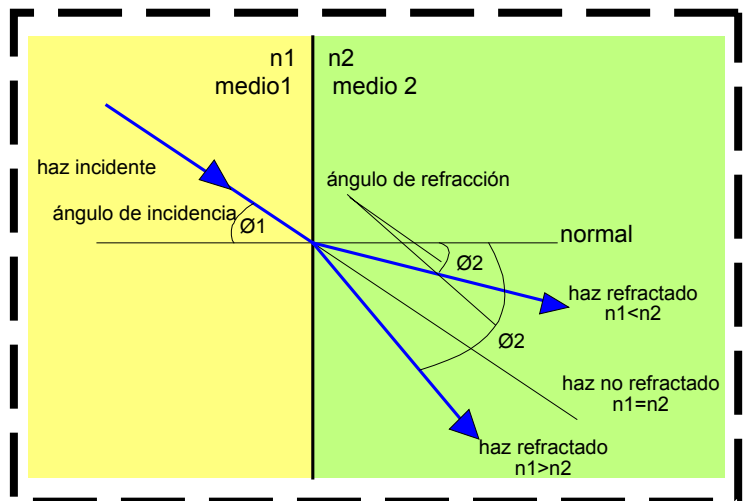
El índice está medido con una longitud de onda en el vacío $\lambda_0 = 589 \text{ nm}$

- La relación que existe entre el ángulo con el cual el haz incide en la unión de uno y otro medio y el ángulo del haz refractado, viene determinada por la *ley de Snell*:

$$n_1 \sin \theta_1 = n_2 \sin \theta_2$$

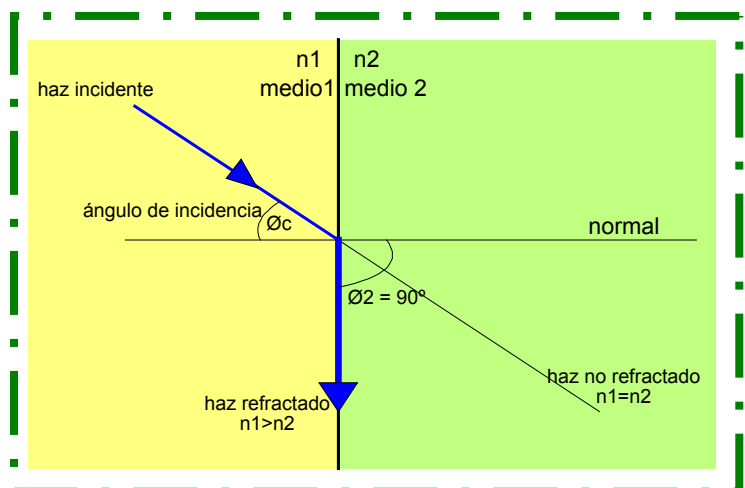
Los ángulos de los haces están siempre medidos respecto a la normal.

En la figura se muestra la refracción que presenta un haz para los casos en que el haz pasa de un medio menos denso a otro mas denso ($n_1 < n_2$) y para el caso contrario ($n_1 > n_2$). En el primer caso se ve como la nueva dirección del haz tiende hacia la normal, mientras que en el segundo caso, tiende a alejarse de la misma. Para el caso en que las densidades de los medios sean iguales ($n_1 = n_2$), el haz no sufre ninguna refracción.

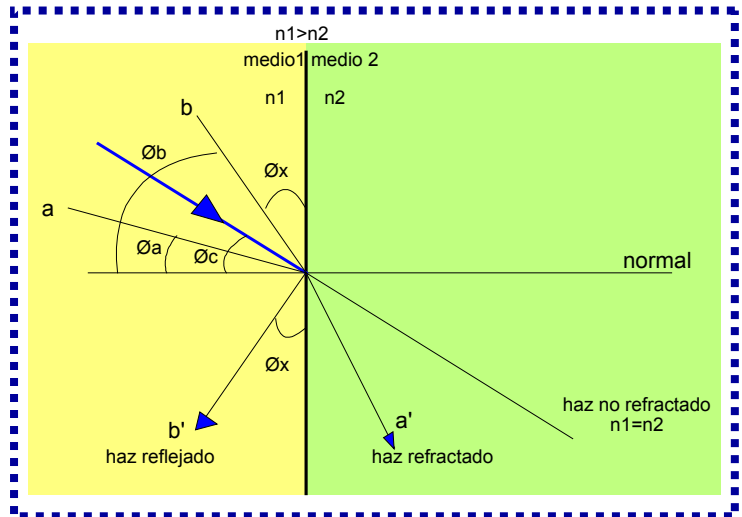


Ángulo crítico (θ_c).

- Cuando la luz incide desde un medio más denso a otro menos denso, el haz refractado tiende a alejarse de la normal. Puede darse el caso, de que la dirección de ese haz refractado siga la línea de unión de los dos medios, es decir, que el ángulo de refracción sea de 90°. En este caso, al ángulo de incidencia se le llama ángulo crítico, θ_c .



- Todos los haces que incidan sobre el medio con un ángulo de incidencia menor que el ángulo crítico (a), serán refractados (a') de la manera explicada hasta ahora. Sin embargo, los haces que inciden con un ángulo mayor que el ángulo crítico (b), serán reflejados (b'). En este caso, los ángulos de reflexión son iguales a los de incidencia de los respectivos haces ($90 - \theta_x$)



11.3.1-ESPECTRO DE FRECUENCIA ELECTROMAGNÉTICA

Hz		LUZ	nm	LUZ VISIBLE	nm
10^{24}	Rayos cósmicos de alta energía	UV extremo	≈ 10 a ≈ 200	Violeta	380
10^{23}					400
10^{22}	Rayos gamma	UV lejano	De ≈ 200 a ≈ 300	Azul marino	440
10^{21}					480
10^{20}	Rayos X	UV próximo	De ≈ 300 a ≈ 380	Verde azulado	520
10^{19}					560
10^{18}	Luz ultravioleta	Espectro luminoso	De ≈ 380 a ≈ 780	Verde Verde amarillo Amarillo Naranja	600
10^{17}					640
10^{16}					680
10^{15}					720
10^{14}	Luz visible	IR próximo	De ≈ 780 hasta $\approx 2 \cdot 10^3$	Rojo claro	760
10^{13}					780
10^{12}					
10^{11}					
10^{10}	Luz infrarroja	IR medio	$\approx 2 \cdot 10^3$ hasta $\approx 8 \cdot 10^3$	Rojo oscuro	
10^9					
10^8					
10^7					
10^6	Radio FM y TV	IR lejano	$\approx 8 \cdot 10^3$ hasta $\approx 6 \cdot 10^4$		
10^5					
10^4					
10^3					
10^2	Radio AM ultrasónico audio	IR muy lejano	$\approx 6 \cdot 10^4$ hasta $\approx 10^6$		
10^1					
10^0					
	subsónico				

11.4-FIBRA ÓPTICA

11.4.1-COMPARATIVA CON LÍNEAS METÁLICAS.

VENTAJAS.

- La capacidad (ancho de banda) de las líneas de fibra óptica es muy superior a la de los cables metálicos. Esto es debido a que en estas líneas no existen las inductancias y capacitancias distribuidas en la línea, que actúan como filtros, reduciendo la frecuencia de trabajo como pasa en las líneas metálicas.
- Estas líneas no son conductoras de electricidad, por lo tanto no tienen un campo magnético asociado a ellas. Esto trae como ventajas:

Inmunidad a las interferencias por inducción magnética.

Inmunidad a las interferencias estáticas producidas por rayos, motores y fuentes de ruido eléctrico.

No causan interferencias radioeléctricas en otros sistemas.

Más seguridad en la manipulación. Se pueden manipular cerca de líquidos y gases volátiles sin el peligro inherente a las líneas portadoras de electricidad de que haya una explosión.

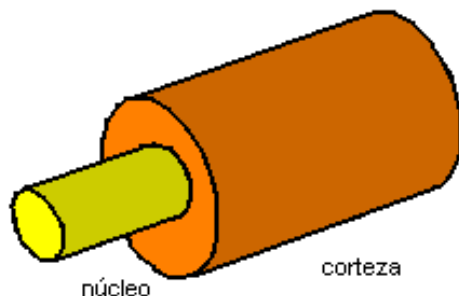
- ❑ Más resistentes a las condiciones climáticas o ambientales que las líneas metálicas. También menos peso y tamaño que los cables metálicos.

DESVENTAJAS.

- ❑ Un sistema basado en fibra óptica tiene un coste inicial mucho mayor que el sistema basado en las líneas metálicas. También el mantenimiento y reparación requiere de una tecnología más costosa.

11.4.2-COMPOSICIÓN.

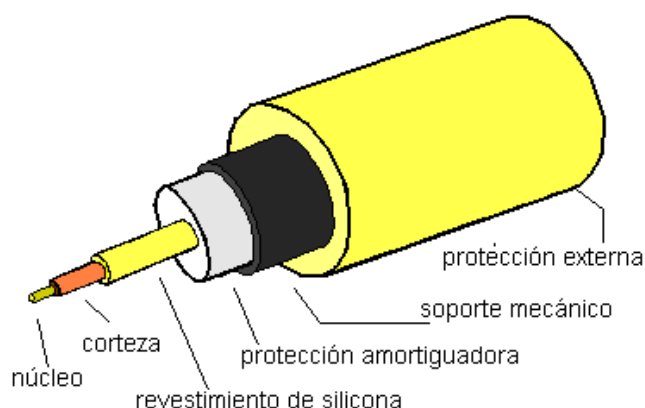
- ❑ En la forma más básica la fibra óptica esta formada por un núcleo con $\varnothing$ que puede variar entre 1 y 200 μm , y un revestimiento o corteza con $\varnothing$ que puede variar entre 125 y 300 μm . El núcleo es el camino para la señal luminosa modulada, la cual se refleja contra la corteza, de tal manera que evite la pérdida de energía.



El núcleo y la corteza, pueden estar fabricados de vidrio (sílice), o de plástico según las siguientes combinaciones:

NÚCLEO vidrio	CORTEZA vidrio
NÚCLEO vidrio	CORTEZA plástico
NÚCLEO plástico	CORTEZA plástico

- ❑ Las fibras de plástico tienen mejores cualidades mecánicas que las de vidrio, tales como menor peso, más flexibilidad, económicamente son menos costosas, pero como contrapartida, atenúan más la luz que las de vidrio.



- ❑ La estructura de una fibra óptica típica es la que se muestra en la figura.

El revestimiento de silicona evita rozamientos con la fibra por parte de la protección amortiguadora de vibraciones. El soporte mecánico aporta la rigidez para evitar roturas o cortes en la manipulación. Una protección externa de poliuretano o polietileno se encarga de la protección contra impactos, humedades, suciedades, etc.

11.4.3-PROPAGACIÓN DE LA LUZ EN LA FIBRA.

- La luz puede propagarse en un canal de fibra óptica por reflexión o por refracción. La forma de propagarse la luz (información), depende del *modo* de propagación y del *perfil de variación del índice de refracción* de la fibra.

Un modo, es cada una de las distintas posibilidades de propagación de la energía en el interior de una fibra. Simplificando esta definición se puede decir que es la posible trayectoria que un haz de luz puede seguir en una fibra. El número máximo de modos que pueden existir viene dado por

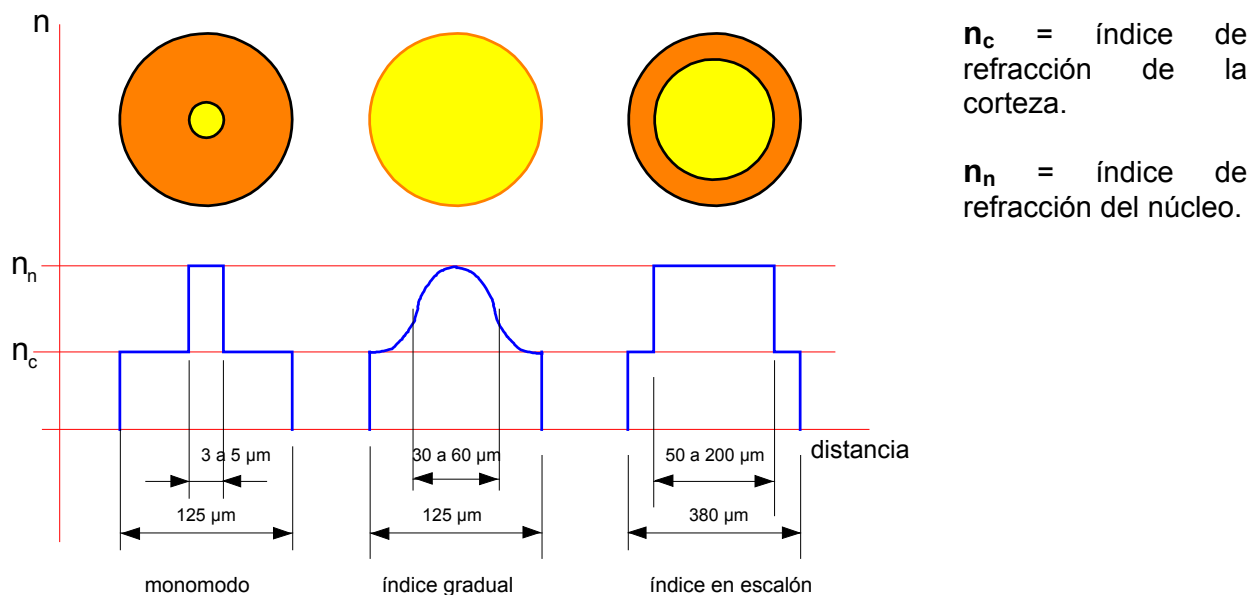
$$N = 1/2 (\pi d/\lambda)^2 (n_n^2 - n_c^2) \text{ donde}$$

d es el diámetro del núcleo
 λ es la longitud de onda
 n_n es el índice de refracción del núcleo
 n_c es el índice de refracción de la corteza

Un haz de luz que entra por un extremo de una fibra con diferentes ángulos, se verá refractado o reflejado en numerosas ocasiones en su camino hacia el otro extremo de la fibra, y por lo tanto llegará con diferentes fases. Los diferentes ángulos de entrada del haz, dan lugar a los diferentes modos de propagación. Una fibra que pueda soportar diferentes modos de propagación, se llama *fibra multimodo*. Aquella que solo soporta un modo de propagación se llama *fibra monomodo*.

El perfil de variación del índice de refracción es una representación gráfica del valor del índice de refracción con relación a la sección transversal de una fibra.

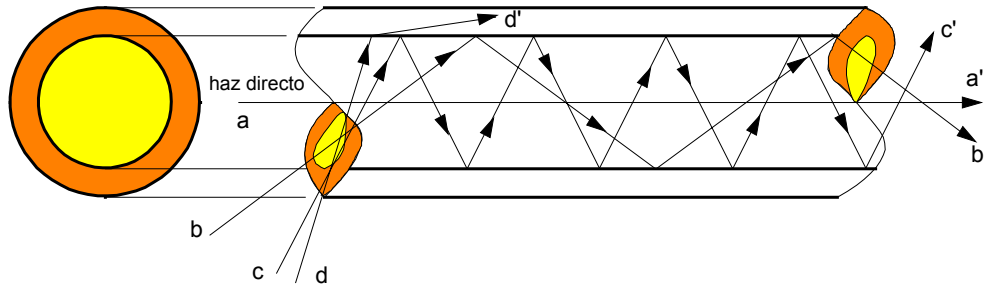
- La forma de transmitir la luz a través de una fibra, responde básicamente a dos métodos: **multimodo** y **monomodo**, y están basados en el tipo de fibra a utilizar. Desde el punto de vista práctico nos interesan las fibras *multimodo de perfil de índice en escalón*, *multimodo de perfil de índice gradual* y las *monomodo*. En la siguiente figura se representan los tres tipos de fibra y la representación de los respectivos perfiles.



Como se observa en la figura, las fibras monomodo, también tienen un perfil de variación del índice de refracción, en escalón.

FIBRA MULTIMODO DE ÍNDICE EN ESCALÓN.

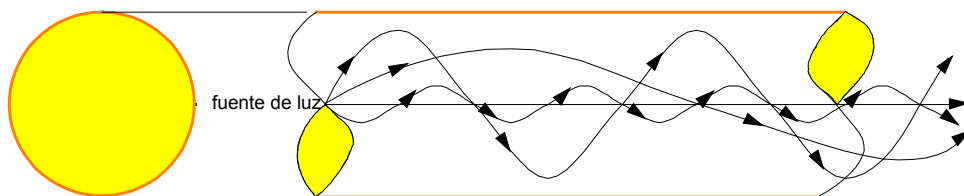
- Los índices respectivos del núcleo (n_n) y de la corteza (n_c), permanecen constantes a cualquier distancia radial del eje de la fibra.



- El perfil de índice en escalón presenta un cambio repentino en el índice de refracción en la unión núcleo-corteza, lo que en función de las leyes de la óptica geométrica, tiende a mantener dentro del núcleo a los rayos, que al propagarse por el mismo, inciden sobre la unión del núcleo-corteza con un ángulo mayor que el ángulo crítico. Estos rayos (en la figura los a, b y c), se propagan por el núcleo en forma de zig-zag, reflejándose continuamente en la frontera núcleo-corteza. Los rayos que inciden en la frontera núcleo-corteza con un ángulo inferior al ángulo crítico, (rayo d, en la figura) entra en la corteza refractado, perdiéndose.
- El rayo puede seguir múltiples trayectorias por el núcleo de la fibra en su propagación por ésta, por lo tanto, el tiempo que invierten en desplazarse de un extremo a otro es diferente. Los rayos que sufren mas reflexiones recorren un camino mas largo, siendo los que tardan más en el desplazamiento por la fibra, esto hace que un impulso de entrada a la fibra salga distorsionado en el otro extremo.

FIBRA MULTIMODO DE ÍNDICE GRADUAL.

- Esta fibra se caracteriza por un núcleo central (sin corteza) cuyo índice de refracción no es uniforme, si no que es máximo en el eje de la fibra, y va disminuyendo de forma gradual conforme nos alejamos de éste en dirección radial.
- Esta fibra se considera una fibra intermedia entre las otras dos, en cuanto a sus ventajas y desventajas.
- La forma de propagación en esta fibra es mediante la continua refracción del haz debido a la continua variación del índice de refracción.

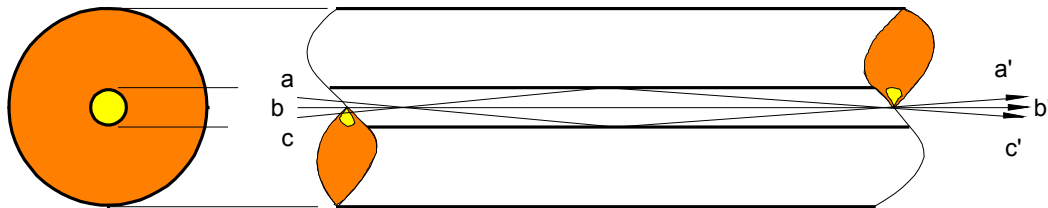


- La luz entra a la fibra con ángulos diferentes. Los haces que sufren mayores desviaciones recorren un camino mas largo que aquellos que se desplazan mas cerca del eje longitudinal de la fibra. Como la velocidad de propagación en un medio, sigue una ley inversamente proporcional al valor puntual del índice de refracción en el mismo, resulta que los rayos que recorren trayectorias mas largas, al ir en algunos momentos por caminos más rápidos (menor índice de refracción), resulta que llegan al final de la línea, casi al

mismo tiempo que los haces que se mueven cercanos al eje. Esta propiedad hace que con frecuencia a esta fibra se le dé el apelativo de “autoenfoque”.

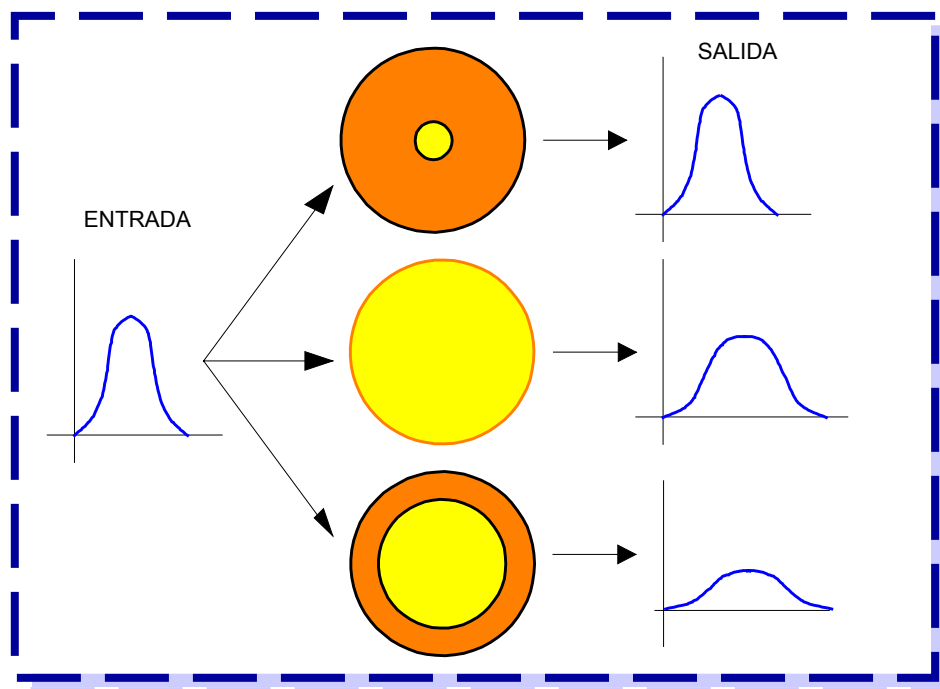
FIBRA MONOMODO.

- ❑ La forma de propagación en esta fibra es por reflexión.
- ❑ El núcleo es lo suficientemente pequeño como para que solo exista un modo de propagación del haz de luz.



- ❑ En este tipo de línea, resulta difícil acoplar la fuente de luz y el receptor a la fibra, debido al diámetro del núcleo. Sabemos que los rayos tienen que incidir sobre la frontera corteza-núcleo con un ángulo mayor que el ángulo crítico. Esto hace que la apertura para la aceptación de la luz de una fuente sea bastante estrecha, por lo que se hace necesaria una fuente de luz muy direccional, como un láser.
- ❑ Los rayos se propagan aproximadamente por la misma trayectoria, por lo que tardan aproximadamente el mismo tiempo en desplazarse por la línea. Esto significa que los impulsos de entrada son fielmente recuperados en el receptor.

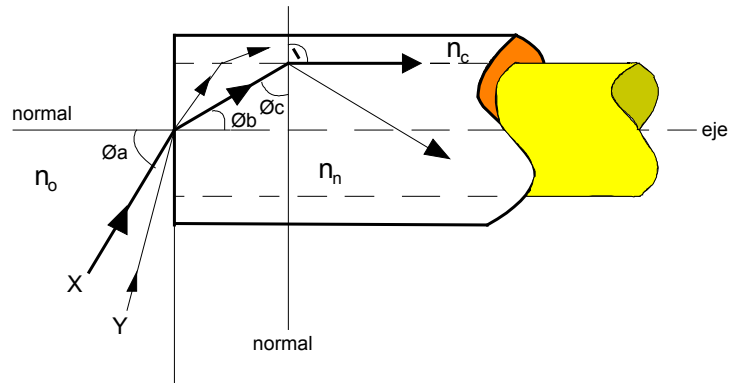
Como dato comparativo se muestra en la figura siguiente la respuesta de cada una de las fibras a un mismo impulso de entrada



11.4.4-CARACTERÍSTICAS

APERTURA NUMÉRICA. (NA)

- Este parámetro define la facultad para captar energía por parte de la fibra. Para evaluarlo, hay que tener en cuenta los efectos de refracción que sufre el haz en el lado de emisión al pasar del medio aire, al núcleo de la fibra y la refracción interna que sufre en la frontera núcleo-corteza.



Para que el haz pueda reflejarse dentro de la fibra, en la frontera núcleo-corteza, el ángulo incidente, debe ser mayor que el ángulo crítico, $\varnothing_c$. Si es menor, el haz se refracta, pasando a la corteza y por lo tanto se pierde.

Aplicando la ley de Snell a la cara de entrada tenemos:

$$n_0 \text{ sen } \varnothing_a = n_n \text{ sen } \varnothing_b \quad (I)$$

siendo n_0 el índice de refracción del medio externo, generalmente aire ($n_0 \approx 1$), y n_n el índice de refracción del núcleo. Como $\varnothing_b = 90 - \varnothing_c$, la ecuación (I) la podemos poner como,

$$n_0 \text{ sen } \varnothing_a = n_n \text{ sen } (90 - \varnothing_c) = n_n \cos \varnothing_c \text{ de donde}$$

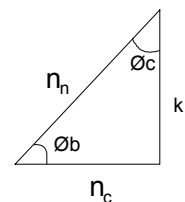
$$\text{sen } \varnothing_a = (n_n / n_0) \cos \varnothing_c \quad (II)$$

Por otro lado, por ser $\varnothing_c$ el ángulo crítico, se cumple (Snell): $n_n \text{ sen } \varnothing_c = n_c$. Es decir,

$$\text{sen } \varnothing_c = n_c / n_n$$

Dando forma geométrica a esta expresión, tenemos el triángulo rectángulo representado, de donde aplicando Pitágoras, obtenemos el valor del segmento k

$$k = (n_n^2 - n_c^2)^{1/2}$$



Podemos poner $\cos \varnothing_c = k/n_n = (n_n^2 - n_c^2)^{1/2} / n_n$

y sustituyendo en (II) tenemos $\text{sen } \varnothing_a = (n_n / n_0) (n_n^2 - n_c^2)^{1/2} / n_n$

$$\text{sen } \varnothing_a = (n_n^2 - n_c^2)^{1/2} / n_0 \text{ de donde}$$

$$\varnothing_a = \arcsen [(n_n^2 - n_c^2)^{1/2} / n_0]$$

- Este ángulo, será el máximo con el que un haz debe entrar en la fibra para que se produzca la reflexión en el interior. Si el ángulo de entrada es superior al dado por este valor (haz Y en la figura), el ángulo incidente desde el núcleo a la corteza, será inferior al ángulo crítico y por lo tanto, se producirá refracción, perdiéndose el haz.

Ángulo de aceptación: es el máximo ángulo al cual el haz externo puede chocar con la unión aire-fibra, y seguir propagándose por el interior de ésta. Viene dado por la fórmula

$$\theta_{a_{\max}} = \arcsen [(n_n^2 - n_c^2)^{1/2} / n_0]$$

Si se gira este ángulo alrededor del eje de la fibra, se obtiene el **cono de aceptación**.

Para evaluar de una forma cuantitativa la capacidad de la fibra para recoger la luz tenemos la apertura numérica (NA) que se define como

$$NA = \sen \theta_{a_{\max}}$$

$$NA = [(n_n^2 - n_c^2)^{1/2} / n_0]$$

y si el haz se recoge en la fibra por medio del aire, $n_0 \approx 1$ nos queda

$$NA = (n_n^2 - n_c^2)^{1/2}$$

Para una fibra de índice gradual, $NA = \sen \theta_n$

- Cuanto más grande es el factor NA, mas capacidad tiene la fibra para aceptar luz procedente de la fuente externa. Visto de otra manera, cuanto más mayor es este factor, más grande es el cono de aceptación y por lo tanto, más luz puede captar.
- La potencia inyectada en la fibra, es función de NA.

Para el caso en que la superficie de la fuente S_s sea menor que la del núcleo S_n , y la potencia que radia la fuente sea P_o , la potencia inyectada será

$$P = S_s P_o (NA)^2$$

Si la superficie del núcleo de la fibra S_n es menor, la potencia será $P = S_n P_o (NA)^2$. Estas expresiones también se aplican de forma aproximada a las fibras con índice gradual.

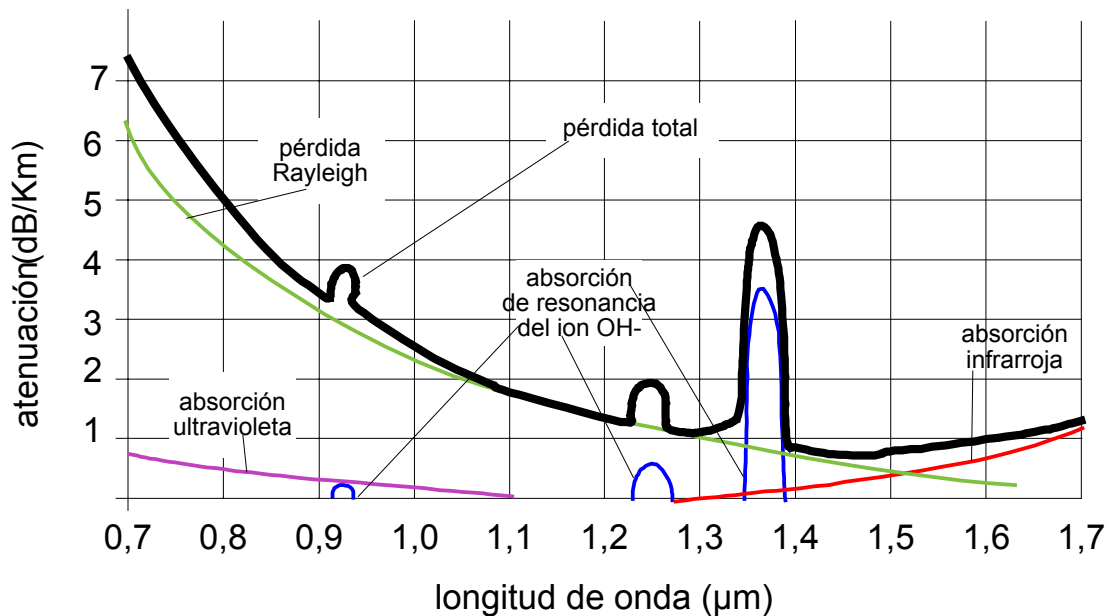
ATENUACIÓN.

Interesa saber que fracción de la potencia que entra en la fibra, va a alcanzar el otro extremo. Para saber esto necesitamos saber que potencia se pierde en la línea. Las causas de esta pérdida son variadas y se pueden clasificar en: *pérdidas por absorción, pérdidas por dispersión de Rayleigh o materiales, dispersión cromática o de longitud de onda, pérdidas de radiación, dispersión modal y pérdidas por acoplamiento*.

PÉRDIDAS POR ABSORCIÓN: La absorción se debe a la interacción luz-materia. Por un lado tenemos la absorción intrínseca del vidrio, entre otros motivos por resonancia mecánica de las moléculas que vibran en torno a su posición de equilibrio. También existen pérdidas por las impurezas en el material, estas absorben la luz y la transforman en calor. Las pérdidas por absorción se pueden clasificar en *absorción ultravioleta, absorción infrarroja y absorción de resonancia de ión OH⁻*.

PÉRDIDAS POR DISPERSIÓN DE RAYLEIGH O MATERIALES: En este caso, la dispersión de la energía tiene su origen en la presencia de irregularidades sobre las características de propagación de la fibra, cuando estas irregularidades son menores o comparables con la longitud de onda de la luz en juego. Estas irregularidades pueden ser defectos de cristalización de la fibra, burbujas microscópicas, modificación del índice de refracción, etc. La luz que choca

con estas irregularidades, se difracta, dispersándose en múltiples direcciones. Parte de la luz difractada se pierde por la corteza, representando una pérdida. A la dispersión también colaboran la distancia entre las irregularidades y la potencia óptica de la señal.



DISPERSIÓN CROMÁTICA: Este fenómeno ocurre por la variación de velocidad de una radiación cuando se propaga a través de un medio tal que el índice de refracción varía para cada longitud de onda. Es el mismo fenómeno de descomposición espectral que ocurre cuando la luz blanca atraviesa un prisma, cada longitud de onda viaja a una velocidad diferente. Si se trabaja con una fuente monocromática como un ILD (Injection Laser Diode), se elimina el problema de la propagación a diferentes velocidades.

PÉRDIDAS DE RADIACIÓN: Estas pérdidas se deben a micro dobleces de origen térmico-mecánico y a dobleces durante la manipulación de la fibra. Estos últimos, empiezan a tener importancia para radios de curvatura relacionados con el diámetro del núcleo de la fibra. Para un diámetro del núcleo comprendido entre 50 y 65 micras, comienza a haber pérdidas para radios de flexión menores de 10 cm. Para las fibras monomodo, este radio es menor.

DISPERSIÓN MODAL: Esta dispersión es debida a la diferencia en los tiempos de propagación de los rayos de luz que toman diferentes trayectorias por una fibra. Por ejemplo, para una fibra multimodo de índice en escalón, un impulso, formado por varios rayos de luz, de tal manera que uno se propague por el eje de la fibra, otro se propague con un número de reflexiones determinado, otro un número mayor de reflexiones, etc., estos rayos alcanzarán el extremo opuesto de la línea en tiempos diferentes. Esto resultará en una dispersión energética respecto al tiempo. Esto se llama dispersión modal y resulta en un impulso estirado y de amplitud reducida respecto del impulso que inicialmente fue emitido. En este caso hay que tener en cuenta que todos los rayos se propagan a la misma velocidad por la fibra, pero unos deben recorrer mas distancia que otros.

Para el caso de una fibra monomodo, como cada rayo debe recorrer la misma distancia, y la velocidad es la misma, no sucede el fenómeno anterior.

Para el caso de la fibra con índice gradual, aunque las trayectorias son diferentes los rayos que se propagan mas alejados del eje de la fibra viajan por caminos menos densos (más rápidos) cuanto más se alejan del eje por lo que el resultado final es que todos alcanzan el extremo opuesto mas o menos en el mismo tiempo.

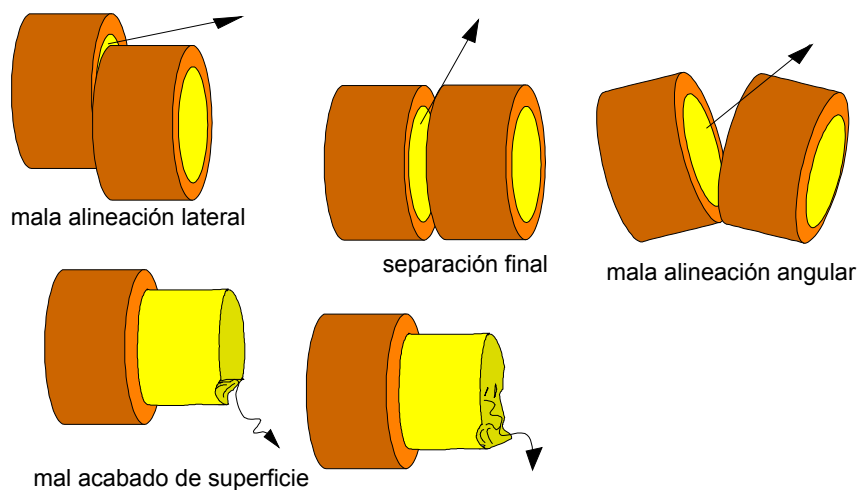
PÉRDIDAS DE ACOPLAMIENTO: Estas pérdidas pueden ocurrir en la unión fuente-fibra, en una unión fibra-fibra o en la unión fibra-receptor y se deben a la mala alineación lateral, mala alineación de la separación, mala alineación angular, o acabado de superficie imperfecta o sucia.

Las pérdidas por mala alineación lateral son imperceptibles si los ejes de la fibra están alineados dentro del 5% del diámetro más pequeño de la fibra.

Mala alineación de la separación. Cuando se realizan empalmes fibra-fibra, estas deben tocarse. Cuanto mas separadas queden, más pérdida habrá. Si el empalme se realiza mediante conector, las puntas no deben tocarse.

Mala alineación angular. Si el desplazamiento angular es menor de 2° , la pérdida será inferior a 0,5dB.

Problemas de superficie. Las puntas de las dos fibras unidas, deben estar muy pulidas y encuadrarse juntas. Si las puntas están a menos de 3° de la perpendicular, las pérdidas serán menores que 0,5 dB.

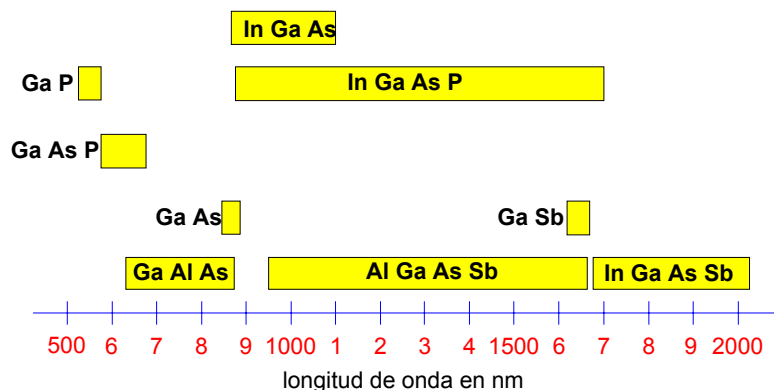


11.4.5-EMISORES.

- ❑ Los elementos utilizados como fuentes de luz en un sistema de información de fibra óptica, son básicamente los LED-s (*Light Emitting Diode*) en su versión IRED (emisión infrarroja) y los LD-s (*Laser Diode*).

Los diodos LED, cuando son atravesados por una corriente en polarización directa, emiten una radiación electromagnética en forma de luz (visible o no). Al contrario que una lámpara de incandescencia, el espectro de radiación está limitado a una estrecha banda de longitudes de onda, la cual está determinada sobre todo por el material semiconductor utilizado en su composición. Por ejemplo un semiconductor formado por Arsénico (grupo V) y Galio (grupo III), en la misma proporción, tendrá un comportamiento parecido a un semiconductor de Ge o de Si intrínsecos, pero con una conductividad inferior (salto de energía de la banda prohibida para el Si 1,12eV, para el Ge 0,807eV y para el Ga As 1,43eV). La formación de un cristal tipo P o tipo N se logra variando la proporción de uno u otro elemento. También, mediante dopado. Las energías puestas en juego según los elementos utilizados para formar los diodos, son las que determinan la longitud de onda de la radiación emitida según la ley de Planck: $E_1 - E_2 = h \cdot f$ donde $E_1 - E_2$ es la diferencia energética entre un nivel y otro, h es la constante de Planck = $6,626075 \cdot 10^{-34}$ Js y f la frecuencia de la radiación.

En la figura se da el espectro de radiación según el tipo de semiconductor utilizado.

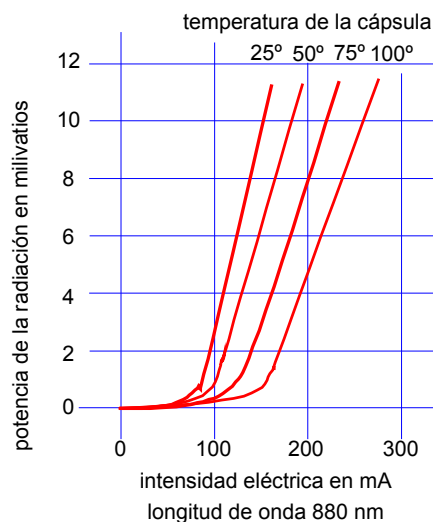
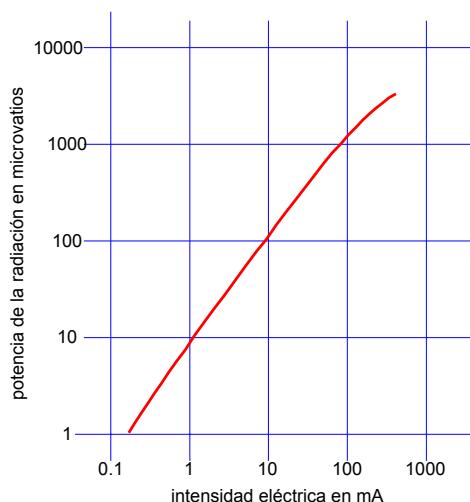


Un diodo láser, por ejemplo de Ga Al As, tiene un comportamiento parecido a un diodo LED por debajo de la corriente umbral, pero muy diferente por encima de ésta. La luz generada por un LED se dice que es de emisión espontánea, mientras que la generada por un LD se dice que es de emisión inducida.

Parámetros característicos para LED y LD.

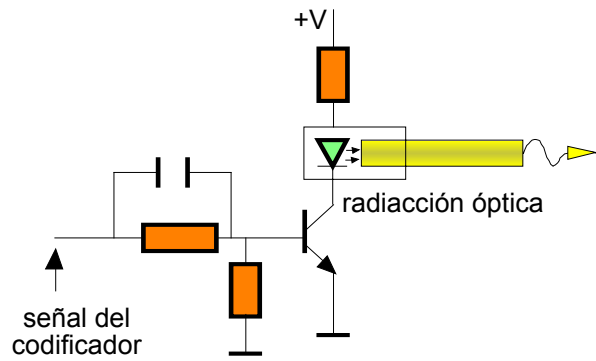
- *Longitud de onda de emisión (λ_e): longitud de onda de trabajo a la máxima potencia emitida.
- *Anchura espectral ($\Delta\lambda$): Ancho de banda espectral al 50% de la máxima potencia emitida.
- *Potencia óptica de emisión: potencia de radiación óptica emitida para un determinado valor de la corriente eléctrica.
- *Emitancia radiante: potencia óptica emitida por unidad de área radiante (W/cm^2)
- *Intensidad radiante: potencia óptica emitida por unidad de ángulo sólido (W/sr)
- *Radiancia: Potencia óptica por unidad de área radiante y unidad de ángulo sólido de emisión ($\text{W}/\text{cm}^2 \text{ sr}$).
- *Lóbulo de emisión: es la magnitud relativa de la potencia radiada en función de la dirección, a partir del eje óptico provisto en el encapsulado del emisor.
- *Corriente umbral: intensidad de la corriente eléctrica en un LD tal que para valores superiores de ella se produce el fenómeno de emisión estimulada.

En las siguientes figuras se muestran las características corriente- potencia de radiación para un LED y para un LD a diferentes temperaturas.



Ventajas de un LD frente a un LED: Es más fácil de acoplar a una fibra óptica debido al patrón de radiación más directo. Potencia radiante superior lo que permite su aplicación para trabajos a distancias superiores. Más rapidez de respuesta con lo que se puede utilizar a mayores velocidades. Reducción de la dispersión cromática.

En la figura se representa el esquema del circuito básico de excitación LED.



- Como desventajas están su mayor coste y su dependencia más alta a la temperatura.

11.4.6-RECEPTORES

- Los elementos utilizados como elementos detectores de luz en un sistema de información de fibra óptica son los fotodiodos PIN (P-Intrínseco-N) y los de avalancha (AFD), trabajando en la zona inversa de sus características, es decir, polarizados inversamente.

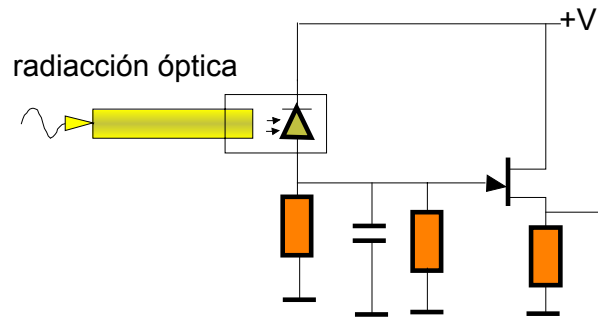
Las características de la radiación óptica emitida por el LED o LD, así como las de la fibra óptica, hacen que el receptor tenga que satisfacer una serie de requisitos para poder ser utilizado en el sistema:

- Sensibilidad a la longitud de onda de emisión para poder obtener una señal eléctrica como consecuencia de la radiación luminosa, incluso en las peores condiciones de potencia de esa radiación.
- Suficiente ancho de banda para evitar distorsiones introducidas por el detector.
- Mínima introducción de ruido a la señal.
- Estabilidad frente a las condiciones ambientales.

Parámetros característicos para los fotodetectores.

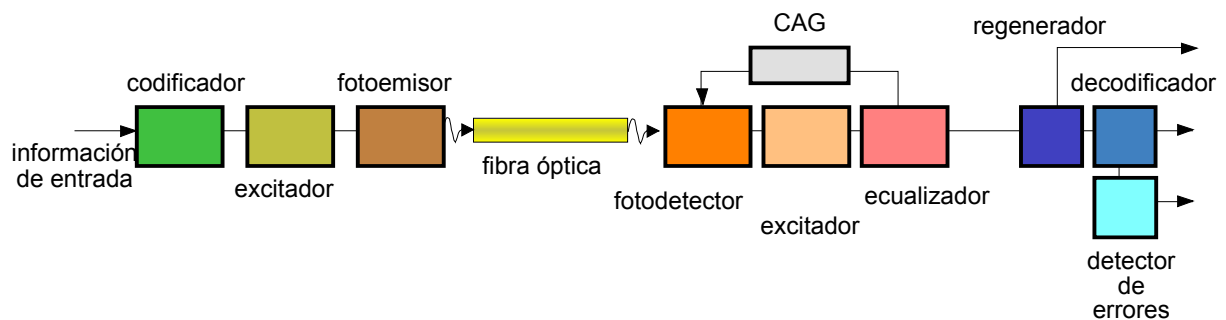
- *Eficiencia cuántica: promedio de electrones generados por cada fotón incidente
- *Sensibilidad espectral: relación entre la magnitud eléctrica de salida y la potencia óptica de entrada.
- *Corriente de oscuridad: corriente de salida del detector en ausencia de radiación óptica incidente.
- *Ancho de banda óptico: anchura espectral a la que corresponde una sensibilidad espectral igual o mayor al 50% del valor máximo.
- *Ancho de banda eléctrico: conjunto de frecuencias eléctricas obtenidas de la radiación óptica, que pueden ser detectadas con un nivel de potencia igual o superior al 50% del nivel máximo.
- *Potencia de ruido equivalente (NEP): Potencia óptica de entrada que se requiere para producir una señal eléctrica de salida igual al ruido del fotodetector por unidad de ancho de banda eléctrico.
- *Apertura de entrada: máximo ángulo de incidencia de la luz para el cual la superficie fotodetectora está totalmente iluminada.

En la figura se representa el esquema del circuito básico de captación.



CONFIGURACIÓN BÁSICA DE UN SISTEMA DE TRANSMISIÓN POR FIBRA ÓPTICA.

- Esquema básico de bloques de un sistema de transmisión por fibra óptica.



CODIFICADOR: se encarga de convertir la información a un código de línea.

EXCITADOR: se encarga de proporcionar los niveles de corriente necesarios para excitar al fotoemisor.

FOTOEMISOR: LED o LD.

FIBRA ÓPTICA.

FOTODETECTOR: PIN o AFD.

EXCITADOR: se encarga de amplificar las variaciones eléctricas procedentes del fotodetector. Puede ser un amplificador de transimpedancia.

ECUALIZADOR: se encarga de conformar la señal obtenida del excitador.

CAG, Control Automático de Ganancia: se encarga de mantener la señal recibida en los niveles óptimos para acoplarlos a la entrada del excitador y su posterior tratamiento.

DECODIFICADOR: recupera el código primitivo.

REGENERADOR: recupera el reloj.

DETECTOR DE ERRORES: detecta posibles errores para proceder a su corrección o respuesta conveniente.

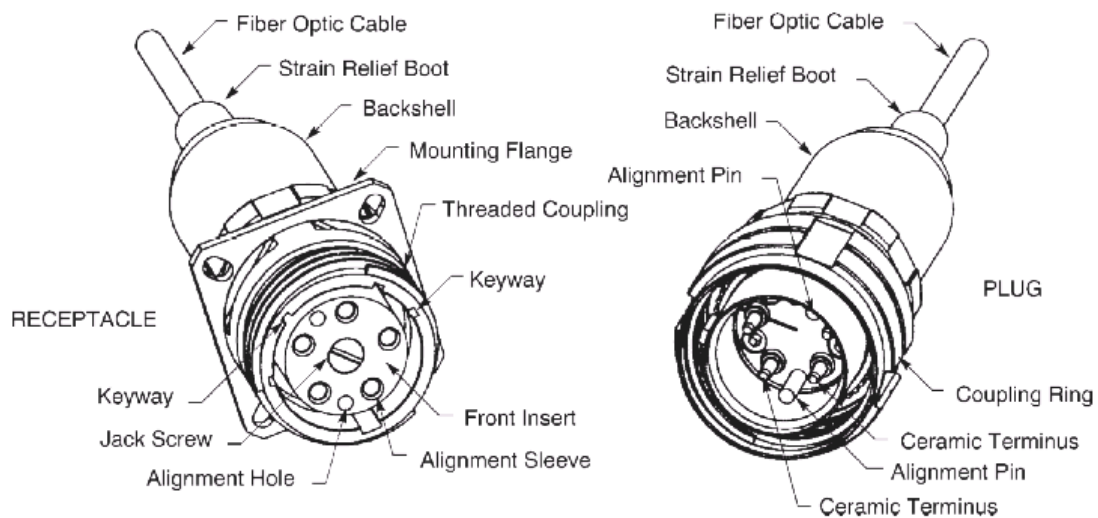
EMPALMES FIBRA-FIBRA

- Generalmente los conectores salen de fábrica con una longitud determinada de fibra óptica ya conectada a ellos. Esto garantiza las condiciones mas optimas de un empalme conector-fibra, dejando la conexión fibra-fibra para el trabajo en línea. Esta conexión no es tan crítica como la de conector-fibra.

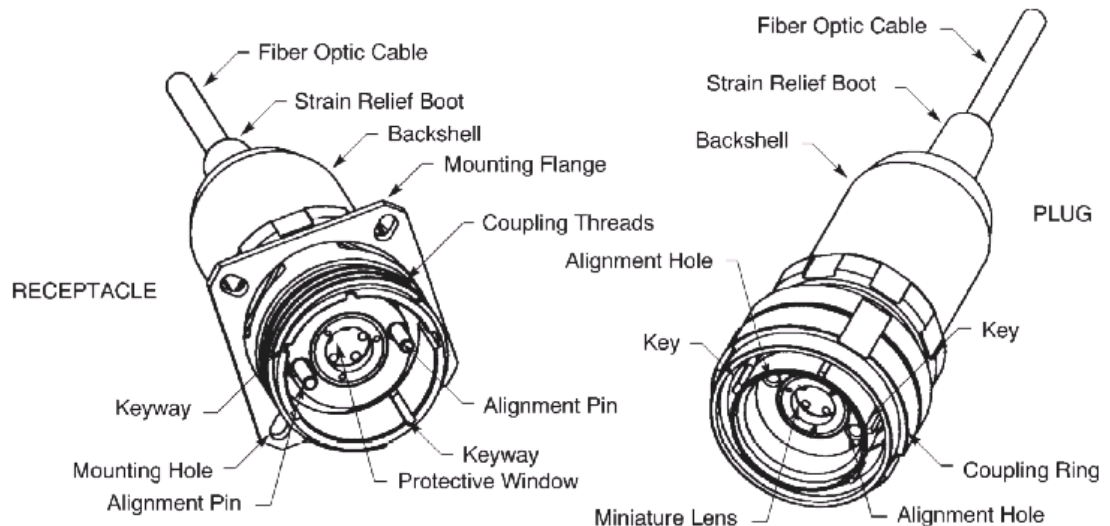
El empalme fibra-fibra puede realizarse bien mediante conectores desmontables o enchufables o bien mediante uniones fijas. Las uniones fijas se realizan generalmente mediante arco eléctrico o mediante llama con microscopio.

11.4.7-TIPOS DE CONECTORES. EJEMPLOS.

Plug y receptacle tipo A.



Plug y receptacle tipo B.





Data Buses

12

12.-BUSES DE DATOS EN AERONAVES.

12.1-ARINC 429 (AERONAUTICAL RADIO INC. 429)

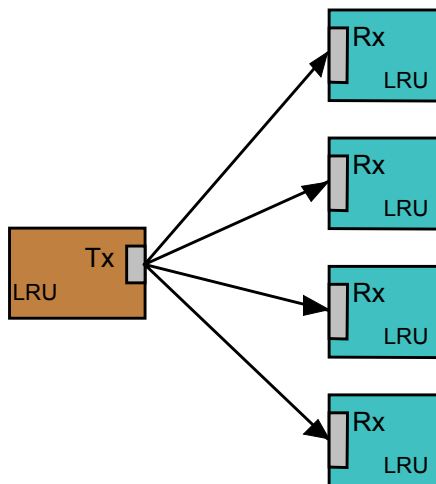
ARINC es una corporación en la que los accionistas son principalmente compañías de líneas aéreas americanas. También son accionistas, compañías de transporte aéreo, fabricantes de aeronaves, fabricantes de equipos para aeronaves americanos y líneas aéreas de otros países.

Entre sus actividades, ARINC patrocina la **AEEC** (Airlines Electronic Engineering Committee) para la especificación de los equipos aviónicos y sistemas de telecomunicación utilizados en líneas aéreas comerciales.

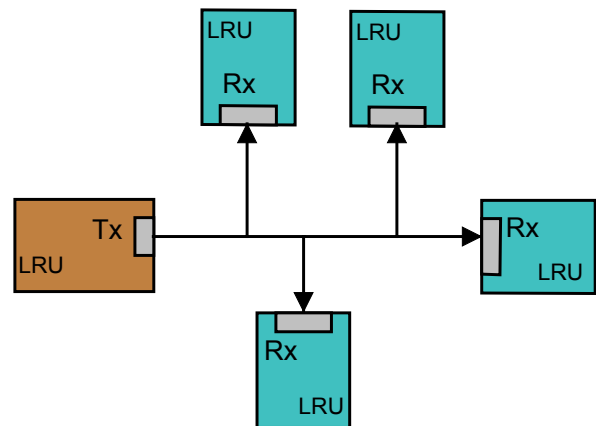
Una de estas especificaciones es ARINC 429, formalmente conocida como "Mark 33, Digital Information Transfer System (**DITS**)". Esta norma establece las especificaciones para la transferencia de datos digitales entre elementos y componentes de los sistemas aviónicos. La primera revisión de esta especificación se realizó en abril de 1978.

12.1.1-CARACTERÍSTICAS GENERALES.

- La especificación ARINC 429, define dos topologías diferentes para la interconexión de las LRU-s (Line Replaceable Unit): en Estrella y Lineal. La conexión en estrella, aunque es más segura por tener cada LRU su propia conexión a la fuente y en caso de que exista una rotura del bus, solamente se vería afectada la correspondiente unidad, tiene el inconveniente del mayor volumen de cable. Esto trae como inconveniente además del peso, la complejidad de instalación en la aeronave. Aunque en diseños antiguos se optó por la configuración en estrella, actualmente los diseños se realizan en configuración lineal.



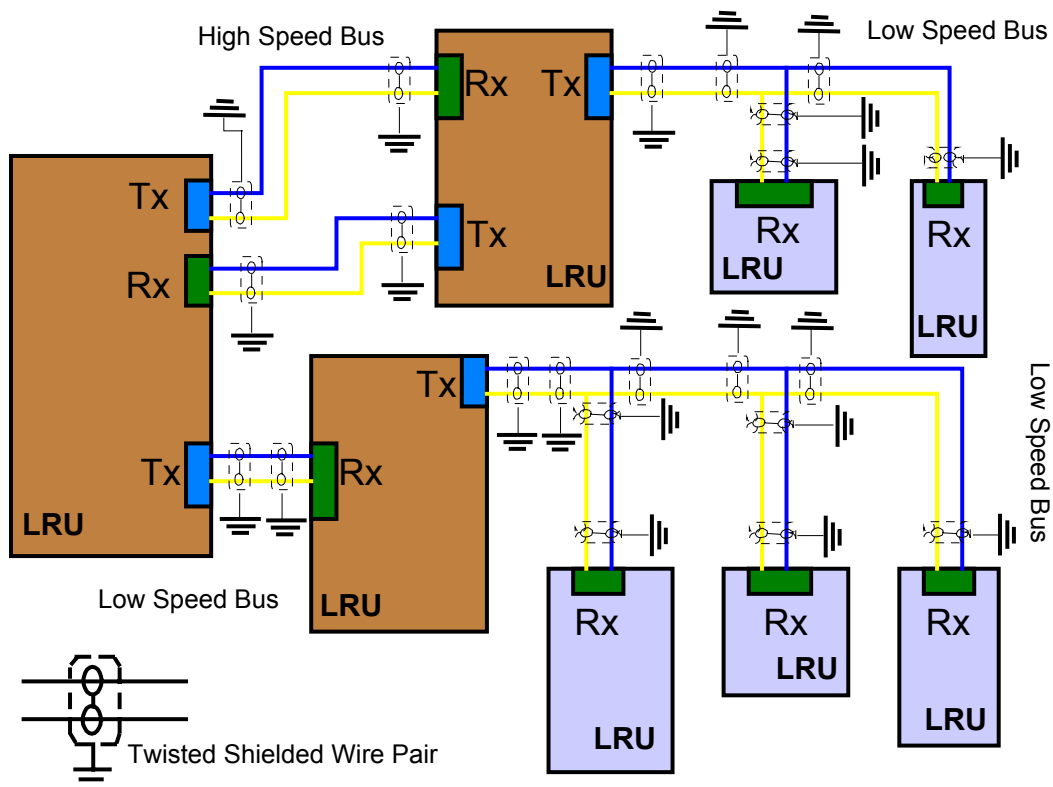
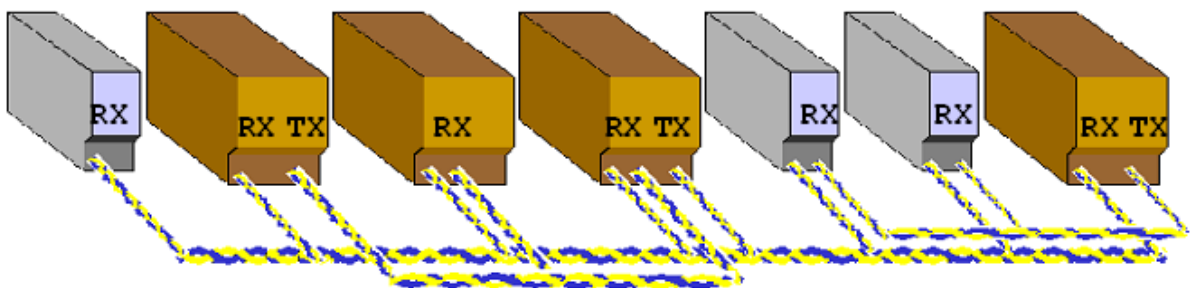
Configuración en estrella



Configuración lineal.

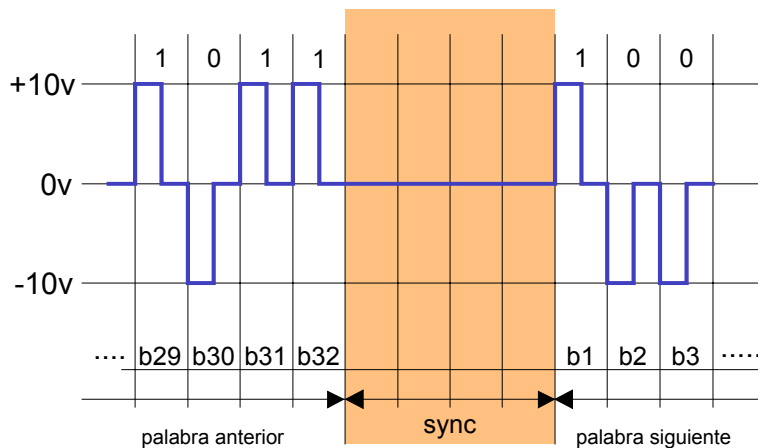
- El camino (bus de datos) utilizado para la transmisión periódica de las señales, consiste en un par de cables trenzados y apantallados (*Twisted Shielded Wire Pair*) con impedancia de 78 Ω .
- El flujo de datos en el bus es *unidireccional* (*Emisor-Receptor*). Si se necesita flujo de información en dos sentidos, se debe hacer uso de dos buses diferentes, es decir, dos caminos realizados mediante *Twisted Shielded Wire Pair*, uno para cada sentido de transferencia de la información.

- ❑ Cada bus está conectado a un solo emisor, pero puede ir conectado hasta a un máximo de 20 receptores. Cada uno de los receptores está programado para escuchar en el bus, y será sensible a la información específica que le atañe, pero no comunicará que ha recibido esa información, es decir, la transmisión se realiza en lazo abierto (*open-loop*), el receptor no “dialoga” con el emisor. Las LRU-s, no tienen una dirección determinada en el bus, poseen un número de equipo mediante el cual son agrupadas en tipos de sistemas/equipos.
- ❑ Por otro lado, cuando se transmiten grandes cantidades de información, es necesario cierto dialogo, como puede ser la indicación por parte del receptor, de que ha recibido la información. Este diálogo se realiza mediante un handshake basado en software. En este caso, como ya se ha indicado antes, son necesarios dos buses, uno para cada sentido. Una LRU puede tener uno o varios receptores y/o transmisores pero estarán unidos a diferentes líneas del bus.
- ❑ Cuando se realiza la conexión de las LRU-s mediante Twisted Shielded Wire Pair, el apantallado de este tipo de bus, debe conectarse a masa a ambos lados de la conexión, y también, en todos los conectores intermedios del bus.



SEÑAL CODIFICADA (MODULADA)

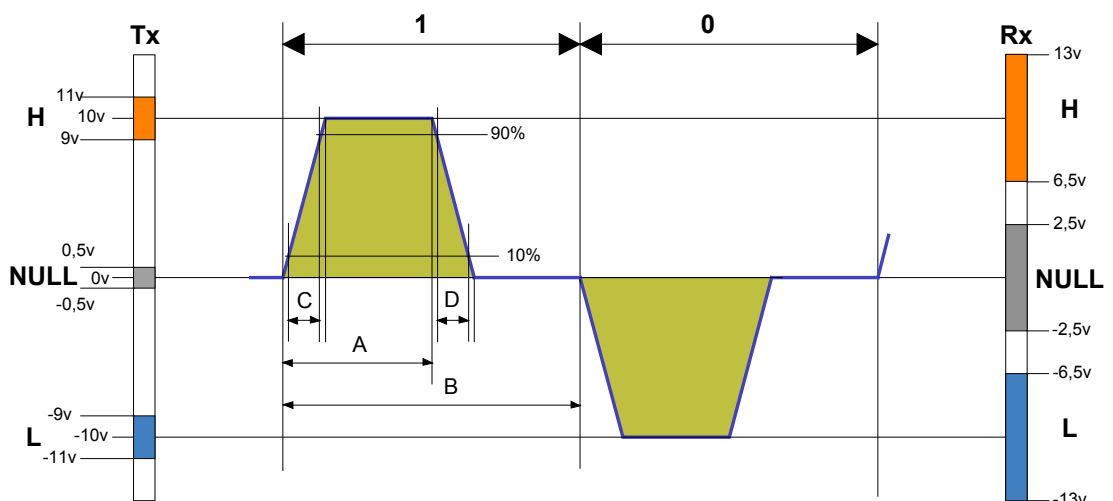
- ❑ La unidad de información está compuesta por palabras de 32 bits. El emisor envía cada palabra a través del bus, comenzando por su LSB (byte menos significativo), de forma codificada o modulada. El método de codificación utilizado es el *Bipolar R-Z*.
- ❑ La señal codificada tiene tres estados; *High*, *Low* y *Null*.



- La primera mitad de cada pulso contiene la información de BIT (1 o 0). La segunda mitad es siempre *null*. Sirve para facilitar la separación entre bits.
- La información de reloj se obtiene de las transiciones al comienzo y a la mitad del periodo de duración del BIT. El retorno siempre a cero en la segunda mitad de cada BIT, mantiene lo que se llama “*self-clocking*”.

- ❑ La sincronización de las palabras transmitidas viene dada por un espacio de tiempo igual al que corresponde a 4 periodos de BIT, con el valor *null*. El primer BIT después de estos “huecos” indica el comienzo de una nueva palabra.
- ❑ La velocidad de transmisión puede ser de *100 Kb/s* en el modo High Speed o de *12-14.5 Kb/s* en el modo Low Speed, dependiendo de los sistemas que estén interconectados. Cada régimen de velocidad está asociado a un bus determinado, es decir, no se comparte el mismo bus para transmitir a velocidades diferentes. En el diseño, se trata de evitar la velocidad de *13.6 Kb/s* para minimizar las interferencias con los sistemas “*Omega*” y “*Loran C*”.
- ❑ La señal modulada que llega a un puerto del receptor, es descompuesta para que la etiqueta sea decodificada y de esta manera seleccionar el registro apropiado para la decodificación de la señal y encauzamiento de la misma. También el receptor chequea el BIT de paridad.

FORMA DE ONDA.



	PARÁMETRO	HIGH SPEED	LOW SPEED
	Bit rate	100 kb/s $\pm$ 1%	12-14,5 kb/s
A	1ª mitad del pulso	5 μ s $\pm$ 5%	B/2 $\pm$ 5%
B	ciclo completo	10 μ s $\pm$ 2,5%	Z* μ s $\pm$ 2,5%
C	tiempo de subida	1,5 $\pm$ 0,5 μ s	10 $\pm$ 5 μ s
D	tiempo de bajada	1,5 $\pm$ 0,5 μ s	10 $\pm$ 5 μ s

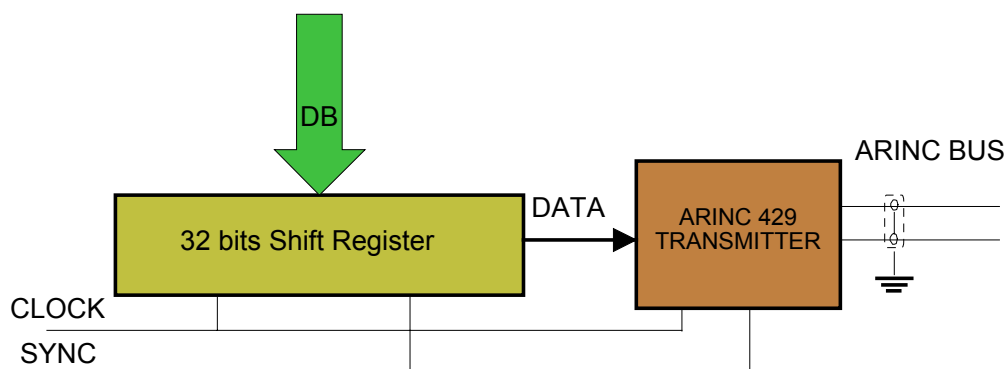
* Z=1/velocidad seleccionada entre 12-14,5kb/s.

TRANSMISOR ARINC 429

- ❑ Los emisores deben diseñarse para que puedan dar los correspondientes niveles de tensión diferencial, dependiendo del estado lógico:
 - El estado lógico 1 debe ser generado por el transmisor mediante el paso del estado NULL (0 $\pm$ 0,5 v) al valor 10 $\pm$ 1 v, y paso al valor NULL de nuevo.
 - El estado lógico 0, lo genera el transmisor mediante el paso del estado NULL al valor -10 $\pm$ 1 v y paso de nuevo al valor NULL.

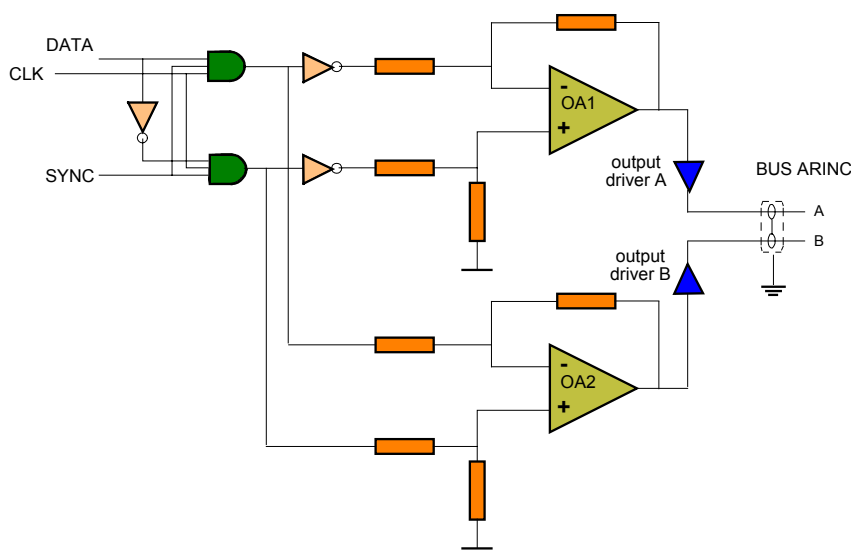
Estos valores de tensión se refieren a los que debe generar el emisor en circuito abierto.

- ❑ La impedancia nominal de salida debe ser, según la especificación, de 75 $\pm$ 5 Ω , aunque valores entre 60 Ω y 80 Ω , están dentro de los valores esperados.
- ❑ La CPU, según rutina programada, forma la palabra digital a ser enviada. Para ello obtiene los datos oportunos de la RAM, la etiqueta la obtiene de una tabla en ROM, y el código SDI lo toma de la entrada de *pins programables*. Esta información es tratada para formar el SSM (Sign Status Matrix) y calcular el BIT de paridad.
- ❑ La palabra formada, es puesta en la interfaz de salida, el cual la envía en forma serie al transmisor ARINC.
- ❑ La línea de sincronización debe asegurar el intervalo de cuatro espacios entre palabras. Esta línea estará en estado alto mientras se esté transmitiendo la información de cada palabra, y pasará a estado bajo para crear el intervalo de tiempo equivalente a cuatro periodos de BIT con valor *null* para separar dichas palabras.



CIRCUITO TRANSMISOR ARINC 429.

La figura representa de forma esquemática el modo de operación de un transmisor.

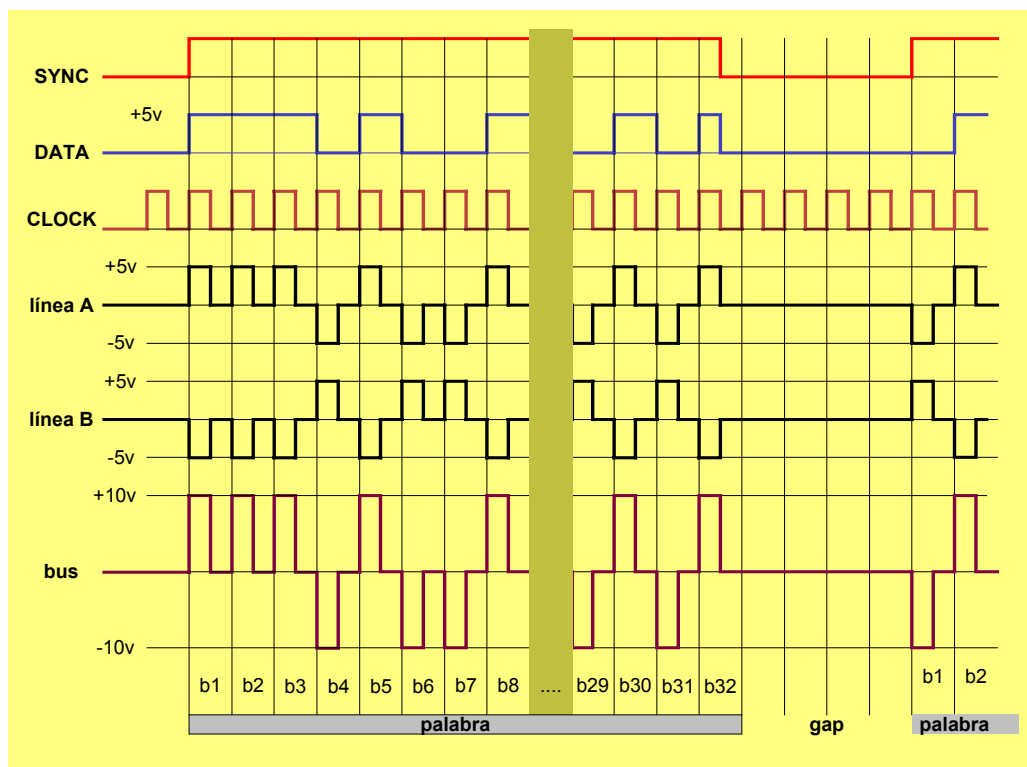


Si la señal de CLOCK o SYNC están a nivel bajo, los datos quedan inhabilitados puesto que las dos puertas AND mantienen sus salidas desactivadas es decir, a nivel bajo. En este caso tanto la línea A como la línea B están a 0 v.

Si se aplica señal de sincronismo, y señal de reloj, las salidas de las puertas AND tomarán el nivel correspondiente, en función de los datos que se apliquen a la

línea DATA. Entonces, cuando coincidan a nivel alto las señales de CLK y SYNC, y se aplique un 1 a la entrada DATA, la salida de la AND superior se activará, mientras la de la inferior permanecerá inactiva. Esto hace que el OA1 realice la diferencia entre una tensión positiva en su entrada no inversora y una tensión nula en la entrada inversora, dando como resultado una salida positiva. Al mismo tiempo, el OA2 realiza la diferencia de tensiones contrarias a las anteriores, es decir, entre una tensión nula en la entrada no inversora, y una tensión positiva en la entrada inversora, dando como resultado en la salida una tensión negativa. Cuando en la entrada de datos se aplica un cero, los fenómenos ocurren al contrario de lo descrito anteriormente.

El resultado de todo este proceso es que con un 1 lógico en la entrada de datos, en la línea A



del bus se va a tener una tensión positiva con respecto a masa, y en la línea B, una tensión negativa del mismo valor absoluto que la anterior, también con respecto a masa.

Para un 0 lógico, los valores de tensión serán justamente los contrarios en cada línea.

Con el diseño conveniente del transmisor, las tensiones que aparecen en cada una de las líneas del bus deben ser, según el caso de 5 V, de -5 V o de 0 V.

La información presente en el bus, se obtiene al tomar la tensión entre las dos líneas, es decir, en modo *diferencial* ($V_A - V_B$). Como ya sabemos, en este modo, el voltaje común en las dos líneas (como puede ser el ruido inducido) dará como resultado un valor nulo.

Para el caso en el cual la señal CLK o la señal SYNC están a nivel bajo, la tensión diferencial que se obtiene es de 0 v, correspondiente al valor NULL. Si la línea A está a 5 V y la B a -5 V, la tensión diferencial será de 10 V, y para el caso de que la línea A tenga -5 V y la B 5 V, la tensión será de -10 V.

- ❑ Debe quedar claro que un nivel lógico H no se representa por una tensión de 10 V, si no que se representa por una tensión de 10 V, seguido de un valor NULL con la respectiva relación de tiempos (DC $\approx$ 50%), según lo expuesto en la sección “forma de onda”. Lo mismo se aplica para un valor L.

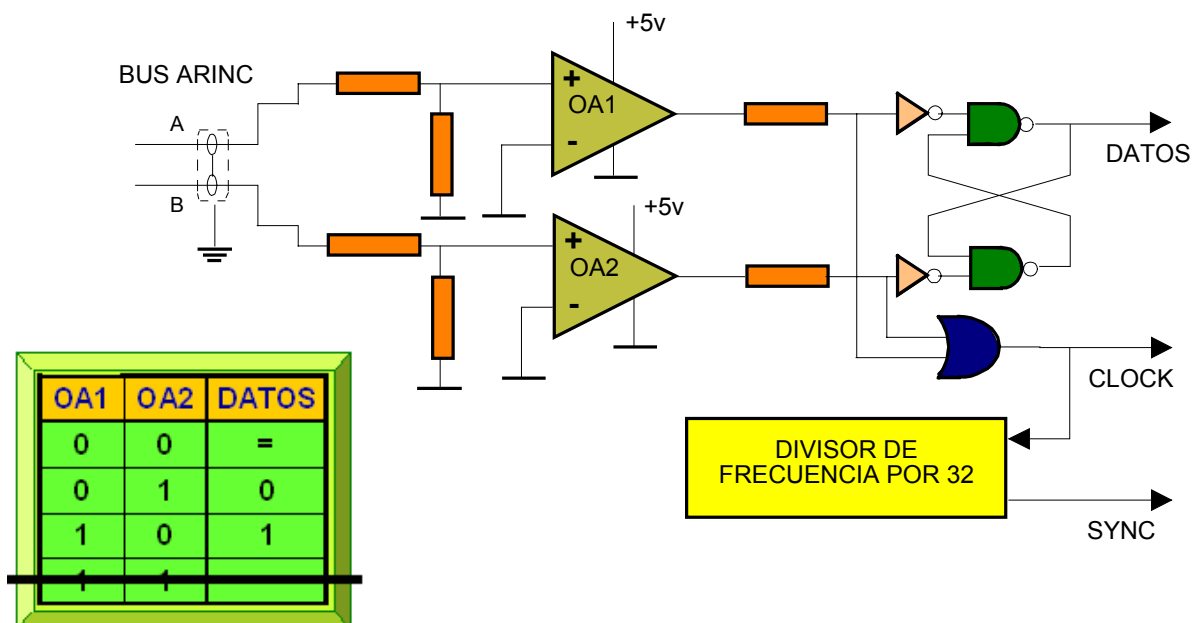
RECEPTOR ARINC 429.

En la figura se representa de forma esquemática el modo de trabajo de un receptor.

La señal procedente del bus es aplicada a unos circuitos que comparan la tensión en cada línea con una tensión de referencia.

Cuando el dato recibido es un 1 lógico, en la línea A habrá una tensión positiva, y en la línea B una tensión igual en magnitud a la de la línea A, pero negativa. Con estas condiciones, el OA1, entregará a su salida una tensión positiva, mientras que el OA2, entregará una tensión nula. Estas tensiones aplicadas a la báscula RS harán que esta ponga en la salida de datos un 1 lógico.

Cuando el dato que se recibe es un 0 lógico, la salida del OA1 tendrá un nivel nulo de tensión, mientras que la salida del OA2 tendrá un nivel alto. Estos niveles de tensión, aplicados a la báscula RS, hacen que esta ponga su salida de datos a un nivel de 0 lógico.

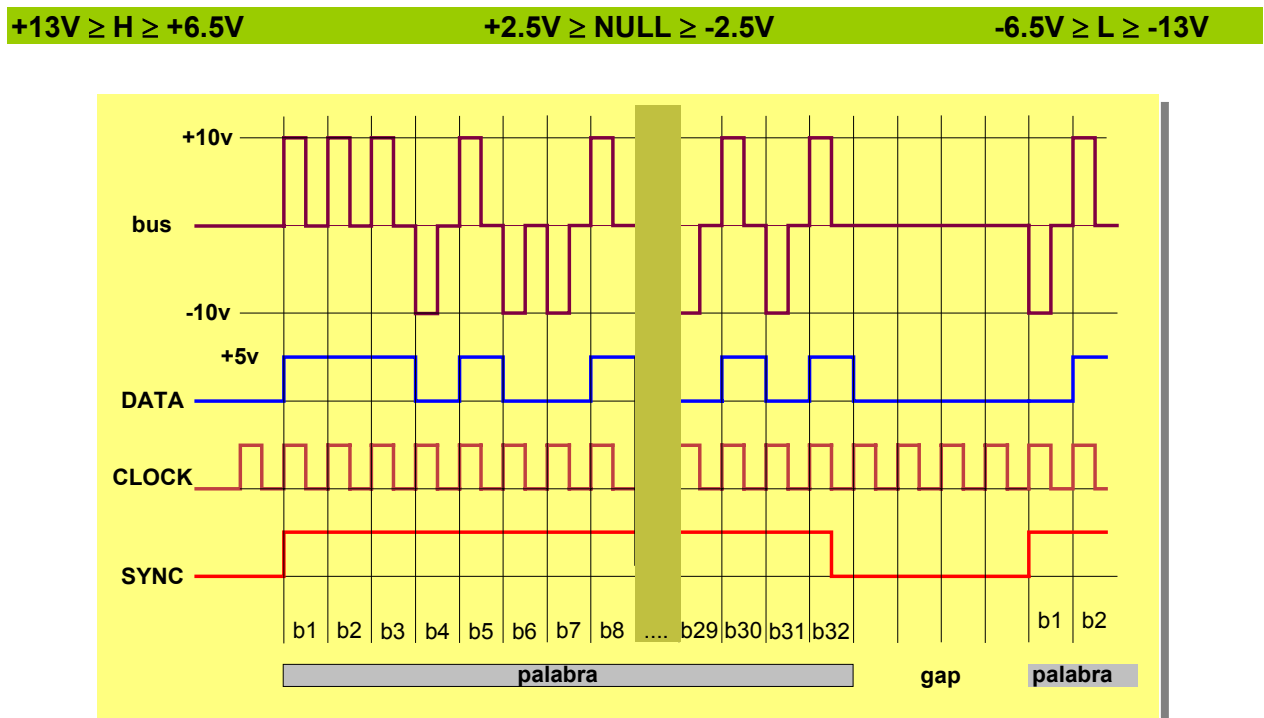


Aunque como ya hemos visto, los niveles de tensión recibidos, que marcan el cero y uno lógicos, solamente aparecen durante un tiempo igual a la mitad de la duración de un BIT, a la salida de datos del receptor, esta información de ceros y unos, aparece durante todo el periodo de BIT. Esto se debe a que las dos líneas que llevan la información al receptor, pasan por un valor nulo de tensión en cada periodo de BIT, durante un tiempo igual a la mitad de este periodo. El que las dos líneas del bus (A y B) tengan un nivel nulo de tensión, hace que el valor en las entradas de la báscula sea de 0, dando como resultado la continuidad del estado adquirido durante la primera mitad del periodo de BIT.

Se puede observar también, que por cada periodo de BIT, se va a obtener un ciclo de la señal de reloj. El estado alto de esta señal, coincidirá con la primera mitad del periodo de BIT, es decir, cuando la línea aporta un nivel de tensión correspondiente a un cero o uno lógicos. El estado bajo de la señal de reloj, coincidirá con la parte donde la línea aporta cero voltios, es decir, en la segunda mitad del periodo de BIT.

Un recuperador de sincronismo, va a generar un impulso cuya duración en estado alto va a ser igual a la de 32 bits (palabra transmitida) y cuya duración en estado bajo, será el equivalente en tiempo a 4 periodos de BIT.

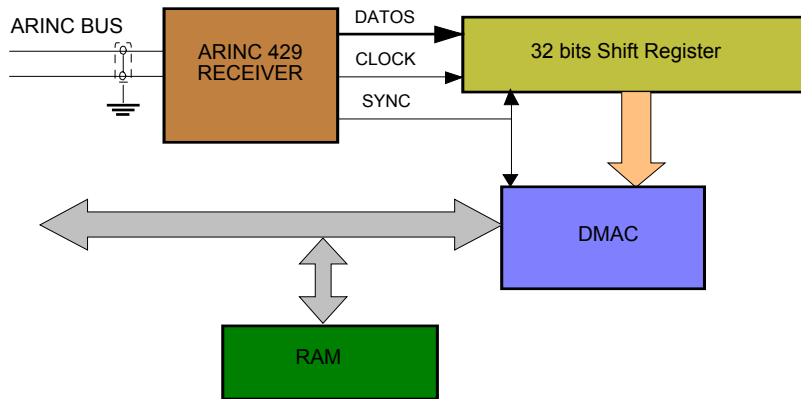
- El voltaje diferencial que va a aparecer en la entrada del receptor, se verá alterado en función de la longitud de la línea, número de receptores conectados, ruido y distorsiones. Por este motivo, los receptores deben ser diseñados para que en condiciones normales de trabajo puedan decodificar para cada nivel lógico, los rangos de voltaje que se dan a continuación.



- Es común encontrar receptores cuyo margen para el estado *null* es de $\pm 3,5$ V. También se pueden encontrar receptores que decodifican un estado H a partir de 4,5 V y un estado L a partir de $-4,5$ V.
- Por otro lado los receptores deben diseñarse para ser capaces de soportar tensiones en AC de 30 V en el modo diferencial, o de ± 29 V en modo común.
- La impedancia de entrada en ningún caso debe ser menor de $8k\Omega$, esto asegura que pueda haber hasta 20 receptores en el bus, sin sobrecargarlo. La capacidad diferencial de entrada, y la capacidad a masa no debe ser mayor de 50 pF.

DECODIFICACIÓN DE LA PALABRA

- De una forma genérica, el proceso consiste en recoger la palabra que llega del transmisor, convertirla de formato serie a formato paralelo y escribir esta información en la memoria RAM del sistema, donde será procesada de acuerdo a la rutina establecida.



- No todas las palabras que llegan al receptor deberán ser procesadas. Hay que tener en cuenta que a un solo bus pueden estar conectados hasta 20 receptores, por lo que unas veces la información irá destinada a unos y otras veces a otros. Para discriminar las palabras, el sistema hace uso de

una etiqueta identificativa que tiene asignada cada palabra.

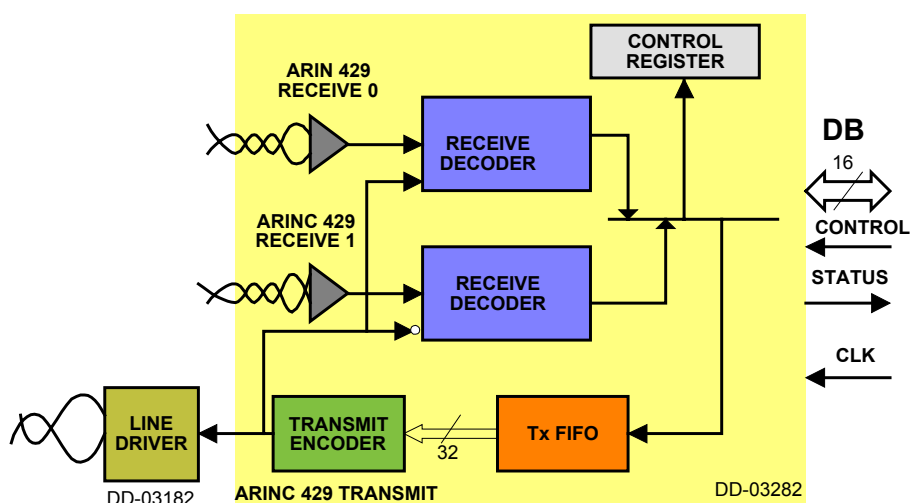
Además de la etiqueta identificativa, el procesamiento de la palabra estará condicionada a que el SDI sea el adecuado, para ello se compara el dato definido por el *pin program* de la instalación, con el recibido en las palabras aceptadas.

- Existen en el mercado diversos circuitos ASIC que realizan este tipo o similares tareas:

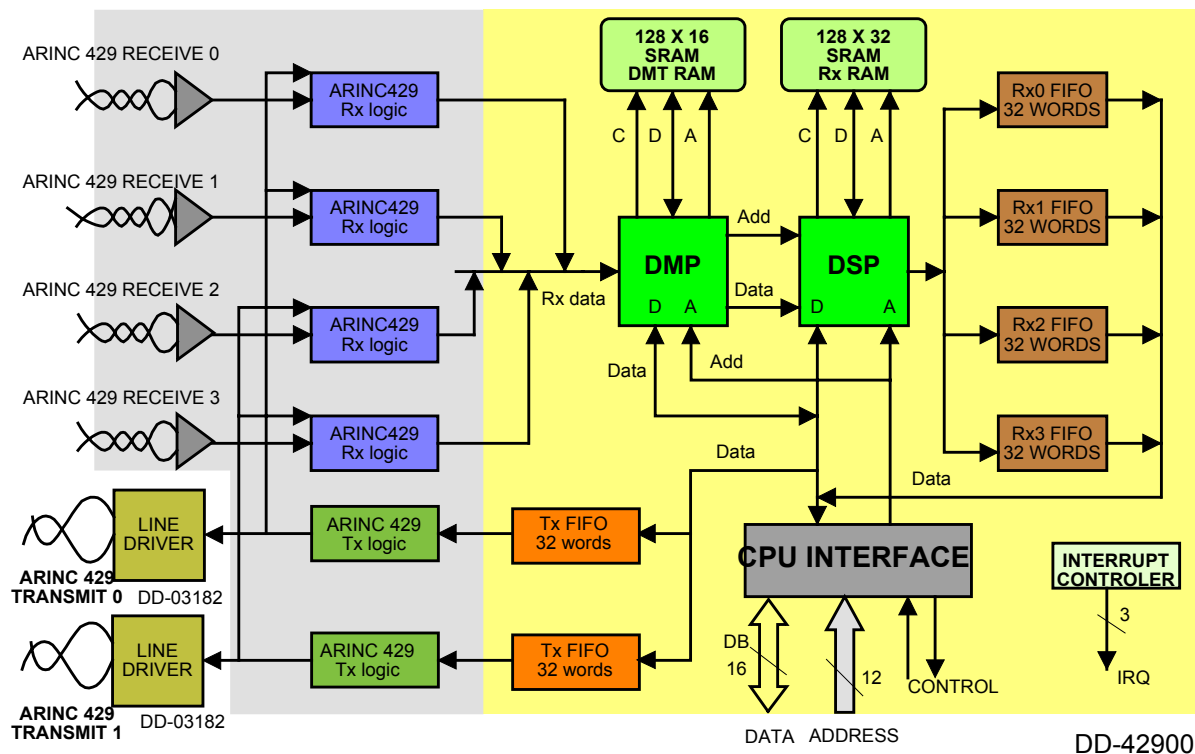
DD-03282 ARINC 429 TRANSCEIVER.-DDC (ILC Data Device Corporation)-.

Consiste en un receptor de dos canales y un transmisor de un canal. El transceptor cumple con la especificación ARINC 429. Proporciona una interfaz entre los estándares de buses digitales de datos serie, utilizados en los equipos aviónicos y buses de datos de 16 bits.

La salida del transmisor viene dada en niveles TTL. El formato de esta señal de salida no es el utilizado en el bus ARINC 429, si no que necesita ser aplicada a un driver de línea (DD-03182 de DDC) que la codificará al formato bipolar RZ.



DD-42900 ARINC 429 MICROPROCESSOR INTERFACE DEVICE. -DDC (ILC Data Device Corporation)-



Consiste en cuatro canales ARINC 429 de recepción (Rx0, Rx1, Rx2 y Rx3), cada uno de los cuales recibe datos de forma independiente. La velocidad de recepción (high speed o low speed) puede ser programada por separado para los canales Rx0 y Rx1 por un lado y Rx2 y Rx3 por otro.

Puede decodificar y ordenar los datos basándose en la etiqueta y en los bits SDI mediante el Data Match Processor (DMP) y guardarlos en una RAM y/o en una FIFO a través del Data Store Processor (DSP).

Tiene también, dos canales de transmisión (Tx0 y Tx1), que pueden transmitir de forma independiente y a velocidades programadas de forma también independiente. Los datos son enviados a través de dos FIFO-s de 32 x 32 bits (32 palabras de 32 bits cada una).

Tiene la posibilidad de generar tres interrupciones de propósito general y una basada en una condición de error.

PRUEBAS DE LOS BUSES DE DATOS.

- ❑ La fuente y los receptores de datos se incorporan dentro de las LRU, por lo que la comprobación de estos elementos se realizará en el taller.
- ❑ En algunos casos pueden ser necesarias pruebas en línea.
 - Analizadores digitales de buses ARINC 429: pueden trabajar en modo emisor o en modo receptor, además, suelen disponer de probadores de cableado (Interface Technology, mod 429 HBA, JC AIR 429).
 - En aviones como el 747-400, 777, A340, los datos que llegan a algunos sistemas pueden ser comprobados utilizando las MCDU-s, mediante la función de "monitoreo de datos de entrada" perteneciente al sistema de control de mantenimiento.

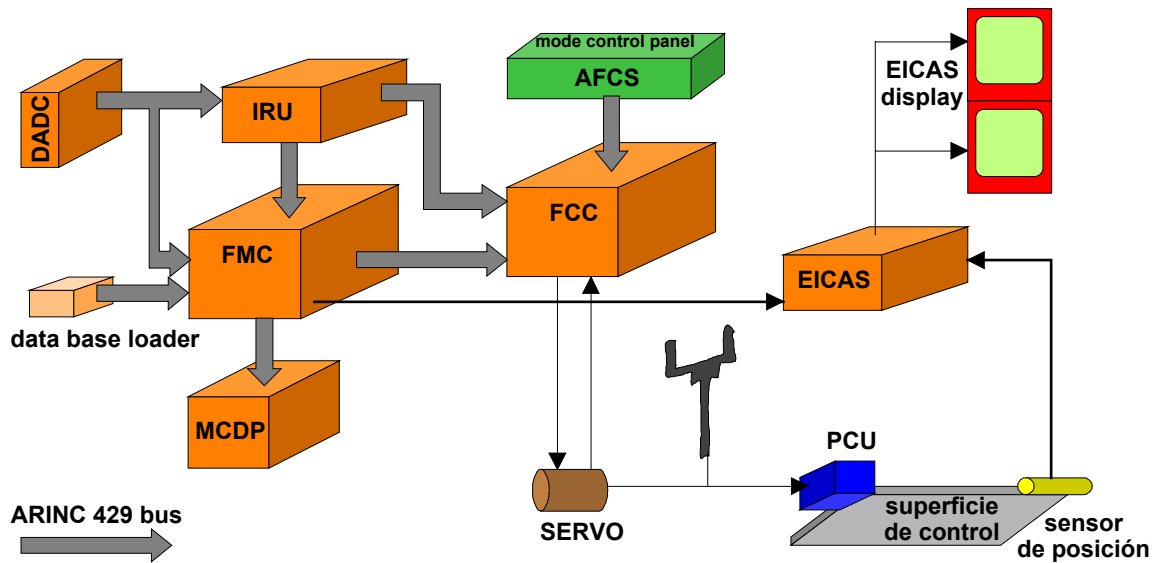
EJEMPLO.

La figura siguiente muestra la operación automática y manual de una superficie de control, con las diferentes LRU-s y la intercomunicación entre ellas mediante los buses ARINC-429.

En la forma manual, el capitán y el primer oficial controlan la superficie mediante las columnas de control. Estas columnas mueven cables que actúan sobre las PCU-s (Power Control Units), las cuales mueven la superficie de control. Un sensor de posición informa de la posición de la superficie, vía EICAS (Engine Indication and Crew Alerting System) el cual la muestra en el correspondiente display.

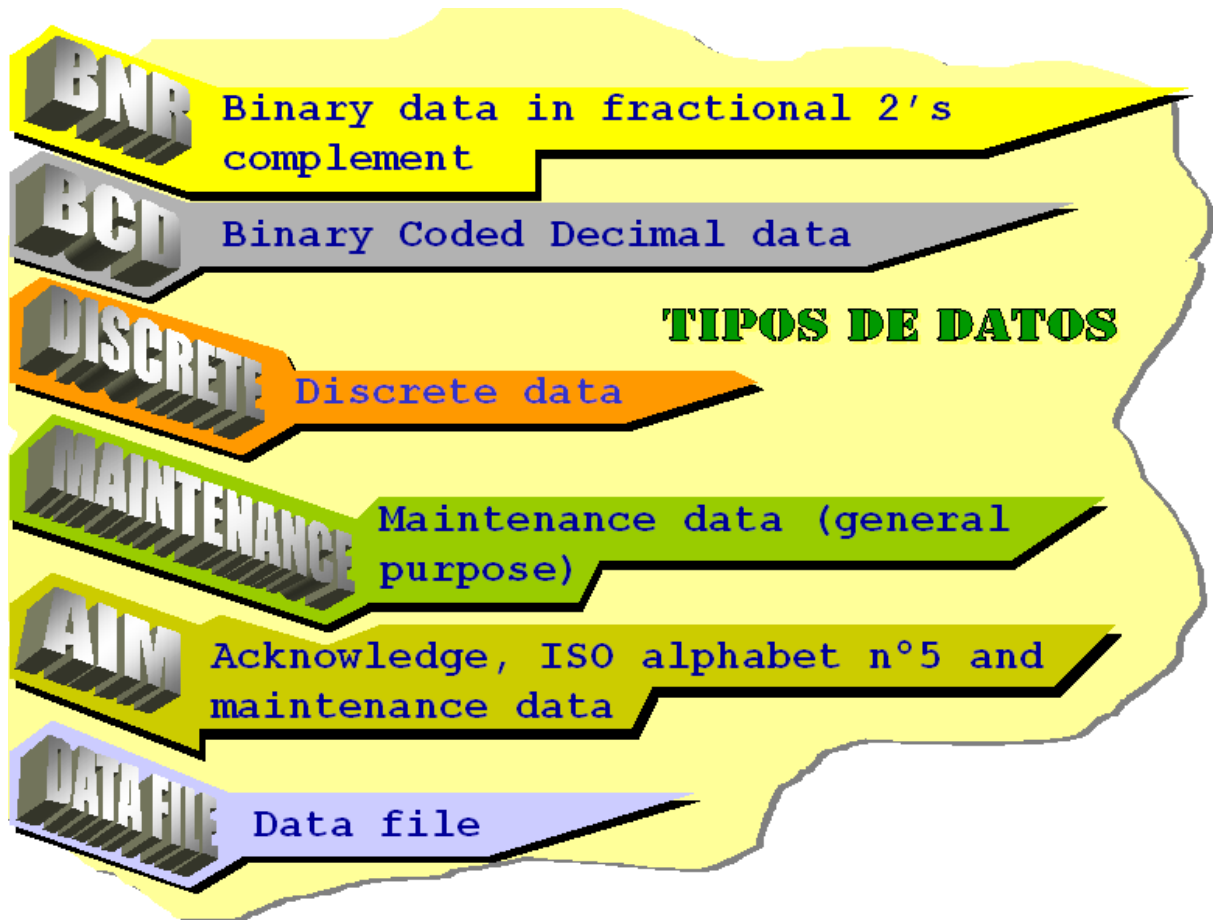
En la operación automática el DADC (Digital Air Data Computer) envía información digital sobre la altitud, y la velocidad del aire a la IRU (Inertial Reference Unit) y al FMC (Flight Management Computer). La IRU utiliza estos datos junto con la posición y actitud de la aeronave, generados internamente, para computar la posición actual, rumbo, y actitud. La posición actual y el rumbo se envían al FMC, y la actitud se envía al FCC. La información procesada por la IRU y FMC es enviada al FCC (Flight Control Computer) que genera las señales oportunas para el servo, el cual está conectado mediante cables mecánicos a la PCU. La actuación del servo mueve los cables que operan sobre la PCU y mueven la superficie. La información sobre la posición de la superficie es enviada al EICAS.

El FMC contiene un circuito BITE que realiza auto-test y guarda los posibles fallos durante el vuelo. Cuando la aeronave aterriza, los fallos se transfieren automáticamente al MCDP (Maintenance Control and Display Panel).

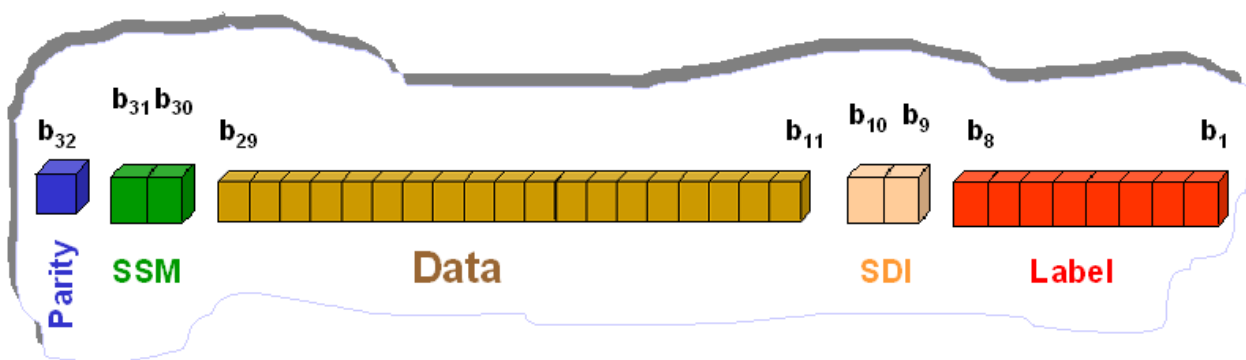


12.1.2-PROTOCOLO.

- La información que se transmite por un bus ARINC 429, puede ser de seis tipos:



- El elemento básico de información es una palabra digital compuesta por 32 bits. La transmisión de la misma, se realiza en intervalos prefijados, independientemente de que los datos contenidos en la palabra hayan cambiado o no. El orden de la transmisión de los bits es $b_8, b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_9, b_{10}, b_{11}, \dots, b_{32}$.

ESTRUCTURA DE LA PALABRA DIGITAL:

SSM: Sign Status Matrix. (b_{30}, b_{31})

Data area. ($b_{11}-b_{29}$)

8 bits label (b_1-b_8)

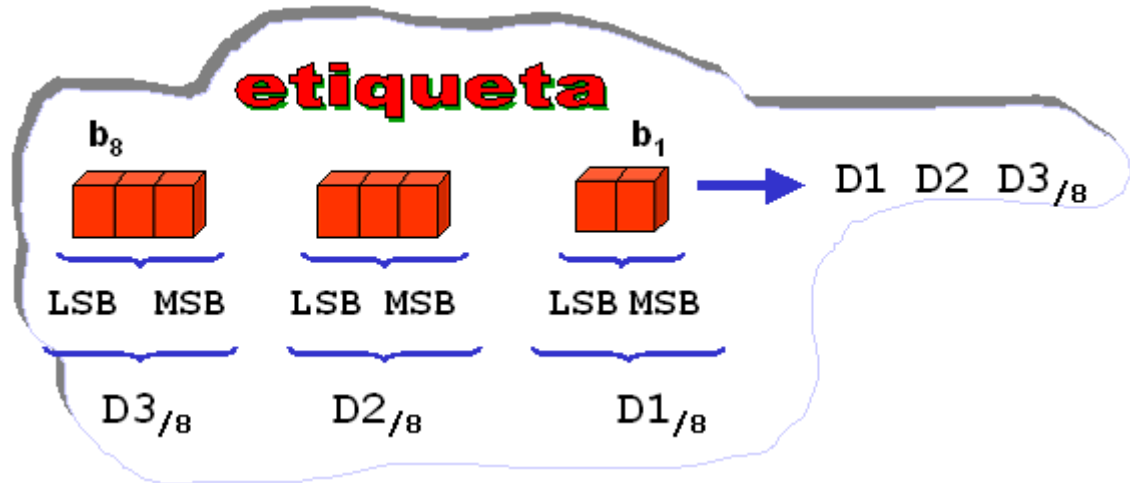
SDI: Source/Destination Identifier (b_9, b_{10}).

Odd parity bit (b_{32}).

(Parity y Label son los dos únicos campos obligatorios).

--LABEL

- Los primeros 8 bits que se transmiten de cada palabra, componen la etiqueta. La etiqueta identifica la clase de información contenida en la palabra (*DISTANCIA DME*, *ALTITUD*, *KNOTS*, *COURSE*, *MHz*, *BNR*, *BCD...*). También identifica la aplicación de la palabra en datos discretos, de mantenimiento y de métodos de transferencia de archivos.



- La etiqueta se convierte a octal. El LSB de cada grupo de tres bits para cada dígito octal, comienza por la izquierda. La etiqueta se lee en octal de derecha a izquierda.

Ej.

El dígito menos significativo se obtiene del grupo de tres bits de la izquierda $110_{/2}$ ($3_{/8}$). El siguiente dígito será $011_{/2}$ ($6_{/8}$). El MSD será $01_{/2}$ que equivale a $2_{/8}$.

b8						b1		
1	1	0	0	1	1	0	1	
1	2	4	1	2	4	1	2	

El valor de esta etiqueta es por lo tanto

$263_{/8}$

- El valor máximo por el que puede comenzar una etiqueta es el 3, el valor más alto de la etiqueta será el 377. Los receptores pueden ser programados para aceptar hasta las 255 etiquetas.

Ejemplos de etiquetas:

CODE Nº / ₈	EQPT ID / ₁₆	TRANSM ORDER. BIT POST								PARAMETER	DATA	
		1	2	3	4	5	6	7	8		BNR	BCD
0 0 0	X X	0	0	0	0	0	0	0	0	NOT USED		
0 0 1	0 2	0	0	0	0	0	0	0	1	DISTANCE TO GO		X
1 7 7	0 3	0	1	1	1	1	1	1	1	ECONOMICAL FLIGHT LEVEL	X	
	2 9	0	1	1	1	1	1	1	1	OIL QUANTITY (APU)	X	
2 0 0	0 4	1	0	0	0	0	0	0	0	DRIFT ANGLE		X
2 0 1	0 9	1	0	0	0	0	0	0	1	DME DISTANCE		X
	0 9	1	0	0	0	0	0	1	0	DME DISTANCE	X	
2 0 2	2 9	1	0	0	0	0	0	1	0	CABIN COMPARTMENT TEMP (GROUP #1)	X	
3 7 4	0 5	1	1	1	1	1	1	0	0	E-W VELOCITY-MAGNETIC	X	
	0 5	1	1	1	1	1	1	0	1	ALONG HEADING ACCELERATION	X	
3 7 5	3 3	1	1	1	1	1	1	0	1	SPARE DC1	X	
	0 5	1	1	1	1	1	1	1	0	CROSS HEADING ACCELERATION	X	
3 7 6	3 3	1	1	1	1	1	1	1	0	SPARE DC2	X	
	0 2	1	1	1	1	1	1	1	1	EQUIPMENT IDENTIFICATION		X
3 7 7	3 0	1	1	1	1	1	1	1	1	EQUIPMENT IDENTIFICATION		X

LABEL CODES (ARINC 429 SPEC – ATTACHMENT I)

Cuando se transmiten bloques de datos o ficheros de datos (*WILLIAMSBURG/BUCKHORN PROTOCOL*), la etiqueta definida anteriormente se sustituye por otro tipo llamado Etiqueta de Dirección del Sistema [System Address Label (SAL)]. Esta forma de transmisión de información (en forma de ficheros o bloques) sustituye al tipo de datos AIM (Acknowledge, Iso Alphabet N°. 5 And Maintenance Data).

	0	1	2	3	4	5	6	7
00								
01								
02								
03								
04								
05								
06								
07								
10								
11								
12								
13								
14								
15								
16								
17								
20								
21								
22								
23								
24								
25								
26								
27								
30								
31								
32								
33								
34								
35								
36								
37								

ASIGNACIÓN DE ETIQUETAS.

En la tabla se presenta la distribución de las posibles etiquetas en función del tipo de dato.

BNR –*Binary*.

BCD –*Binary Coded Decimal*.

Maint Disc= *Maintenance Discretes*.

Maint Dat = M Dt = *Maintenance data*.

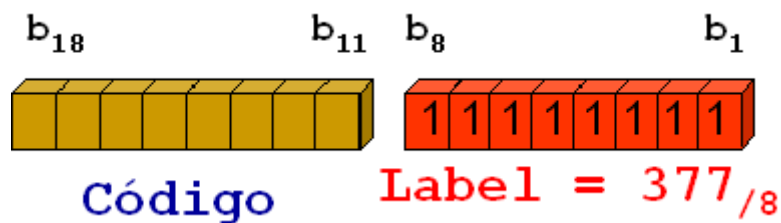
Ack = *Acknowledgement*
 M ISO = *maintenance ISO*
 ISO 5 = *ISO alphabet nº 5 (ASCII)*

AIM

EQ ID = Código de equipo.

--CÓDIGOS DE EQUIPO.

- ❑ Vienen dados por 8 bits, con 255 códigos diferentes (el 00H no se utiliza)
- ❑ Se utilizan con la etiqueta 377, enviada con anterioridad a la palabra que requiere la identificación del equipo.
- ❑ Cuando diferentes equipos pueden mandar la misma etiqueta, el computador receptor debe saber que parámetro recibe. En este caso, el computador transmisor, envía primero una palabra que contiene la etiqueta 377, y en el campo de datos (bits 11 al 18), envía el código de equipo. Todas las palabras que se reciban a partir de este momento, con un código de etiqueta común para dos equipos, serán asociadas por el receptor, al equipo cuyo código ha recibido previamente.



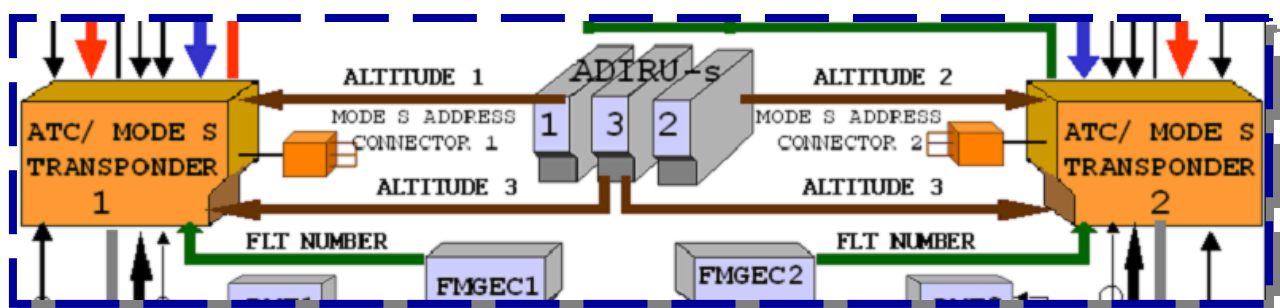
Equipment ID (HEX)	Type	Equipment ID (HEX)	Type
01	Flight Control Computer	40	Cockpit Printer
02	Flight Management Computer.	47	Digital Flight Data Recorder
1F	Fuel Quantity Totalizer	4A	Landing Gear Position Interface Unit
27	Microwave Landing System	4B	Main Electrical System Controler
2A	Thrust Management Computer	5C	Cabin Pressure Unit
35	TCAS	6B	Battery Charger Limiter

Ejemplos de identificación de equipos.

--SDI (*Source-Destination Indicator*).

- ❑ Codificado en los bits 9 y 10. Se usa, cuando haya que identificar la fuente de información o cuando la información se dirige a una ubicación específica.
- ❑ En instalaciones multi-sistema, es decir, múltiples receptores en un bus o interconexión de LRU-s por diferentes buses, cuando palabras específicas necesitan ser enviadas a un sistema concreto, se va a indicar dicho sistema mediante el SDI correspondiente. Por otro lado, también será necesario transmitir el SDI cuando el sistema origen de una instalación multi-sistema necesite ser reconocido por el destinatario.

Para el caso de la siguiente figura, cada ADIRU tendría su propio SDI.



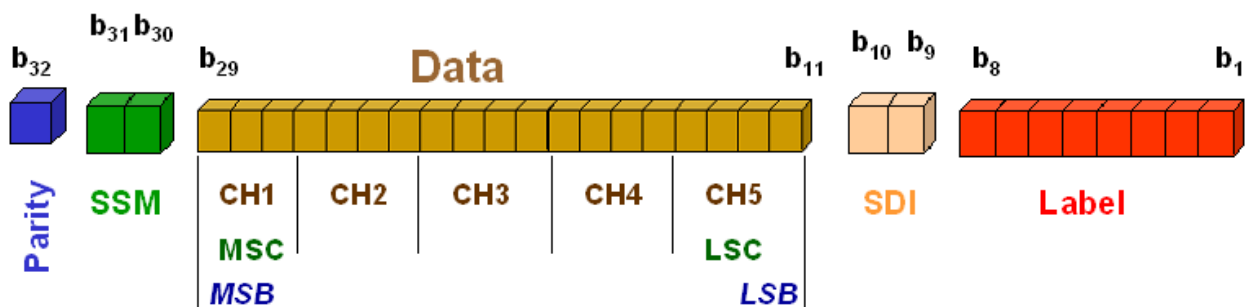
- El código SDI, puede ser establecido mediante la función de *pin programs*.
- El SDI no es obligatorio. Hay casos en los que esta función no se utiliza (transmisión de datos alfanuméricos ISO nº5). En estos casos, estos bits pueden ser utilizados para componer datos discretos. En otros casos, la necesidad de resolución de los datos numéricos (BNR/BCD) requiere el uso de estos bits para dar datos válidos.
- Un receptor conectado a un bus, podrá reconocer palabras conteniendo el código de su propia instalación (01, 10, 11) o "00". Este último caso (00) puede indicar varias cosas: a) que no se usa la función SDI, b) que la información se dirige a todas las instalaciones (*all-call*) del bus, es decir, todos los sistemas conectados al bus deben decodificar la información recibida, y c) asignado a una cuarta LRU.

--DATA.

- Los datos que tiene asignados la etiqueta, vienen dados mediante 19 bits, los que van desde el b11 hasta el BIT b29. Sin embargo, como el campo SDI es opcional, quedan 21 bits disponibles para el uso de datos. Algunas compañías que desarrollan este tipo de configuraciones, utilizan solamente los campos de etiqueta y paridad, usando los 23 bits restantes para los datos.
- No todos los bits van a ser siempre significativos. En los casos en los cuales no se utilizan todos los bits porque no se necesitan para la resolución establecida, dichos bits son puestos a cero (*pad bits*). Los bits o caracteres más significativos, comienzan a contarse de izquierda a derecha, por lo tanto, los *pad bits* serán situados entre el valor del dato dado y el SDI.

BCD.

- El campo de data se divide en 5 caracteres. El carácter 1 (CH1), o carácter más significativo (MSC), se codifica con 3 *bits* (b29, b28 y b27). Los demás CH2, CH3, CH4, y CH5, carácter menos significativo (LSC), se codifican con 4 bits.

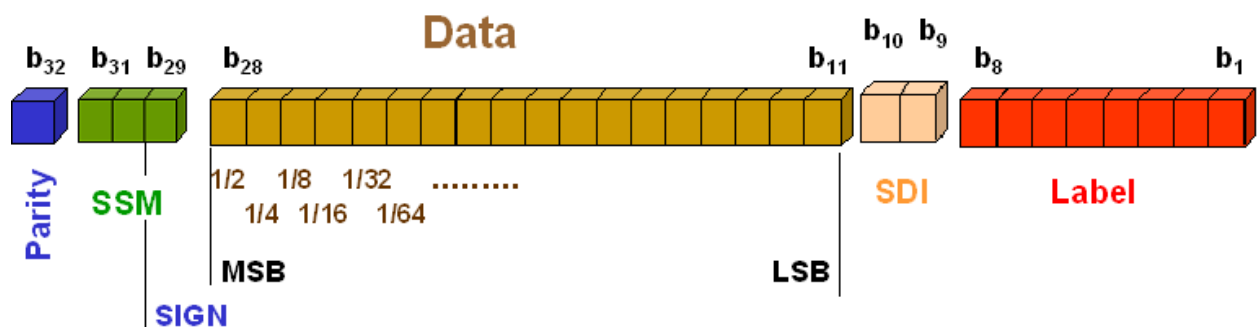


LABEL	EQUIP ID (HEX)	PARAMETER	UNITS	RANGE	SIGN DIGITS	RESOL.
010	02	PRESENT POSITION LATITUDE	DEG-MIN	180N-180S	6	0.1
	04	PRESENT POSITION LATITUDE	DEG-MIN	180N-180S	6	0.1
	38	PRESENT POSITION LATITUDE	DEG-MIN	180N-180S	6	0.1
066	02	LONGITUDINAL CG	% MAC	0-100.00	5	0.01
067		LATERAL CG	% MAC	0-100.00	5	0.01
125	31	GREENWICH MEAN TIME (GMT)	HRS, MIN	0-23.59.9	5	0.1
165	07	RADIO HEIGHT	FEET	± 7999.9	5	0.1
170	25	DECISION HEIGHT SELECTED	FEET	0-2500	4	1.0
200	04	DRIFT ANGLE	DEG	± 180	4	0.1
201	09	DME DISTANCE	NM	-1 - +399.99	5	0.01

EJEMPLO DE BCD DATA VALUES

BNR.

- Los datos son expresados en complemento a dos fraccional. El BIT de signo es el b29. En el campo de la magnitud, el peso del MSB (b28), es 1/2 del valor máximo indicado en el rango, para el parámetro representado. Los bits que le siguen, representan 1/4, 1/8, 1/16, etc., del rango. Para determinar el valor del dato, simplemente sumamos los valores de los pesos cuyo BIT valga 1. Hay que tener en cuenta que cuando el valor es negativo, la magnitud se debe complementar a dos y aplicar el criterio anterior. También podemos calcular el valor del dato aplicando los pesos binarios de cada BIT que valga 1, multiplicados por la resolución y realizando la suma.



Por ejemplo, si tenemos una palabra cuya etiqueta representa el número 366, indica que el parámetro a que se refiere la información contenida en dicha palabra es la velocidad N-S (ver tabla mas abajo). El rango para este parámetro es 0 a 4096 knots, y el número de bits significativos es 15 (sin incluir el de signo). Si en el campo de datos están a 1 los bits b26 y b22, el BIT de signo está a 0 y los demás bits también a 0, quiere decir que el valor que se está dando es:

$$\begin{aligned}
 & \text{b28} \quad \text{b27} \quad \text{b26} \quad \text{b25} \quad \text{b24} \quad \text{b23} \quad \text{b22} \quad \text{b21....etc.} \\
 & 4096 * 0 + 4096 * 0 + 4096 * 1/8 + 4096 * 0 + 4096 * 0 + 4096 * 0 + 4096 * 1/128 + 4096 * 0 + \dots \\
 & = \\
 & = 512 + 32 = 544 \text{ knots.}
 \end{aligned}$$

LABEL	EQUIP ID (HEX)	PARAMETER	UNITS	RANGE	SIGN BITS	RESOL APROX
064	3C	TIRE PRESSURE (NOSE)	PSIA	1024	10	1.0
365	04	VERTICAL ACCELERATION	FT/MIN	32768	15	1.0
	05	VERTICAL ACCELERATION	FT/MIN	32768	15	1.0
366	04	N-S VELOCITY	KNOTS	4096	15	0.125
367	04	E-W VELOCITY	KNOTS	4096	15	0.125
370	25	DECISION HEIGHT SELECTED (EFI)	FEET	8192	16	0.125
	05	DECISION HEIGHT SELECTED (EFI)	FEET	16.384	17	0.125
372	02	WIND DIRECTION – MAGNETIC	DEG/180	± 180	9	0.35
373	05	N-S VELOCITY – MAGNETIC	KNOTS	4096	15	0.125
374	05	E-W VELOCITY – MAGNETIC	KNOTS	4096	15	0.125

EJEMPLO DE BNR DATA VALUES

--SSM (Sign Status Matrix).

- ❑ b30, b31-.Tiene diferente sentido según los datos vengan en BCD o BNR. Para datos en BNR, el b29 también forma parte del SSM.
- ❑ SSM no es obligatorio. Puede ser utilizado cuando los datos numéricos necesiten más resolución que la dada de forma general.
- ❑ El uso del SSM tiene dos fines:
 - a.- Proporcionar el signo o indicación de dirección para los datos contenidos en la palabra.
 - b.- Proporcionar información sobre el estado de los equipos de donde provienen los datos.

BNR.

- ❑ El b29 representa el BIT de signo en los datos dados en complemento a dos fraccional. También, según el estado del mismo representa la siguiente información:

b₂₉	ESTADO	INFORMACIÓN
	0	Plus, north, east, right, to o above
	1	Minus, south, west, left, from o below

- ❑ Los bits b30 y b31, representan información de estado según la tabla siguiente:

-FW (*Failure Warning*): Indica que los monitores del sistema han detectado un fallo en el transmisor y/o en el equipo de procesamiento de señal.

-NCD (*No Computed Data*): Indica que la fuente es incapaz de computar datos fiables por razones que no sean de fallo del sistema. (banderas a la vista)

-FT (*Functional Test*): Indica que la fuente está en modo test, generando información predeterminada.

-NO (*Normal Operation*).

b_{31}	b_{30}	STATUS
0	0	FW
0	1	NCD
1	0	FT
1	1	NO

BCD.

El b_{29} es el MSB del MSC (carácter más significativo).

Cuando los datos no son válidos no se indica de ninguna manera, simplemente se retiran del bus.

b_{31}	b_{30}	INFO
0	0	Plus, north, east, right, to o above
0	1	NCD
1	0	FT
1	1	Minus, south, west, left, from o below

--PARITY.

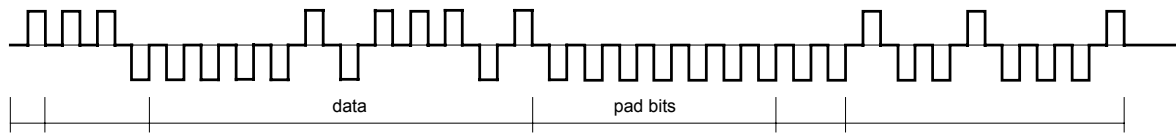
- El BIT de paridad es utilizado como comprobador de error en la transmisión. Se utiliza el tipo de *paridad impar* (*Odd Parity*).



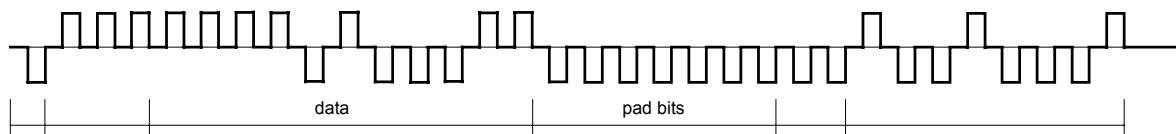
Ejemplos de datos numéricos:

BNR.

Label	parameter	bits sign.	Unit	range	resolution
211	Total Air Temperature	11	°C	±512	0,25°C

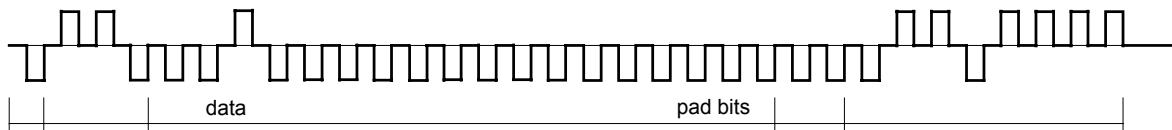


+23,25°C



-23,25 °C

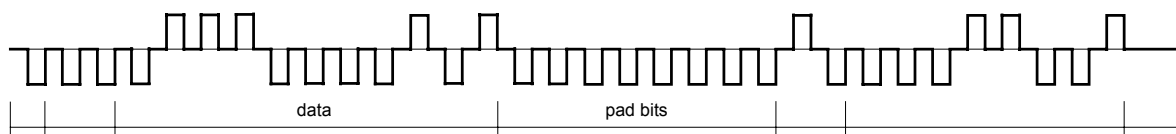
Label	parameter	bits sign.	Unit	range	resolution
366	N-S velocity	15	Knots	4096	0,125



512 Knots

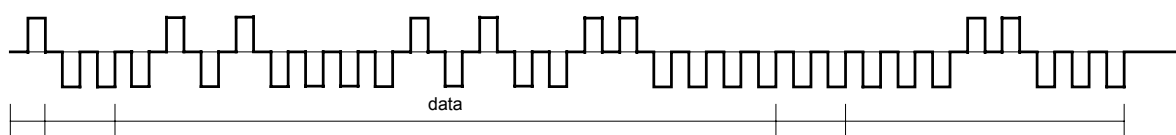
BCD.

Label	parameter	digits sign.	Unit	range	resolution
230	True airspeed	3	Knots	599	1.0



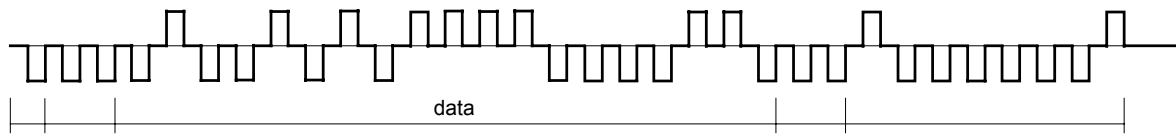
385 Knots

Label	parameter	digits sign.	Unit	range	resolution
030	VHF Com Frequency	5	MHz	79,999	0,001



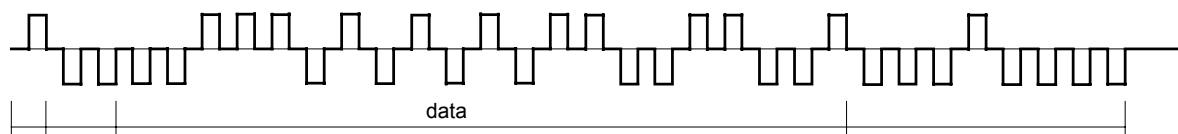
28,530 MHz

Label	parameter	digits	sign.	Unit	range	resolution
201	DME Distance	5		NM	-1 399,99	0,01



257,86 NM

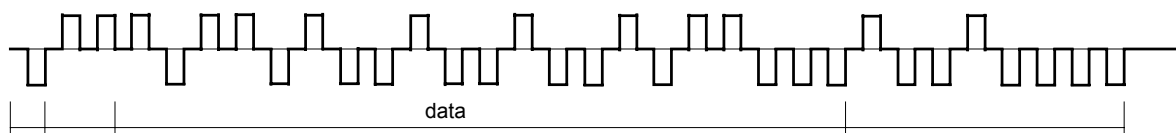
Label	parameter	digits	sign.	Unit	range	resolution
010	Present Position-Latitude	6		Deg-min	180N-180S	0,1



N 75° 59,9'

Observar que en esta palabra no se hace uso de SDI. En este caso, la configuración de los dígitos, también esta modificada respecto a la típica. El b29, hace de dígito más significativo, los otros cuatro dígitos, están codificados con cuatro bits cada uno.

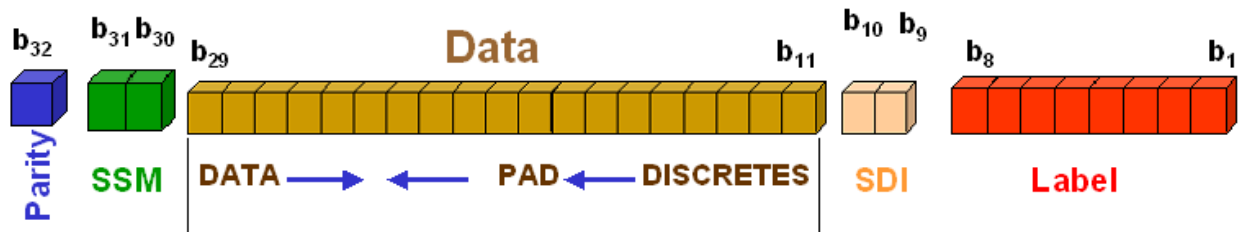
Label	parameter	digits	sign.	Unit	range	resolution
011	Present Position-Longitude	6		Deg-min	180E-180W	0,1



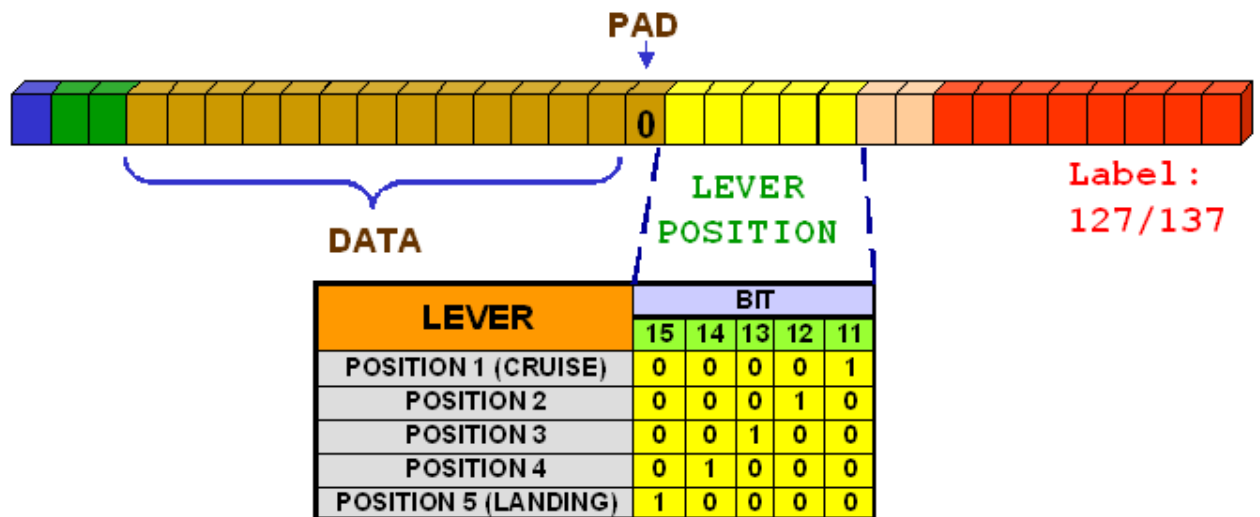
W 169° 25,8'

DISCRETOS.

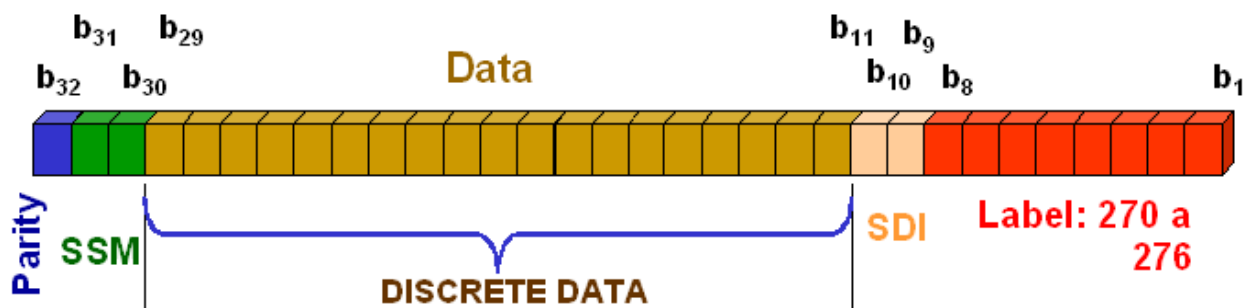
- Los datos formados por discretos, son utilizados de dos maneras:
 - a.- Como parte de una palabra de datos BCD o BNR, en los bits no utilizados. La regla a seguir para asignar los bits libres a discretos es empezar por el LSB (b_{11}) y continuar hacia el MSB. A los bits restantes (pad bits) se les asigna el valor 0.



Ej.



- b.- Usando las etiquetas 270 a 276. En este caso, el campo de datos desde el BIT b_{11} hasta el b_{29} es usado para datos formados por discretos.

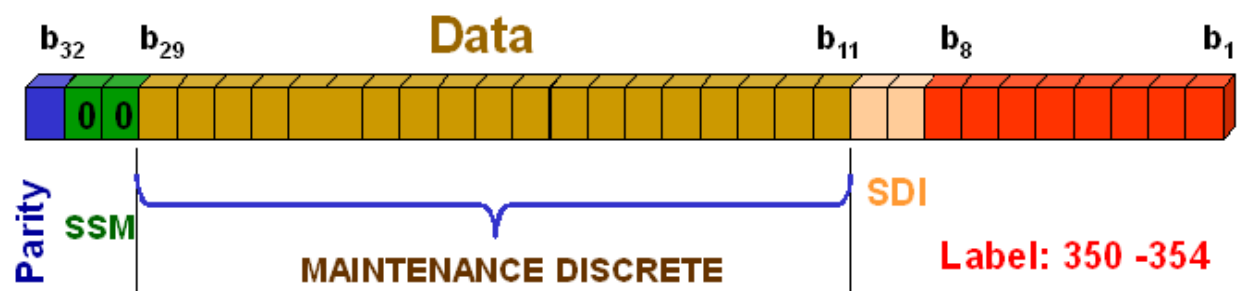


Nº bit	Function	Bit status		Nº bit	Function	Bit status	
		0	1			0	1
1			x	17	Left side angle of attack sensor heat	off	on
2		x		18	Right side angle of attack sensor heat	off	on
3			x	19	VMO/MMO Overspeed Warning	not Warn	W
4			x	20			
5			x	21			
6		x		22			
7		x		23			
8		x		24			
9				25			
10				26			
11	Icing Detector	off	on	27	Angle of attack alternate correction	No	yes
12	Pitot Probe Heat	off	on	28	Baro-correction port A	No	yes
13	ADS Computer Status	good	fail	29	Zero mach SSEC	No	yes
14	Pitot/Static Probre Heat	off	on	30			
15	Static Source Heat	off	on	31			
16	TAT Probe Heat	off	on	32	Odd parity		

Ejemplo de los discretos contenidos en la palabra con la etiqueta 270

MAINTENANCE DATA. (General Purpose).

- Este tipo de información puede estar formado por datos BNR, BCD y Discretos, pero nunca puede contener datos AIM. Las etiquetas que se usan son desde la 350 hasta la 354.



Formato para una palabra de Maintenance (para el caso de discreto)

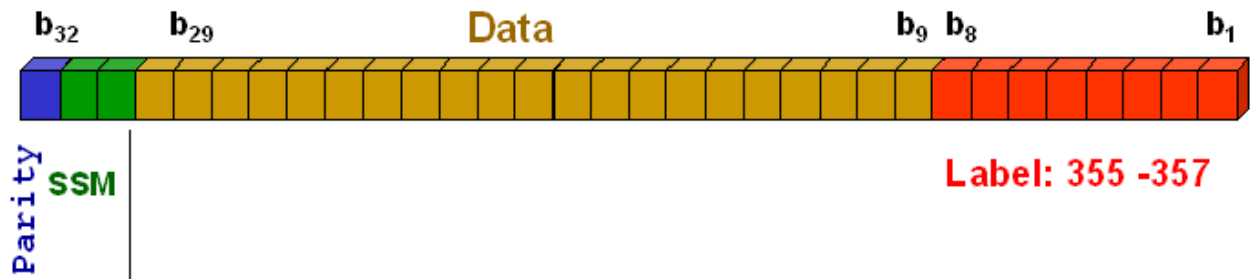
AIM –Acknowledgement, ISO Alphabet nº5 and Maintenance Data.

- La forma de transmisión de la información es en modo transferencia en bloque de las palabras, en contraposición a las formas vistas hasta ahora en las que la información ARINC 429 es transmitida como parte integral de una palabra etiquetada.
- El código dado por la etiqueta determina el tipo de datos:

Etiqueta 355: Acknowledgement Data.

Etiqueta 356: ISO Alphabet nº 5-Maintenance Data.

Etiqueta 357: ISO Alphabet nº 5- Messege.



b ₃₁	b ₃₀	significado
0	0	Intermediate word
0	1	Initial word
1	0	Last word
1	1	Check word

- El código presente en SSM determina el tipo de palabra.
- La primera palabra de cada grupo debe contener en los bits destinados a la SSM el código “01”. También debe contener en los bits b9 hasta b16, la representación en binario del número de palabras que componen el grupo, exceptuando cuando esta palabra es la única a transmitir, donde estos bits se ponen todos a cero.
- Cuando la etiqueta de la palabra es asignada a *Acknowledgement data*, los bits comprendidos desde b17 hasta el b29 ambos incluidos, de la palabra inicial, pueden ser utilizados para transmitir información.

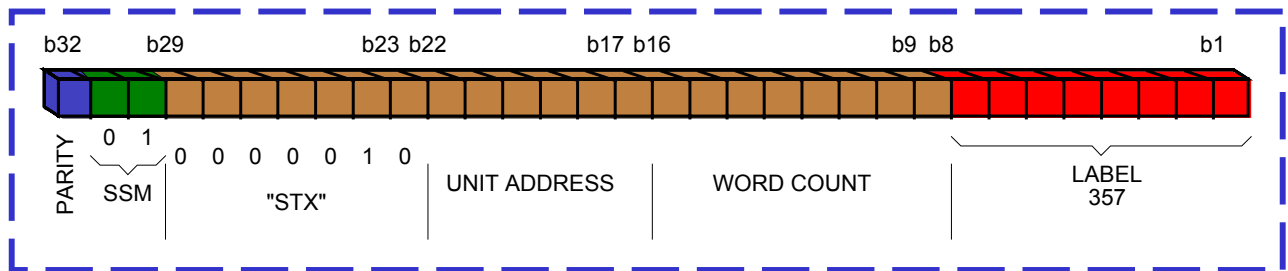
- Si la etiqueta es asignada a *Maintenance data* los bits desde el b17 hasta el b22 deben ser todos ceros.

- Si la etiqueta es para mensajes *ISO Alphabet nº5 (ASCII)* los bits desde el b17 a b22 son usados para direccionar una unidad según la tabla que se muestra.

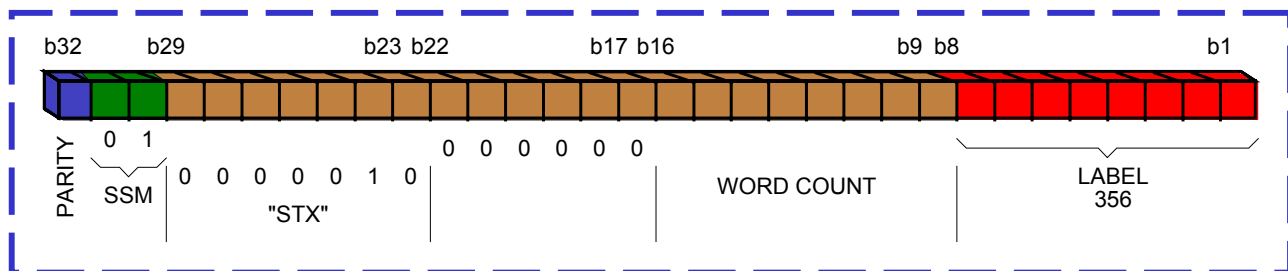
BIT (en ASCII)						Función
22	21	20	19	18	17	
0	0	0	0	0	0	ALL CALL, ALL GROUPS
		X	X	X	X	GROUP 0, UNIT 1-15
0	1	0	0	0	0	GROUP 1, ALL CALL
		X	X	X	X	GROUP 1, UNIT 1-15
1	0	0	0	0	0	GROUP 2, ALL CALL
		X	X	X	X	GROUP 2, UNIT 1-15
1	1	0	0	0	0	GROUP 3, ALL CALL
		X	X	X	X	GROUP 3, UNIT 1-15

- Para mensajes *ISO Alphabet nº 5* y *Maintenance data*, los bits desde el b23 hasta el b29, deben tener el código correspondiente al carácter de control “STX”.

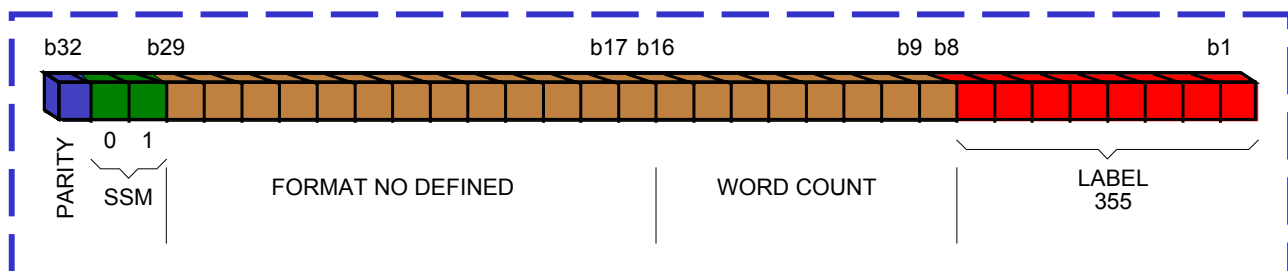
Palabra inicial para mensajes ISO Alphabet nº 5.



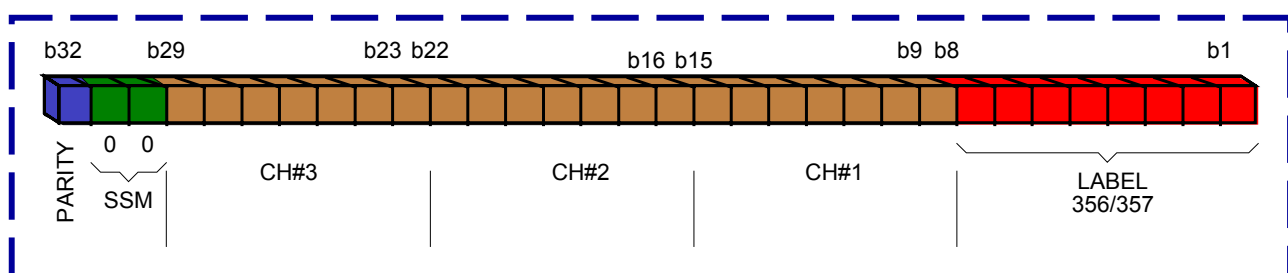
Palabra inicial para Maintenance.



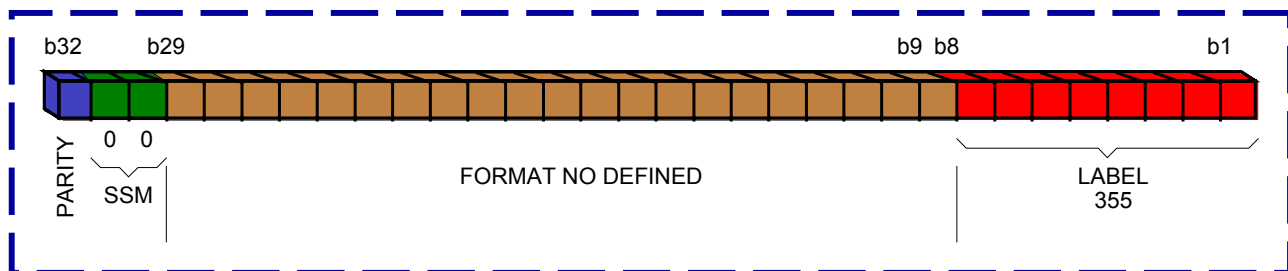
Palabra inicial para Acknowledgement



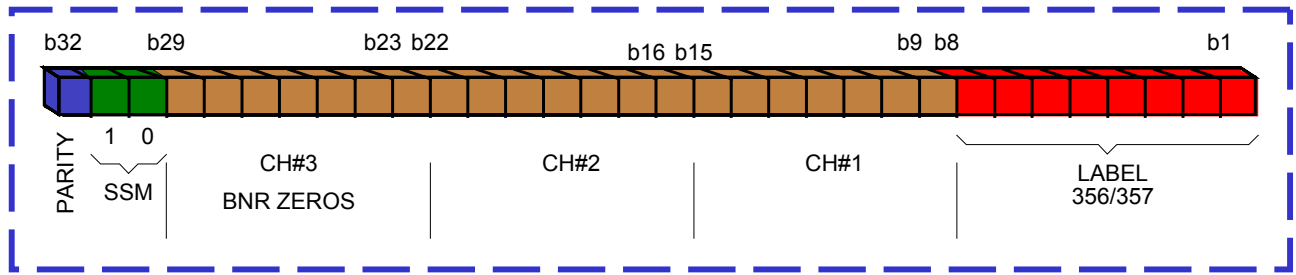
Palabra intermedia para Maintenance y Message.



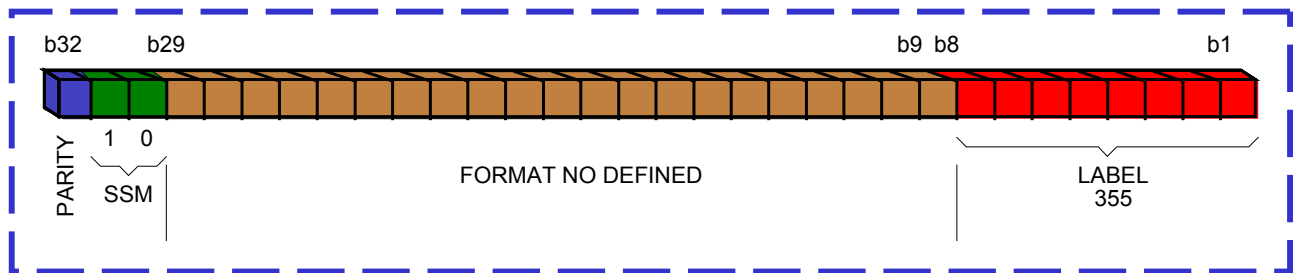
Palabra intermedia para Acknowledgement



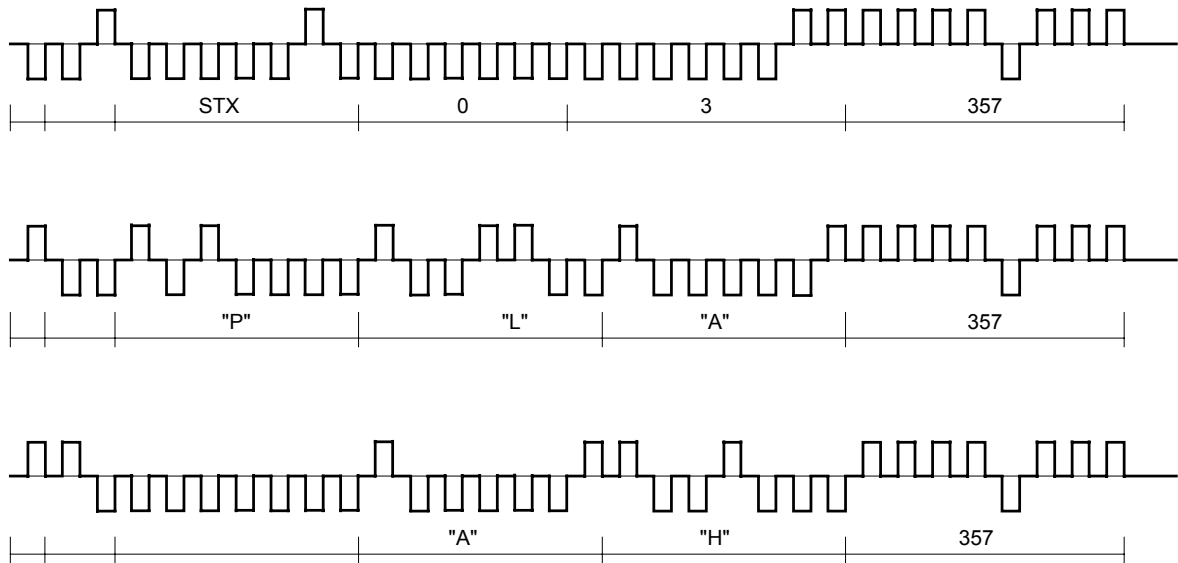
Palabra final para Maintenance y Message



Palabra final para Acknowledgement



Ej. Mensaje ISO Alphabet nº5.



DATA FILE

- ❑ Este grupo, describe el método para transferir ficheros de datos. Es utilizado cuando el terminal tiene capacidad para enviar y recibir datos.
- ❑ Las palabras que se transmiten en el fichero pueden ser cualesquiera de las pertenecientes a los 5 grupos vistos anteriormente (BNR, BCD, Discrete; Maintenance, AIM).
- ❑ Este método es utilizado cuando un sistema necesita dialogo con otro. Cuando un emisor está preparado para transmitir datos a un receptor, aquel envía una palabra "request to send" por un bus que conecta ambos elementos. Si el receptor está preparado, responde con una palabra "clear to send" por un bus diferente al anterior. Se envía una palabra de "data follows" e inmediatamente después, la información. Al final de la transmisión, se envía una *final word* para control de errores. Si después de recibir la *final word*, no existen errores (de paridad o tamaño de fichero), el receptor envía una palabra "data received OK" al emisor. Si se detecta algún error, el receptor pide al emisor que reenvíe el registro donde se detectó el error. En este punto, se establece un pequeño dialogo entre Tx y Rx para asegurar que el nuevo registro se puede volver a retransmitir sin que se produzca un nuevo error.

Debido a la cantidad grande de información que debe manejarse, la velocidad de trabajo del bus es la de 100 Kbaud.

- ❑ Cada fichero de datos puede contener desde 1 hasta 127 registros, y cada uno de estos registros puede contener desde 1 hasta 126 palabras de datos.

La primera palabra de un registro será la *Initial Word*, seguida por la *Intermediate Word/s* y terminará el registro con la *Final Word*. La forma de identificar estas palabras es la misma que la utilizada por el grupo de datos AIM, es decir, mediante la SSM (b_{30} y b_{31}).

- ❑ Existen 8 tipos de Initial Word:

tipo	ORIGEN	DESTINO	comentario
<i>Request to send</i>	Tx o Rx	Rx o Tx	El campo de datos contendrá (en binario) el número de registros que van a ser enviados.
<i>Clear to send</i>	Rx	Tx	Indica que el receptor está listo para recibir datos. El campo de datos puede contener información como es la máxima longitud de los registros o el número de palabras de 32 bits que el Rx puede aceptar.
<i>Data follows</i>	Tx	Rx	El Rx informa a Tx de que la información que sigue son los datos.
<i>Data Received OK</i>	Rx	Tx	Los datos han sido recibidos satisfactoriamente. El Rx informa al Tx del número de palabras y secuencia de registros recibida.
<i>Data Received Not OK</i>	Rx	Tx	Ha habido un error en los datos recibidos. El Rx informa al Tx del número de palabra y secuencia de registros donde se ha producido dicho error.
<i>Synchronization Lost</i>	Rx	Tx	Información de que el Rx está fuera de sincronismo enviando ceros en los bits b_9 a b_{22} .
<i>Header Information</i>	Tx	Rx	Similar a request to send pero sin compromiso de envío.
<i>Poll</i>	Rx o Tx	Rx o Tx	Puede ser usado como forma de interrogación por el Tx o Rx. "Tx: no tengo info. para ti, ¿tienes tú para mí?"

12.2-ESPECIFICACIÓN ARINC 629. (AERONAUTICAL RADIO INC. 629)

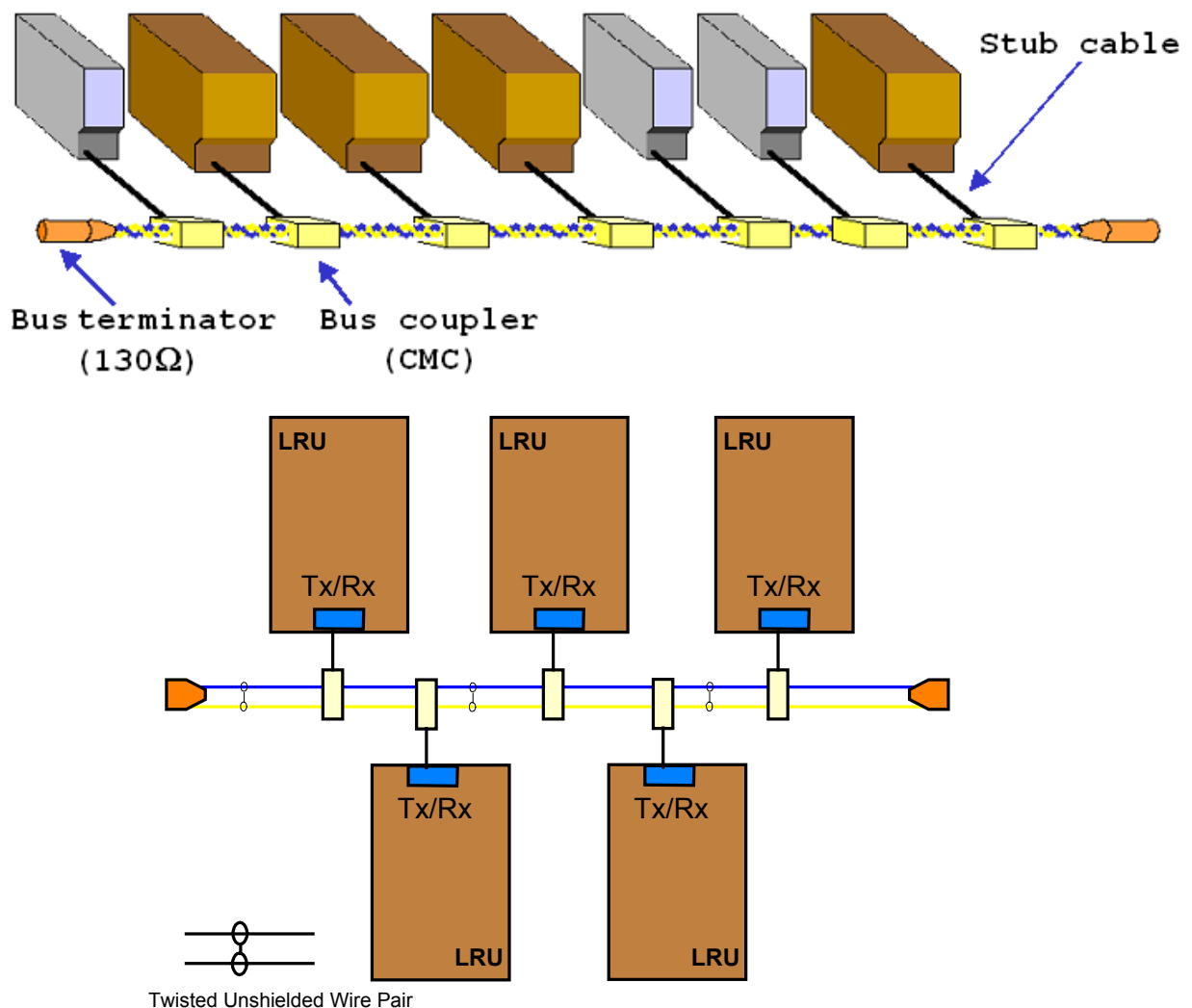
La especificación ARINC 629, define los estándares de la industria de aviación, para la transferencia de datos digitales entre sistemas de aeronaves usando un bus bidireccional y de acceso múltiple. Define el control, temporización, conversión de datos, características de las señales, tipo de etiquetas para los datos, formato de las cadenas de datos, etc.

A finales de los años 70 Boeing desarrolla DATAC (Digital Autonomous Terminal Access Communication). En 1989 ARINC lo estandariza como ARINC 629 BP (Basic Protocol). Este protocolo o MAC (Médium Access Control) permite la operación en el bus de modo periódico o de modo no periódico pero conmutando entre uno y otro modo.

Las características del estándar BP, fueron mejoradas mediante el CP (Combined Protocol), definido por Airbus Industrie. Los mensajes de tipo periódico y no periódico, pueden ser transmitidos de forma combinada.

12.2.1-CARACTERÍSTICAS GENERALES.

- El acceso al bus de datos se lleva a cabo mediante un control autónomo. El control autónomo del bus, permite la conexión al mismo bus, de numerosos terminales en paralelo, sin necesidad de usar un controlador, gestor de bus central o líneas adicionales de control.



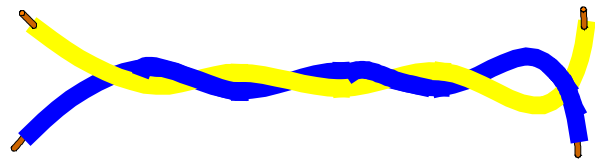
El control de acceso al bus, está distribuido entre los terminales a él conectados. Cada uno de estos terminales, independientemente determina la secuencia de transmisión en el bus. Esta secuencia está basada en que en un momento dado, sólo un terminal puede transmitir, después de la transmisión, el terminal no puede volver a transmitir hasta que los demás terminales hayan tenido una oportunidad para realizar una transmisión.

BUS:

-El soporte físico puede ser de fibra óptica o par trenzado TP (Twister Pair). En este último caso, cada extremo tiene un terminador con impedancia característica de **130Ω**.

-La longitud máxima del bus es de **100m**

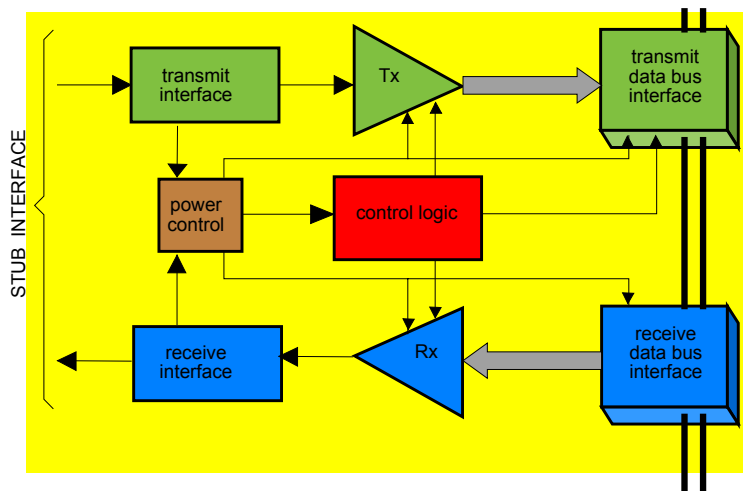
-No lleva ninguna conexión ni soldadura (acoplamiento electromagnético entre bus y terminal)



-El número máximo de terminales que pueden estar conectados al bus es de **120**. Cada uno de los terminales puede direccionar hasta **16** subsistemas.

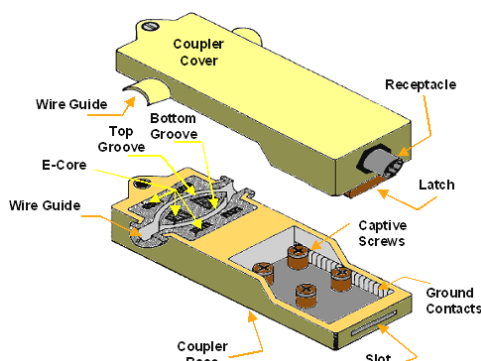
-La velocidad de trabajo del bus es de **2Mb/s**. Esto permite que 100000 palabras de 20 bits cada una, sean transmitidas en cada segundo.

COUPLER:

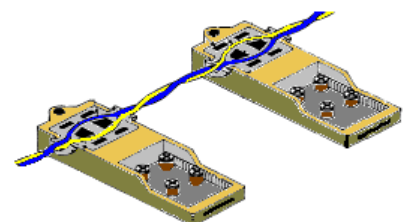


Es el elemento activo que proporciona el acoplamiento magnético entre el SIM (Serial Interface Module) y el bus. Con este sistema no se requiere el corte físico del cable del bus para la transferencia de información. Mediante un transformador, el *receive data bus interface*, detecta las variaciones de corriente en el bus ("dobletes" de corriente) y mediante un amplificador de transimpedancia, estas

variaciones se convierten en las variaciones de tensión ("dobletes" de tensión), que contiene la información. En el *transmit data bus interface*, se produce el proceso inverso.



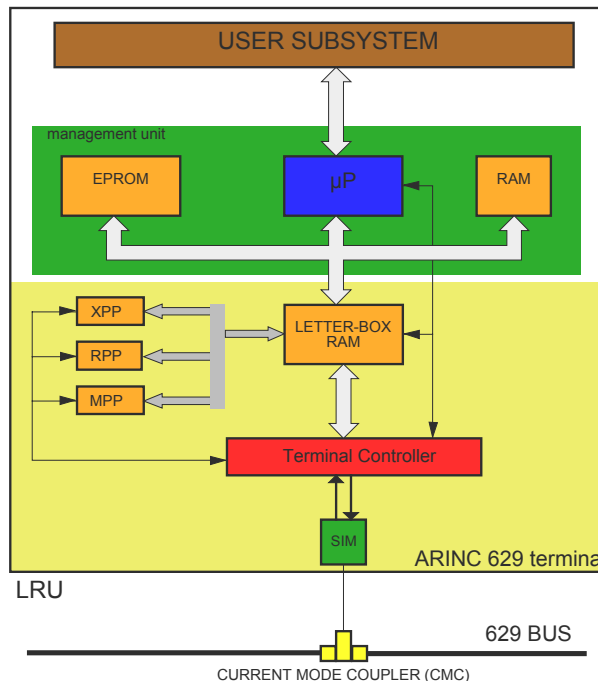
El coupler también recibe la información que él mismo transmite, de esta manera permite que el SIM analice esa información, pudiendo detectar posibles fallos, e informar al sistema BITE.



TERMINAL.

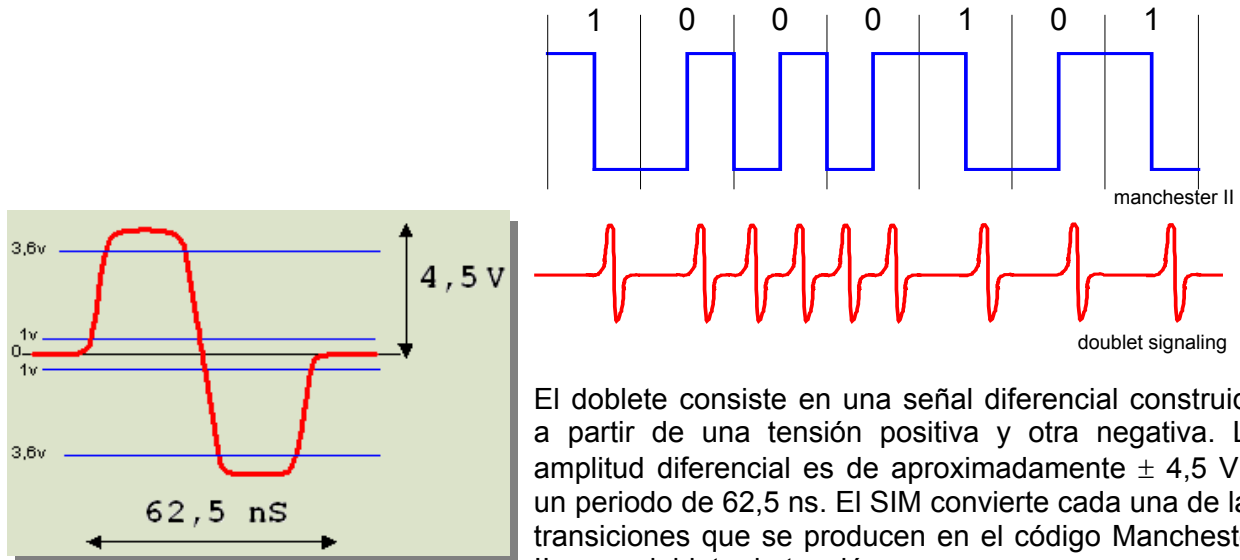
Es la unidad componente de la LRU que implementa el protocolo de acceso y el protocolo de datos. La información a transmitir se toma de la LRU correspondiente, es codificada y se envía a través del SIM y del stub cable al coupler, con la secuencia de tiempos adecuada. Desde el coupler, la información pasa al bus global. La información que se recibe desde el bus global, recorre el camino inverso para ser decodificada por el terminal y ser llevada a la memoria de la LRU correspondiente.

Cada terminal tiene tablas particulares que dan a ese terminal unas características únicas. Estas tablas contienen datos que determinan la frecuencia con la que la información es transmitida, que información se transmite o recibe y que hacer con la misma.



SIM. (Serial Interface Module).

Proporciona la interfaz entre las señales dadas por el Terminal Controller (Código Manchester II) y las que requiere el coupler (Doublets Encoding), en ambos sentidos. También controla el modo de funcionamiento del coupler y además, gestiona posibles fallos de la transmisión y el BITE. El SIM, monitoriza la transmisión de cada "doblete" hacia el coupler para asegurarse de que la amplitud y la simetría son las correctas. Si se detecta un fallo en la transmisión de un doblete, el SIM inhabilita la transmisión para asegurar que el bus de datos no va a tener información falsa. Los dobletes transmitidos, son también monitorizados en la sección receptora del SIM para comprobar la efectividad de la transmisión del coupler.



El doblete consiste en una señal diferencial construida a partir de una tensión positiva y otra negativa. La amplitud diferencial es de aproximadamente $\pm 4,5$ V y un periodo de 62,5 ns. El SIM convierte cada una de las transiciones que se producen en el código Manchester II en un doblete de tensión.

El SIM, determina el final de un mensaje en el bus del par trenzado cuando transcurren 4 periodos de reloj sin que reciba ningún doblete del coupler.

XPP, RPP, MPP.

Son memorias EEPROM-s. Contienen las tablas que dan personalidad única a cada terminal. Los datos a transmitir o recibir, su posición en memoria (direccionamiento), sus identificadores (etiquetas), valores de temporizadores, etc., son controlados por medio de XPP, RPP y MPP. Estas memorias son distintas para la parte receptora RPP (*Receive Personality Prom*) y la parte emisora XPP (*Transmit Personality Prom*), de esta forma se pueden monitorizar las transmisiones de un terminal mediante la parte receptora del mismo. La otra memoria es utilizada para la configuración del terminal y opciones de control, y se designa como *Multiple Personality Prom* (MPP).

XPP. Define la manera en la que los mensajes son creados, ordenados y en que tiempos se transmiten. Cada cadena de palabras que va a ser enviada, tiene una posición en la XPP que define la etiqueta, la posición de memoria donde se encuentra la primera palabra de datos, el número de palabras de la cadena, y otros datos de control. Antes de transmitir un mensaje, el *terminal controler* interroga a la XPP para obtener toda esta información.

RPP. Define la manera en la que los mensajes entrantes son manipulados. Cada una de las 4096 etiquetas tiene una representación en esta memoria. Cuando entra una cadena de palabras, la etiqueta es decodificada por el terminal controler y esta información es usada para direccionar una posición de la RPP. En la posición direccionada, está la información de si los datos entrantes deben ser procesados o ignorados. Si deben ser procesados, también indica la posición de memoria a partir de la cual deben guardarse esos datos, el número de palabras, etc.

Esta memoria también contiene la información de los valores de los timers para Synchronization Gap (SG), Transmit Interval (TI) y Terminal Gap (TG), necesarios para ejecutar el protocolo en el bus.

En un bus ARINC 629 pueden estar conectados varios terminales idénticos, cada uno de los cuales puede estar transmitiendo la misma etiqueta y en los mismos intervalos. Cada uno de estos terminales tendrá un código diferente que lo distinguirá de los otros. Este código es el *Channel Identification Number (CIN)*, el cual, se transmite en la palabra de etiqueta, dentro del campo extensión (Ext.). En el terminal receptor la información con igual etiqueta, será guardada en posiciones de memoria diferentes dependiendo del CIN. La posición en particular donde debe ser guardada se obtiene mediante operaciones con pointers y ciertos valores de offset. La memoria MPP guarda estos valores de offset.

LETTER BOX RAM.

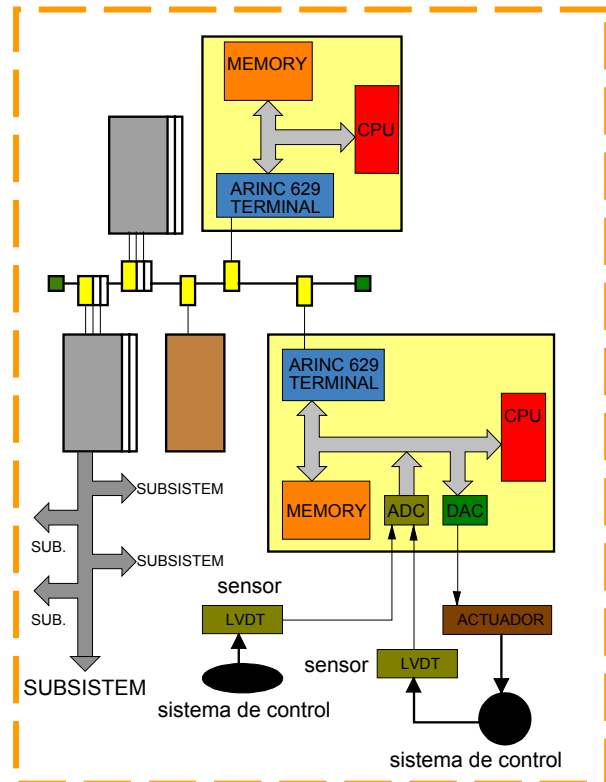
Es la interfaz entre el terminal y la 629 management unit. Consta de una memoria de doble puerto que es accedida por ambas partes para realizar la transferencia de la información entre el terminal y el usuario en ambos sentidos.

629 MANAGEMENT UNIT.

- A un terminal ARINC 629 pueden estar conectados múltiples subsistemas. La management unit gestiona, por un lado, la parte correspondiente a los subsistemas conectados a un terminal, y por otro, mantiene el diálogo con el terminal. Básicamente, está formada por un sistema microcomputador.

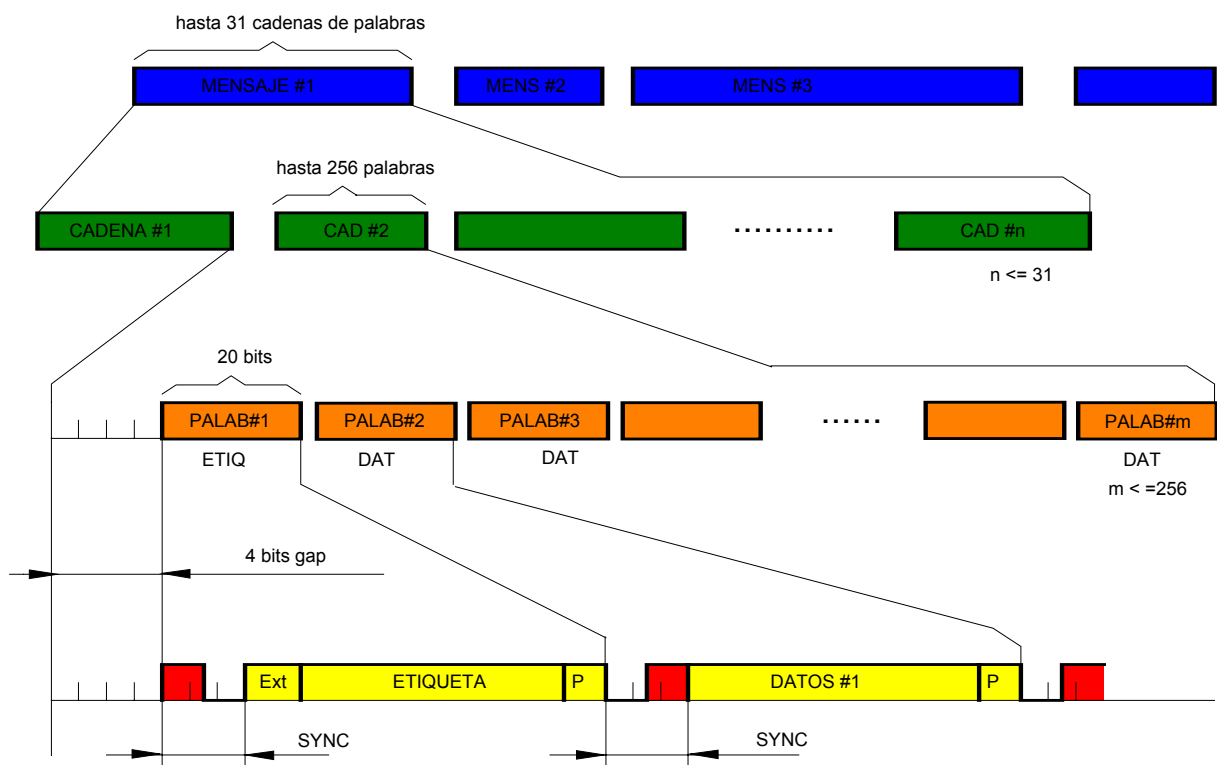
- En un sistema ARINC 629 podemos diferenciar dos lados: el lado local y el lado global. La parte local está formada por el lado del terminal que interactúa directamente con las memorias de los subsistemas en la LRU. En esta parte, la información es recibida y enviada a través de buses formados por líneas en paralelo. A este bus también pueden estar conectados latches para dar paso a los discretos de la aeronave y conversores ADC y DAC.

En el lado global, el terminal está conectado mediante los couplers a los demás terminales del bus.



12.2.2-OPERACIÓN

- ❑ El terminal ARINC 629 ensambla y transmite la información al bus global en forma de *mensajes*. Los diferentes terminales, toman el control del bus según el protocolo de acceso. Una vez que un terminal tiene el control del bus, envía la información.
- ❑ Cada terminal es capaz de transmitir *mensajes* de una longitud fija o variable. Estos mensajes están constituidos por cadenas de palabras de hasta un máximo de **31** cadenas. Cada una de las cadenas de palabras del mensaje, están separadas por un periodo de cuatro bits sin información



Cada cadena puede tener hasta **256 palabras**. La primera palabra de cada cadena es la *etiqueta*, la cual es única para cada función y sirve para identificar las *palabras* que la siguen. Estas palabras contendrán datos relacionados con la función definida por dicha etiqueta. La palabra de etiqueta está formada por **20 bits**. Los bits dedicados al código de la etiqueta son **12**, con lo que se pueden codificar **4096** etiquetas distintas. **3 bits** son de sincronismo, 1 BIT de paridad y los **4** restantes son para el campo de la extensión de la etiqueta (16 asignaciones de canal).

Cada palabra de datos, esta formada por **20 bits** repartidos en **16 bits** de datos, **3 bits** de sincronismo y **1 BIT** de paridad.

Los terminales vigilan todas las transmisiones en el bus (incluida la suya), y todos decodifican las etiquetas para determinar si una cadena de palabras contiene información que se requiere por la LRU correspondiente. Si los datos son requeridos y son válidos (basado en el formato y la paridad), el terminal guarda la información en la memoria de la LRU asociada, donde queda disponible para el subsistema correspondiente. Si la información no es necesaria, se descarta.

- ❑ Cualquier terminal puede transmitir datos hacia el bus o recibirlos desde este.
- ❑ Cada terminal tiene asignado un periodo específico de tiempo en el cual puede transmitir.
- ❑ El hecho de que un terminal esté transmitiendo por el bus, hace que se imposibilite a cualquier otro terminal para realizar una transmisión.
- ❑ La ausencia de cualquier transmisión en el bus (*quiet bus*), indica que el bus está libre y por lo tanto, un terminal puede comenzar a transmitir bajo ciertas condiciones.

PROTOCOLO DE ACCESO.

CSMA/CA (*Carrier Sense Multiple Access / Clash Avoidance*). El control del bus se realiza de forma autónoma basado en el método CSMA/CA. La aplicación de este método evita colisiones en el bus y resuelve quien puede transmitir. Se lleva a cabo mediante la detección y medida de periodos inactivos del bus (*Carrier Sense*) y el control para que solo un terminal a la vez intente transmitir en el bus. (*Clash Avoidance*).

CSMA/CA

- ❑ El CSMA/CA se lleva a cabo cuando se cumplen tres condiciones en cada terminal. Estas condiciones forman el protocolo y está basado en tres contadores, que representan intervalos de tiempo. Estos tres contadores son:

TRANSMIT INTERVAL (TI)

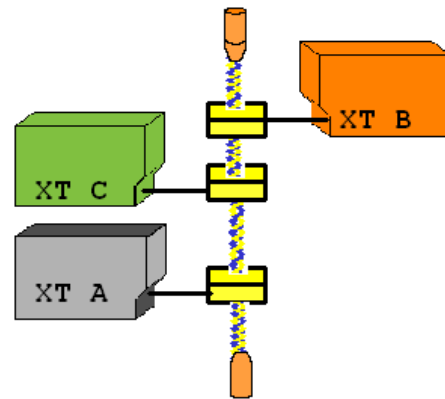
SYNCHRONIZATION GAP (SG)

TERMINAL GAP (TG).

- ❑ Todos los terminales conectados al bus global están programados con el mismo **TI** y el mismo **SG**.
- ❑ Cada uno de los terminales en el bus global esta programado con su propio **TG**.
- ❑ Un terminal podrá transmitir solamente después de que los tres intervalos de tiempo hayan finalizado.

Cada terminal debe tener en cuenta estos intervalos. ("no dialogo"; Control autónomo del bus)

Tomemos como ejemplo un bus con tres terminales para explicar la función de estos tres contadores:

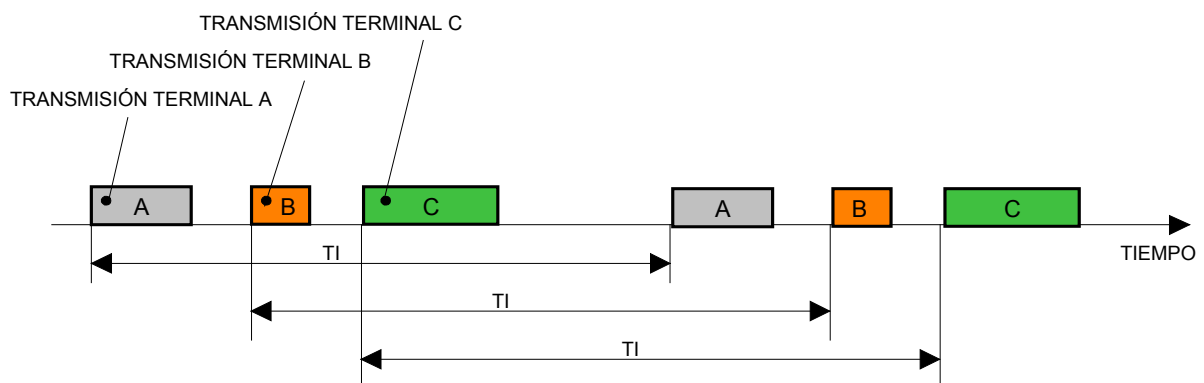


TI. (Transmit Interval)

- TI asegura que cada terminal transmita a partir de intervalos mínimos constantes de tiempo.

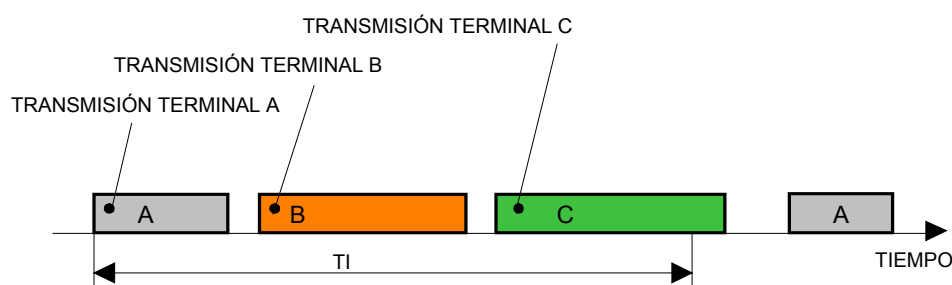
*El TI comienza el conteo en el mismo momento en que el terminal empieza a transmitir.

*Una vez que el terminal ha finalizado la transmisión, no podrá volver a transmitir hasta que como mínimo su contador TI no haya finalizado la cuenta.



El tiempo de conteo del TI va desde 0,5 ms hasta 64,00 ms. Este tiempo se programa teniendo en cuenta la frecuencia de refresco de los datos del sistema.

El bus ARINC 629 trabaja en dos modos diferentes; periódico y no periódico. En una primera aproximación se puede decir que un intervalo periódico ocurre cuando todos los terminales completan su transmisión dentro de un TI. Si el TI es excedido se produce un intervalo no periódico, e indica que uno o más terminales han transmitido un mensaje mayor de lo común.



SG. (Synchronization Gap)

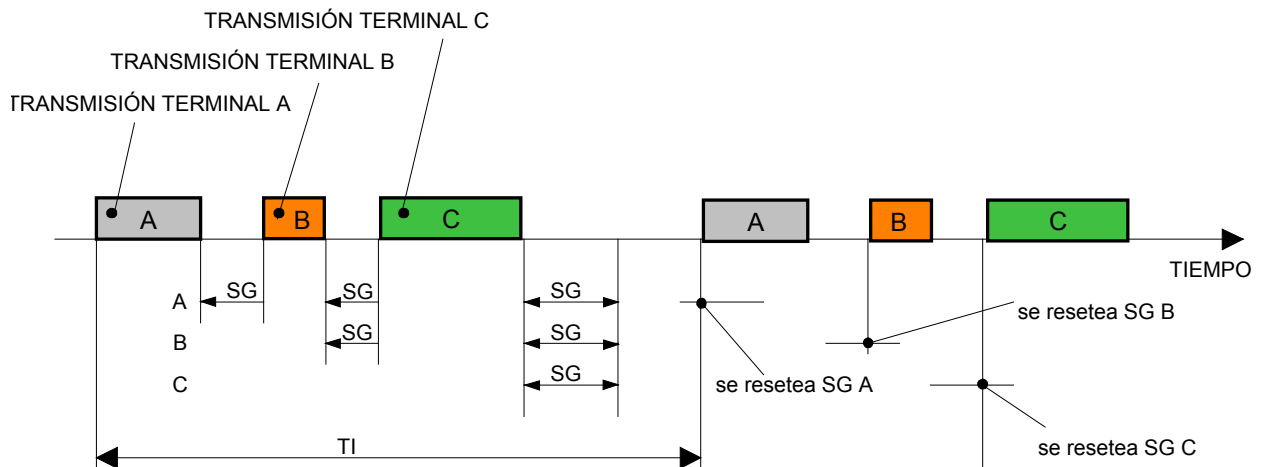
- SG, junto con TG asegura que todos los terminales del bus global, tengan la oportunidad de transmitir antes de que repita de nuevo un terminal.

*EL SG comienza su cuenta desde cero, tan pronto como el bus esté inactivo, es decir, ningún terminal transmitiendo.

*Si cualquier terminal comienza a transmitir antes de que SG haya finalizado su cuenta, éste se pone de nuevo a cero.

*El SG terminará su conteo sólo cuando cada terminal ha tenido la oportunidad de transmitir. Si un terminal no transmite durante un tiempo igual o superior al TG, el bus queda abierto de nuevo a todos los terminales que hayan agotado las tres temporizaciones.

*Una vez que SG haya terminado su cuenta, sólo puede ser puesto a cero cuando su propio terminal comience a transmitir.



El tiempo de conteo del SG va desde 17,69 μ s hasta 128,69 μ s. Este tiempo se programa teniendo en cuenta que debe ser mayor que el mayor de los tiempos del contador TG.

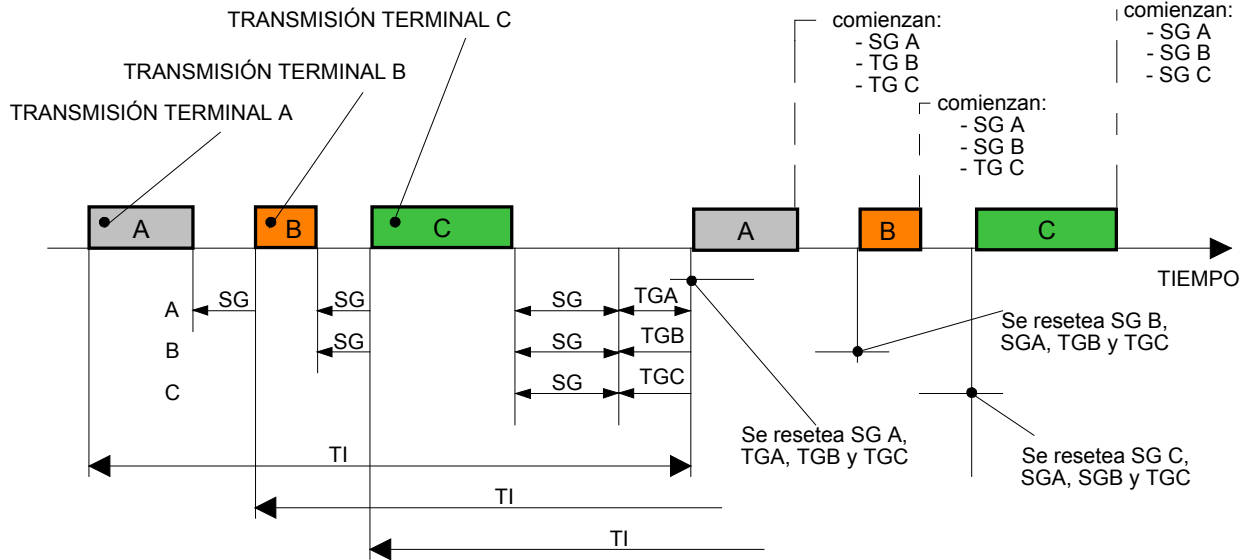
TG. (Terminal Gap)

- TG establece el tiempo mínimo de inactividad del bus entre transmisiones de los terminales, y el orden en el cual los terminales transmiten, es decir, marca una prioridad. Un terminal con una alta prioridad, tiene un TG muy corto y al contrario, cuanto menos prioridad se asigne a un terminal, más largo será su TG. Este tiempo es específico de cada terminal, y no puede haber dos de ellos con el mismo valor de TG.
- También se usa junto con SG para asegurar que todos los terminales del bus global, tengan la oportunidad de transmitir, antes de que repita de nuevo un terminal.

*TG comienza a contar después de que SG complete su cuenta y el bus sigue inactivo.

*Se pone a cero cada vez que un terminal transmite en el bus.

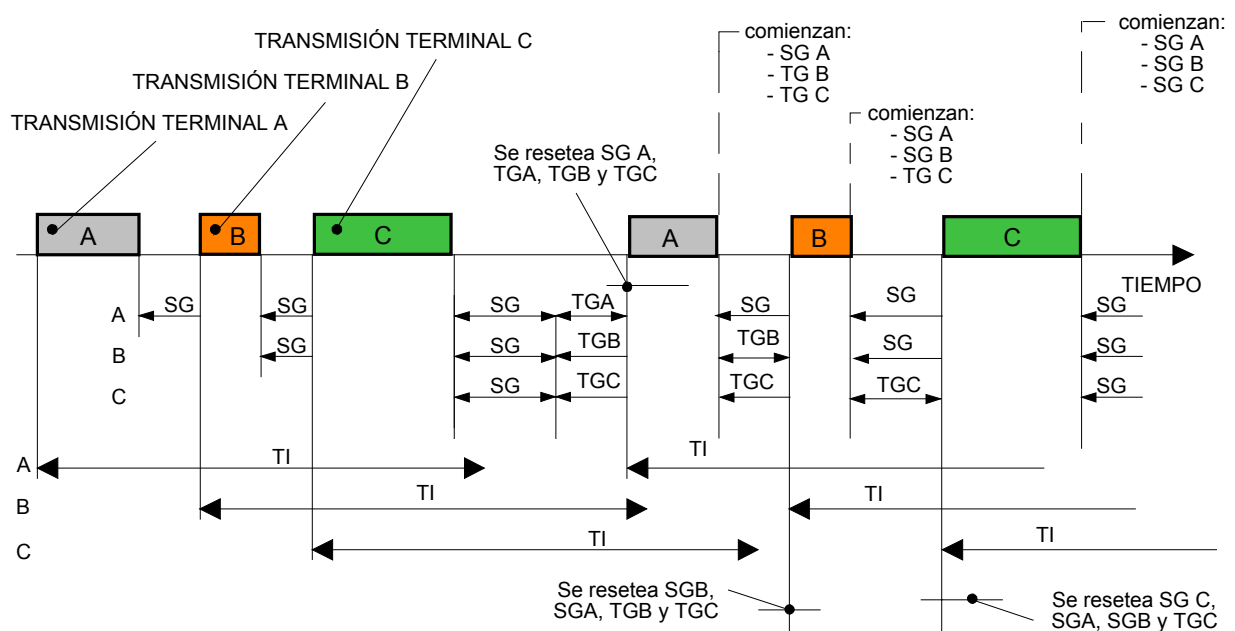
*TG puede contar a la vez que TI, pero debe contar de forma consecutiva a SG, es decir, TG y SG no pueden contar a la vez.



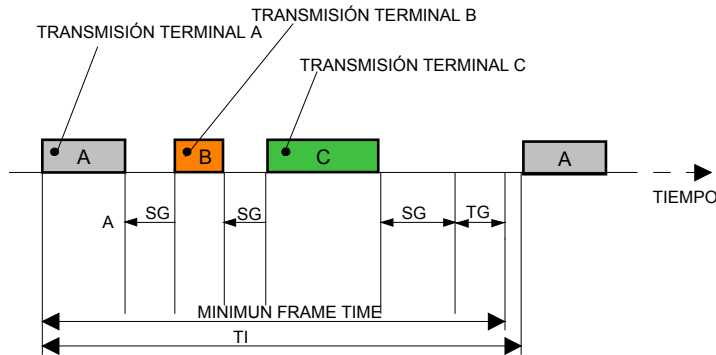
El tiempo de conteo del TG va desde $3,69 \mu s$ hasta $127,69 \mu s$. Este tiempo se programa teniendo en cuenta que debe ser menor que el intervalo de tiempo del contador SG.

Conclusión: después de que un terminal haya transmitido, puede volver a transmitir si se han cumplido las tres condiciones:

- 1ª Por lo menos ha transcurrido un periodo de TI.
- 2ª El bus ha estado inactivo por un periodo mayor que el tiempo del SG.
- 3ª El bus ha estado inactivo por un periodo igual que el periodo del TG propio del terminal, desde la última transmisión en el bus y desde que el periodo de SG se ha cumplido.

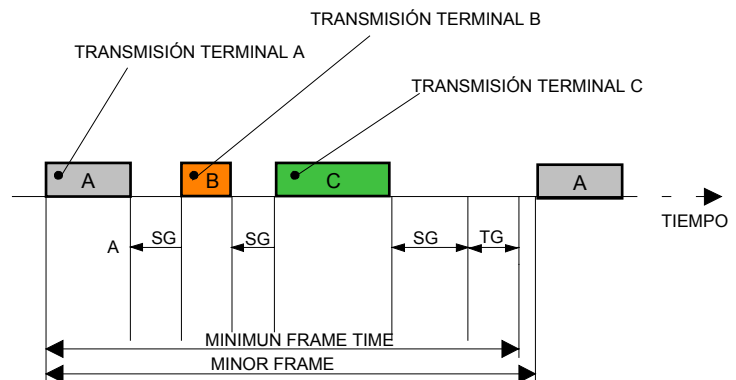


MODOS PERIÓDICO Y NO PERIÓDICO.



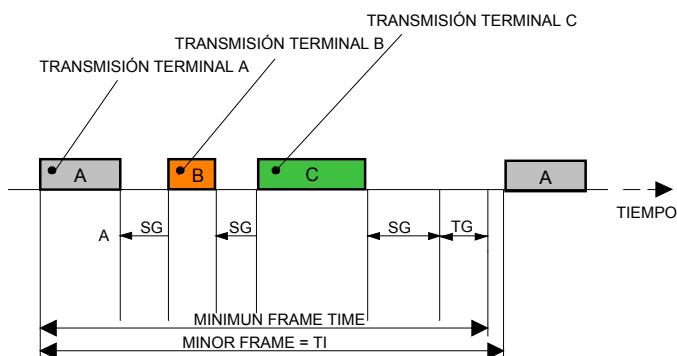
- El *tiempo mínimo de la trama*, se define como la suma del tiempo requerido para transmitir por todos los terminales en el bus global, más cada uno de sus tiempos TG, más un SG.

- La *trama menor* se define como el tiempo que transcurre desde que un terminal transmite, hasta el momento que comienza a transmitir de nuevo.
- La *trama mayor* se define como el tiempo en que un terminal repite una secuencia de cadenas de palabras.



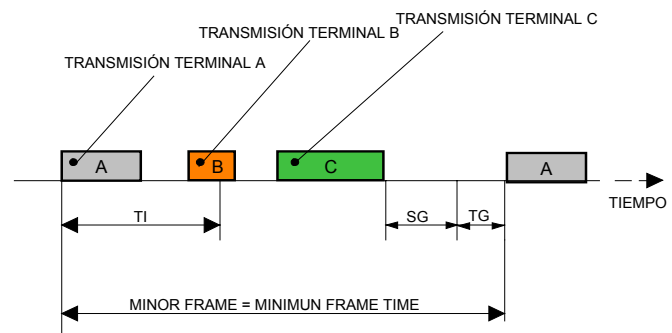
La información a transmitir puede ser periódica o no periódica. La información como la actitud procedente de un sensor inercial es transmitida de forma periódica. Información como datos de mantenimiento y datos de inicialización se transmiten de forma no periódica.

El modo periódico garantiza que todos los terminales en el bus van a transmitir en intervalos regulares de tiempo. El modo no periódico garantiza el acceso para todos los terminales en el bus, en un cierto orden.



Modo periódico.

Modo no periódico.



12.3-MIL-STD-1553

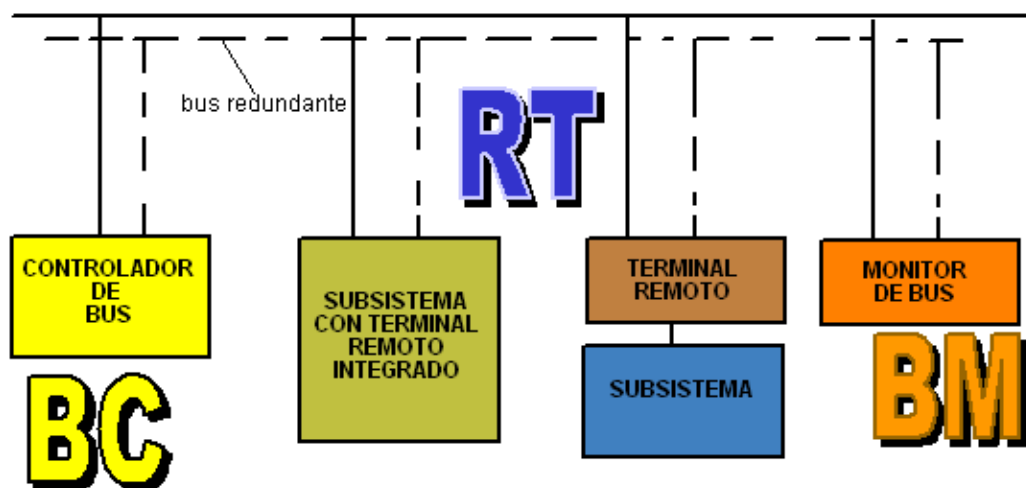
La designación completa es “MIL-STD-1553, Aircraft Internal Time División Command/Response Multiplex Databus”.

En 1968 la Society of Automotive Engineers (SAE) formó un subcomité de personal del gobierno e industria para desarrollar un bus de datos digital. En 1973 MIL-STD-1553 fue publicado como un estándar de la fuerza aérea. Este estándar fue revisado dos veces; MIL-STD-1553A que fue publicado en 1975 y MIL-STD-1553B publicado en 1978.

Este tipo de bus de datos es ampliamente utilizado en aeronaves militares y también en vehículos tales como, submarinos, barcos e incluso tanques.

12.3.1-CARACTERÍSTICAS GENERALES.

- El bus está formado por líneas de pares de cables trenzados y apantallados (Twisted Shielded Wire Pair) y una serie de cables (cabos) que unen estas líneas y los terminales. La norma MIL-STD-1553 especifica que todos los dispositivos en el sistema estén conectados a un par de buses redundante. En caso de fallo en uno de ellos, quedaría el otro bus como camino para la transferencia de información.



- Existen solamente tres tipos funcionales de terminales en el bus: el Controlador de Bus (**BC**), el Monitor de Bus (**BM**) y el Terminal Remoto (**RT**). Estos dispositivos pueden ser capaces de realizar más de una función.

-*Controlador de Bus.* El controlador de bus es el terminal que inicia y controla las transferencias de información en el bus de datos. Este control lo realiza mediante una serie de órdenes (guardadas en su memoria local) a los terminales remotos en puntos específicos de tiempo.

También en el caso de un evento no planeado como por ejemplo el fallo de un terminal remoto, el controlador de bus tiene la capacidad de detectar ese error y realizar la acción necesaria para recuperar el sistema de esa condición de error.

El bus puede tener varios controladores, pero sólo uno de ellos puede estar activo en un momento dado.

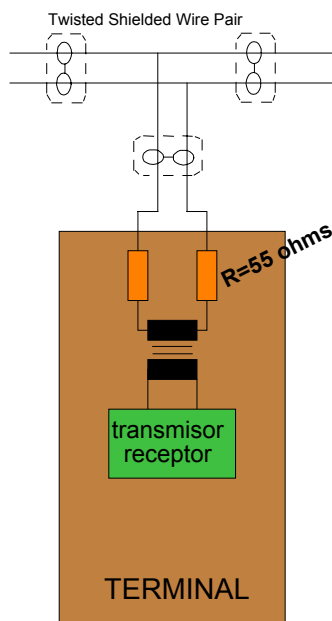
-*Terminal Remoto.* Es el sistema de control que formatea los datos que le llegan desde el bus para que puedan ser utilizados por el sistema aviónico a él asociado o al revés, los datos que le llegan procedentes de un sistema aviónico o simplemente de un sensor, los formatea y transmite por el bus de datos, para uso por otros terminales. Tienen, además, capacidad para detectar errores en la transmisión desde el controlador de bus.

Puede estar localizado en un subsistema aviónico, o como unidad independiente. Todo el movimiento de datos desde y hacia el subsistema asociado, lo realiza bajo la dirección del controlador de bus. Pueden tener suficiente capacidad de proceso como para realizar la función de controlador de bus, en caso de que el controlador primario de bus falle.

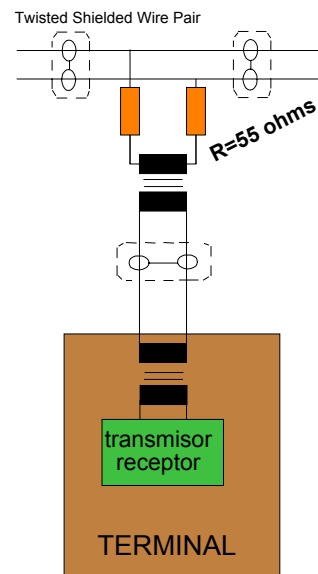
Cualquier terminal que no está operando en el modo de controlador de bus o monitor de bus, está en modo de terminal remoto. Según el estándar puede haber hasta 32 terminales remotos conectados al bus de datos.

-*Monitor de Bus.* Este terminal es opcional y no puede transmitir mensajes al bus de datos. Tiene la misión de “escuchar” todos los mensajes que circulan por el bus para extraer determinada información y guardarla en memoria para poder utilizarla en momentos posteriores, por ejemplo en mantenimiento. También puede servir como fuente de información para el controlador de bus de respaldo.

- Existen dos maneras de conectar un terminal al bus de datos: acoplamiento directo y acoplamiento por transformador. Las siguientes figuras representan las dos formas de conexión.



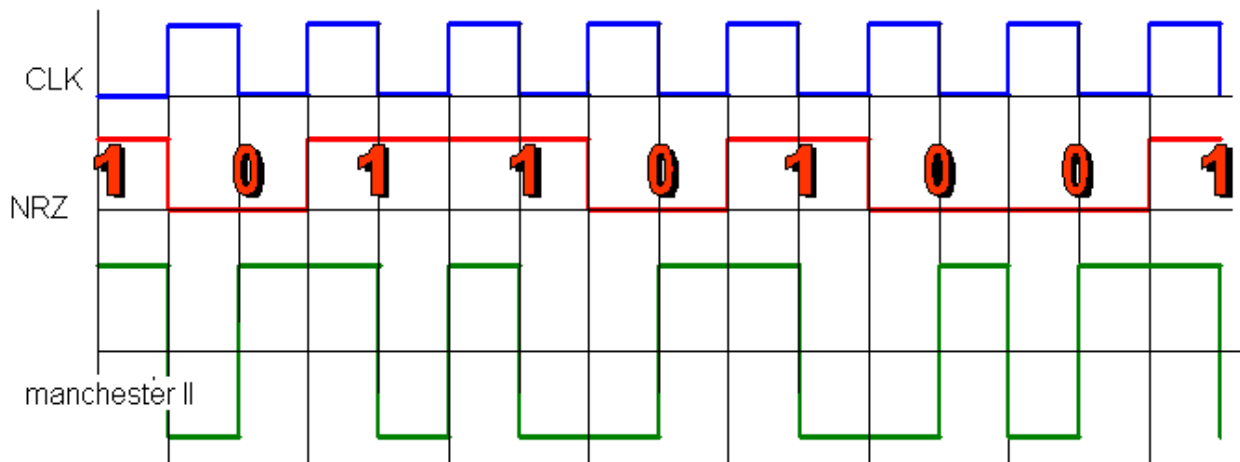
Acoplamiento directo.



Acoplamiento por transformador

El acoplamiento directo se utiliza para longitudes de cabo inferiores a los 33 cm y el acoplamiento por transformador hasta aproximadamente 6 m.

CODIFICACIÓN DE LOS DATOS



La información está codificada en código Manchester II.

En la figura se compara la información dada en código NRZ y código Manchester II, en función de una señal de reloj (un ciclo de reloj corresponde a un periodo de BIT).

VELOCIDAD DE TRANSMISIÓN. La velocidad de transmisión es de 2 Mb/s.

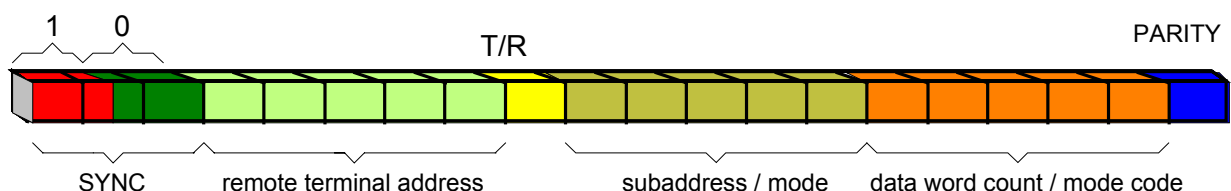
FORMATO DE LA INFORMACIÓN.

- Los mensajes son transmitidos por el bus en forma de palabras. Una palabra está formada por 20 bits. De ellos, 3 periodos de BIT son utilizados para la forma de onda de sincronismo, 16 bits componen el campo de la información, y 1 BIT para la paridad. Los terminales del bus son los encargados de añadir el sincronismo y el BIT de paridad a la palabra de 16 bits procesada. También son los encargados de analizar el sincronismo y el BIT de paridad en la recepción.

El sincronismo se transmite primero, después los 16 bits de información y por último se transmite el BIT de paridad. Este último BIT debe ser tal que se cumpla que la paridad sea siempre **impar**, sin tener en cuenta los bits de sincronismo.

- Existen tres tipos de palabras: palabra de orden (*Command Word*), palabra de estado (*Status Word*) y palabra de datos (*Data Word*).

COMMAND WORD



La palabra de orden (*Command Word*) puede ser solamente transmitida por el controlador del bus. Esta palabra contiene la dirección (*remote terminal address*), dada por 5 bits, del terminal remoto al cual implica en la transferencia, bien sea para que éste transmita o bien para que reciba. Cada terminal remoto en un sistema tiene asignada una única dirección.

- Cada palabra de comando que llega a los terminales remotos, es primeramente analizada por el campo de *dirección de terminal remoto (remote terminal address)*. Aquel terminal que coincida con la dirección dada en ese campo, será el que actuará según la palabra de comando recibida. Los demás terminales, simplemente ignoran la palabra.

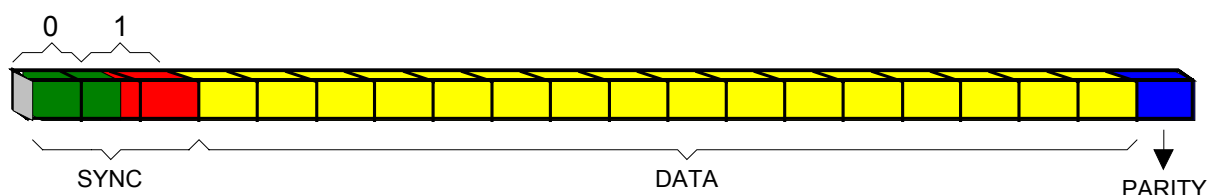
Existe un tipo de palabra de orden que va dirigida a todos los terminales simultáneamente (broadcast messages). Para este caso, la dirección que se da en el campo de terminal remoto en la palabra de orden es 11111 (31).

- *T/R*. Este BIT indica la acción que debe realizar el terminal remoto direccionado. Si el BIT *T/R* vale 1, el terminal remoto transmitirá datos en respuesta a la orden. Si el BIT vale 0, el terminal actuará como un receptor de datos.
- Otro campo de la palabra de orden es el dedicado a la dirección específica del subsistema conectado al terminal direccionado (*subaddress /mode*). Se pueden direccionar hasta 30 subsistemas por cada terminal.

Si en el campo de subaddress/mode la información contenida son todo ceros 00000 (0), o todo unos 11111 (31), indica que en el campo siguiente de data word count/mode code no se envía un número que indica la cantidad de palabras a mover, sino que se envía un código de modo que sirve para configuración o gobierno del bus de datos o control eléctrico de los terminales remotos, como por ejemplo, un reset de un terminal remoto (01000), inicialización de un self-test (00011), desconexión de un terminal redundante (10100)...etc.

- El siguiente campo (*data word count /mode code*) indica el número de palabras de datos (hasta 32 palabras, codificado el 32 con 00000) que van a ser movidas con la presente orden. También, según lo explicado mas arriba, puede indicar que orden de modo está siendo transmitida.

DATA WORD

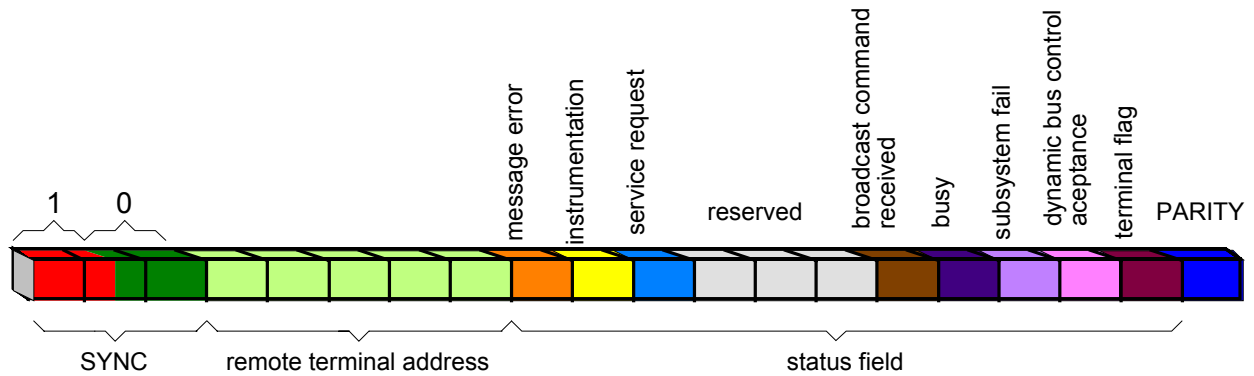


- Las palabras de datos (Data Words) pueden ser transmitidas tanto por el controlador del bus como por los terminales remotos en respuesta a una petición del controlador. El número máximo de palabras de datos que se permiten por cada palabra de orden (Command Word) es de 32.
- Este tipo de palabras, aunque son las menos estructuradas de las que utiliza el estándar, contienen toda la información real (en 16 bits) que va a ser transferida entre los terminales remotos y el controlador de bus, o entre dos terminales remotos.

STATUS WORD

- La palabra de estado (Status Word) es transmitida por el terminal remoto en respuesta a los mensajes de orden dados por el controlador del bus.

- Al igual que la palabra de orden, esta palabra comienza con los bits de sincronismo y la dirección del terminal remoto, en este caso, que envía la palabra de estado. Existen dos motivos para que el terminal que transmite la palabra de estado, envíe su dirección: primero, permite al controlador del bus verificar que el terminal que transmite la palabra de estado es el correcto, y por lo tanto el último comando enviado por el BC fue interpretado. Segundo, evita que cualquier otro terminal remoto, confunda la respuesta de estado con un comando (ya que el patrón de sincronismo es el mismo para ambas palabras). Si un terminal remoto intenta por error decodificar la palabra de estado de otro terminal tratándola como una palabra de orden, encontrará que la dirección no le corresponde.



- Inmediatamente después de la dirección del terminal remoto, está el BIT que indica error en el mensaje (*message error*). Este BIT es puesto a uno lógico, indicando error, cuando se da una de las siguientes situaciones:

Una palabra de datos recibida del controlador de bus, contiene error.

Se detecta un hueco en el tiempo entre palabras de datos.

Un comando recibido por el terminal remoto no está implementado en este.

El terminal remoto recibe un número de palabras de datos erróneo.

Si se produce un error en un mensaje, no se transmite la palabra de estado. El controlador de bus, solicita entonces mediante un comando que se transmita la última palabra de estado para analizar la falta de respuesta anterior.

- El siguiente BIT es el BIT de instrumentación (*instrumentation bit*). Es un BIT opcional que sirve para ayudar a diferenciar palabras de comando y palabras de estado. Debe ser siempre cero.
- El BIT de petición de servicio (*service request BIT*) indica que el terminal remoto requiere servicio. Cuando este BIT está puesto a uno lógico, hace que el BC realice una transferencia predefinida. Este BIT también es opcional, si no es utilizado debe ser puesto a 0.
- Los tres siguientes bits están reservados y deben ser puestos siempre a 0.
- Cuando un terminal remoto recibe una orden válida dirigida a todos los terminales (broadcast), el BIT denominado *broadcast command received BIT* es puesto a 1. Si no se implementa, este BIT debe ser puesto a 0.
- El *busy BIT* indica si está puesto a 1, que el terminal remoto no es capaz de transferir datos desde o hacia el subsistema, en respuesta a una orden. Si no se utiliza, debe ser puesto a 0.

- ❑ El *subsystem fail BIT* es usado por el terminal remoto para indicar (con un uno lógico) al controlador del bus que existe una condición de fallo en el subsistema y por lo tanto, los datos transmitidos pueden no ser válidos. Si no es utilizado este BIT debe ser siempre 0.
- ❑ El *dynamic bus control acceptance BIT* cuando está puesto a uno, indica al controlador del bus, que el terminal remoto ha aceptado la orden de control dinámico de bus e inmediatamente se hará cargo del bus como controlador de bus de respaldo. También este BIT es opcional y debe ser puesto a 0 si no es utilizado.
- ❑ El *terminal flag BIT* cuando está puesto a uno indica que existe un fallo en el propio terminal remoto. Si no es utilizado también como los anteriores, se pone a 0.

Hay que tener presente que el sincronismo aunque tiene una duración de tres bits, y solamente dos estados, no tiene un valor diferente para cada duración del periodo de BIT. Para el caso de command word y status word, el sincronismo comienza por un estado alto durante **1,5** periodos de BIT, seguido por un estado bajo durante los otros **1,5** periodos de BIT restantes.

Para el caso de la data word, los niveles son los contrarios a los anteriores, pero la duración es la misma.

12.3.2-OPERACIÓN BÁSICA.

- ❑ El protocolo MIL-STD-1553 establece diez diferentes tipos de mensajes divididos en dos grupos: formato de transferencia de datos (data transfer format) y formato de información al bus (broadcast information format), que cumplen las siguientes cuatro premisas:

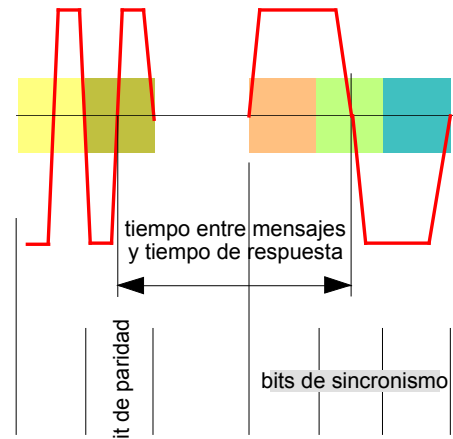
1-Todos los mensajes comienzan con una palabra de orden procedente del controlador del bus.

2-Debe existir un hueco mínimo en el tiempo entre los mensajes (4µs).

3-Si un terminal remoto debe contestar una orden, la respuesta debe comenzar con una palabra de estado.

4-No pueden existir huecos entre las palabras de datos.

Aparte del tiempo mínimo entre mensajes (no hay tiempo máximo especificado entre mensajes), existe otro tiempo a tener en cuenta, que es el tiempo de respuesta del terminal remoto. Este tiempo viene dado como el valor dentro del cual, un terminal remoto debe responder. Esta ventana temporal tiene un valor comprendido entre los 4 y 12 microsegundos y se mide como muestra la figura.



TIPOS DE MENSAJES.

Formato de transferencia de datos:

- Controlador de bus a terminal remoto (recepción). *BC to RT (receive)*.
- Controlador de bus a terminal remoto (transmisión). *BC to RT (transmit)*
- Terminal remoto a terminal remoto. *RT to RT*
- Modo de orden (sin datos). *Mode command (no data)*
- Modo orden (con transmisión de datos). *Mode command (with Tx data)*
- Modo de orden (con recepción de datos). *Mode command (with Rx data)*

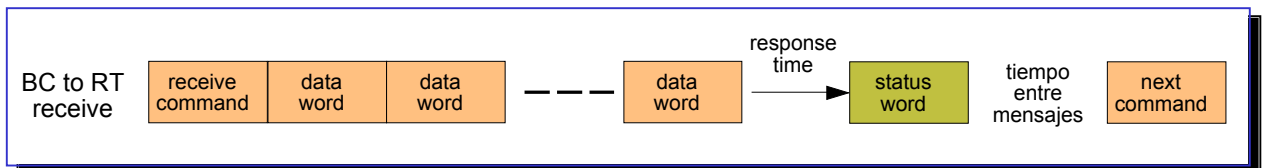
Formato de información al bus:

- Controlador de bus a terminal remoto. *BC to RT*
- Terminal remoto a terminal remoto. *RT to RT*
- Modo de orden (sin datos). *Mode command (no data)*
- Modo de orden (con datos). *Mode command (with data)*

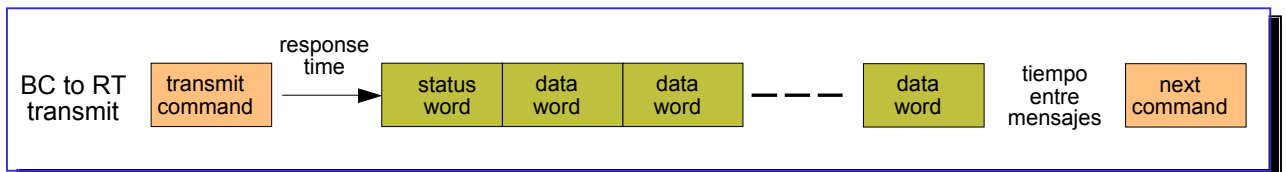
El proceso de transmisión para cada uno de los tipos de mensaje es el siguiente:

(en las figuras que siguen, los bloques más oscuros representan información transmitida por los RT, y los más claros a la transmitida por el BC)

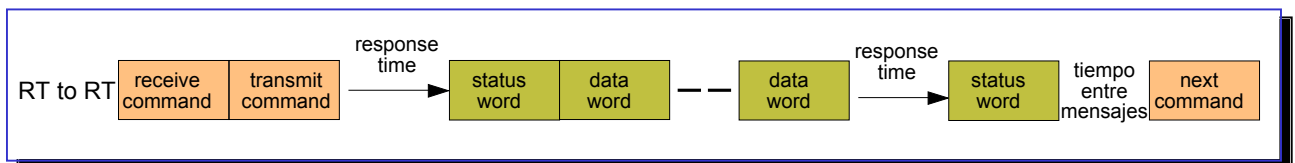
TRANSFERENCIA DE DATOS.



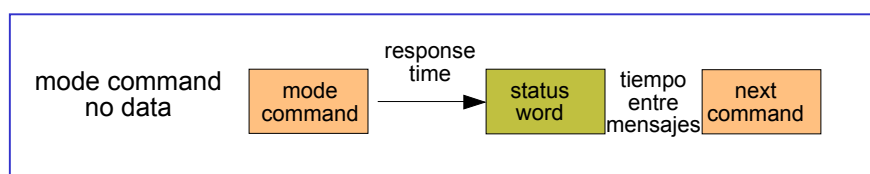
El controlador del bus transmite datos a un terminal remoto. El bit T/R en la palabra de orden, vale 0. Si el mensaje cumple con todos los requerimientos, el terminal remoto responde con la palabra de estado, reconociendo la recepción de la orden y los datos.



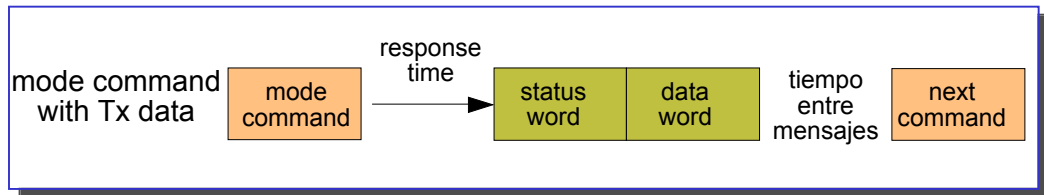
El controlador del bus recibe datos de un terminal remoto. El BIT T/R en la palabra de orden, vale 1. En este caso el controlador de bus ordena a un terminal remoto que le envíe un número concreto de palabras de datos. El terminal responde con una palabra de estado y el correspondiente número de palabras solicitadas.



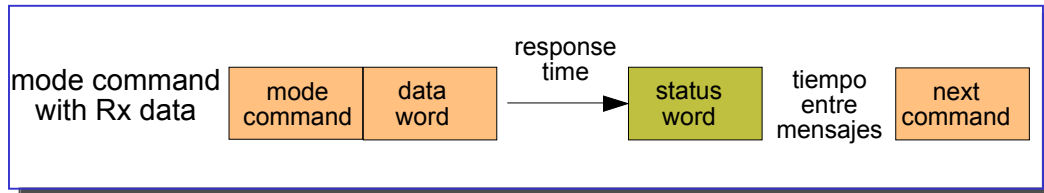
El controlador del bus transmite dos órdenes. Una para el terminal que debe recibir los datos, seguido de otra para el terminal remoto que debe transmitir los datos. Acto seguido el transmisor transmite una palabra de estado (transmisor) seguida del número de palabras indicadas por el controlador del bus. Para finalizar, el receptor transmite la palabra de estado correspondiente (palabra de estado del receptor).



El controlador del bus transmite una orden con el BIT T/R igual a 1, y el campo de subaddress mode field se llena con todo ceros o todo unos. El campo de word count/mode code contiene el código de modo para la orden a ejecutar. Un comando de este tipo puede ser un reset a un terminal remoto, o un self-test.

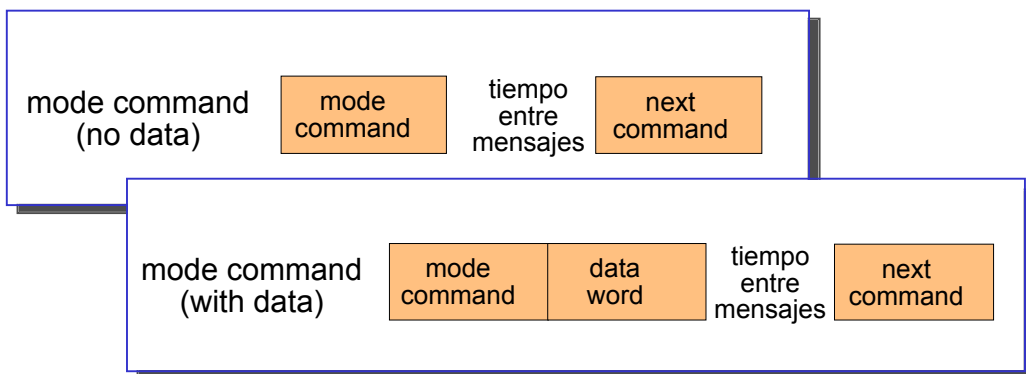
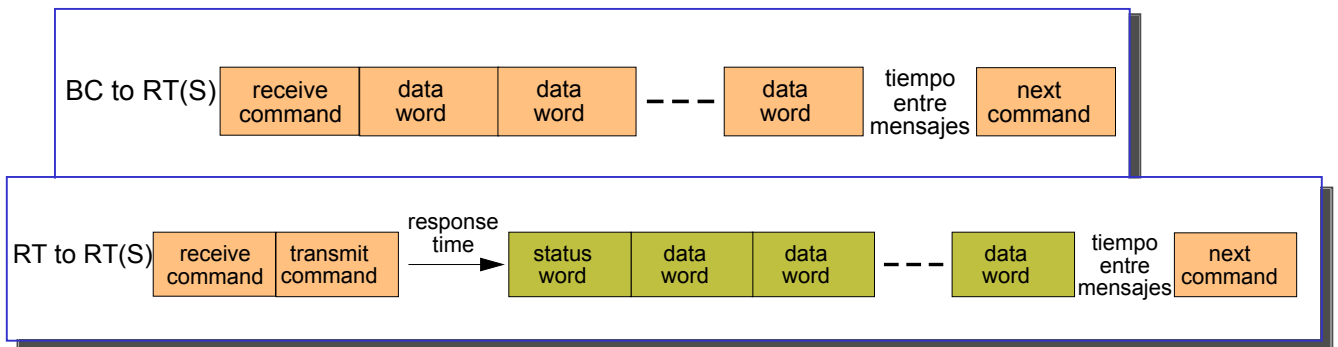


Este modo es igual que el anterior, pero requiere (definido en el campo word count/mode code) que el terminal remoto devuelva una palabra de datos junto con la palabra de estado. Esta palabra de datos puede ser una originada por un built-in-test (BIT), la última orden recibida, etc.



El controlador del bus transmite una orden con el BIT T/R igual a 0 indicando que el terminal remoto va a recibir una palabra de datos.

BROADCAST.



El controlador del bus, siempre transmite este tipo de comandos (a todos los terminales simultáneamente) poniendo en el campo de remote terminal address el valor $11111_{1/2}$ (31). Cuando los terminales remotos ven este tipo de mensaje, lo reciben junto con los datos (si hay) como si hubiera sido dirigido a ellos.

Para el caso de RT to RT(s), el receive command es el que implica a todos los terminales remotos (salvo el transmisor) como receptores del mensaje. El transmit command y la status word, es para el terminal remoto que va a transmitir el mensaje en forma broadcast.

El terminal remoto, pone a 1 el BIT de broadcast command received en la palabra de estado. El controlador del bus determina si un comando broadcast ha sido recibido de forma correcta por los terminales, haciendo un sondeo a cada una de esos terminales, demandando la palabra de estado de cada uno de ellos, y haciendo el chequeo del broadcast command received BIT.

ERRORES EN LOS MENSAJES.

- Existen dos clases de palabras y mensajes que originan errores: **inválidos e ilegales.**

Una orden o palabra inválida es una orden o palabra de datos que tiene un carácter de sincronismo no definido para esa orden o palabra, un BIT que no cumple con la codificación Manchester II o que no cumple la paridad impar requerida.

Un mensaje inválido, es un mensaje que posee una palabra inválida, un hueco temporal entre palabras, o un número de palabras diferente al establecido para ese mensaje.

Una orden ilegal es una orden para la cual un terminal no ha sido diseñado. Por ejemplo, una orden hacia una dirección que no está implementada, es una orden ilegal.

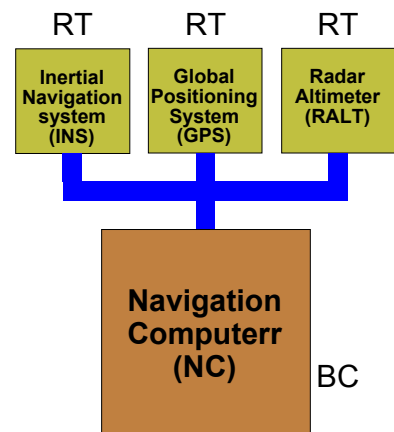
- Si el controlador de bus envía una orden a un terminal remoto y no recibe una palabra de estado por parte de ese terminal remoto dentro del tiempo especificado, el controlador asume que la orden no ha llegado correctamente o que la orden no está dentro de la que reconoce ese terminal. En este último caso, el terminal remoto puede realizar dos acciones; una no hacer nada y esperar que pase el tiempo de respuesta especificado, y dos, pone a uno el *message error BIT* en la palabra de estado y la envía al controlador.

En ambos casos el terminal remoto no tiene en cuenta la palabra de orden y el controlador realiza un procedimiento de recuperación de este error.

Si el error lo detecta el terminal remoto en la parte de datos del mensaje, pondrá el BIT de *message error*, en la palabra de estado, a uno lógico. Aún así, la palabra de estado no es enviada. El controlador del bus, puede demandar del terminal remoto, una vez que ha transcurrido el tiempo específico de respuesta de este, la palabra de estado mediante el modo de orden Transmit Status Word, o si se esta utilizando un bus redundante, puede transmitir la orden de nuevo por el otro bus.

EJEMPLO:

- En esta figura se muestra un ejemplo de aplicación MIL-STD-1553. Corresponde a un formato de *bus simple*. Está constituido por un controlador de bus y tres terminales remotos que corresponden a los sistemas indicados.

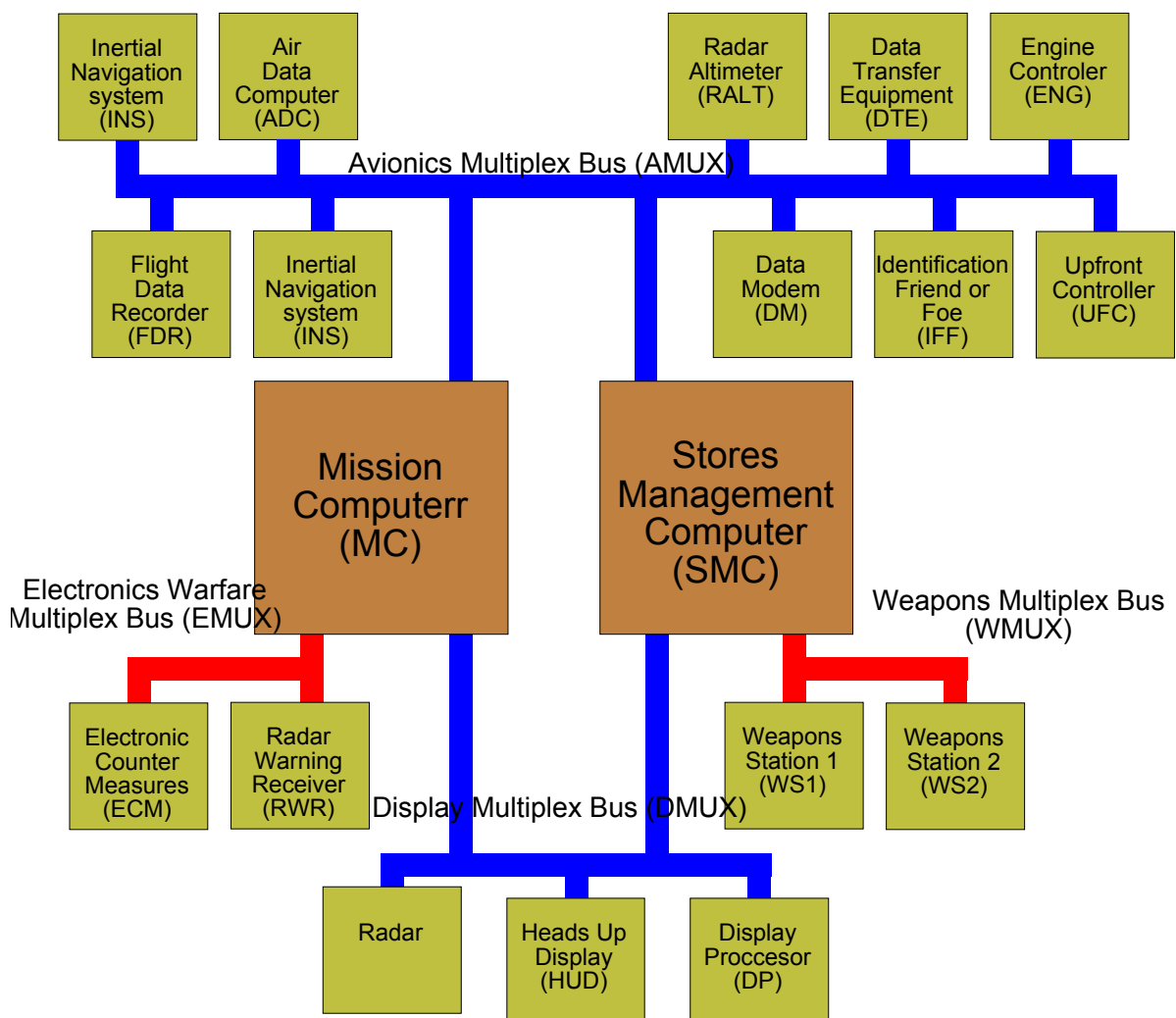


- La figura siguiente, corresponde a un formato de *bus múltiple*, en donde están implicados cuatro buses diferentes en cuanto a la funcionalidad de los sistemas a cada uno de ellos conectados.

Tenemos el WMUX (Weapons Multiplex Bus), relacionado con el armamento de la aeronave, el DMUX (Display Multiplex Bus) principalmente relacionado con los terminales de presentación de la información en los diferentes displays, el EMUX (Electronic Warfare Multiplex Bus), cuyos terminales son principalmente sistemas de medida y sensores electrónicos y por último el AMUX (Avionics Multiplex Bus) con los terminales encargados de la navegación y control de la aeronave.

En este caso, el MC proporciona el control de bus para los buses AMUX, EMUX, y DMUX, y el SMC proporciona control de bus para el WMUX. El criterio para la utilización de un controlador de bus para todos los buses o uno para cada bus u otra configuración diferentes depende de los requerimientos de redundancia, y vulnerabilidad.

En el caso de fallo en el MC, el SMC tomaría el mando como controlador de bus de respaldo para los buses AMUX y DMUX. Se observa que no funcionaría como controlador de bus para el bus EMUX. Se considera en este diseño que este bus no contiene terminales críticos para la seguridad y por lo tanto no necesitan un BC de respaldo para su atención.



12.4-MIL-STD-1773.

- ❑ MIL-STD-1773 define los requerimientos para utilizar un cableado de fibra óptica (*Fiber Optic*) en lugar del par trenzado y apantallado (*Twisted Shielded Wire Pair*) como medio de transmisión en un sistema con protocolo MIL-STD-1553.
-



Displays

13

13.- DISPLAYS.

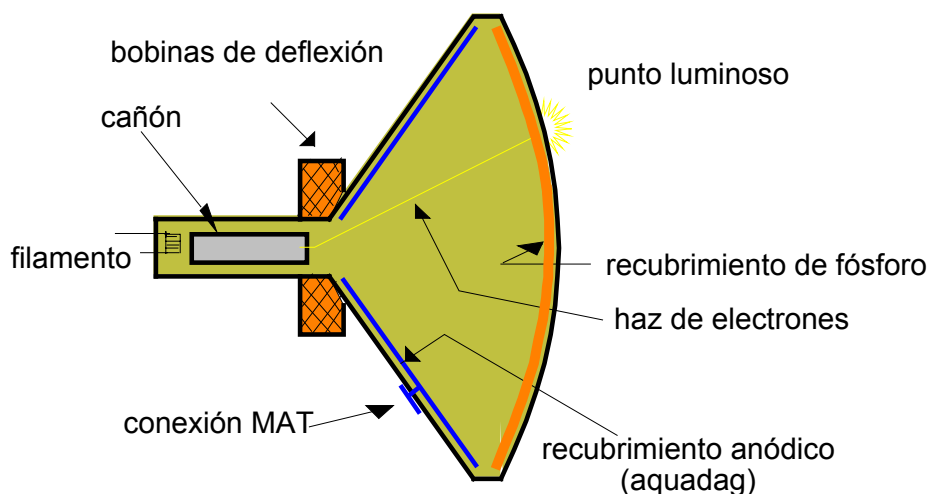
13.1-CRT (CATHODE RAY TUBE).

En su aspecto físico más básico un tubo de rayos catódicos consiste en una ampolla de vidrio con un conjunto de electrodos en su interior, en la cual se ha hecho el vacío.

13.1.1-CRT MONOCROMO.

Las partes fundamentales, son las siguientes:

PANTALLA.



Es la parte frontal o parte más ancha de este tubo. Está recubierta por su parte interna de material fosforescente (compuesto de fósforo), el cual emite radiación luminosa cuando es excitado por electrones (por un haz en nuestro caso), con energía suficiente. La naturaleza del compuesto de fósforo, determina las propiedades fundamentales del tubo como son el color (generalmente blanco, verde o ámbar) y la persistencia de la luz.

CAÑÓN.

Es un elemento situado dentro del tubo, en la parte posterior o parte mas estrecha, llamada comúnmente *cuello del tubo*. El cañón está formado por un cátodo que es calentado mediante un filamento con el fin de que pueda emitir electrones. Hay también un conjunto de electrodos que se encargan de acelerar y concentrar el chorro de electrones en un haz muy delgado y enfocararlo en la pantalla, para poder producir el punto luminoso lo mas fino posible. También se encuentra en el cañón un electrodo que permite controlar la intensidad del haz, lo que se traduce en una mayor o menor luminosidad del punto sobre la pantalla.

CUELLO.

Como se ha indicado, es la parte más estrecha del tubo y en la cual se realizan las conexiones eléctricas a excepción de la conexión MAT (Muy Alta Tensión).

BOBINAS DE DEFLEXIÓN.

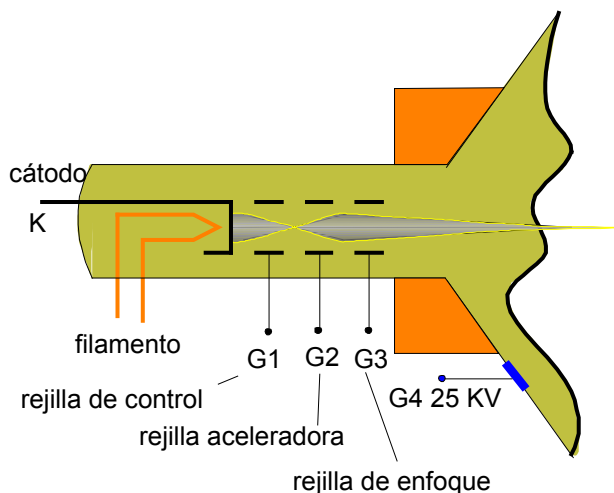
Estas bobinas están situadas alrededor del cuello del tubo. Cuando circula corriente por estas bobinas (corriente con forma de dientes de sierra), se generan campos magnéticos que producen la desviación del haz, logrando así, que este alcance cualquier punto de la pantalla. El conjunto de las bobinas recibe el nombre de *yugo*.

BOBINA DE DEFLEXIÓN VERTICAL. La corriente que circula por esta bobina hace que el haz de electrones se desplace en la dirección vertical.

BOBINA DE DEFLEXIÓN HORIZONTAL. La corriente que circula por esta bobina hace que el haz de electrones se desplace en la dirección horizontal.

FUNCIONAMIENTO

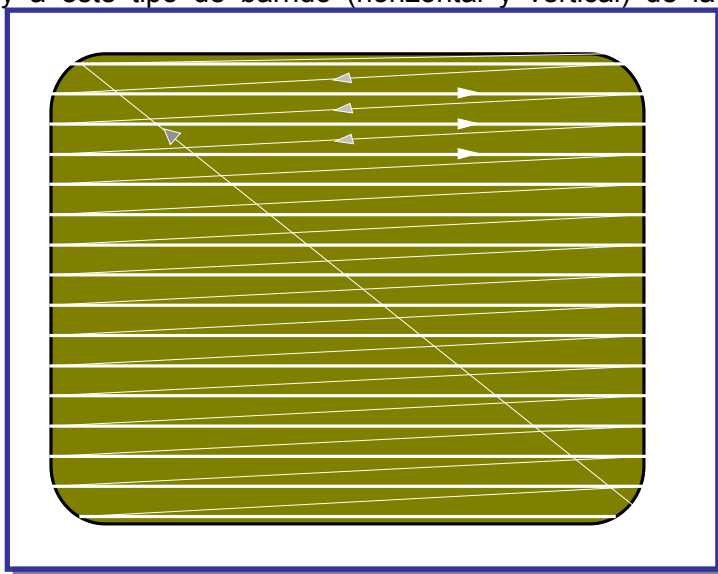
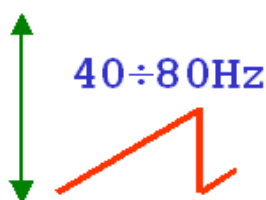
- El funcionamiento básico del CRT es muy simple: el filamento que hay dentro del tubo, alimentado aproximadamente a 6V y con un consumo de corriente que puede variar de 0,5 hasta 2 A, calienta el cátodo, energizando los electrones y facilitando la emisión de un haz. Un potencial alto (de varios cientos de voltios) de DC, más positivos que el cátodo, se aplica a la rejilla G2, haciendo que los electrones del haz sean acelerados hacia la pantalla. Como el haz de electrones que emerge del cátodo tiende a diverger, se aplica un potencial negativo respecto del cátodo en la primera rejilla G1, llamada rejilla de control, haciendo que el haz converja hacia un punto central en la dirección del eje del movimiento. Modulando la diferencia de potencial entre el cátodo y la rejilla de control, la intensidad del haz también se modula, o lo que es lo mismo, varía el brillo en la pantalla. Finalmente el haz es enfocado de forma electrostática, mediante el potencial aplicado a la rejilla de enfoque (G3), para dirigirlo hacia el ánodo (pantalla) conectado a un potencial de muy alta tensión (MAT), que puede variar desde 10KV hasta 30KV.



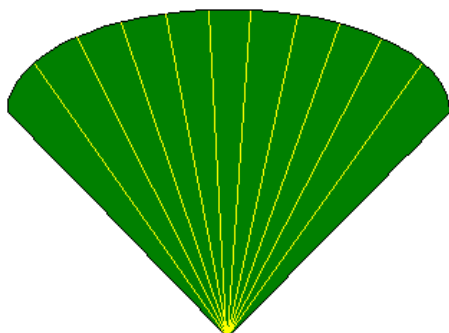
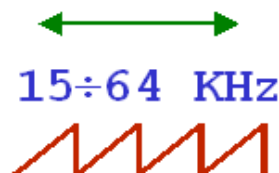
BARRIDOS.

Las bobinas son excitadas por corrientes periódicas en forma con dientes de sierra. Estas señales se llaman *señales de barrido* y a este tipo de barrido (horizontal y vertical) de la pantalla se llama *Raster*.

Barrido Vertical. Esta señal es la que se encarga del movimiento vertical del haz. La frecuencia de esta señal suele estar comprendida entre 40 y 80 Hz. Determina las veces que se recorre la superficie de la pantalla en cada segundo. Si esta frecuencia es demasiado baja, el ojo humano puede percibir el efecto de parpadeo.

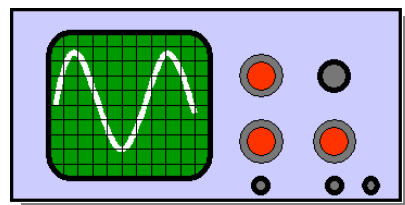


Barrido Horizontal. Se encarga del movimiento horizontal del haz. La frecuencia de esta señal suele estar entre 15 y 64 kHz. Esta frecuencia define cuantas veces por segundo se recorre la pantalla en sentido horizontal, mientras se realiza un recorrido completo de arriba abajo. Determina cuantas líneas va a contener la pantalla.



Otro tipo de barrido llamado barrido ρ - θ (RHO-THETA) se ha venido usando hasta la aparición de los CRT en color, para representar, por ejemplo, información procedente del radar. En este tipo de barrido, el origen del haz es siempre el punto medio de la línea inferior de la pantalla. La información se representa en los radios que salen desde este punto y que hacen un recorrido en forma de abanico.

Existe también el barrido vectorizado, en el cual el haz de electrones, únicamente recorre la zona de la pantalla que representa la imagen, esto es, el haz de electrones es como un pincel que dibuja en la pantalla la imagen. Para ello, el movimiento del haz en las dimensiones de anchura y altura de la pantalla es producido por campos eléctricos generados por dos pares de placas. Un par de ellas, a las que se les aplica una onda en diente de sierra generada internamente y que forma la base de tiempos, hace que el haz se desplace en el sentido horizontal, mientras que el otro par de placas a las que se aplica la señal a representar, hacen que el haz se desplace en sentido vertical. La combinación de los campos producidos por los dos pares de placas, hace que el haz se desplace dibujando la señal en la pantalla. Este método, es el utilizado por los osciloscopios analógicos.



SINCRONISMOS.

Las señales de barrido deben estar perfectamente sincronizadas para poder componer la imagen. Para ello, a los generadores de barrido se les suministran las señales de sincronismo consistentes en impulsos de la frecuencia de barrido deseada. En TV, estas señales de sincronismo vienen con la señal que se recibe por la antena. En otros sistemas de presentación con CRT, estas señales son generadas por tarjetas controladoras del tubo de rayos catódicos.

Las señales de sincronismo fuerzan a los generadores de barrido del sistema a funcionar a la frecuencia adecuada. En ausencia de estas señales, los generadores de barrido oscilan en su frecuencia propia, próxima a la de sincronismo presentándose en la pantalla una imagen distorsionada e inestable.

MODULACIÓN DEL HAZ.

El haz de electrones que incide en la pantalla, será visto como un punto si la incidencia se realiza con la energía suficiente y por supuesto, si el haz permanece estático. Dependiendo de cómo sea el control del haz el efecto sobre la pantalla será diferente:

Control Analógico. La intensidad del punto, podrá ser cualquiera entre la máxima intensidad (blanco) y apagado (negro), es decir, blanco, negro y todos los niveles de grises intermedios.



Control Digital. La intensidad del punto dependerá del número de bits. Para un solo BIT, solamente tendrá dos valores que serán el punto apagado (negro) y el punto iluminado (blanco). Para 2 bits, se obtendrán cuatro posibles colores, para 3 bits ocho colores, etc.

Para 1 BIT

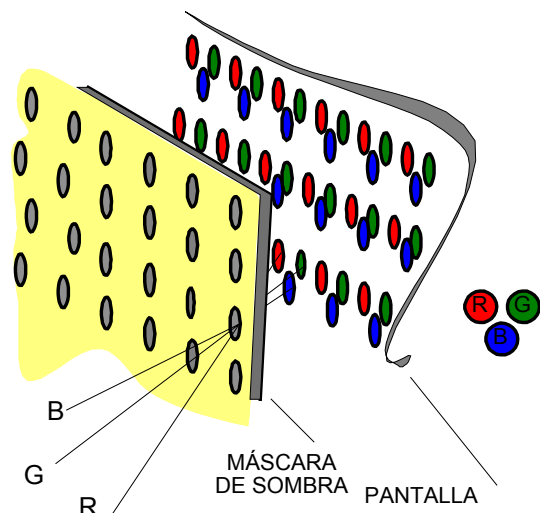
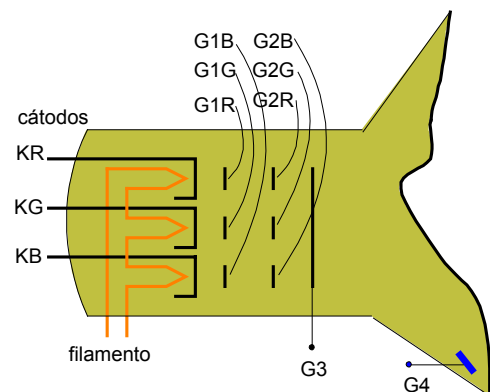


Para 2 bits



13.1.2-CRT COLOR.

- ❑ Este CRT es especial. En él existen tres haces de electrones, uno para cada color de luz básico RGB (Red, Green, Blue).
- ❑ La pantalla está dividida en diminutos puntos (*luminóforos*). Estos puntos se agrupan en bloques de tres (*triadas*). Cada uno de los luminóforos de una triada está recubierto por fósforo de uno de los tres colores básicos. Estos recubrimientos están destinados a emitir una radiación luminosa en su color correspondiente. En una triada se tiene por lo tanto los tres colores básicos de luz: rojo(R), verde (G) y azul (B).
- ❑ Cada uno de los haces, controla la porción de fósforo correspondiente a su color. Esto quiere decir que cada uno de los haces debe estar ajustado para que impacte en la zona de la triada que corresponde a su color; haz de electrones del rojo controla el fósforo rojo, haz de electrones del verde, el verde y haz de electrones del azul, fósforo azul. Cualquier color diferente de los tres colores básicos, se consigue por mezcla aditiva de estos.
- ❑ Una delgada lámina de acero (*shadow mask*), situada entre la pantalla y los cañones, se encarga de que cada uno de los haces solamente impacte en el punto de la



triada correspondiente a su color. La máscara tiene un orificio para cada triada, en cada instante, los tres haces inciden sobre la misma triada, y cada haz en el luminóforo correspondiente a su color. De esta manera se evita que se iluminen zonas de la pantalla, que no forman triada y por lo tanto no pueden aportar información de color a la imagen.

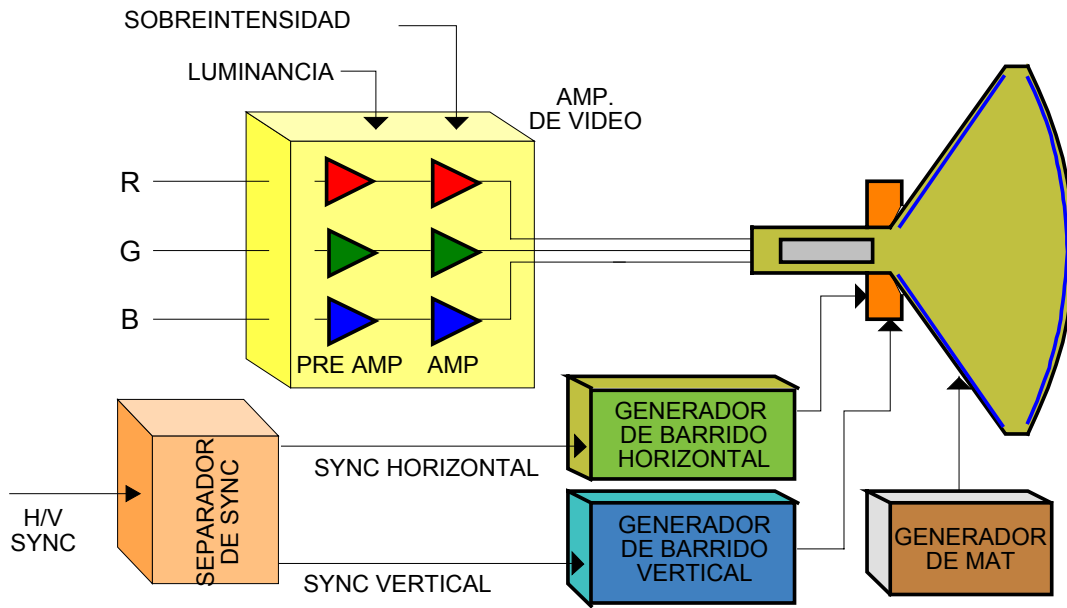
- ❑ La intensidad de cada haz se modula de forma independiente.
- ❑ En un CRT cuyo control digital de la intensidad del haz se realiza con un solo BIT (haz encendido o haz apagado), solo se podrán representar los ocho 8 colores que se indican en la tabla.
- ❑ Si la intensidad de los haces se controla de forma analógica, la posibilidad de colores por mezcla de los tres básicos es infinita.

R	G	B	
0	0	0	NEGRO
0	0	1	AZUL
0	1	0	VERDE
0	1	1	CIAN
1	0	0	ROJO
1	0	1	MAGENTA
1	1	0	AMARILLO
1	1	1	BLANCO

13.1.3-MONITORES BASADOS EN CRT.

- ❑ Toda la circuitería interna de un monitor, se puede agrupar dentro de tres categorías principales que son: (a) el procesamiento y amplificación de la señal de video, (b) sincronismos y deflexiones H y V y por último, (c) la alimentación.
- ❑ La señal que llega al CRT suele ser de una amplitud muy baja, alrededor de 1Vpp por lo cual necesita ser amplificada antes de ser aplicada a los cátodos. Una primera amplificación se realiza en el pre-amplificador, el cual, además, proporciona el control de contraste y brillo. El control de contraste permite variar la ganancia del amplificador de video. Un incremento del contraste, hace que se incremente el nivel de la señal de video, lo cual hace que las partes con más luz de la imagen se hagan más brillantes y las partes más oscuras, menos brillantes. La variación del brillo de la imagen se realiza variando el nivel de la componente DC de la señal de video. Si se aumenta el brillo, las partes más iluminadas, y también las más oscuras brillarán más.
- ❑ El amplificador se encarga de amplificar la señal del preamplificador hasta el nivel que se requiere para que el cátodo pueda energizar cada punto de fósforo de la pantalla. En un monitor de color, cada amplificador amplifica la señal correspondiente a su color para energizar cada uno de los luminóforos de la triada, que como se he indicado, está compuesta por los fósforos de cada color básico de luz (RGB).

La luz emitida por cada punto de fósforo es proporcional al número de electrones que incide sobre ese punto. Modulando el voltaje de cada uno de los tres cátodos en un monitor de color, el luminóforo correspondiente será energizado a diferente intensidad, produciendo los diferentes colores en los pixels que forman la imagen.

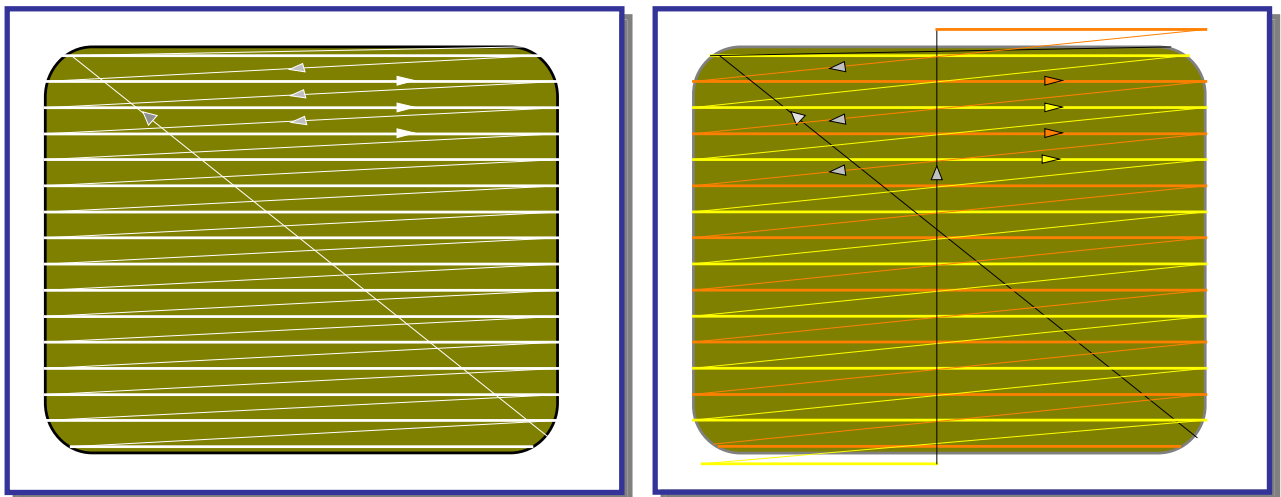


FORMACIÓN DE LA IMAGEN EN LA PANTALLA.

- ❑ Los generadores de barrido horizontal y vertical hacen que el haz recorra la pantalla continuamente. El generador de barrido horizontal hace que el haz se desplace de izquierda a derecha en cada línea visible, y de derecha a izquierda en lo que se denomina retrazado horizontal, de forma no visible en pantalla. El generador de barrido vertical hace que el haz tenga un desplazamiento desde la parte de arriba hacia la de abajo de forma visible y de abajo arriba (retrazado vertical) de forma no visible.

La información en la pantalla viene dada mediante la modulación de la intensidad de los haces electrónicos RGB, iluminando a los niveles correspondientes, los puntos adecuados.

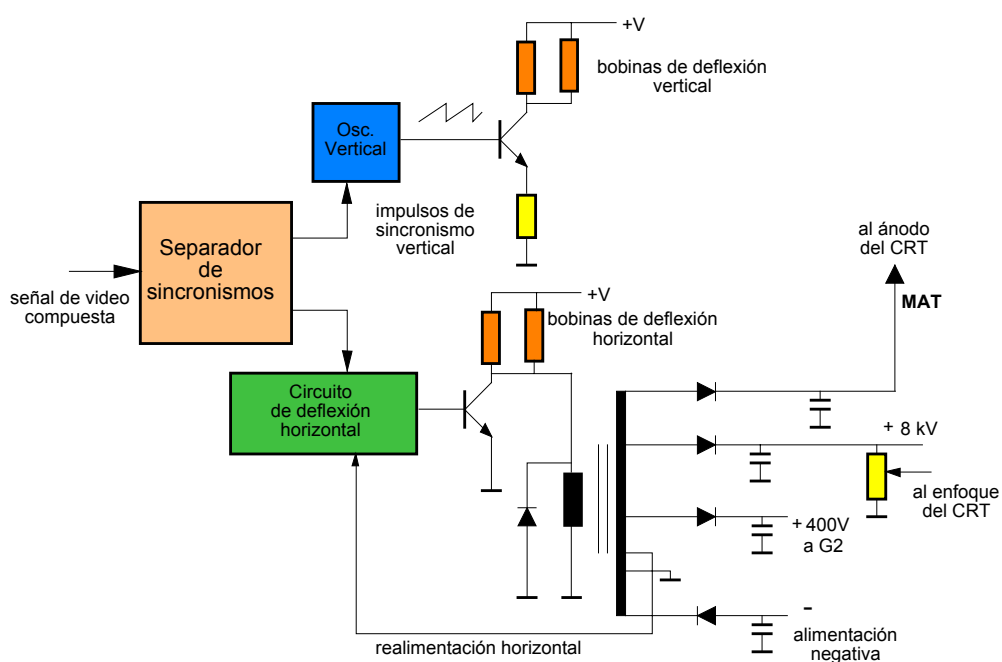
- ❑ Como la luminiscencia del fósforo va desapareciendo poco a poco después del impacto del haz, la imagen necesita ser formada de forma periódica. En los monitores por raster, la información en la pantalla debe ser re-escrita constantemente, del orden de 40-80 veces por segundo (fósforo de persistencia baja o media), de esta manera se obtiene la sensación de continuidad en la imagen, y no la sensación de una sucesión de fotogramas. Este proceso es lo que se llama *refresco de pantalla*, y el circuito encargado de esta misión es el controlador de pantalla.

BARRIDOS NORMAL Y ENTRELAZADO.

NO ENTRELAZADO

ENTRELAZADO

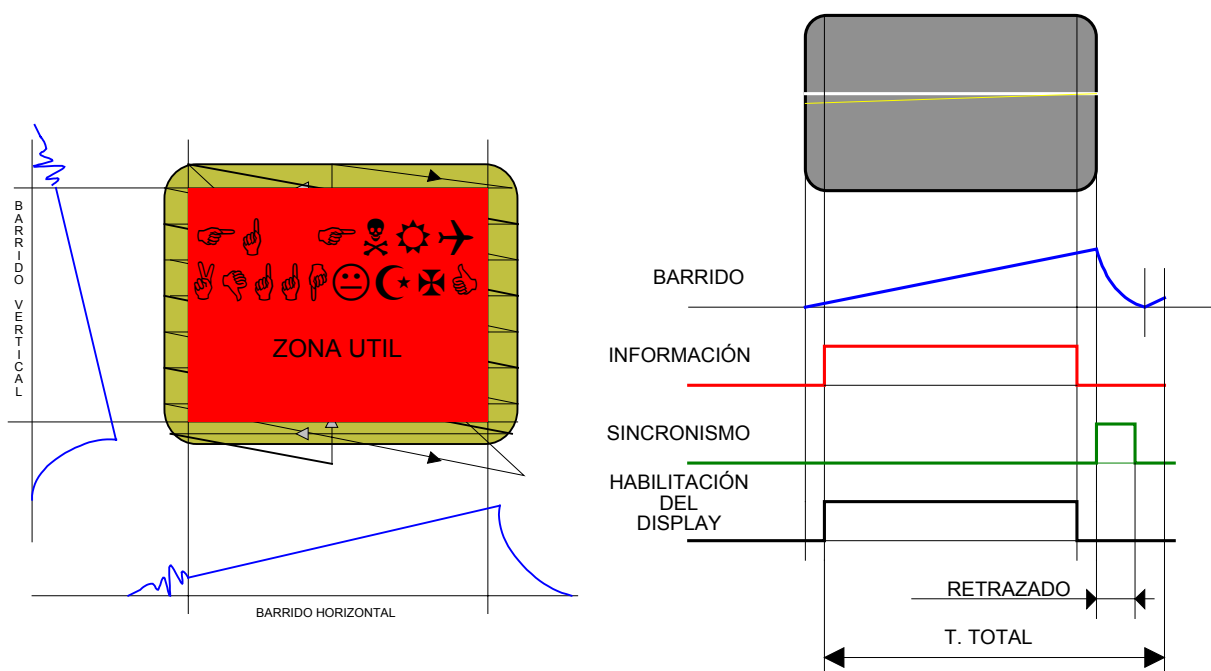
- ❑ El recorrido completo de la pantalla de arriba hacia abajo, determinado por la frecuencia de barrido vertical, origina una *imagen* o un *cuadro*.
- ❑ Cuando un monitor funciona en modo entrelazado, cada imagen está contenida en dos campos. El primer campo solo está formado por las líneas pares, dejando el espacio entre cada una de estas líneas para que el segundo campo trace las líneas impares. Dicho de otra forma, cada imagen se forma al realizarse dos barridos verticales de la pantalla. En el modo no entrelazado, la imagen se forma de un solo barrido.
- ❑ Cuanto más grande es la frecuencia de barrido vertical, menor es la sensación de parpadeo que siente el observador.

SEÑALES DE BARRIDO Y SINCRONISMO.

- ❑ El separador de sincronismos se encarga de aislar de la señal de video, los impulsos de sincronismo horizontales y verticales y dirigir cada uno de estos impulsos a los circuitos de deflexión horizontal y vertical, según corresponda.

Los impulsos de sincronismo vertical disparan el oscilador vertical, el cual genera una señal en diente de sierra que una vez amplificada es aplicada a las bobinas que causan la deflexión vertical del haz de electrones, haciendo que éste pase de una línea a la siguiente con una velocidad uniforme. La amplitud de la señal de corriente aplicada a las bobinas es ajustable de forma manual, permitiendo de esta manera seleccionar la altura de la imagen presentada en la pantalla.

- ❑ La pérdida del sincronismo vertical se manifiesta por el desplazamiento en sentido vertical, de la imagen formada en la pantalla.
- ❑ El circuito de barrido horizontal trabaja de forma similar al de barrido vertical, con la diferencia de que al utilizar alta frecuencia necesita más precaución en el diseño de los circuitos para mantenerlos libres de cualquier ruido. El oscilador horizontal, en lugar de ser un oscilador de relajación disparado por la señal de sincronismo como ocurre con el oscilador vertical, es un VCO (Voltage Controlled Oscillator). Un AFC (Automatic Frequency Control) compara la fase de la señal de sincronismo horizontal con una muestra de la que existe en la salida horizontal, generando una tensión de error de DC proporcional al desfase de las dos señales. Esta señal gobierna el VCO para ajustar su frecuencia a la de la señal de sincronismo.
- ❑ La pérdida de sincronismo horizontal se manifiesta por una distorsión de la imagen en forma de bandas diagonales indefinidas.
- ❑ Las altas tensiones necesarias en las rejillas y ánodo del CRT se obtienen de la acción de un transformador elevador, cuyo primario está gobernado por el mismo transistor que gobierna las bobinas de deflexión horizontal. Durante el retrazo horizontal, la variación de corriente es muy rápida, provocando las grandes tensiones en el secundario del transformador.
- ❑ La complejidad para obtener linealidad en todo el recorrido de las corrientes en diente de sierra que se aplican a las bobinas de deflexión, hace que ni el comienzo de las rampas ni el final de las mismas, sean utilizadas en la zona visible de la pantalla.



- ❑ La información que debe modular la intensidad del haz para formar la imagen en la pantalla, solamente se aplica en los tiempos en los cuales las rampas de las señales de barrido están en la zona útil de presentación.
- ❑ El final de cada línea viene determinado por un impulso de sincronismo horizontal.
- ❑ Una señal de habilitación de display, evitará que sea visible la zona inútil de cada línea. Esta señal mantiene apagada la pantalla durante el principio de la rampa, el final de la rampa y durante el tiempo de retrazado.

CLASIFICACIÓN DE MONITORES.

MONITORES MONOCROMO.

SIMPLE: Utiliza un BIT para modulación del haz (encendido o apagado). Las señales de sincronismo horizontal y vertical se suministran por conexiones independientes.

CON SOBREINTENSIDAD: Como el anterior, pero una señal independiente permite controlar la intensidad del haz entre los valores de normal y sobre-intensidad. Los puntos podrán aparecer apagados, encendidos con luminosidad normal, o encendidos con mayor luminosidad.

DE ESCALA DE GRISES: Admite una señal de intensidad (luminancia) de tipo analógico. Los puntos pueden tener cualquier intensidad luminosa.

DE VÍDEO COMPUESTO: Las señales de sincronismo se mezclan con las señales de video (luminancia). Circuitos internos del monitor se encargan de separar estas señales.

MONITORES COLOR.

SIMPLE: Existen tres señales, RGB controladas por un BIT. Se pueden presentar hasta 8 colores.

AMPLIADO: Dispone de una señal adicional de sobre-intensidad ($I=0$, normal, $I=1$, sobre-intensidad). Con las tres señales de color, más esta de intensidad se pueden presentar hasta los 16 colores que se especifican en la siguiente tabla:

R	G	B	COLOR ($I=0$)	COLOR ($I=1$)
0	0	0	NEGRO	GRIS OSCURO
0	0	1	AZUL	AZUL CLARO
0	1	0	VERDE	VERDE CLARO
0	1	1	CIAN	CIAN CLARO
1	0	0	ROJO	ROJO CLARO
1	0	1	MAGENTA	MAGENTA CLARO
1	1	0	MARRON	AMARILLO
1	1	1	GRIS CLARO	BLANCO

ANALÓGICO: Cada una de las señales, RGB, se pueden controlar de forma analógica (infinitos valores), por lo que en estos monitores se pueden obtener infinitos colores.

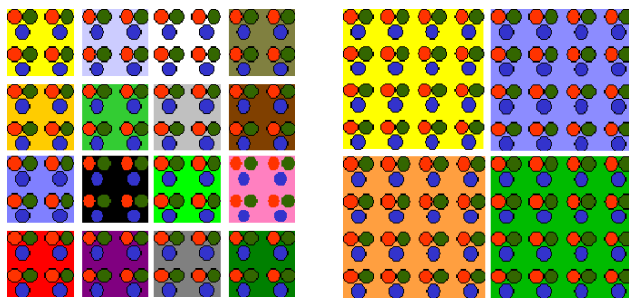
VÍDEO COMPUESTO: En una sola señal se combinan las señales RGB (analógicas o digitales) y las señales de sincronismo. Se adaptan a los métodos estándar de transmisión TV-C (PAL, SECAM, NTSC).

MONITORES MULTISINCRONISMO.

- Se adaptan a cualquier tipo de frecuencia de sincronismo que se les envíe (dentro de un cierto margen), para poder generar los barridos.

PARÁMETROS

RESOLUCIÓN. El elemento de imagen más pequeño que un sistema de video es capaz de reproducir es el píxel. Cuanto más pequeño es el tamaño de píxel, mayor es la resolución del sistema. Desde el punto de vista del monitor, el tamaño de píxel, solo está limitado por el tamaño de las triadas (cada luminóforo, y en realidad, cada triada ha de ser igual o más pequeño que el tamaño del píxel para poder aportar toda la información al mismo) y por el ancho de banda del sistema (cuanta más alta es la resolución de un monitor, más pequeño es el tiempo disponible para energizar cada píxel pues el número de estos es mayor). La resolución en pixels, se da mediante el valor del número de estos en dirección horizontal, por el número en dirección vertical. (p. ej. 1024 X 768).



En las figuras se representan dos resoluciones diferentes. En una de las figuras (la de la izquierda) tenemos para un área dada, 16 pixels, con una información de color diferente para cada uno de ellos. En la otra (la de la derecha) tenemos para la misma área, 4 pixels, en la que solo se pueden representar cuatro colores. Obviamente, la formación de

cualquier imagen será mucho más pobre con esta última resolución que con la primera.

DOT PITCH. Expresa la distancia entre dos triadas adyacentes.

ERRORES DE GEOMETRÍA: COJÍN Y BARRILETE. La imagen toma forma de estas figuras por defectos en la deflexión.

LOW RADIATION. Las bobinas de deflexión, recorridas por fuertes corrientes emiten ondas electromagnéticas de baja frecuencia. Para reducir la posibilidad de que estas ondas salgan al exterior, se deben proveer apantallamientos que absorban en la medida de lo posible dichas radiaciones.

ERROR DE CONVERGENCIA. Cuando los haces electrónicos no tienen la trayectoria adecuada, y no inciden en la zona del luminóforo correspondiente a su color, se produce el error de convergencia. Generalmente este ajuste se realiza de forma automática al conectar el monitor.

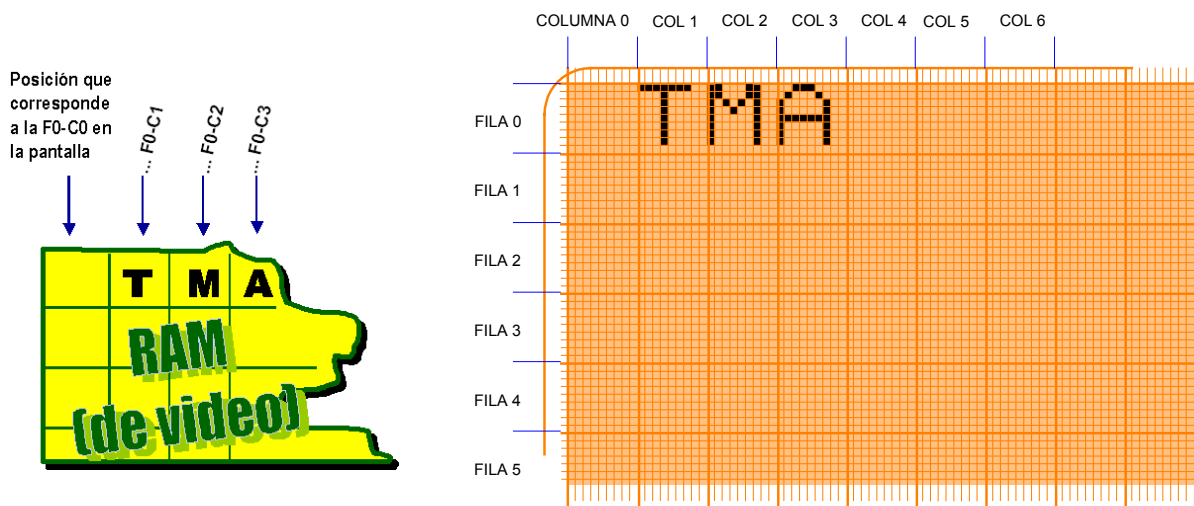
13.1.4-CONTROLADORES DE PANTALLA.

- ❑ Un controlador de pantalla engloba el conjunto de circuitos que permiten la formación de una imagen en la pantalla. Generan la señal de video y las de sincronismo correspondientes, se encargan también, del refresco de la pantalla para que la información que se presenta en ella aparezca estable al ojo humano.
- ❑ Toda modificación que la CPU quiera realizar sobre la información presentada en la pantalla, la hace efectiva a través del controlador.
- ❑ El controlador puede ser alfanumérico o gráfico. Los alfanuméricos son aquellos que sólo pueden representar en la pantalla un conjunto determinado de caracteres. Los controladores gráficos pueden representar cualquier forma gráfica en la pantalla.

CONTROLADOR ALFANUMÉRICO.

- ❑ La pantalla se organiza en forma de matriz. El caso más típico para pantallas alfanuméricas es de 25 filas de 80 columnas (2000 caracteres)

Una fila es una secuencia horizontal de caracteres. Cada fila se compone de un cierto número de líneas de barrido.



EN LA FIGURA, CADA CARÁCTER OCUPA UN RECTÁNGULO DE 9 X 9 PÍXELS. LA REPRESENTACIÓN VISIBLE DE UN CARÁCTER DETERMINADO, SE REALIZA SOBRE UNA MATRIZ DE 7 X 8 PÍXELS.

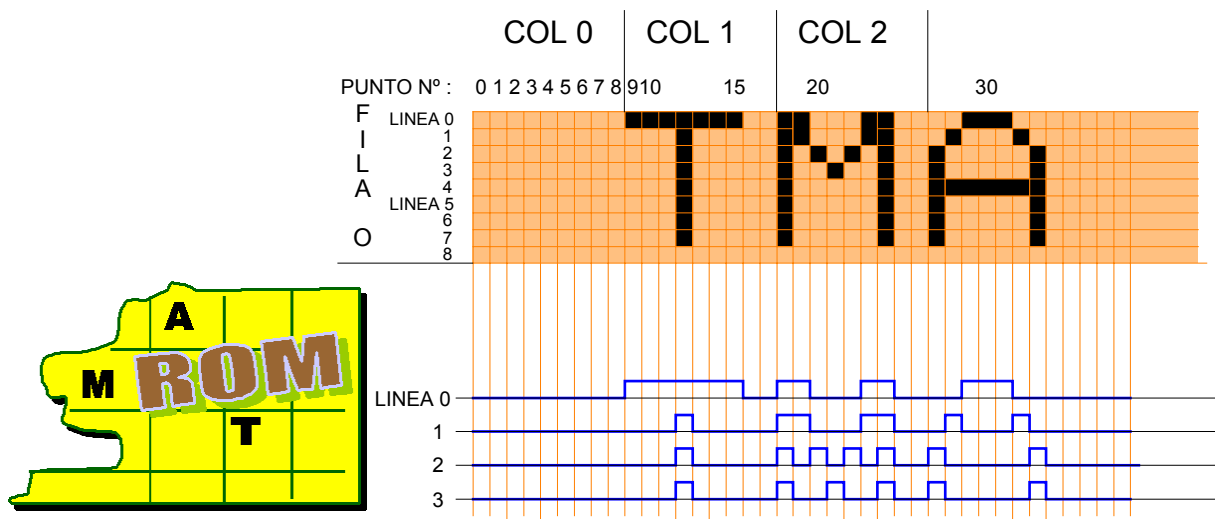
Una columna es el espacio en sentido vertical con anchura igual a la de un carácter. Está constituida por un número determinado de puntos.

Cada carácter se compone de un cierto número de pixels formando un rectángulo. Incluye también los espacios en blanco necesarios para separarlo de otros caracteres, tanto en sentido horizontal como en vertical.

- ❑ Los caracteres que en un momento dado se van a visualizar en la pantalla, son guardados por el microprocesador en una memoria RAM (memoria de pantalla) mediante el formato de su código ASCII correspondiente.
- ❑ El tamaño de esta memoria debe ser tal que pueda contener todos los caracteres de la pantalla. Por ejemplo, para una pantalla de 25 filas x 80 columnas la memoria debe contener 2000 caracteres.

- ❑ Cada posición de esta memoria, se corresponde directamente con una posición concreta (fila-columna) que ocupa un carácter en la pantalla, y contiene el código ASCII del carácter que se va a representar.
- ❑ Una vez que se tiene la información que hay que representar en pantalla, el problema reside en cómo se debe representar cada carácter. Cada carácter está contenido en una matriz de pixels. Las filas de esa matriz coinciden con las líneas de barrido, y las columnas con lo que llamamos puntos. Para representar en pantalla un carácter de 8 x 8 pixels en B/N, necesitamos 64 bits, es decir, la información de cada píxel está contenida en un BIT cuyo valor como ya sabemos puede ser 0 o 1, dando como resultado en la pantalla, que un píxel esté apagado o esté iluminado.
- ❑ El conjunto de caracteres representable en pantalla, se encuentra definido en una memoria ROM llamada **GENERADOR DE CARACTERES**. Esta memoria contiene la información final que corresponde a la manera en la cual cada carácter va a ser representado en pantalla.
- ❑ El tamaño del generador de caracteres depende del número de caracteres que es capaz de definir en pantalla. Por ejemplo, si para representar cada carácter necesitamos 64 bits, para representar todo el conjunto de caracteres ASCII ampliado (256 caracteres), necesitamos una memoria ROM de

$$256 \times 64/8 = 2 \text{ KB.}$$
- ❑ La formación de cada carácter se realiza mediante el recorrido del haz de izquierda a derecha de la pantalla, línea a línea. En cada trazado se va formando la parte concreta de la línea correspondiente a cada uno de los caracteres que han de ser representados en una fila determinada. Dicho de otra forma, en cada trazado se forma una pequeña parte de todos los caracteres que van a estar presentes en una fila.



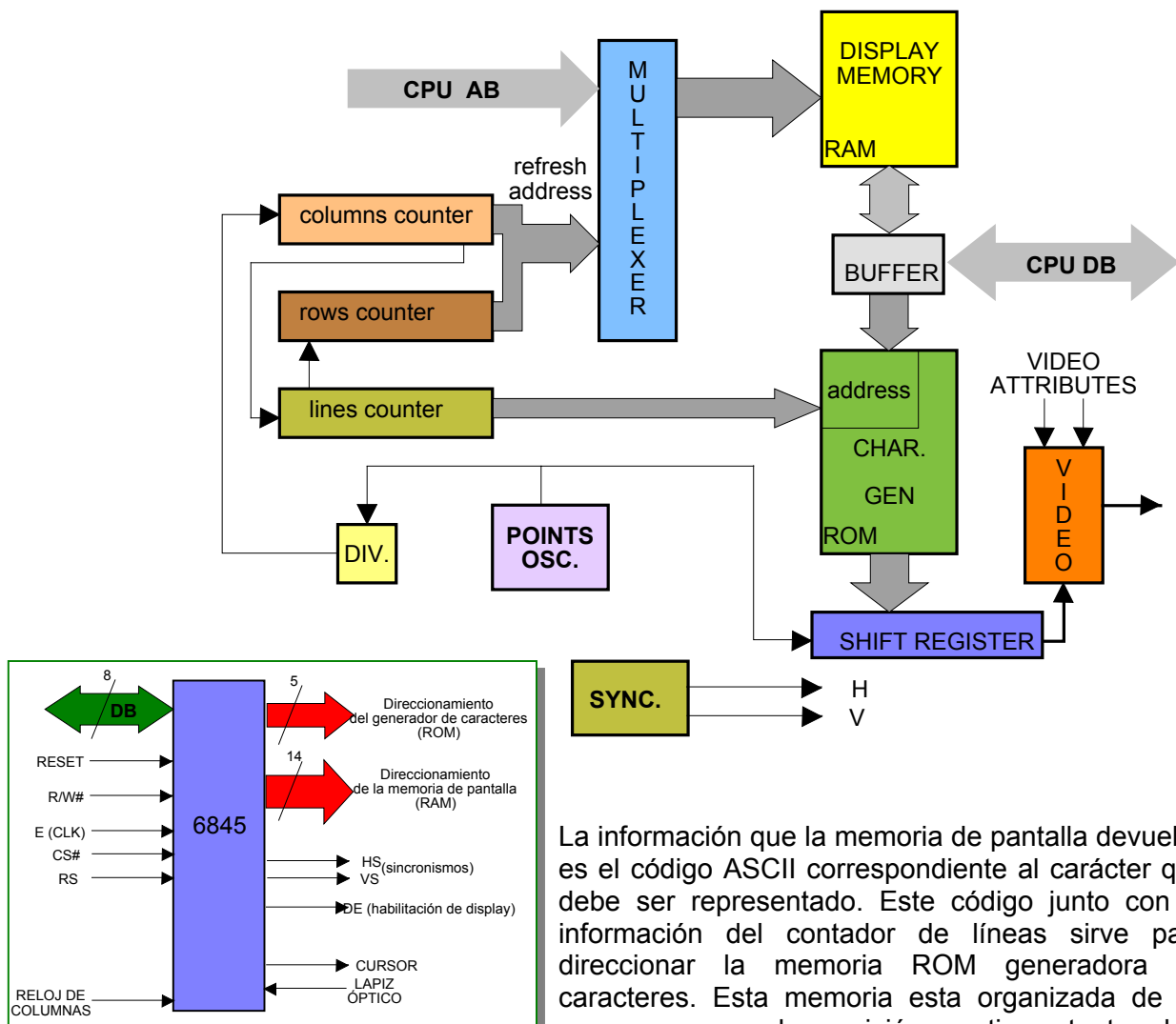
- ❑ El controlador de pantalla determina en que fila y columna se encuentra el haz. También, dentro de cada carácter, en qué **línea** y en qué **punto** se encuentra dicho haz, de esta manera realiza el control para saber si el píxel ha de estar iluminado o apagado.
- ❑ La forma de determinar todos los parámetros anteriores es mediante una serie de contadores que posee el controlador de pantalla. La unidad básica de medida la establece el *reloj de puntos*, cuya frecuencia es la inversa del tiempo que tarda el haz en pasar de un punto al siguiente.

El funcionamiento básico de un controlador alfanumérico de pantalla es como sigue (ver figura siguiente):

La frecuencia del oscilador de reloj de puntos se divide entre el número de puntos que define cada carácter en sentido horizontal. De esta división, se obtiene una frecuencia que será la del reloj llamado de columnas o de caracteres, la cual permitirá contar las columnas mediante el contador de columnas. El módulo de este contador es el del número de columnas del sistema, por lo tanto, cuando llegue al final de su cuenta indicará que se ha recorrido una línea. La señal de indicación de columnas se lleva a un contador de líneas, cuyo módulo es el número de líneas que compone cada carácter. El fin de cuenta de este contador, indicará que se ha recorrido toda la fila de caracteres, con lo que esta señal de fin se lleva al contador de filas.

El estado de los contadores de fila y columna determinan la posición de un carácter en la pantalla, por lo que estos contadores se usan para direccionar la memoria de *RAM de pantalla*. Cuando estos contadores tengan un valor 0, estarán generando la dirección de memoria RAM donde la CPU, según el programa, ha guardado el código ASCII correspondiente al carácter que debe ser representado en la esquina superior derecha de la pantalla (fila cero, columna cero).

Para otros valores de estos contadores se generan las diferentes direcciones de la memoria de pantalla que guardan los caracteres que deben ser representados en posiciones específicas de dicha pantalla.



La información que la memoria de pantalla devuelve es el código ASCII correspondiente al carácter que debe ser representado. Este código junto con la información del contador de líneas sirve para direccionar la memoria ROM generadora de caracteres. Esta memoria está organizada de tal manera que cada posición contiene tantos bits como puntos horizontales definen un carácter, por lo que cuando se direcciona una

posición correspondiente a la representación de un carácter determinado y una línea determinada de ese carácter, la memoria devuelve de forma simultánea la información de la línea a representar. Esta información es enviada a un registro de desplazamiento (carga paralelo-lectura serie) que a ritmo del reloj de puntos entrega el valor del BIT correspondiente para modular el haz y que en la pantalla aparezca el píxel iluminado o apagado.

- ❑ En todo el proceso explicado, la memoria de pantalla es direccionada mediante la información procedente de los contadores de filas y columnas internos del controlador, y la información guardada en esta memoria va apareciendo repetidamente en la pantalla. Este proceso corresponde al refresco de la pantalla y hace que la imagen parezca estable al ojo humano.
- ❑ Cuando se quiere modificar el contenido que se presenta en pantalla, se debe variar el contenido de la memoria de pantalla. En este caso, esta memoria es direccionada mediante la información del bus de direcciones de la CPU, la cual puede ahora realizar operaciones de escritura o lectura sobre la memoria RAM de pantalla.
- ❑ Existen tres métodos para coordinar los accesos a la memoria de pantalla por parte de la CPU, y el acceso por parte de los circuitos de refresco. Un método consiste en dar prioridad total a la CPU frente a los circuitos de refresco. Cuando la CPU necesita cambiar la información, accede a la memoria de pantalla y modifica los datos, parando el proceso de refresco. Con este método pueden aparecer transitorios luminosos en la pantalla.

Otro método consiste en diseñar el sistema de forma compacta con una relación íntima entre CPU y sistema de video, haciendo que el reloj sea el mismo para la CPU y el controlador. En un sistema de este tipo se puede sincronizar los accesos de refresco con los de actualización de la información.

El tercer método utiliza los periodos en los que el barrido se realiza en las zonas no visibles de la pantalla para realizar la actualización de la información. La CPU retarda los accesos a la memoria de pantalla hasta que el haz no es visible. Este método es el utilizado en los PC-s.

SISTEMAS GRÁFICOS.

- ❑ En los sistemas gráficos la información que se presenta en pantalla no está previamente definida y guardada en una memoria ROM generadora de caracteres, con una limitación en cuanto al número de formas a representar. En estos sistemas, las formas que se pueden representar no están limitadas ya que son generadas por software y por lo tanto dependen en su mayor parte de la capacidad de diseño del programador.
- ❑ En estos sistemas el contenido de la memoria de pantalla con la información oportuna en cada momento, se lleva directamente al registro de desplazamiento que genera la señal de video. En la memoria se guarda la información que en forma binaria determina que píxel debe estar encendido o apagado (caso monocromo, B/N) o el color que debe tener (caso RGB).
- ❑ La memoria en estos sistemas es como un mapa de la pantalla. Una posición determinada en esta última, tiene asignada una posición específica en la memoria de video. En un sistema gráfico, no tiene sentido hacer una división de la pantalla en filas y columnas, como en el modo texto, pero sí se mantiene cierta analogía para el caso de presentación en modo gráfico en B/N. En estos casos, se hace una división de la pantalla en forma vertical en bandas (columnas) que tienen una anchura en píxeles, igual a la capacidad en bits de una posición de memoria. Cada píxel tiene asociado un BIT de esa posición de memoria.

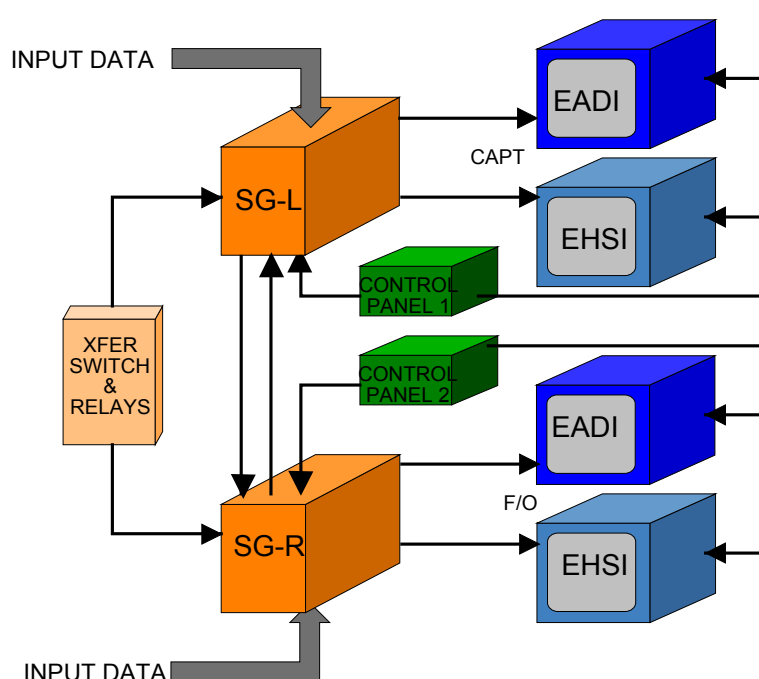
- Para representar formas en colores, es necesario dedicar más de un BIT para dar información a cada uno de los píxeles (además, por supuesto, de presentar las formas en una pantalla de color). Si cada píxel tiene asignados 2 bits de la memoria, cada píxel podrá tener hasta cuatro colores (blanco, negro y dos más). Si cada píxel tiene asignado 4 bits de la memoria, cada uno podrá tener hasta 16 colores, etc. La memoria de pantalla debe tener la capacidad suficiente para albergar esta cantidad de bits. Por ejemplo una pantalla con una resolución de 640 X 480 píxeles y 256 colores necesitará una memoria con una capacidad de:

$$640 \times 480 = 307200 \text{ píxeles} \times 8 \text{ bits/píxel} = 2457600 \text{ bits} = 300 \text{ KB.}$$

- Para aportar información a cada píxel, deben ser leídos de forma simultánea todos los bits correspondientes a ese píxel. Esta información es aplicada a los circuitos de video, los cuales generan las señales (RGB) correspondientes, que se aplican a través de un DAC a cada una de las entradas de color del monitor. Teóricamente se pueden representar un número infinito de colores al ser modulado cada haz (RGB) con una señal analógica.
- Si en cada representación que aparece en la pantalla, existen formas, que siempre han de estar presentes y que no cambian nunca, como puede ser la representación de un indicador por aguja, en la cual lo único que cambia es la posición de dicha aguja, estas formas fijas no son generadas por software de forma repetida. La información pertinente para que estas figuras aparezcan tal y como las vemos en pantalla está almacenada en una memoria ROM llamada **GENERADOR DE SÍMBOLOS**. En estos casos, la información que vemos en la pantalla, es una combinación OR adecuadamente ordenada de la información que la CPU manda a la memoria de video y la información que aporta el generador de símbolos.

EJEMPLO. EFIS.

El siguiente diagrama de bloques representa una visión general de un sistema de presentación basado en CRT para representar la actitud y situación horizontal de la aeronave. Este sistema sustituye a los indicadores mecánicos (ADI y HSI) que realizaban la misma función. El sistema se denomina **EFIS** (*Electronic Flight Instrument System*) y está compuesto por los siguientes elementos:



-Dos monitores con funciones **EADI** (*Electronic Attitude Director Indicator*) y dos monitores con funciones **EHSI** (*Electronic Horizontal Situation Indicator*). La distribución en cabina es un EADI y un EHSI frente al piloto, y los otros dos monitores del mismo tipo y funciones, enfrente del copiloto. Estos monitores tendrán una pantalla basada en CRT o en LCD. Para componer la imagen, pueden usar una combinación de sistema raster (256 líneas X 512 píxeles /línea X 40 Hz) para ciertas imágenes, y sistema vectorizado (80 Hz) para ciertos símbolos, dentro de la misma representación.

-Dos **SG-s** (*Symbol Generators*) que son las unidades encargadas de procesar la información que les llega procedente de los diferentes sistemas de la aeronave (ARINC 429, discretos, señales analógicas), así como de los paneles de control del EFIS y de la unidad XFER Switch, para poder enviar las señales de video y sincronismo correspondientes, a los monitores.

Dos **EFIS Control Panels**, que son los encargados de realizar cierto control sobre la información que se representa en los monitores (tipo de información a representar: ILS, VOR, NAV, DH..., rango o escala en que se representa, etc.), control manual de brillo (sensores de luz ambiental mantienen de forma automática el brillo en los monitores para adecuarlo al nivel aceptable para los pilotos)

Una unidad **XFER SWITCH** y los correspondientes relés donde se encuentran los circuitos necesarios para la realización de las correspondientes transferencias y enlaces entre los SG-s y los monitores según la combinación necesaria en cada caso, para mantener la correcta información para los pilotos. Las posibles combinaciones pueden realizarse de forma manual o automática (como consecuencia de un fallo en un SG, por ejemplo).

La situación en cabina de los displays del sistema básico y antiguo de EFIS, es la que se representa en la siguiente figura.



13.2-LCD (LIQUID CRYSTAL DISPLAY).

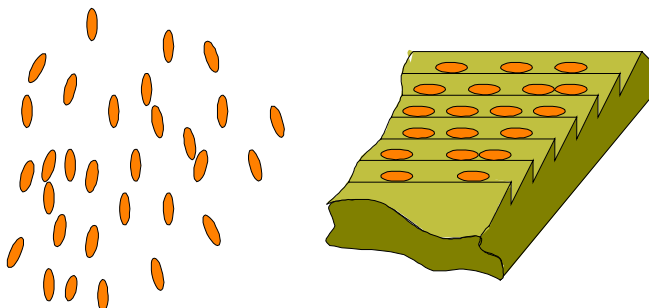
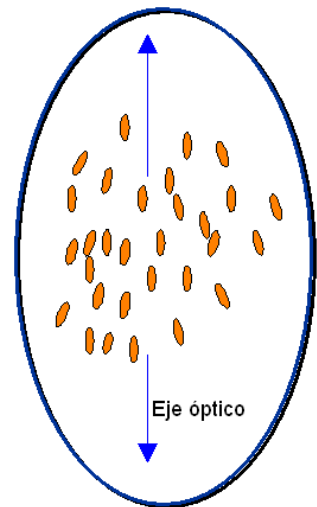
El estado “líquido cristalino” de la materia está situado entre el estado sólido de un cristal y el estado líquido de un líquido normal. Este efecto fue descubierto primeramente por Friedrich Reinitzer en 1888, dos años más tarde el físico Otto Lehmann denominó a este estado “líquido cristalino”. Este nuevo estado puede ser logrado de dos maneras: con temperaturas especiales o mediante la adición de solventes especiales.

La mayor parte de la sustancia de cristal líquido es compuesto orgánico consistente en moléculas largas parecidas a bastones. Poseen gran fluidez y elasticidad.

- ❑ Los dispositivos LCD, trabajan a partir de una señal de corriente alterna de bajo voltaje (comúnmente entre 3 y 15 v) y baja frecuencia (de 25 a 60 Hz). Necesitan muchísima menos corriente para su operación, que los dispositivos LED equivalentes.
- ❑ A diferencia también de los LED-s, los LCD-s no son fuentes de luz directa. Dependen de una luz externa para que puedan visualizarse las formas. En muchos casos la luz ambiental es suficiente para la aplicación. La luz requerida para la visualización, puede venir desde la parte frontal (luz ambiental) o desde la parte posterior del display.
- ❑ En la actualidad las tecnologías de LCD-s se pueden dividir en dos grupos según el tipo de material utilizado en los cristales, siendo estos esmécticos (utilizados en displays ferroeléctricos -FLCD) y nemáticos. Estos últimos son los mayoritariamente utilizados.

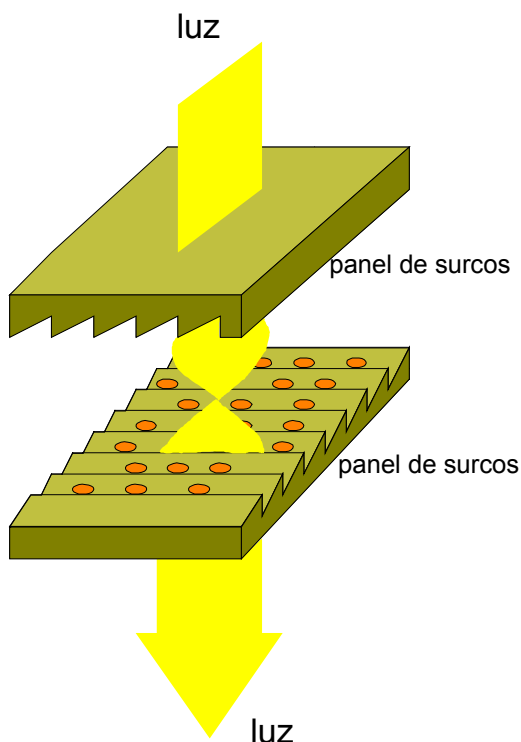
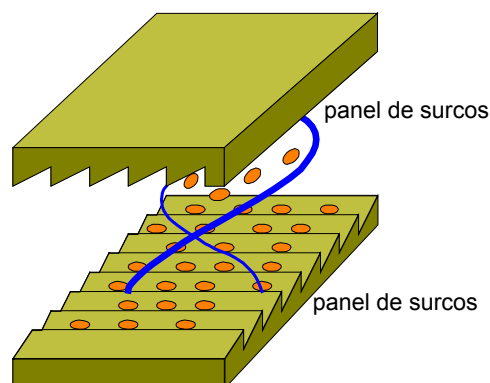
13.2.1-DISTRIBUCIÓN NEMÁTICA DE LOS CRISTALES.

- ❑ A temperatura ambiente, las moléculas cuyo tamaño es aproximadamente de 10 x 1 nm, penden alrededor de una dirección principal llamada eje óptico. Para cada muestra de material existen diferentes ejes ópticos.
- ❑ Aplicando un campo magnético, un campo eléctrico o una estructura mecánica (panel de surcos) se puede obtener una orientación uniforme de las moléculas. Debido a la característica de elasticidad del líquido, la orientación original vuelve cuando estas fuerzas externas dejan de actuar.
- ❑ Si las moléculas de cristal líquido se ponen en contacto con un panel de microsurcos de cierta dirección, estas moléculas cambian su posición y orden para ponerse paralelas a dichos microsurcos.



ESTRUCTURA NEMÁTICA RETORCIDA.

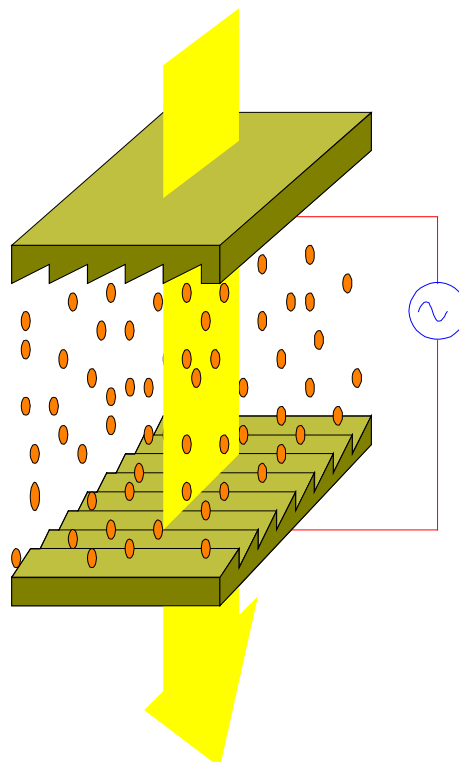
- Esta estructura se consigue emparedando el compuesto de cristal líquido entre dos paneles de microsurcos, cuya orientación relativa entre dichos microsurcos sea ortogonal.
- La estructura molecular que parte de un panel en una dirección, se irá “retorciendo” hasta acabar en la dirección de los microsurcos del otro panel.



- Si a través de la estructura se hace pasar luz, esta luz se irá “retorciendo” siguiendo la ordenación de las moléculas, hasta terminar en una dirección ortogonal de salida respecto a la que tenía de entrada.

- Si se aplica un voltaje a los paneles (electrodos), las moléculas se alinean verticalmente, siguiendo el campo eléctrico.

La luz atraviesa la estructura sin necesidad de giro o retorcimiento, siguiendo la dirección de orientación de las moléculas y pasa al exterior sin girar los 90° como en el caso anterior.

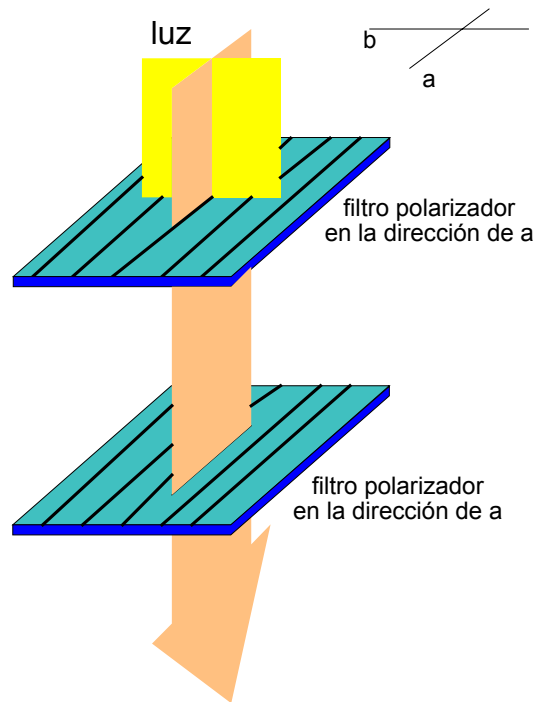


POLARIZACIÓN DE LA LUZ.

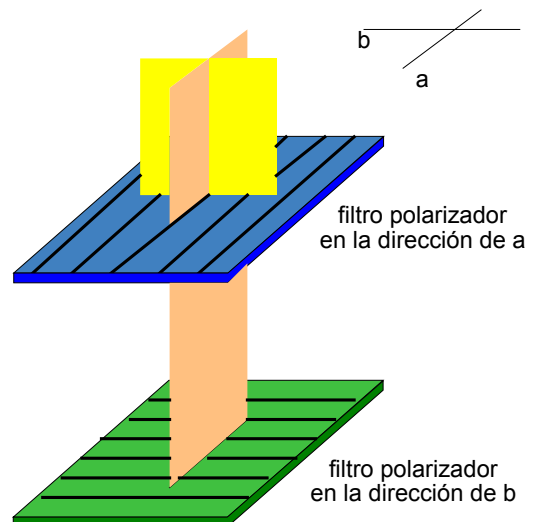
- ❑ La luz de cualquier lámpara normal, la luz del sol, etc. no está polarizada. Su forma de onda tiene una componente horizontal y una componente vertical.
- ❑ Un filtro polarizador de luz colocado delante de una lámpara, permitirá que una parte de la forma de onda atraviese el filtro, mientras que la otra componente sea totalmente reflejada.

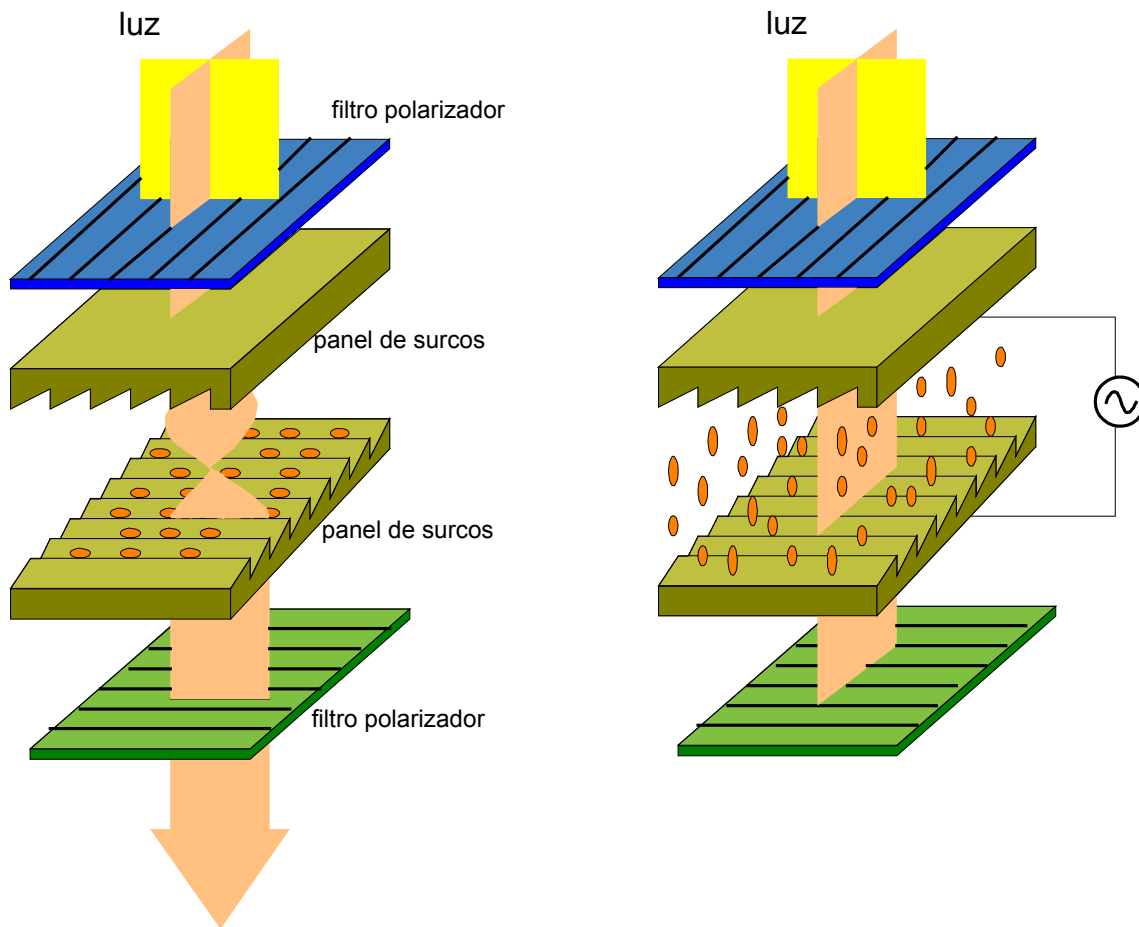
En la figura, los dos filtros están situados en la misma dirección de polarización. La componente de la luz correspondiente a esa dirección atravesará los dos filtros, mientras que la componente perpendicular a la dirección de polarización será reflejada.

- ❑ El efecto de polarizado disminuye la luz en un 50%, por lo tanto en un principio la luz que atraviesa el filtro simplemente parece ser más oscura.



- ❑ Si colocamos los filtros de tal manera que sus direcciones de polarizado estén situadas de forma ortogonal, uno de ellos polarizará la luz en una dirección, y el otro en la otra. El efecto será que un filtro reflejará una componente de la onda y el otro reflejará la componente ortogonal a la anterior, por lo tanto, no habrá luz después de los filtros.



NEMÁTICA RETORCIDA CON FILTROS POLARIZANTES.

- ❑ Si suponemos una estructura (célula de cristal líquido) como la de la figura de la izquierda, el primer filtro polarizador dejará pasar una componente de la onda de luz.

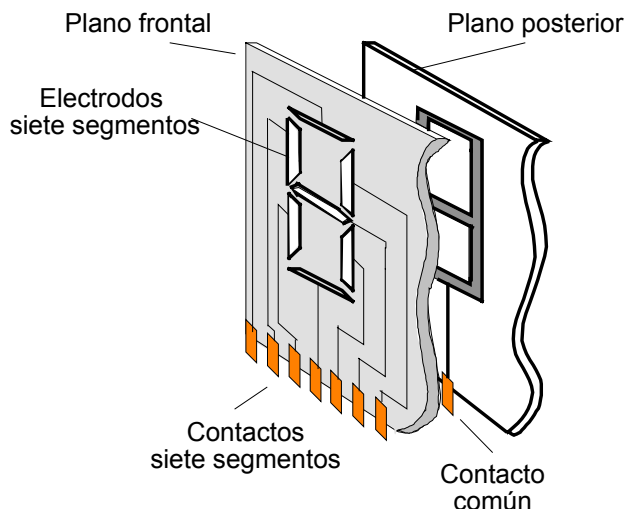
Esa componente, al pasar por el "sándwich" de los paneles de microsurcos, será retorcida, con lo cual la primitiva componente de la onda habrá girado 90° .

Un filtro polarizador en la dirección de esta última componente no afectará a la onda, por lo que la luz (parte) atravesará la estructura.

Si se aplica un voltaje a los electrodos (Fig. derecha), las moléculas en su nueva orientación no retorcerán la luz 90° , con lo cual ésta pasará a través de la célula directamente hasta el segundo filtro polarizador, el cual la bloqueará totalmente.

Regulando convenientemente el valor eficaz del voltaje aplicado, hará que la luz que atraviesa la estructura sea mayor o menor.

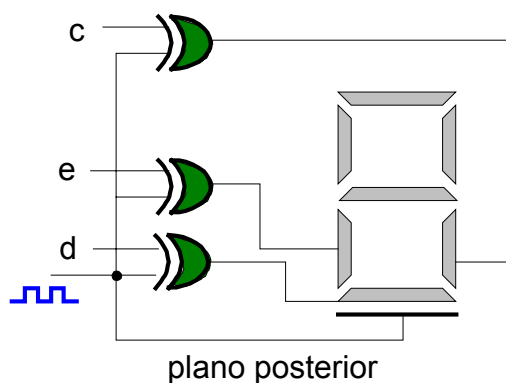
13.2.2-TIPOS.

NEMÁTICO DE DIRECCIONAMIENTO DIRECTO.

- ❑ Cada elemento en el dígito o carácter es direccionado directamente por el chip controlador.
- ❑ El material del plano frontal o superior está segmentado con arreglo al patrón de siete segmentos, cada uno de los cuales forma un electrodo transparente.
- ❑ En el plano posterior existe una imagen del dígito sin segmentar que forma el otro electrodo, también transparente.

Entre los dos planos (5 μ m) está el compuesto de cristal líquido.

- ❑ Un segmento se activa cuando se aplica un voltaje de AC entre ese segmento y el contacto del plano posterior, común a todos los segmentos. Este voltaje controla la orientación física de las moléculas comprendidas entre los dos electrodos. El segmento y el plano posterior forman como un condensador por lo que la corriente que necesita la excitación de cada segmento se mantendrá baja mientras la frecuencia que se aplica entre ambos electrodos sea baja.
- ❑ Esta frecuencia está limitada en la parte inferior por el efecto de parpadeo a unos 25 Hz.
- ❑ La tensión variable que se necesita suele obtenerse a partir de una onda cuadrada desfasada aplicada entre el segmento y el plano posterior.



La forma práctica de aplicar esta onda cuadrada es la mostrada en la figura. A cada una de las puertas X-OR, se aplica la salida de un decodificador BCD-siete segmentos en una de sus entradas.

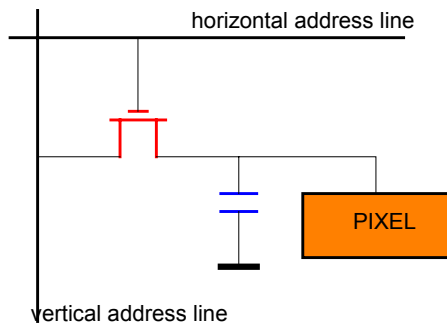
En la otra entrada, se aplica una señal cuadrada. De esta forma se obtiene una señal desfasada entre segmento y plano posterior, en la salida de aquellas puertas cuya entrada correspondiente al segmento, sea activa. Si esta entrada no es activa, las señales del segmento y plano posterior están en fase; de esta manera el segmento queda inactivo.

- ❑ La presentación en este tipo de LCD puede ser del tipo reflexiva o tipo transmisiva. En el primer caso, se produce la lectura del carácter por *segmentos oscuros*. La organización de las moléculas del cristal líquido, hace que la luz que atraviesa dicho cristal quede reflejada mediante un espejo posterior en aquellos segmentos que no han sido activados, siendo por lo tanto invisibles. Para los segmentos activados, la orientación de las moléculas es tal que la luz que entra es absorbida por un polarizador posterior, impidiendo el reflejo de la luz y por lo tanto creando los segmentos oscuros visibles en contraste con el fondo claro.

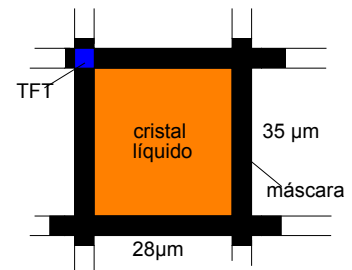
La forma transmisiva de visualización es por *segmentos iluminados*. Se proporciona una fuente de luz posterior y se dirige la luz hacia el observador. En este caso no existe espejo en la parte posterior. La formación del dígito es función de los segmentos activos. Se produce un dígito luminoso contra fondo oscuro. Si el segmento es activo, bloquea la luz y cuando no lo es deja pasar esta con lo que el segmento correspondiente queda iluminado.

NEMÁTICO DE MATRIZ ACTIVA DIRECCIONADA.

- Mediante este sistema de presentación, que pertenece a la forma transmisiva de

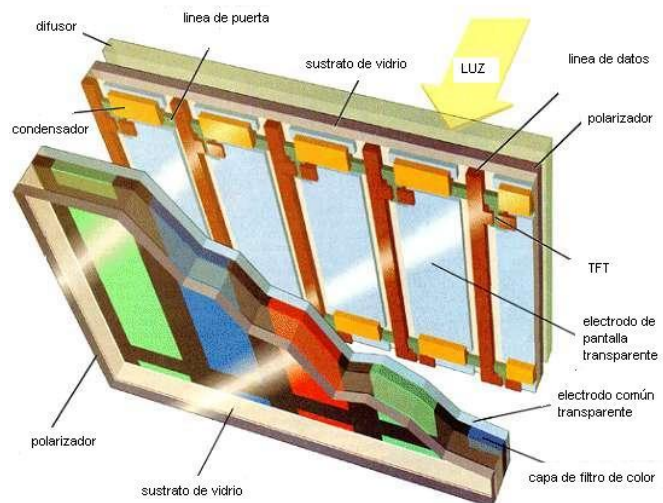


visualización, se pueden realizar tanto sistemas basados en segmentos como pantallas estructuradas según una matriz de miles de pixels. En este caso, cada una de las células nemáticas retorcidas, con sus correspondientes polarizadores de luz descritas anteriormente, sería un píxel (figura) monocromo. La forma de activar cada uno de los pixels es mediante un transistor extremadamente plano llamado TFT (Thin Film Transistor). Este transistor, va a permitir el paso de carga eléctrica cuando sea direccionado mediante la línea horizontal y vertical. Según la cantidad de carga aportada por la línea vertical, bloqueará mas o menos la luz que ha de incidir en la pantalla, mostrando mas o menos luz para su píxel correspondiente y por lo tanto, mostrando el píxel en escala de grises.



- Para una pantalla color, el principio es el mismo que para los CRT (excitado de los tres colores primarios), pero en este caso no existen haces para excitar las triadas.

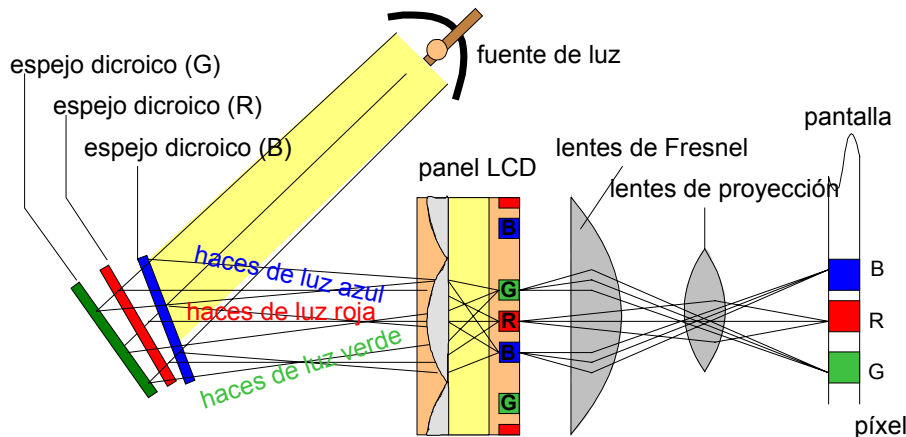
Para formar la imagen en la pantalla, cada uno de los pixels debe ser activado o desactivado de forma independiente mediante los tres TFT-s correspondientes, uno para cada color primario. El TFT pasa a ON cuando los electrodos horizontal y vertical correspondientes a un transistor, son activos al mismo tiempo. La imagen en la pantalla se forma mediante el aporte de luz a cada uno de los colores de cada píxel.



13.3-CAÑÓN PROYECTOR DE IMAGEN.

- Existen básicamente dos tecnologías usadas para el diseño de proyectores de video, la **LCD** (Liquid Crystal Display) y la surgida posterior a esta, llamada **DLP** (Digital Light Processing) de Texas Instruments.

13.3.1-PROYECTORES LCD.

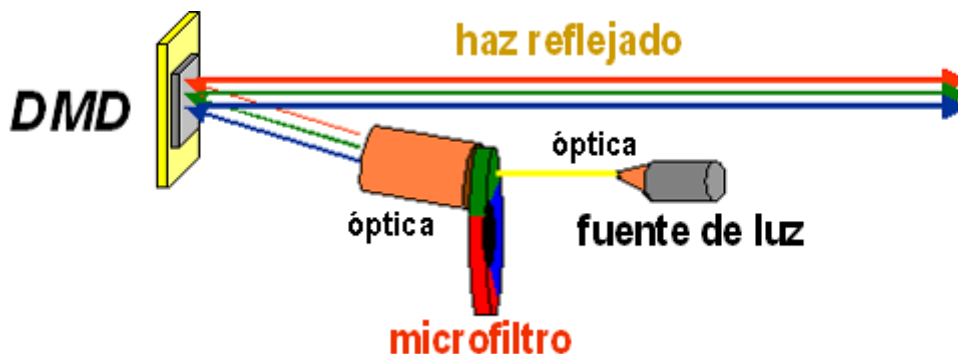


En estos proyectores la luz procedente de una lámpara de descarga de alta intensidad, es dividida en tres haces y conducida por tres caminos diferentes (rojo, verde y azul), cada uno de los cuales va a través de su propio array de LCD, el cual modulara con la información requerida cada uno de los haces,

haciendo que pase mas o menos luz del color correspondiente para formar el color final deseado. Posteriormente la luz es recombinada mediante un prisma antes de alcanzar la lente de salida. En este tipo de presentación no se necesita filtro de color, pues los tres colores son separados previamente de forma óptica.

13.3.2-PROYECTORES DLP.

Estos proyectores se basan en un chip llamado Digital Micromirror Device (DMD), que tiene su superficie cubierta con miles de espejos microscópicos. Cada uno de estos espejos tiene la capacidad mediante un pequeño movimiento gobernado de forma digital, de reflejar o bloquear la luz, dependiendo del color del píxel correspondiente. De la frecuencia con la que refleje o bloquee la luz dirigida hacia el píxel, va a depender la luminosidad de ese píxel.



La luz procedente de la lámpara, pasa a través de una rueda o microfiltro, que tiene filtros con los colores rojo, verde, azul. Esta rueda está situada entre la lámpara y el chip DMD y gira a 7200 rpm. La luz pasa esta rueda alternativamente roja, verde y azul, alcanzando al DMD. Los microespejos del chip se encargan de desviar o encauzar la luz que les llega hacia las lentes, dependiendo de la cantidad de color que necesita un píxel en un momento dado. En versiones mas sofisticadas de proyectores DLP, se utilizan tres chips DMD (uno para cada color), en lugar de la rueda de color.

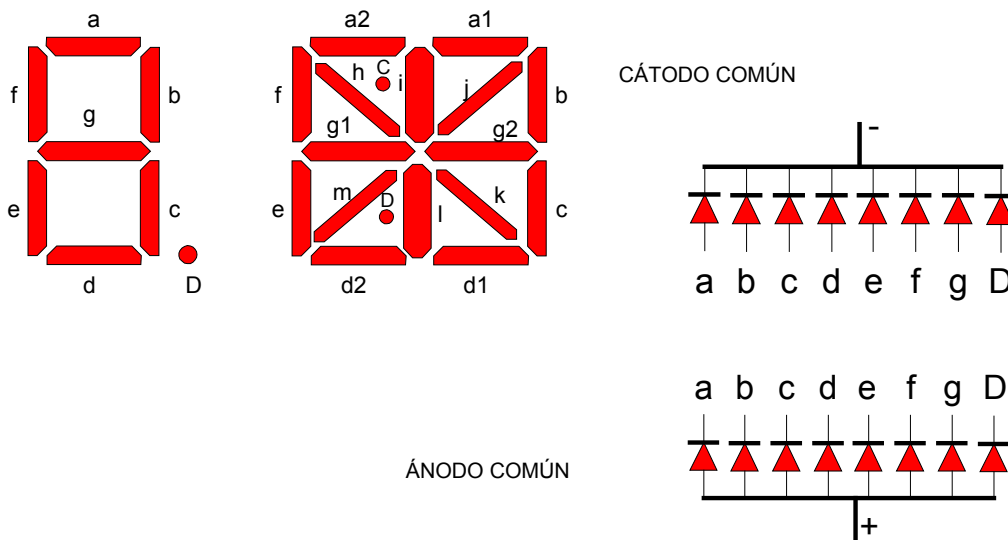
13.4-LED (LIGHT EMITTING DIODE) DISPLAYS.

Los displays basados en LED-s, utilizan un conjunto de diodos tipo LED ordenados de forma determinada, que son seleccionados de forma individual por grupos para formar diferentes caracteres alfanuméricos.

Existen dos tipos de displays LED: tipo segmentado y matriz de puntos. Cada uno de estos displays puede estar configurado con conexión ánodo común o cátodo común.

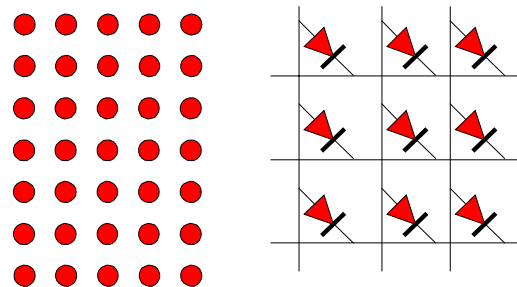
13.4.1-DISPLAYS SEGMENTADOS.

- Las configuraciones pueden ser de 7 segmentos para la presentación de números y algunas letras, y 16 segmentos para la presentación de números y letras. Tienen, además, uno o dos segmentos para la representación de uno o dos puntos separadores.



13.4.2-DISPLAYS DE MATRIZ DE PUNTOS.

- Estos displays están formados por una matriz de diodos LED que pueden ser direccionados de forma individual. Según la configuración de la matriz y el número de diodos (puntos) se pueden representar caracteres alfanuméricos con más o menos complejidad.



13.4.3-UTILIZACIÓN EN UN SISTEMA.

DISPLAYS 7-s. CONEXIÓN BÁSICA.

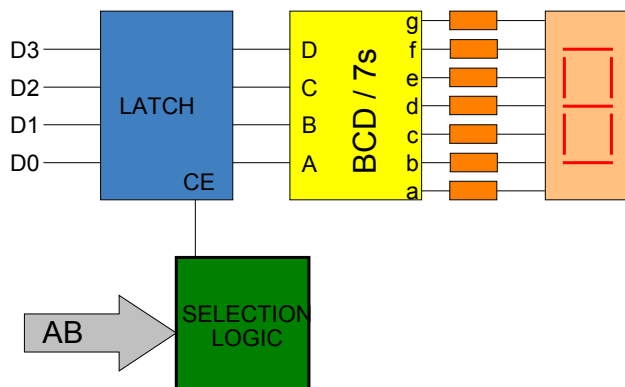
- Para la utilización de los displays de 7 segmentos la primera precaución a tener en cuenta es la de colocar la resistencia limitadora correspondiente, según la tensión a la que vaya a estar sometido cada uno de los segmentos (diodos LED).
- En muchos casos el circuito de control, no podrá entregar al display la corriente necesaria para poder excitar de forma conveniente los segmentos. En estos casos tendremos que colocar un driver entre circuito de control y display.

- La conexión más común de un display de 7-segmentos, y a la que hay que recurrir cuando se quieren presentar números decimales codificados en binario en sistemas cableados, es a través de un decodificador BCD 7-segmentos. Este decodificador se encarga de convertir cualquier combinación de entrada dada en código BCD (Binary Coded Decimal) a una combinación de salida que excitará el segmento correspondiente para que en el display sea visualizado el número decimal correspondiente al código BCD dado.

Hay que prestar atención al tipo de display utilizado (ánodo o cátodo común) para la elección del decodificador, o teniendo el decodificador ya definido, para conectar el display apropiado.

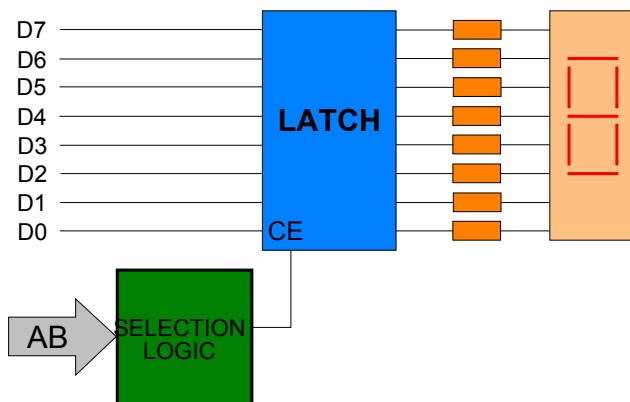
Un display de ánodo común es necesario que sea excitado con un decodificador BCD/ 7 –s con salidas activas a nivel bajo. Para un display de cátodo común, el decodificador debe tener salidas activas a nivel alto.

- El origen de la información BCD puede ser cualquiera: microswitches, counters, etc. o un sistema computador como en la figura. En este caso toda esta información BCD, será generada por programa y puesta en las líneas del Data Bus de donde pasará a un latch cuando el programa lo decida, y desde este circuito, al decodificador BCD/7-s.



La información puesta en el latch permanecerá estable en el display aunque la entrada BCD del latch cambie (siempre y cuando el latch este inhabilitado, lógicamente)

- En principio, para el manejo de cada display se necesitan 4 líneas de DB. También se puede recurrir al multiplexado de la información, y sincronizado del proceso, utilizando de esta manera siempre las mismas cuatro líneas.



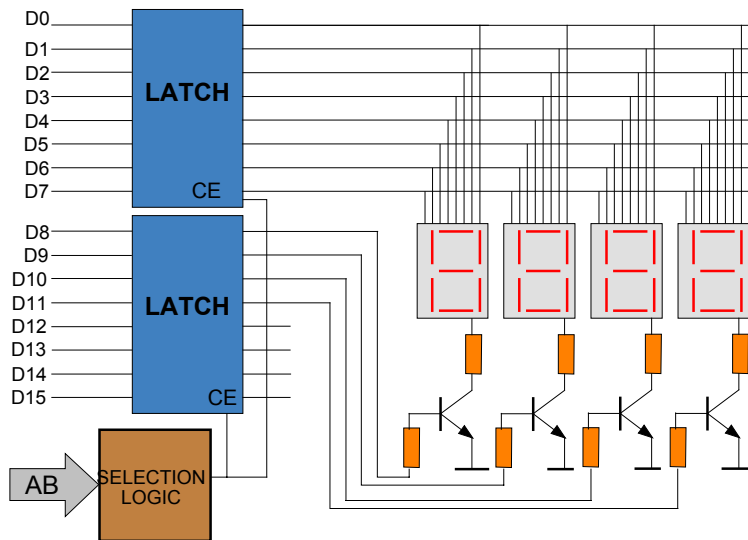
- El proceso de la decodificación del valor BCD (o cualquier otro) a 7-s lo puede realizar directamente el sistema computador, con lo cual se elimina el decodificador externo. Los segmentos que deben ser iluminados para cada valor concreto de un determinado código, se pueden obtener mediante programa según una rutina adecuada, o estar guardados en una memoria ROM (Look-up table) que entrega dichos valores según requerimientos.

DISPLAYS 7-s. MULTIPLEXADOS.

- En esta forma de trabajo, la información que se envía a los displays es la misma para todos ellos (4 de cátodo común en este ejemplo de la figura siguiente) y es entregada mediante las 8 líneas (una para el punto) menos significativas de un puerto paralelo de 16 bits. Las siguientes cuatro líneas del puerto (D8, D9, D10 y D11) se van a encargar de excitar en los momentos precisos uno de los cuatro transistores, que cuando estén en saturación conectarán el cátodo correspondiente de cada display a masa a través de una resistencia limitadora.

El proceso de trabajo es el siguiente:

Supongamos que se quiere representar el número 1961 en los cuatro displays. El sistema, mediante programa, carga en el latch correspondiente a las 8 líneas menos significativas del DB el valor para que se iluminen los segmentos b y c del display menos significativo (el de la derecha). También pone un nivel alto en la línea del DB, D11. Cuando mediante la lógica de selección, los latches sean activados, el valor correspondiente al 1 se entregará a todos los displays, pero solamente el display de la derecha, que tendrá su cátodo a masa, representará el número 1.



Para representar el número 6 en el segundo display comenzando desde la derecha, el programa coloca en las 8 líneas menos significativas del DB los valores correspondientes que excitarán los segmentos a, f, e, d, c y g. También pone un nivel alto en la línea D10 (y pone un nivel bajo en la D11). Cuando se direccionen los latches, el valor correspondiente al 6 se entregará a todos los displays, pero solamente aquel cuyo cátodo esté

conectado a masa a través del transistor en saturación, presentara dicho número. En este caso el segundo transistor comenzando por la derecha.

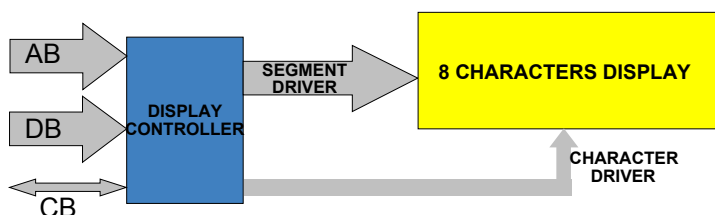
El proceso se repite para todos y cada uno de los dígitos.

Para conseguir que el número completo sea visible y a la vez estable, este proceso explicado debe realizarse a una velocidad tal que el ojo humano no perciba que los displays están continuamente encendiéndose y apagándose. Con una frecuencia de encendido-apagado de cada display de 50 veces por segundo, es suficiente.

- ❑ Existen en el mercado IC-s que realizan la misma función de multiplexado de la información en los displays para aplicaciones cableadas.

DISPLAYS 16-s.

- ❑ Generalmente este tipo de displays vienen agrupados en paneles de 4 a 8 dígitos. Estos paneles pueden ser a su vez agrupados para conseguir el número requerido de caracteres.
- ❑ El control de estos paneles suele realizarse mediante circuitos específicos (controlador de display).

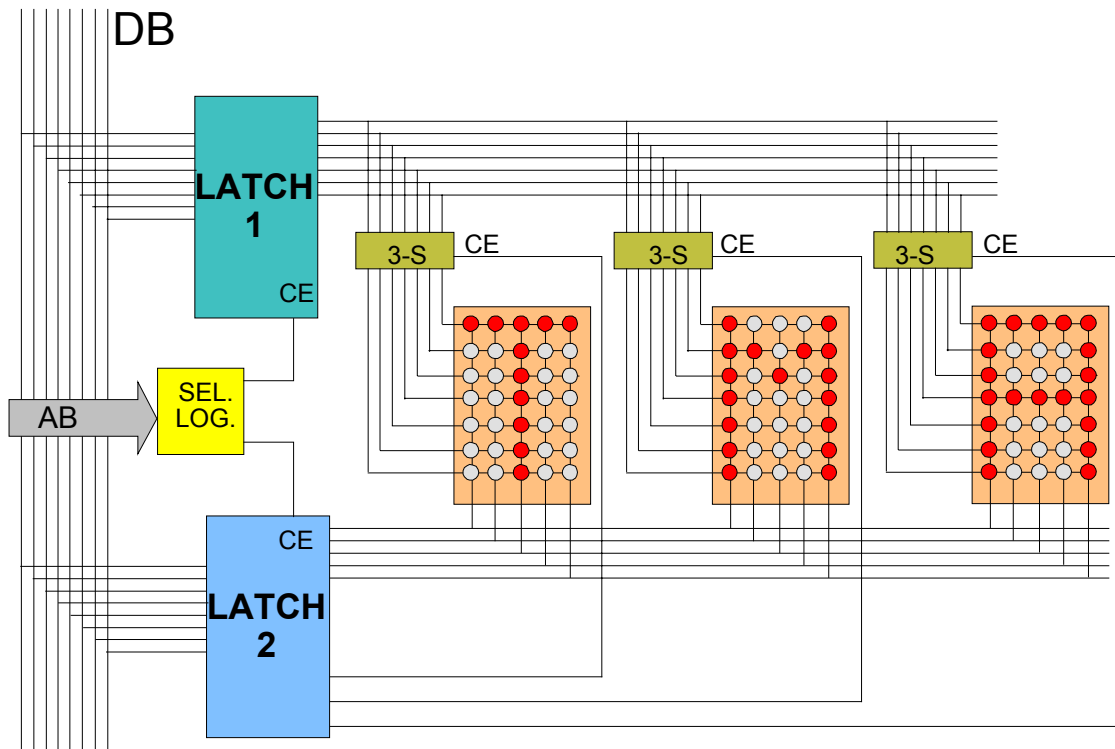


Estos controladores permiten el uso no sólo de caracteres alfanuméricos ASCII, si no también de códigos que sirven para mover un cursor por los caracteres, borrarlos, etc.

- El proceso de presentación es parecido al explicado en el ejemplo anterior. La información se va presentando de forma multiplexada.

DISPLAYS DE MATRIZ DE PUNTOS.

- En este tipo de displays, las columnas se van iluminando de una en una. Cuando el sistema se implementa con más de un display, las columnas de cada uno de ellos se conectan para el trabajo en paralelo, es decir, la misma columna de cada uno de los displays es activada al mismo tiempo.



- El sistema computador pone en el latch 1 la información de línea que necesita la columna que es gobernada por el latch 2. En este esquema de funcionamiento, la información para la línea y para la columna no es simultánea, sino que, se multiplexa en el tiempo. Primeramente se pone en el bus de datos la información para la fila que queda en el latch1. En el siguiente paso se pone en el bus de datos a través del latch 2, la información para la columna y a la vez para el display correspondiente a través de los buffers triestado. De esta manera ya se tiene toda la información necesaria para una columna de un display. Este proceso se repite para cada una de las columnas de cada uno de los displays.
- Se puede ver que en este caso también se trata de un proceso de multiplexado, por lo que requiere la velocidad apropiada para que la información parezca estable al ojo humano.



Software

14

14.-SOFTWARE.

Como ya hemos visto anteriormente la diferencia fundamental entre un sistema cableado y un sistema programado, está en el hecho de que el hardware (parte mecánica o tangible del equipo) utilizado en el sistema programado puede ser utilizado para realizar diferentes funciones, funciones, que van a depender de una serie de instrucciones que componen el programa (software de programa). En un sistema cableado, el cambio de función del mismo, lleva consigo un cambio en la estructura física que en algunos casos origina un sistema completamente diferente al originario.

14.1-EVOLUCIÓN.

Aunque pueda parecer que la “revolución electrónica” es algo tan cercana en el tiempo como puede ser la última década del siglo pasado, la realidad es que varias décadas antes, ya se estaba produciendo esa revolución, sobre todo en el entorno aeronáutico.

El primer sistema electrónico de control incorporado a la aviación civil, data de 1956. Se trataba de un sistema que controlaba actuadores hidromecánicos para el ajuste del flujo de combustible, entradas de aire, etc. en los motores. Anteriormente, todo este control se realizaba de forma puramente hidromecánica. Este sistema de control, era un sistema totalmente analógico. En 1976, se incorporó la electrónica digital como un subsistema para control, de ciertos parámetros de los motores del famoso Concorde. En 1984 apareció el primer motor que utilizaba electrónica digital para controlar sus funciones. Este motor, Pratt and Whitney, se incorporó en un B757. El Concorde (1976), fue también el primer avión civil en utilizar electrónica analógica para gobierno de los controles de vuelo alimentados hidráulicamente. La electrónica digital para este fin entró en servicio con el A310, en el cual, los flaps, slats y spoilers eran controlados por computadores digitales. Mas tarde, en 1988, la tecnología digital, se extendió también a los controles primarios de vuelo en el A320.

La primera noticia que se tiene del uso de software de abordo por un operador civil, data de 1968. En aquel año, un INS (Inertial Navigation System) montado en un B707 poseía un software cuyo peso era de 4KB.

14.2-SOFTWARE STANDARDS. DO-178B

El desarrollo de software para sistemas aviónicos cumple con el estándar **DO-178B** establecido por RTCA (Radio Technical Comisión for Aeronautics) o con estándares equivalentes como **ED-12B** de EUROCAE (para Europa). DO-178B “Software Considerations in Airborne Systems and Equipment Certification”, describe los requerimientos de certificación relacionados con cinco niveles de software. Estos niveles de software se corresponden con la forma en la que el software contribuye a las condiciones de fallo de la aeronave según la FAA, organización que acepta el uso del estándar DO-178B como un medio de certificación del software de sistemas aviónicos.

DO-178B / ED-12B

Los niveles de software mencionados anteriormente están clasificados por sus efectos en la aeronave, tripulación y pasajeros.

NIVEL A (CATASTRÓFICO): Condición de fallo que va a impedir la continuidad de un vuelo y aterrizaje seguros.

NIVEL B (PELIGROSO / GRAVE-IMPORTANTE): Condición de fallo que va a reducir la capacidad de la aeronave o la habilidad de la tripulación para afrontar condiciones adversas de operación, hasta un punto tal que pueda provocar:

- Gran reducción en los márgenes de seguridad o capacidad funcional.
- Dificultades físicas o aumento de la carga de trabajo de tal manera que la tripulación no pueda afrontarlo para desarrollar sus tareas de forma precisa y completa.
- Efectos adversos en ocupantes tales como daños serios a un número relativamente bajo de los ocupantes.

NIVEL C (IMPORTANTE): Condición de fallo que va a reducir la capacidad de la aeronave o la habilidad de la tripulación para afrontar condiciones adversas de operación, hasta un punto tal que pueda provocar por ejemplo, una reducción significativa en los márgenes de seguridad o capacidades funcionales, un incremento en la carga de trabajo de la tripulación o condiciones que afecten a la eficiencia de la tripulación o comodidad para los pasajeros.

NIVEL D (NO IMPORTANTE): Condición de fallo que no va a reducir de forma significativa la seguridad de la aeronave, y que conlleva acciones de la tripulación que están dentro de sus capacidades normales de trabajo. Las condiciones de fallo no importantes; puede incluir por ejemplo, una leve reducción en los márgenes de seguridad o capacidad funcional, un incremento leve de la carga de trabajo de la tripulación como puede ser un cambio en los planes de vuelo o alguna incomodidad a los ocupantes.

NIVEL E (SIN EFECTO): Condición de fallo que no afecta a la capacidad operativa de la aeronave o incrementa la carga de trabajo de la tripulación.

14.2.1-PROCESOS.

El desarrollo de software para ser certificado según D0-178B, exige una serie de procesos que generan una mayor o menor cantidad de documentos y registros, en función del nivel al cual se quiere certificar dicho software. Estos procesos, llamados procesos principales, son según establece DO-178B:

Planificación. Este proceso, establece los diferentes planes de software, estándares, procedimientos, actividades y herramientas necesarias para desarrollar, verificar, controlar, asegurar y producir los datos del ciclo de vida del software. Se generan documentos tales como el Plan sobre aspectos de software para certificación (PSAC), Plan de desarrollo de software (SDP), Plan para verificación de software (SVP), Plan de garantía de calidad del software (SQAP), Requerimientos del sistema, etc.

Desarrollo. Este proceso se puede dividir en los siguientes subprocesos: requerimientos, diseño, código e integración.

Los documentos que genera este proceso son: Datos de requerimientos del software (SDR), Descripción del diseño del software (SDD), Código fuente y Código objeto ejecutable. Dependiendo del nivel del software, se requiere la trazabilidad desde los requerimientos del sistema hasta el código fuente o código ejecutable.

Verificación. Este proceso se compone generalmente de una combinación de inspecciones, demostraciones, repasos, análisis y pruebas. Se generan documentos tales como: Casos y procedimientos para la verificación del software (SVCP), Resultados de la

verificación del software (SVR). También conlleva requerimientos para herramientas de test y análisis de código.

Gestión de configuración. Los documentos que se generan en este proceso son: Índice de configuración del software (SCI), Índice de configuración de entorno del ciclo de vida del software (SECI). Este proceso, lleva a cabo informe de problemas, cambios y actividades relacionadas, además de proporcionar archivo e identificación de revisión de: el entorno de desarrollo del código fuente y otros entornos de desarrollo como herramientas de test y análisis, herramienta de integración del software, todos los demás documentos, software y hardware.

Seguridad de calidad. Los documentos que se generan son: Registros de seguridad de calidad del software (SQAR), Revisión de conformidad de software (SCR), Resumen de consecución de software (SAS). Este proceso lleva a cabo revisiones y auditorias para mostrar conformidad con D0-178B.

Enlace en la certificación. Este proceso es el vehículo que establece comunicación y entendimiento entre el candidato (el que busca la certificación) y la autoridad competente en la certificación. Típicamente es un representante de ingeniería designado (DER), trabajando para FAA o EASA en una compañía fabricante de aeronaves.

El proceso de la certificación D0-178B es mas demandado para los niveles altos. Un software certificado para Nivel A, tendrá el potencial más alto en el mercado, pero requerirá una tarea dura y una preparación meticulosa de los documentos requeridos en los procesos. Por el contrario, si el software se certifica para el Nivel E, requerirá menos trabajo, menos documentos, etc., pero el rango de aplicación de este producto estará mucho más limitado.

Desde la publicación de DO-178B en diciembre de 1992, ha habido numerosas peticiones a las agencias de certificación sobre aclaraciones al documento. Como consecuencia de dichas peticiones, el 12 de octubre de 2001 RTCA publicó el **DO-248B**, “*Annual Report for Clarification of DO-178B*”.

14.3-CAMBIOS EN EL SOFTWARE.

- a.- El candidato a la certificación, debe identificar los cambios de software que se incorporarán al producto y realizar un análisis de impacto de cambio. Este análisis debe seguir un proceso definido para determinar el impacto potencial del cambio en la seguridad operacional continua de la aeronave. Para los equipos autorizados por TSO (Technical Standard Order) el análisis debería identificar el entorno buscado de la aeronave objeto que forma la base del análisis. El análisis también proporciona una base para determinar el punto de implicación de la autoridad de certificación. En el análisis de impacto de cambio, se deberían tratar los siguientes puntos:

1.- El análisis de la trazabilidad identifica áreas que pueden ser afectadas por el cambio del software. Esto incluye el análisis de requerimientos afectados, diseño, arquitectura, análisis de código, procedimientos de pruebas y análisis de casos.

2.- El análisis de margen de memoria asegura que los requerimientos de reparto de memoria y los márgenes aceptables son mantenidos.

3.- El análisis de margen de temporización asegura que los requerimientos de tiempos, los requerimientos establecidos de tareas de la CPU, las características de mantenimiento de los recursos del sistema, requerimientos de los tiempos del interfaz y los márgenes de temporización, son mantenidos.

4.- El análisis del flujo de datos identifica cambios en el flujo de datos y acoplamiento entre componentes y asegura que no hay un impacto adverso.

5.- El análisis del control de flujo identifica cambios en el control de flujo y acoplamiento entre componentes y asegura que no hay un impacto adverso.

6.- El análisis de I/O asegura que los cambios no han impactado de forma adversa los requerimientos de entrada y salida (incluido el acceso a memoria, carga del bus y las interfaces hardware de entrada y salida) del producto.

7.- El análisis del entorno de desarrollo y proceso identifica cualquier cambio que puede afectar de forma adversa al producto como puede ser las opciones del compilador, versión, linker, ensamblador, etc.

8.- El análisis de las características operacionales evalúa que cambios (tales como cambios en el manejo de interrupciones y excepciones, validación de datos, límites) no resultan en efectos adversos.

9.- El análisis de los requerimientos de mantenimiento de certificación (CMR) determina si se necesitan nuevos o cambiados CMR-s por el cambio de software.

10.- El análisis de particionado asegura que los cambios no tienen efecto en ningún mecanismo de protección incorporado en el diseño.

- b.- El análisis de impacto de cambio debería determinar si el cambio puede afectar de forma adversa a la operación segura del producto o sistema. A continuación se indican una serie de áreas que pueden tener un impacto adverso en la seguridad u operación:

1.- Se cambia información relacionada con la seguridad.

2.- Cambios en características operacionales o procedimentales de la aeronave que pueden afectar de forma adversa la seguridad del vuelo.

3.- Se añaden nuevas funciones a las funciones existentes que pueden impactar de forma adversa a la seguridad del vuelo.

4.- Se cambian en el entorno componentes hardware de tal manera que pueden afectar a la seguridad.

5.- Se cambian de forma significativa los datos del ciclo de vida del software (requerimientos, código y arquitectura) de tal manera que puede afectar de forma adversa la seguridad.

- c.- Adicionalmente, los siguientes aspectos se deberían identificar en el análisis de impacto del cambio:

1.-Actualizaciones necesitadas para asegurar que los cambios de software se incorporan en los datos del ciclo de vida del software apropiado, incluyendo requerimientos, diseño, arquitectura, código fuente y código objeto y trazabilidad.

2. Actividades de verificación necesitadas para verificar los cambios y que no hay efectos adversos en el sistema. El análisis de impacto de cambio debería cubrir como se

verificarían los cambios que pueden afectar de forma adversa la operación del sistema o aeronave, de tal manera que el software cambiado, y el no cambiado continuará cumpliendo los requerimientos para una operación segura. Estas actividades de verificación pueden incluir revisiones, análisis, pruebas de vuelo, etc., incluyendo la reevaluación de análisis ya existentes, reejecución de pruebas ya realizadas y nuevos procedimientos de test.

14.4-REDUNDANCIA.

En las aplicaciones de software utilizadas en aeronaves civiles, se puede ver como la realización de funciones iguales es llevada a cabo por caminos diferentes en cuanto al software. Se habla entonces del término REDUNDANCIA DISIMILAR. El software que realiza la función por un camino determinado, generalmente es escrito por un equipo diferente, al software que la realiza por el otro camino, incluso el hardware es diferente, utilizando también diferentes variedades de microprocesador. Los resultados del procesamiento por uno y otro camino son comparados, y en caso de que existan diferencias, el subsistema se desconecta.

Este método se utiliza como una precaución adicional, donde el mal funcionamiento de un sistema puede causar consecuencias inaceptables, pero donde otros sistemas de respaldo (back up) están disponibles.

14.5-CONTROL DEL SOFTWARE DE PROGRAMA.

- ❑ Los sistemas cableados se han estado utilizando en las aeronaves durante muchos años bajo el nombre de *computadores analógicos*. Estos computadores basan su función en el análisis de niveles tensiones y corrientes de tipo analógico como variables de entrada, para generar unas tensiones y corrientes como variables de salida, que van a ser función de las anteriores. El sistema entero está implementado utilizando componentes tradicionalmente analógicos como transistores, amplificadores operacionales, sincros, servos, etc. Un cambio de función en uno de estos computadores, requiere generalmente una modificación sustancial de la estructura del mismo. De hecho, computadores analógicos que realizan exactamente la misma función en una aeronave y en otra con diferente configuración, son implementados de diferente manera, originando diferentes computadores.
- ❑ Actualmente, toda la parte de cómputo, análisis y procesamiento de variables, es realizada por los computadores digitales, que son, como hemos visto anteriormente, sistemas programados por una serie de instrucciones. Esto significa que un cambio de función en el computador para que este se adapte a una nueva configuración aeronave / motor, básicamente requiere un cambio en el software de programa. El control del software de programa permite por lo tanto que un mismo computador (LRU) pueda ser utilizado en diversas configuraciones.

14.6-MÉTODOS DE CONTROL DEL SOFTWARE DE PROGRAMA.

El acceso al software de programa instalado, va a depender de la función a la que esté destinado dicho software, de tal manera que, en unos casos, este se encontrará en memorias de programa tipo EAROM o Flash Memory, relativamente fáciles de alterar (**UMS** -User-Modifiable Software-). En otros casos, dicho software tendrá un acceso limitado y se encontrará guardado en forma de Firmware (inalterables) en memorias tipo PROM u OTP (One Time Programmable Rom). En cualquiera de los dos casos, la forma en que ese software de programa puede ser configurado para que trabaje de una forma o de otra, puede ser realizado mediante el método de *particionado de software* o mediante el método de *pinos programables*

14.6.1-PARTICIONADO DE SOFTWARE.

- Durante el proceso de diseño, el software y la memoria asociada se organizan teniendo en cuenta que se pueda proporcionar el aislamiento requerido entre las funciones que necesita cada instalación particular, de forma que el cambio de un modo de operación en una instalación no cause problemas en la otra instalación. Esto puede lograrse realizando la correspondiente división del software y su memoria entre los sistemas que deben ser operados. Cuando el tiempo es el apropiado para un cambio de software, el computador es enviado al laboratorio para realizar el tipo de modificación requerido.

El nuevo software es introducido en la memoria de programa del computador, memoria del tipo semiconductor implementada en la propia unidad y que suele ser del tipo EAROM (Electrically Alterable ROM), también llamada EEPROM (Electrically Erasable Programmable ROM).

14.6.2-PINES PROGRAMABLES.

- Este tipo de control no cambia el software con el que trabaja el sistema, sino que, el software se sirve de la configuración de estas señales para operar en la manera correspondiente. La programación de los pines se implementa a través de un conector o bloque de terminales externo a las LRU-s; Se realiza a nivel de cableado y mediante dos tipos diferentes de señales de entrada: (1) *por discretos* y (2) *por nivel de voltaje*.

1. DISCRETOS.

La mayoría de los sistemas tales como EICAS, EFIS, FMCS, IRS, etc. usa entradas de discretos para su control lógico. Estas señales están normalizadas para asegurar la compatibilidad de los equipos. Las entradas de discretos, pueden ser programadas de forma individual o en grupos, para determinar la combinación apropiada de ceros y unos. De esta manera se pueden seleccionar parámetros como configuración de aeronave / motor, límites, versiones de software, etc., disponibles para una LRU.

Las entradas de discretos correspondientes al "0" y "1" son generadas de acuerdo a las siguientes lógicas:

"1"		"0"	
<i>Voltaje estándar aplicado</i>	<i>Señal estándar abierta</i>	<i>Voltaje de 28 V DC / señal abierta</i>	<i>Señal estándar de masa</i>
28 V DC	OPEN	28 V DC/OPEN	GROUND
Señal nominal de 27,5 V DC. El rango de voltaje va desde 18,5 V hasta 31 V medido desde el pin hasta la masa de la aeronave.	La impedancia en un abierto es mayor de 100 K Ω , medida respecto a la masa de la aeronave.	Los criterios para este estado lógico pueden ser cualquiera de los anteriores.	Puesta a masa en un punto de la aeronave. Un potencial de 3,5 V o menos respecto a la masa de la aeronave, representa una condición de masa. O también, la impedancia de este pin respecto a la masa de la aeronave es de 10 Ω o menos

2. NIVELES DE VOLTAJE.

El control es realizado mediante una serie de niveles de voltaje, determinando cada nivel una condición de configuración.

- Los diferentes tipos de aplicación de la utilización de pines programables, caen dentro de una o más de las siguientes categorías:

*.-Source/Destination Identifier (SDI).

En la palabra ARINC-429, los bits 9 y 10, cuando son utilizados como SDI, codifican el número de una instalación en una aeronave. El SDI utiliza combinaciones de los discretos “ground” y “open” para definir la fuente y el destino de la señal en una instalación multisistema. Por ejemplo, en una instalación con tres EFIS, se le asigna a cada uno de ellos un código diferente entre las combinaciones 01 (left EFIS), 10 (right EFIS) o 11 (center EFIS) para que pueda ser identificado. Un equipo, reconocerá palabras que contengan su propio código de instalación y palabras que contengan el código 00, que es el código para “all call”

*.-Source selector switch.

La configuración de estos pines permite que un sistema seleccione mediante circuitos internos, entradas de señal entre diferentes fuentes. Esto hace que el equipo sea más versátil a la hora de poder adaptarlo a necesidades específicas de filosofía de funcionamiento de cada aerolínea.

*.- Air/Ground logic pin.

Este pin, está asignado a una lógica de aire / tierra para indicar cuando la aeronave está en el aire y cuando está en tierra. Normalmente, presenta un “open” cuando la aeronave está en el aire, y presenta un “ground” cuando la aeronave está en tierra.

*.-Performance Data Base.

Mediante la programación de los pines dedicados a este fin, se puede hacer que el software de programa tenga en cuenta para sus operaciones unas bases de datos u otras diferentes guardadas en la memoria interna. Por ejemplo, el FMCS guarda bases de datos que incluyen características de actuaciones aerodinámicas y de propulsión (valores máximos o extremos) para mas de una combinación de aeronave / motor. La base de datos que debe aplicarse para una configuración determinada, es seleccionada mediante pines programables.

*.-Option selections.

En algunos casos, el diseño del software de programa permite que la ejecución pueda ser variada (dentro de los márgenes de las normas de regulación aérea) de forma que se adapte a requerimientos que mejor convengan a una aerolínea en particular. Esta variedad de ejecución se realiza mediante la configuración de pines programables. Por ejemplo, en el sistema EFIS el modo MAP puede representar datos de navegación según una selección en particular, junto con la pantalla del radar meteorológico.

Como ejemplo de algunos parámetros que pueden ser programados mediante este método tenemos el SDI en una configuración multisistema de FMC FCC, etc., opción de Kg como unidad de medida, entradas de navegación referidas a norte magnético o verdadero, diferentes configuraciones aeronave / motor, etc.



Electromagnetic Compatibility

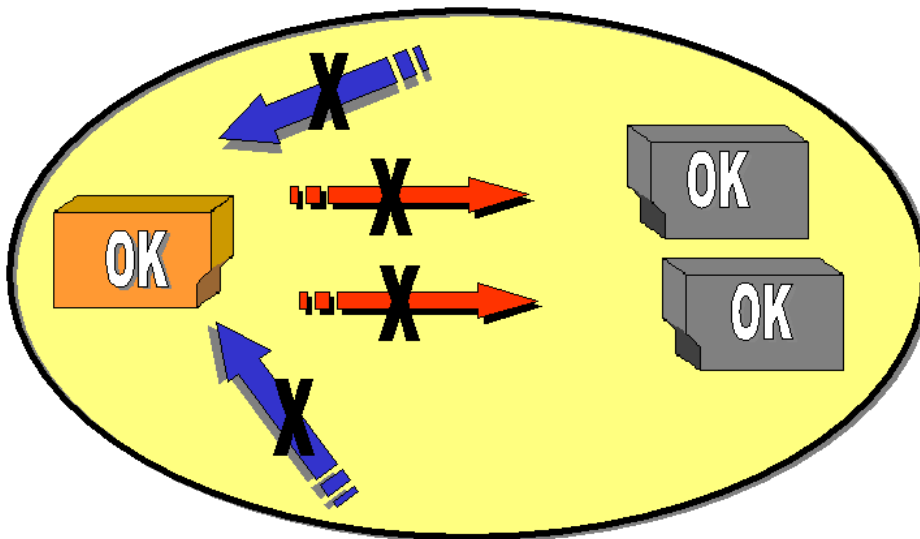
15

15.-ENTORNO ELECTROMAGNÉTICO.

El entorno electromagnético se puede definir como la combinación de la energía electromagnética, procedente de una fuente realizada por el hombre, o natural, a la cual un sistema o equipo será expuesto en la realización de su función. El entorno queda definido para un lugar y tiempo determinado.

15.1-ELECTROMAGNETIC COMPATIBILITY (EMC).

- La compatibilidad electromagnética (EMC) describe la capacidad de un dispositivo o sistema eléctrico o electrónico para funcionar de forma satisfactoria en su entorno electromagnético, sin provocar perturbaciones electromagnéticas sobre otros sistemas que trabajan en ese mismo entorno. También describe la capacidad de dichos sistemas para funcionar de forma correcta, sin ser afectados por el entorno electromagnético específico de trabajo.



Una EMC eficaz requiere que el sistema esté diseñado, fabricado y comprobado según el entorno electromagnético operativo determinado. La EMC es problemática especialmente, cuando muchos sistemas eléctricos y/o electrónicos coexisten en una

instalación muy compacta, como es el caso de los compartimientos de los sistemas aviónicos y en el cockpit en las aeronaves.

En la práctica, la definición de la EMC integra todo el espectro de frecuencias, desde corriente continua, hasta radiofrecuencia y microondas, incluyendo las vías de acople.

15.1.1-ANTECEDENTES.

- **RTCA** (Radio Technical Commission on Aeronautics) hoy en día RTCA Inc., tomando como base de partida las experiencias acumuladas en el campo militar, estableció después de la II Guerra Mundial una serie de estándares de actuación para los equipos aviónicos. En 1975 el comité de trabajo SC-123 (con la participación de **EUROCAE** -European Organization for Civil Aviation Electronics-) emitió el primer documento enfocado hacia este aspecto bajo la denominación DO-160, internacionalmente aceptado. En 1977 otro comité de trabajo, el SC-135, comenzó la ampliación de ese documento sobre todo en el área de test electrónicos, originando en 1981 el documento DO-160B. Este documento fue el de referencia por las autoridades de aviación civil para asegurar la compatibilidad electromagnética entre sistemas de abordo en transporte comercial.

La compatibilidad electromagnética fue considerada como un problema de entorno, por lo cual emisiones y susceptibilidad además de la reacción de los equipos aviónicos a sobrevoltajes, fallos de tensión e irregularidades en la generación de energía eléctrica en la aeronave, fueron partes integrantes de este documento.

En 1986 EUROCAE como parte de JAA (Joint Airworthiness Authority) y **SAE** (Society of Automotive Engineers) como parte de FAA (Federal Aviation Administration), fueron encargados, entre otras tareas, de definir el entorno electromagnético HIRF para operaciones de aeronaves civiles. EUROCAE formó el comité denominado *Grupo de trabajo 33* y SAE formó *AE4R* con representación de la industria y autoridades de aviación. La misión de estos comités fué:

- Definir el entorno electromagnético HIRF (High Intensity Radiated Field).
- Redactar el borrador de criterios de conformidad en forma de Advisory Circular (AC) por SAE, o en forma de Advisory Material Joint (AMJ) por EUROCAE, describiendo los caminos para demostrar compatibilidad en los sistemas de aeronaves.
- Redactar el borrador de una Guía de Usuario / Manual de Usuario con todos los detalles técnicos requeridos para llevar a cabo los AC / AMJ

Además, se procedió a la modificación del documento DO-160B, culminando con la aparición en 1989 del documento DO-160C (conocido como **ED-14C** en Europa), en el cual el tema HIRF y el tema relativo a los rayos (lightning) pasaron a formar parte del documento como extensiones EMI del propósito de compatibilidad electromagnética original.

En 1993, los procedimientos utilizados por la unión de la industria y la FAA, llevaron a la formación de un comité llamado **ARAC** (Aviation Rulemaking Advisory Committee), para ayudar a la creación de documentos relativos a reglas de aviación. En 1994 se creó un grupo de trabajo de ARAC llamado **EEHWG** (Electromagnetic Effects Harmonisation Working Group) cuya misión era la de crear de una forma concordante o armónica, la documentación que SAE y EUROCAE estaban produciendo.

EEHWG tomó el control sobre todas las actividades de SAE relacionadas con HIRF y fue encargado de la producción de un requerimiento HIRF común (XX.1317) y el análisis económico para llevarlo a cabo. Ambos, EUROCAE y SAE suspendieron los trabajos para la creación de AC / AMJ y Manual de Usuario / Guía de Usuario.

El 29 de julio de 1997 apareció la versión "D" del documento (**DO-160D / ED-14D**) en el cual se apreciaba un aumento en las especificaciones de los equipos relacionados con EMC. Los cambios reflejan una apreciación de la severidad del entorno HIRF.

DO-160D / ED-14D

En noviembre de 1998 ARAC EEHWG presentó el documento AC / AMJ 20.1317. Este documento incorpora información y referencias relativas a entornos HIRF externos así como la valoración, análisis, certificación y procedimientos de verificación relativos a sistemas eléctricos y electrónicos que proporcionan funciones para operaciones seguras de las aeronaves. Estos sistemas pueden ser sistemas de generación de energía eléctrica y su distribución, dispositivos electrónicos y electromecánicos, displays, cableado, etc.

En diciembre de 2000, apareció el cambio 1 y en junio de 2001 el cambio 2, del documento DO-160D / ED-14D, cambios que afectaban a diferentes secciones del mismo.

- En resumen, el documento DO-160D / ED-14D, define una serie de procedimientos tipo y condiciones mínimas de prueba ambientales para la comprobación de equipos electrónicos de abordo (aviónica). El propósito de estas pruebas es determinar el comportamiento de los equipos aviónicos en condiciones ambientales representativas de aquellas que pueden encontrarse en las operaciones de abordo.

El documento incluye 25 secciones y 3 apéndices; las secciones 15 hasta la 23 y la sección 25 cubren los temas relacionados con la EMC. Estos temas son:

Sección 15	Magnetic Effect.
Sección 16	Power Input.
Sección 17	Voltage Spike.
Sección 18	Audio Frequency Conducted Susceptibility- Power Inputs.
Sección 19	Induced Signal Susceptibility.
Sección 20	Radio Frequency Susceptibility (Radiated and Conducted).
Sección 21	Emission of Radio frequency Energy.
Sección 22	Lightning Induced Transient Susceptibility.
Sección 23	Lightning Direct Effects.
Sección 25	Electrostatic Discharge.

La finalidad de los tests que se realizan en cada una de las secciones, en forma muy resumida, es la siguiente

Sección 15. Magnetic Effect.

Este test se realiza para determinar hasta que punto el equipo bajo prueba (*equipment under test-EUT-*) va a desviar la aguja de una brújula. Se definen según el caso, 4 clases de equipos (A, B, C, y Z)

Sección 16. Power Input.

Este test se realiza para determinar si el EUT puede operar según los requerimientos durante todas las condiciones diferentes de variación de AC y DC que ocurren durante la operación de la aeronave (fluctuación del voltaje, de la frecuencia suministro de energía de emergencia, etc.). Quedan definidas 3 categorías de equipos (A, B y Z)

Sección 17. Voltage Spike.

Mediante este test, se determina si el EUT es capaz de operar como se espera durante o después de que transitorios de voltaje (impulsos de forma definida) sean aplicados a las entradas de la alimentación de AC y DC. Hay dos categorías de equipos (A y B)

Sección 18. Audio Frequency Conducted Susceptibility- Power Inputs.

El fin de este test es determinar si el EUT podrá operar como se requiere cuando se aplica una interferencia de audiofrecuencia a las entradas de alimentación AC y/o DC. Los niveles de las señales que se aplican pueden llegar a ser hasta el 8% del valor nominal del voltaje de alimentación, y en un margen de frecuencias que van desde 10 kHz hasta 150 kHz. Existen cuatro categorías de equipos (A, B, J y Z)

Sección 19. Induced Signal Susceptibility.

Se trata de determinar si el EUT puede operar como se requiere cuando este y los cables que lo interconectan son sometidos a campo magnético, campo eléctrico de audiofrecuencia y transitorios de voltaje. Existen cuatro categorías de equipos (A, B, C y Z)

Sección 20. Radio Frequency Susceptibility (Radiated and Conducted).

Se trata de determinar si el EUT puede operar como se requiere cuando este y los cables que lo interconectan son expuestos a una interferencia de RF. Se requiere señales de radiofrecuencia de tipo onda continua (CW), onda cuadrada (SW), AM y modulación por impulso (PM). La frecuencia para el test de Conducted Susceptibility va desde 10 KHz hasta los 400 MHz y para el test de Radiated Susceptibility va desde los 100 MHz hasta los 18 GHz. Existen 16 categorías de equipos para Conducted Susceptibility y 17 para Radiated Susceptibility. Estas categorías indican el nivel de severidad del test y/o el tipo de modulación usado.

Sección 21. Emission of Radio frequency Energy.

Se trata de determinar que el EUT no emite interferencias de RF mas allá de los límites especificados. Existen cuatro categorías de equipos (B, L, M y H)

Sección 22. Lightning Induced Transient Susceptibility.

El fin de este test es determinar si el EUT puede operar como se requiere durante o después de que varias señales transitorias inducidas por rayos sean inyectadas en los pines de los conectores, en los cables de interconexión, y en los terminales de alimentación.

Sección 23. Lightning Direct Effects.

El fin de este test es determinar si el EUT puede soportar los efectos directos de un rayo. Existen cinco categorías de equipos (1A, 1B, 2A, 2B y 3)

Sección 25. Electrostatic Discharge.

El fin de este test es determinar si el EUT puede operar como se requiere durante o después de ser sometido a una descarga electrostática del aire. Existe una única categoría de equipos (A)

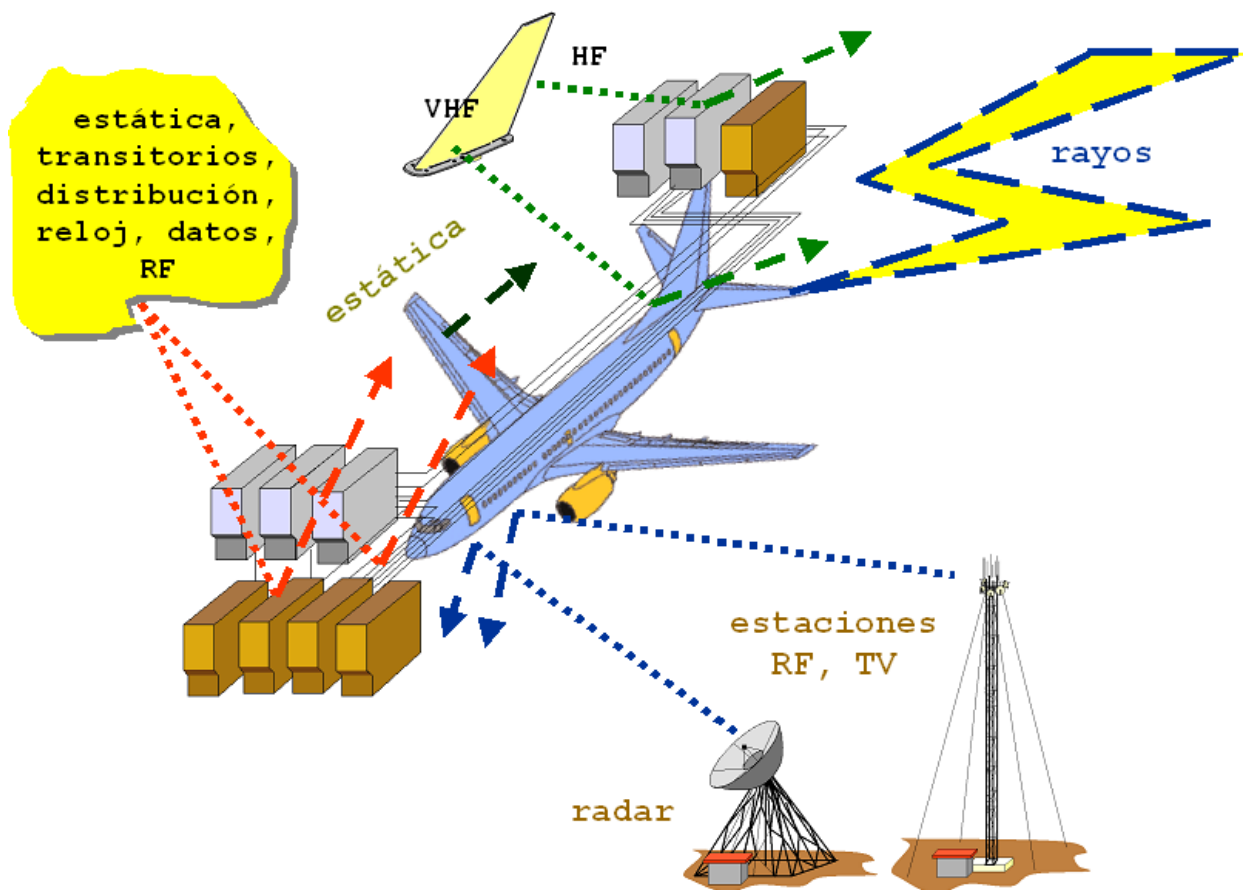
15.1.2-ELECTROMAGNETIC INTERFERENCE (EMI).

- La interferencia electromagnética (EMI) es la emisión electromagnética no planeada procedente de un dispositivo o sistema eléctrico o electrónico y que potencialmente puede interferir en el funcionamiento normal de otros dispositivos o sistemas.

Un ejemplo muy común y cotidiano de EMI, son las interferencias ocasionadas en la recepción de radio debidas a una computadora o a un teléfono móvil. En este caso la EMI no implica más que una desagradable molestia, pero cuando la interferencia es ocasionada en la recepción de una señal de radiofrecuencia para comunicación o navegación en una aeronave, el problema, lógicamente, es más serio.

Otros ejemplos de problemas relacionados con EMI muy cotidianos, son la distorsión que aparece en una imagen de TV cuando se conecta una aspiradora, el ruido provocado en la recepción AM cuando el receptor se sitúa bajo una línea de alta tensión, conversaciones que escuchamos a través de nuestro equipo de música cuando un vehículo equipado con una emisora de CB, pasa cerca de nuestra casa, etc.

- Las fuentes generadoras de EMI para una aeronave, caen dentro de una de las tres categorías básicas siguientes:



- a. Transitorios, debidos a descarga electrostática (Electrostatic Discharge, **ESD**), los rayos (**LIGHTNING**) y elementos de conmutación para conexión y desconexión de cargas eléctricas.
- b. Campos eléctricos y magnéticos provenientes del suministro de energía eléctrica alterna de 400 Hz.

- c. Radiofrecuencias debidas a elementos de abordo como las señales de reloj de los computadores, la conmutación de las señales de datos, equipos de HF y VHF, y equipos de tierra como estaciones de radio, televisión y radar (High Intensity Radiated Fields, **HIRF**).

El cableado es el factor más importante en cuanto a la interferencia electromagnética, dado que interconecta los equipos aviónicos y transfiere entre unos y otros señales de datos e información.

El cableado puede actuar como antena en el sentido de captar interferencias electromagnéticas y a la vez radiar interferencias hacia otros cables o equipos.

De menos importancia es el camino de EMI a través de las carcasas construidas de metal donde están inmersos los equipos aviónicos.

15.1.3-ELEMENTOS EN UN PROBLEMA DE EMC.

- A un problema de EMC están asociados tres elementos: una fuente que origina una perturbación electromagnética, un receptor de dicha perturbación y por lo tanto víctima de la misma, y un camino entre fuente y receptor que permite que la perturbación interfiera de forma negativa con el receptor.



La solución a un problema de EMC pasa por identificar al menos dos de estos tres elementos y reducir al mínimo posible, o eliminar, uno de ellos.

El acoplamiento, o lo que es lo mismo, la transferencia de la energía electromagnética desde la fuente al receptor, puede producirse por conducción (corriente eléctrica), inducción (campo magnético), acoplamiento capacitivo (campo eléctrico) y radiación (campo electromagnético), de forma aislada, o como combinación entre estos acoplamientos.

La entrada de energía electromagnética a la estructura de una aeronave, puede realizarse a través de caminos no intencionados, como pueden ser discontinuidades presentes en la superficie conductora de la estructura, tales como ventanas, entradas de aire y canalizaciones de cables. Esta forma de entrada de la energía electromagnética es llamada *back door coupling*. En contraposición, existe también la entrada de energía electromagnética a la estructura a través de antenas y sistemas sensores diseñados específicamente para captar dicha energía. En este caso se habla de *front door coupling*.

15.1.4-EFECTOS PRODUCIDOS POR LA INCOMPATIBILIDAD ELECTROMAGNÉTICA.

- Un aspecto que hace que se incremente el interés por la EMC, es el uso cada vez mas generalizado de microprocesadores. Estos componentes son ampliamente responsables de la generación de interferencias de radiofrecuencia en los sistemas electrónicos, además, ellos mismos también son susceptibles a fenómenos de este tipo.

- Otro aspecto que obliga a que se haga una mayor observación de la EMC en algunos sectores, es la progresiva sustitución de cajas y chasis metálicos que contienen los circuitos electrónicos, por cajas de algún tipo de plástico, con la consiguiente reducción de la protección que aquellas ofrecían. Afortunadamente esto no ocurre con los equipos aviónicos.

FUNCIONAMIENTO DEFECTUOSO DE LOS SISTEMAS.

Cada vez más, se están implantando sistemas basados en microprocesadores para realizar tareas que antes eran encomendadas a elementos analógicos. Estos sistemas trabajan con frecuencias muy elevadas de conmutación, y con niveles de consumo energético muy inferiores a los que necesitaban los sistemas analógicos anteriores para realizar la misma función. Estos factores hacen que dichos sistemas sean, por un lado, elementos perturbadores del entorno electromagnético donde trabajan debido a esas frecuencias de conmutación, y por otro, más susceptibles a la interferencia, debido a los bajos valores energéticos de trabajo necesarios para el cambio de información.

Algunos fenómenos más probables que se pueden dar en un sistema, en un entorno electromagnético no compatible son:

- fluctuaciones transitorias en las líneas de alimentación, en las líneas de control o líneas de información.
- campos de radiofrecuencia pulsantes (radar) o continuos acoplados al sistema o a sus cables de conexión.
- descarga electrostática.
- campos magnéticos o eléctricos de baja frecuencia.
- puesta de un sistema microcomputador a un estado de paro por falsedad en datos.

INTERFERENCIAS EN LA RECEPCIÓN DE SEÑALES DE RADIO.

Señales perturbadoras que se superponen a las señales útiles de recepción, modificando su misión.

PERTURBACIONES EN LA ALIMENTACIÓN ELÉCTRICA.

Este tipo de perturbaciones como pueden ser los transitorios y sobretensiones, se pueden producir por las operaciones de conmutación de elementos inductivos conectados a la red de alimentación. También se pueden producir distorsiones de la forma senoidal de la onda de alimentación y fluctuaciones.

IGNICIÓN DE ATMÓSFERAS INFLAMABLES.

Como consecuencia de una energía de RF, se puede producir la ignición de una atmósfera inflamable causando una explosión. Un campo electromagnético fuerte puede inducir corrientes en grandes estructuras metálicas, que al entrar en contacto con otras estructuras o por proximidad con ellas, pueden ocasionar chispas, y estas, si tienen la suficiente energía, incendiarán los vapores de la atmósfera.

15.1.5-MEDIDAS DE PROTECCIÓN EN LA AERONAVE.

CONDUCTIVIDAD DE LA ESTRUCTURA.

La conductividad es la primera consideración EMC en los materiales para obtener una estructura unificada. Las funciones eléctricas que dependen en gran medida de la conductividad son las siguientes:

- a.- Estabilidad eléctrica. Establecimiento de una masa de referencia de bajo ruido, con un cero estable para los equipos eléctricos y electrónicos. Esta masa eléctrica incluye la estructura, estantes para los equipos aviónicos, superficies externas, chasis de los equipos, etc.
- b.- Inmunidad contra las perturbaciones de radiofrecuencia internas y externas, y campos debidos a la distribución eléctrica de 400 Hz. Mediante el apantallado de cables, carcasas metálicas, paneles metálicos y la estructura de la aeronave.
- c.- Eliminación de riesgos y daños debidos a la ESD y Lightning. La estructura de la aeronave, ofrece el camino y control de la corriente debida a estos fenómenos.
- d.- Retorno para las señales y retorno para la alimentación. Utilizando la estructura, se puede ahorrar la instalación de cableado de retorno.

BLINDAJE.

- El Aluminio después del cobre es uno de las mejores pantallas electromagnéticas. Su efectividad varía con su grosor y la frecuencia de la radiación. Siempre que se pierdan propiedades de puesta a tierra o conductivas en uniones, o no estén disponibles por parte de la estructura (comprometiendo la continuidad), se debe añadir blindaje en forma de lámina de aluminio o pinturas metálicas que contienen agentes conductores.

SEGURIDAD EN LOS RETORNOS DE TIERRA.

- La estructura es una masa eléctrica y un camino de vuelta para las corrientes. En una aeronave podemos encontrar en los equipos eléctricos / electrónicos masas o caminos de vuelta, tales como las carcasas donde están inmersos los equipos, retorno para las corrientes de 115 V/400Hz, 28 V/400Hz, 28 V DC, retorno para los circuitos analógicos y circuitos digitales y diferentes apantallados. Algunos cerramientos de equipos requieren doble puesta a tierra en zonas expuestas a agua como puede ser la zona del ala, cargo o cola para ofrecer mayor seguridad en caso de fallo eléctrico.

RESISTENCIA.

- El valor máximo de resistencia más a menudo citado para una terminación o conexión en un diseño basado en EMC es de 2,5 mΩ. Este valor se aplica a terminaciones del cableado, carcasas de equipos, estantes, puertas y cualquier componente que forme parte del plano de tierra, apantallado o retorno eléctrico.

Todos los elementos conductores en el compartimiento de vuelo están enlazados para reducir la descarga electrostática. Todas las válvulas, bombas, rebordes, aperturas y entradas en el depósito de combustible están especialmente enlazados para evitar arcos procedentes de los rayos.

15.2-TESTS DE EMC. ASPECTOS BÁSICOS.

15.2.1-INSTRUMENTACIÓN.

- Dentro de la instrumentación utilizada en el tratamiento de la EMC, están el *receptor de medida* y el más utilizado, *analizador de espectro*. El analizador de espectro muestra como se distribuye la energía de una señal en función de la frecuencia (en un osciloscopio vemos como varia el voltaje de una señal en función del tiempo). El visualizador instantáneo del espectro es muy útil para confirmar las frecuencias y la naturaleza de las emisiones perturbadoras, así como la capacidad de capturar una pequeña parte del espectro.

Algunas de las especificaciones típicas de un analizador de espectro son: *rango de frecuencia*, que indica el rango de frecuencias que pueden ser medidas; *rango de la pantalla*, que indica en dB-s, la relación entre la señal más grande (*nivel de referencia*) y la señal más pequeña (*nivel de la línea de base*) que pueden ser mostradas juntas; *resolución del ancho de banda*, que representa el ancho de banda del filtro de la etapa IF del analizador y la habilidad para separar señales cercanas en frecuencia, etc.

El analizador de espectro, es también un importante instrumento para la resolución de problemas en equipos aviónicos como transponder, DME, radar meteorológico, radio-altímetro, etc.

- Para poder acoplar la variable medida al instrumento de medida en cuestión se necesita el transductor. Dependiendo de la forma que tomen las variables, el transductor será diferente. Las variables que se miden toman una de estas tres formas: (1) campo electromagnético radiado, (2) tensión conducida propagada por cable y (3) corriente inducida propagada por cable.

Para el caso en el que la variable sea un campo electromagnético radiado, el transductor será una antena. La medida de campos radiados se pueden hacer de la componente del campo eléctrico (E) o del magnético (H). En el campo lejano ambos están relacionados por la impedancia del espacio libre:

$$E/H = Z_0 = (\mu_0/\epsilon_0)^{0,5} = 120 \pi = 377 \Omega$$

En el campo cercano no existe relación, y las medidas del campo magnético dan mejor reproducibilidad que las mediciones del campo eléctrico el cual se ve fácilmente perturbado por objetos cercanos. En cualquier caso, se necesita una antena para acoplar el campo al receptor que efectúa la medición.

- A parte de medir las emisiones directamente como un campo radiado, se pueden medir también las corrientes de interferencias en cables de conexión y relacionarlas con los niveles aceptados de intensidad de campo. Los transductores utilizados para este tipo de mediciones son *la pinza de absorción* y *la sonda de corriente*.

- Cuando se quiere localizar físicamente el origen de las emisiones, se utiliza un juego de *sondas de campo cercano* (rastreador). Este juego consiste en dos sondas, una para el campo eléctrico (construcción de varilla) y otra para el campo magnético (construcción en bucle).

15.2.2-MEDICIONES.

Para simular de forma real una condición de EMC (referente tanto a la emisión como a la inmunidad), el equipo bajo prueba (EUT – Equipment Under Test-), debe representar una instalación típica. Si un EUT tiene diferentes configuraciones, las pruebas deben ser realizadas con todas aquellas configuraciones que representen un uso normal del equipo.

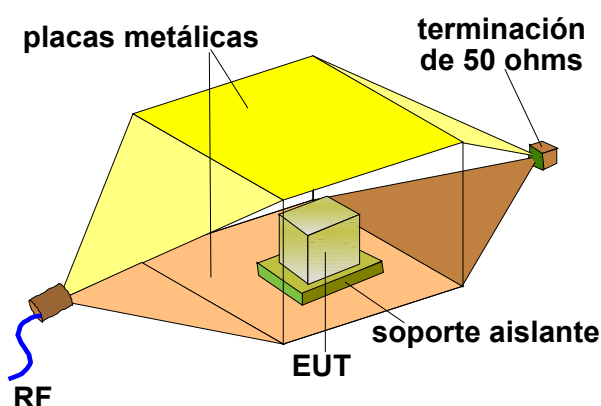
Para la medición de la susceptibilidad a la RF existen principalmente las siguientes técnicas:

Línea Plana, Célula TEM (Transverse Electromagnetic), Célula GTEM (Gigahertz TEM), Inyección de corriente en masa (BCI-Bulk Current Injection-), Cámara de modo agitado (Mode Stirred Chamber), Test en espacio abierto (OATS- Open Area Test Site).

Los aspectos básicos de cada una de las técnicas son los siguientes:

- La línea plana consiste básicamente, en dos placas paralelas (ver figura) entre las cuales se desarrolla el campo, alimentado por un extremo mediante un amplificador. El conjunto es como la parte final de una línea de transmisión, con su terminación típica pero formada por las placas citadas.

El equipo bajo pruebas (EUT), se coloca dentro del volumen, sobre un soporte aislante.



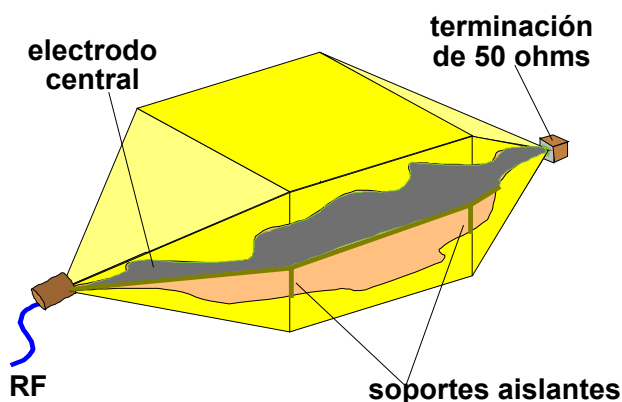
Las placas proporcionan un campo electromagnético transversal (TEM), uniforme, con las mismas características que el espacio abierto, siendo el campo eléctrico de placa a placa y el campo magnético paralelo a ellas.

La precisión de la prueba depende en gran medida de las dimensiones del EUT. Estas dimensiones están definidas en las normas correspondientes, así como los factores de corrección aplicables según dichas dimensiones. En cualquier caso, el volumen de los EUT debe ser relativamente pequeño, dado que el volumen de la Línea

Plana abarcado por las placas paralelas es de 80 x 80 x 80 cm.

La frecuencia máxima de las pruebas está limitada a los 200 MHz. ($\lambda = 1,5$ m). Por encima de esta frecuencia la separación de las placas es mayor que una longitud de media onda, con lo que el campo se vuelve más variable.

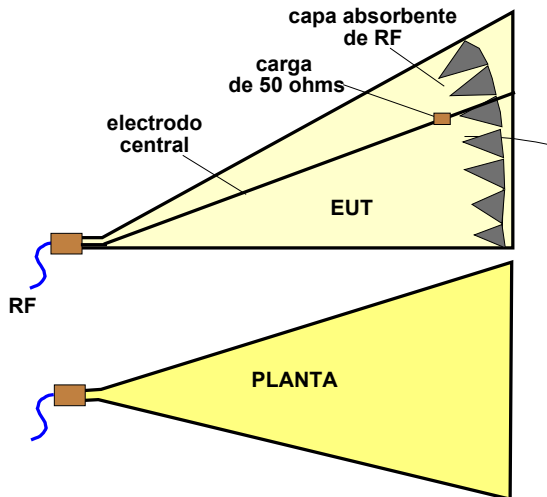
- La Célula TEM (o célula Crawford) es una alternativa a la Línea Plana. La célula TEM, diseñada por Myron Crawford en 1970, a partir de una línea coaxial.



Está basada en la idea de que una onda transversal se propaga dentro de una línea de transmisión. Si la línea de transmisión se hace lo suficientemente grande, se pueden emplazar EUT-s dentro de ella y probar su susceptibilidad a la onda TEM incidente. Básicamente es un Línea Plana como la vista mas arriba en la cual una de las placas se ha ampliado de tal manera que rodea a la otra por completo.

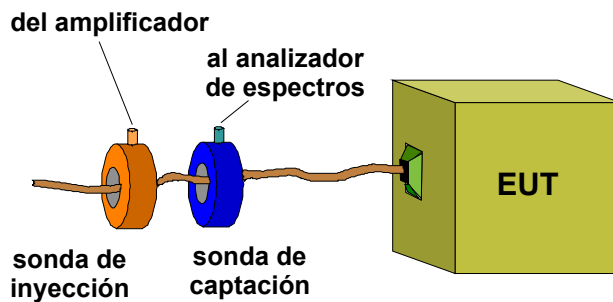
Los campos producidos en esta célula son bastante uniformes, y están totalmente metidos dentro de la estructura de la célula. Al igual que la Línea Plana, las dimensiones de la célula limitan la frecuencia superior de las pruebas; está indicada también para equipos relativamente pequeños.

- La célula GTEM supera algunas de las desventajas inherentes a la Línea Plana y a la Célula TEM. Esta célula es un híbrido entre la célula TEM tradicional y una cámara anecoica. Mediante el uso de una capa absorbente de RF, (conos anecoicos de carbón dopado y paredes de ferrita), se reducen las reflexiones e interferencias.



El ancho de banda útil en modelos comerciales es desde DC hasta 20 GHz para la prueba de inmunidad y desde 9 kHz. hasta 5 GHz. para la prueba de emisiones.

- Bulk Current Injection (BCI), es una técnica desarrollada primeramente en la industria aeronáutica.



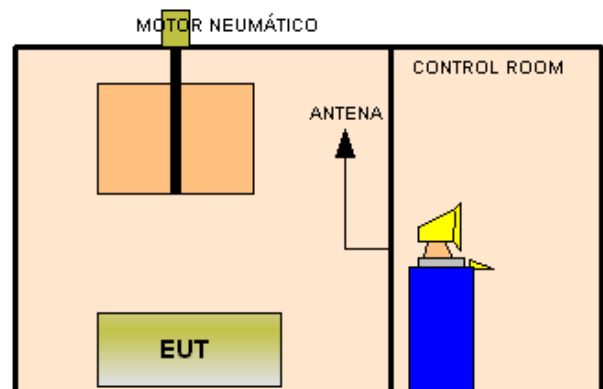
Con esta técnica se puede evaluar la inmunidad de un sistema, mediante la inducción de un nivel controlado de corriente en el cableado de dicho sistema.

Utiliza sondas de corriente (de inyección y de captación), para acoplar los sistemas de análisis y medida a un conductor o mazo de conductores. Las sondas funcionan como transformadores de RF. El primario de estos transformadores es la sonda de inyección, la cual acopla la energía de RF procedente de un amplificador, al mazo de

cables del equipo bajo prueba. Este mazo actúa como secundario. La sonda de captación funciona de manera semejante a la sonda de inyección, pero en este caso, aquella (secundario), se encarga de monitorizar la corriente en el mazo de cables (primario), mediante un analizador de espectros.

Los niveles de inyección de corriente son los equivalentes a aquellos que van a ser encontrados en el cableado de los sistemas instalados en la aeronave expuesta a un entorno específico de RF.

- La técnica Mode Stirred Chamber utiliza una paleta de metal, rotando a baja velocidad mediante un sistema mecánico, dentro de una habitación apantallada. Con estas paletas se pretende “agitar” la energía electromagnética de RF radiada por una antena y repartirla de forma uniforme en la habitación de pruebas. Mediante esta técnica, todas las caras y elementos del EUT son expuestos a todas las polarizaciones y ángulos de incidencia, con el nivel determinado por el tipo de test en ejecución.



- OATS, mediante este método se trata de medir las emisiones radiadas procedentes de equipos eléctricos y electrónicos (EUT), en un espacio abierto, libre de radiaciones ajenas al propio equipo.

15.3-HIGH INTENSITY RADIATED FIELDS (HIRF).

- ❑ En los últimos años el avance e incremento de las comunicaciones por radio y otras tecnologías electrónicas han introducido en el entorno operacional el fenómeno HIRF. Existen más de 500.000 emisores en USA y el oeste de Europa que contribuyen al entorno electromagnético. Los sistemas eléctricos y electrónicos de las aeronaves están expuestos a los entornos HIRF que emanan de transmisores de radio y televisión de alta potencia, radar, enlaces de satélite y sistemas de comunicaciones por microondas.
- ❑ En el pasado, las aeronaves civiles eran evaluadas en cuanto al entorno electromagnético, en términos de la habilidad de los sistemas de abordaje para coexistir sin problemas de compatibilidad. Una de las fuentes más importantes generadoras de problemas de compatibilidad era el equipo transmisor de HF.

Factores como la incorporación de elementos electrónicos, en los sistemas críticos de seguridad en aeronaves de nueva generación, posibilidad de la reducción del apantallamiento electromagnético por el uso de materiales compuestos, expansión en el uso del rango de frecuencias, sobre todo por encima de 1 GHz, etc., hicieron que las autoridades de aviación, se interesaran por la potencial interferencia de HIRF generada por fuentes externas de transmisión sobre esos sistemas críticos. La primera amenaza en cuanto HIRF proviene de transmisores de comunicación y radar, emplazados en tierra mar o aire.

- ❑ Desde el punto de vista aeronáutico, bajo el acrónimo de HIRF se define la energía electromagnética, no ionizante, externa a la aeronave, que puede causar efectos adversos en los sistemas eléctricos y electrónicos de la misma, y que afecten a la seguridad de vuelo.

Como se aprecia, bajo esta definición no se incluye la interferencia entre los sistemas de abordaje, materia esta que es tratada bajo el acrónimo EMC. Tampoco incluye los efectos causados por la electricidad estática en la aeronave, materia tratada bajo el acrónimo ESD, ni el tema correspondiente a rayos (lightning).

El entorno electromagnético está cambiando continuamente acorde a como cambia la tecnología de los transmisores y las antenas, el reparto administrativo de las emisiones en diferentes bandas y nuevos productos y servicios que hacen uso del espectro de RF. Los campos de RF interactúan con los sistemas eléctricos y electrónicos de la aeronave a través de aperturas intencionadas (front door coupling) tales como las antenas y no intencionadas (back door coupling) tales como las ventanas del cockpit, huecos en escotillas, apantallado. Las altas frecuencias de los campos de RF que se acoplan en el interior de la aeronave, pueden resonar en ciertos espacios de la misma, incrementando su nivel y acoplándose posteriormente a través de los apantallados de los cables a los equipos eléctricos o electrónicos.

El conocimiento y control sobre HIRF, implica a una serie de diferentes áreas. La primera es el conocimiento del entorno amenazante, el acoplamiento de señales electromagnéticas dentro de la estructura de la aeronave y después en el sistema de cableado de la misma. La segunda trata con los efectos de señales HIRF indeseadas, en el comportamiento del sistema y la especificación de protección HIRF. La tercera implica el conocimiento para definir una estrategia de protección HIRF con metodología para asegurar que la protección ha sido tomada. Por último, la verificación mediante el análisis y las pruebas requeridas por el proceso de certificación.

15.3.1-PROTECCIÓN EFECTIVA DE LOS SISTEMAS.

- Un programa de certificación HIRF puede ser descompuesto en cinco actividades:
 1. Establecimiento del entorno HIRF externo aplicable.
 2. Identificación de los sistemas que deben ser valorados.
 3. Establecimiento del entorno para las pruebas a sistemas instalados.
 4. Consideraciones de requerimientos.
 5. Verificación de conformidad.

1. ESTABLECIMIENTO DEL ENTORNO HIRF EXTERNO APLICABLE.

- Tomando como referencia los cálculos realizados sobre 500.000 emisores con base en USA y cinco países europeos (Francia, Alemania, Suecia, Holanda y UK) se han modelado los entornos HIRF. El resultado, es la representación de la fuerza del campo electromagnético, sobre un rango de frecuencias de 10 kHz a 40 GHz.

Están definidas cuatro categorías: *entorno severo para ala fija, entorno severo para ala rotatoria, entorno para certificación y entorno normal*. La aplicación de cada uno de los entornos depende de la categoría de la aeronave y de lo crítico que sea el sistema que se valora.

Para el cálculo de los entornos, se tomaron como base unos supuestos generales y otros específicos. Los supuestos específicos para cada entorno, tienen que ver con los criterios de distancias entre aeronave y transmisor. Entre los generales, tenemos la división del espectro en bandas de frecuencias definidas, presentación de valores de pico y medios de campo, no se consideró la modulación de la señal transmitida, aunque se utilizó el Duty Cycle para calcular la media de potencia para transmisores de impulsos, etc.

VALORES PARA EL ENTORNO HIRF SEVERO. (ALA FIJA)

Este entorno representa la fuerza del campo electromagnético en el espacio aéreo en el cual las operaciones de vuelo están permitidas.

El entorno HIRF severo por encima de los 18 GHz, debería ser usado solamente si el criterio pasa/fallo para los sistemas de nivel A no se encuentra durante las pruebas en el rango 12-18 GHz, o si los sistemas están diseñados para operaciones en la banda de frecuencias de 18 a 40 GHz como parte de su proceso funcional.

NOTA: Se ha encontrado que existen campos de alta impedancia de 1000V/m en la banda de frecuencias de 10KHz-100KHz. Investigaciones demuestran que estos campos inducen corrientes insignificantes en el cableado de la aeronave y por lo tanto pueden ser ignorados.

FRECUENCIA	FUERZA DEL CAMPO (V/m)	
	PICO	MED.
10-100 kHz	50	50
100-500 KHz.	60	60
500 KHz.- 2 MHz	70	70
2-30 MHz	200	200
30-70 MHz	30	30
70-100 MHz	30	30
100-200 MHz	90	30
200-400 MHz	70	70
400-700 MHz	730	80
700 MHz-1 GHz	1400	240
1-2 GHz	3300	160
2-4 GHz	4500	490
4-6 GHz	7200	300
6-8 GHz	1100	170
8-12 GHz	2600	330
12-18 GHz	2000	330
18-40GHz	1000	420

VALORES DEL ENTORNO HIRF SEVERO. (ALA ROTATORIA)

Este entorno representa la fuerza del campo en el espacio aéreo en el cual las operaciones de vuelo de la aeronave de ala rotatoria están permitidas.

El entorno HIRF severo por encima de los 18 GHz, debería ser usado solamente si el criterio pasa/fallo para los sistemas de nivel A no se encuentra durante las pruebas en el rango 12-18 GHz, o si los sistemas están diseñados para operaciones en la banda de frecuencias de 18 a 40 GHz como parte de su proceso funcional.

NOTA: Se ha encontrado que existen campos de alta impedancia de 1000V/m en la banda de frecuencias de 10KHz-100KHz. Investigaciones demuestran que estos campos inducen corrientes insignificantes en el cableado de la aeronave y por lo tanto pueden ser ignorados.

FRECUENCIA	FUERZA DEL CAMPO (V/m)	
	PICO	MED.
10-100 kHz	150	150
100-500 kHz	200	200
500 kHz-2 MHz	200	200
2-30 MHz	200	200
30-70 MHz	200	200
70-100 MHz	200	200
100-200 MHz	200	200
200-400 MHz	200	200
400-700 MHz	730	200
700 MHz-1 GHz	1400	240
1-2 GHz	5000	250
2-4 GHz	6000	490
4-6 GHz	7200	400
6-8 GHz	1100	170
8-12 GHz	5000	330
12-18 GHz	2000	330
18-40 GHz	1000	420

VALORES PARA ENTORNO HIRF CERTIFICACIÓN.

FRECUENCIA	FUERZA DEL CAMPO (V/m)	
	PICO	MED.
10-100 kHz	50	50
100-500 kHz	50	50
500 kHz-2 MHz	50	50
2-30 MHz	100	100
30-70 MHz	50	50
70-100 MHz	50	50
100-200 MHz	100	100
200-400 MHz	100	100
400-700 MHz	700	50
700 MHz-1 GHz	700	100
1-2 GHz	2000	200
2-4 GHz	3000	200
4-6 GHz	3000	200
6-8 GHz	1000	200
8-12 GHz	3000	300
12-18 GHz	2000	200
18-40 GHz	600	200

Esta tabla muestra el entorno requerido usado para certificar sistemas de seguridad críticos en ala fija y en helicópteros para operaciones IFR.

El entorno HIRF severo por encima de los 18 GHz, debería ser usado solamente si el criterio pasa/fallo para los sistemas de nivel A no se encuentra durante las pruebas en el rango 12-18 GHz, o si los sistemas están diseñados para operaciones en la banda de frecuencias de 18 a 40 GHz como parte de su proceso funcional.

FRECUENCIA	FUERZA DEL CAMPO (V/m)	
	PICO	MED.
10-100 KHz.	20	20
100-500 KHz.	20	20
500 KHz.-2 MHz	30	30
2-30 MHz	100	100
30-70 MHz	10	10
70-100 MHz	10	10
100-200 MHz	30	10
200-400 MHz	10	10
400-700 MHz	700	40
700 MHz-1 GHz	700	40
1-2 GHz	1300	160
2-4 GHz	3000	120
4-6 GHz	3000	160
6-8 GHz	400	170
8-12 GHz	1230	230
12-18 GHz	730	190
18-40 GHz	600	150

VALORES PARA ENTORNO HIRF NORMAL.

Este entorno representa la fuerza del campo en el espacio aéreo de los aeropuertos y alrededores en los cuales las operaciones de salidas y llegadas tienen lugar.

El entorno HIRF severo por encima de los 18 GHz, debería ser usado solamente si el criterio pasa/fallo para los sistemas de nivel A no se encuentra durante las pruebas en el rango 12-18 GHz, o si los sistemas están diseñados para operaciones en la banda de frecuencias de 18 a 40 GHz como parte de su proceso funcional.

El rango de frecuencias consideradas en los entornos HIRF externos, se puede dividir en dos partes dependiendo del uso que de los distintos rangos se hace; desde 10 KHz. a 400 MHz, el mayor uso que se hace del espectro electromagnético es por dispositivos de comunicaciones y navegación, que generalmente son poco direccionales. Por encima de los 400 MHz el uso está enfocado hacia dispositivos que trabajan con haces estrechos y frecuentemente pulsatorios como radar, señales de datos de alta velocidad, señales de mando y control para satélites y misiles.

La siguiente tabla muestra las distintas bandas según la distribución en el espectro, así como algunas de las aplicaciones en cada banda.

3 KHz		
VERY LOW FREQUENCY (VLF)	OMEGA (10 - 14 KHz) [descatalogado]	
30KHz		La transmisión se realiza mediante <i>ondas de tierra</i> .
30 KHz		
LOW FREQUENCY (LF)	LORAN C (100 KHz)	Estas ondas siguen la curvatura de la tierra, alcanzando toda la superficie.
300KHz		
MEDIUM FREQUENCY (MF)	NDB-s (190 - 1750KHz) AM BROADCAST (530 – 1600 KHz)	
3 MHz		
HIGH FREQUENCY (HF)	HF COMM (2 – 30 MHz)	En este rango las ondas son de <i>cielo</i> . Tienden a viajar en línea recta y se reflejan en la ionosfera para volver a la tierra mas allá de la línea del horizonte.
30 MHz		
30 MHz	MARKER BEACONS (75 MHz) FM BROADCAST (88 – 108 MHz) VHF NAV –VOR-LOC (108 – 118 MHz) VHF COMM (118 – 137 MHz)	
VERY HIGH FREQUENCY (VHF)		
300 MHz		A partir de HF, la transmisión se realiza a través de <i>ondas espaciales</i> . Estas ondas viajan también en línea recta, pero no son reflejadas por la ionosfera. El rango de recepción entre estaciones de tierra para este tipo de ondas es de alrededor de 100 Km., a partir de estas distancias, las ondas quedan bloqueadas por la curvatura de la superficie de la tierra.
300 MHz	GLIDESLOPE (328 – 336 MHz) DME (960 – 1215 MHz) TRANSPONDER TCAS (1030 MHz Y 1090 MHz)	
ULTRA HIGH FREQUENCY (UHF)		
3 GHz	GPS (1575,42 MHz) AES SATCOM (1,5 – 1,6 GHz)	
3 GHz	GES SATCOM (4 – 6 GHz)	
SUPER HIGH FREQUENCY (SHF)	RADAR ALTIMETER (4,3 GHz) MLS (5,0 – 5,1 GHz) DOPPLER NAV (8,8 GHz) WEATHER RADAR (9,375 GHz)	
30 GHz		
30 GHz		
EXTREMELY HIGH FREQUENCY (EHF)		
300 GHz		

2. IDENTIFICACIÓN DE LOS SISTEMAS QUE DEBEN SER VALORADOS.

- ❑ Solamente los sistemas identificados como sistemas que realizan o contribuyen a funciones cuyo fallo podría resultar en efecto Catastrófico, efecto Peligroso/Grave-Importante, o efecto Importante en la aeronave, son requeridos para la demostración de conformidad con los requerimientos de FAA/JAA.

CLASIFICACIÓN DE LOS SISTEMAS.

- ❑ Los requerimientos FAA/JAR establecen niveles de pruebas para los sistemas eléctricos y electrónicos dependiendo de la clasificación de la condición de fallo de los mismos. Esta clasificación es:

CATASTRÓFICO: Condición de fallo que va a impedir la continuidad de un vuelo y aterrizaje seguros.

PELIGROSO / GRAVE-IMPORTANTE: Condición de fallo que va a reducir la capacidad de la aeronave o la habilidad de la tripulación para afrontar condiciones adversas de operación, hasta un punto tal que pueda provocar:

- Gran reducción en los márgenes de seguridad o capacidad funcional.
- Dificultades físicas o aumento de la carga de trabajo de tal manera que la tripulación no pueda afrontarlo para desarrollar sus tareas de forma precisa y completa.
- Daños serios a un número relativamente bajo de los ocupantes.

IMPORTANTE: Condición de fallo que va a reducir la capacidad de la aeronave o la habilidad de la tripulación para afrontar condiciones adversas de operación, hasta un punto tal que pueda provocar por ejemplo, una reducción significativa en los márgenes de seguridad o capacidades funcionales, un incremento en la carga de trabajo de la tripulación o condiciones que afecten a la eficiencia de la tripulación o comodidad para los ocupantes.

NO IMPORTANTE: Condición de fallo que no va a reducir de forma significativa la seguridad de la aeronave, y que conlleva acciones de la tripulación que están dentro de sus capacidades normales de trabajo. Las condiciones de fallo no importantes; puede incluir por ejemplo, una leve reducción en los márgenes de seguridad o capacidad funcional, un incremento leve de la carga de trabajo de la tripulación como puede ser un cambio en los planes de vuelo o alguna incomodidad a los ocupantes.

SIN EFECTO: Condición de fallo que no afecta a la capacidad operativa de la aeronave o incrementa la carga de trabajo de la tripulación.

- ❑ El requerimiento del nivel de certificación está relacionado con la clasificación de la condición de fallo del sistema. Las definiciones de los niveles de certificación son:

NIVEL A: Sistemas eléctricos y electrónicos cuyo fallo puede causar o contribuir al fallo de una función, dando como resultado una condición de *fallo catastrófico* para la aeronave.

NIVEL B: Sistemas eléctricos y electrónicos cuyo fallo puede causar o contribuir al fallo de una función, dando como resultado una condición de *fallo Peligroso / Grave-Importante* para la aeronave.

NIVEL C: Sistemas eléctricos y electrónicos cuyo fallo puede causar o contribuir al fallo de una función, dando como resultado una condición de *fallo Importante* para la aeronave.

NIVEL D: Sistemas eléctricos y electrónicos cuyo fallo puede causar o contribuir al fallo de una función, dando como resultado una condición de *fallo No Importante* para la aeronave. Una vez que el sistema ha sido reconocido como NIVEL D, por parte de la autoridad de certificación de aviación competente, no es necesaria ninguna demostración posterior de conformidad HIRF.

NIVEL E: Sistemas eléctricos y electrónicos cuyo fallo puede causar o contribuir al fallo de una función, dando como resultado una condición de *fallo Sin efecto* para la aeronave. Una vez que el sistema ha sido reconocido como NIVEL E por parte de la autoridad de certificación de aviación competente, no es necesaria ninguna demostración posterior de conformidad HIRF.

Una valoración final de la clasificación de fallos de los sistemas, muestra que las funciones no están al mismo nivel de clasificación de condición de fallo a través de todas las fases de operación de vuelo, y por lo tanto justifica la aplicación del entorno apropiado para la fase correspondiente.

3. ESTABLECIMIENTO DEL ENTORNO PARA LAS PRUEBAS DE SISTEMAS INSTALADOS.

- ❑ El entorno HIRF externo a la aeronave, puede penetrar y establecer en el interior de la misma un entorno de RF, al cual los sistemas eléctricos y electrónicos instalados estarán expuestos. El entorno de RF que se puede desarrollar en el interior, va a ser específico para cada aeronave y cada zona.

El entorno de RF interno se deriva del apropiado entorno HIRF externo para sistemas o equipos que desempeñan funciones clasificadas como NIVEL A.

Para sistemas o equipos que desempeñan funciones clasificadas como NIVEL B el entorno normal puede ser utilizado para determinar el entorno de RF interno. Esta determinación viene marcada en los procedimientos de test del documento ED-14/DO-160 y mediante la utilización de curvas genéricas de acoplamiento y atenuación. El entorno derivado de RF interno, que está relacionado con el entorno externo aplicable a la clasificación de condición de fallo de los equipos, es entonces utilizado para establecer los niveles de pruebas a los sistemas y equipos.

Para equipos y sistemas que desempeñan funciones clasificadas como NIVEL C se utilizan los procedimientos marcados en ED-14/DO-160.

4. CONSIDERACIONES DE REQUERIMIENTOS.

REQUERIMIENTOS PARA SISTEMAS DE NIVEL A:

- ❑ Estos equipos requieren protección hasta el punto de que la función que desempeñan no debe verse afectada cuando la aeronave es expuesta al entorno HIRF apropiado.

REQUERIMIENTOS PARA SISTEMAS DE NIVEL B:

- ❑ Las funciones para los equipos de este nivel, requieren protección hasta el punto de que cuando los equipos que componen el sistema son expuestos al nivel de test definido en la sección 8.6.12 (ED-14/DO-160), los sistemas eléctricos y electrónicos que realizan esas funciones no se verán afectadas de forma adversa.

REQUERIMIENTOS PARA SISTEMAS DE NIVEL C:

- ❑ Las funciones para los equipos de este nivel, requieren protección hasta el punto de que cuando los equipos que componen el sistema son expuestos al nivel de test definido en la sección 8.6.12 (ED-14/DO-160), los sistemas eléctricos y electrónicos que realizan esas funciones no se verán afectadas de forma adversa.

CONDICIÓN DE FALLO	NIVEL	REQUERIMIENTOS DE TEST	ENTORNO
CATASTRÓFICO	A	A nivel de aeronave (excepto para los sistemas que ejecutan una función donde se puedan utilizar sistemas de test usando curvas genéricas)	SEVERO/CERTIFICACIÓN/NORMAL (depende de la categoría de la aeronave)
PELIGROSO /GRAVE-IMPORTANTE	B	Test de equipos, incluyendo sensores o simulación representativa de sensores.	ENTORNO DE TEST DE EQUIPOS
IMPORTANTE	C	Test de equipos.	ENTORNO DE TEST DE EQUIPOS
NO IMPORTANTE	D	Sin requerimiento	Sin requerimiento
SIN EFECTO	E	Sin requerimiento	Sin requerimiento

5. VERIFICACIÓN DE CONFORMIDAD.

- ❑ La verificación de conformidad debe mostrarse mediante la confirmación de que los niveles de los entornos HIRF internos que aparecen en las interfaces y recintos de los equipos, no exceden los niveles de test HIRF de los sistemas/equipos.
- ❑ El plan para el acercamiento de conformidad en la certificación debería incluir los aspectos siguientes:
 - *Identificación de los sistemas.* Desempeño de funciones de nivel A, B o C.
 - *Entorno HIRF de los sistemas instalados.* El entorno HIRF externo aplicable y/o la clasificación de la condición de fallo del sistema/equipo.
 - *Métodos de verificación.* Descripción de los métodos propuestos para la verificación de conformidad. La verificación debe ser demostrada mediante los tests apropiados según la clasificación de la condición de fallo de la función que desempeñan los equipos.
 - *Criterios de Pasa / Falla.* Criterios específicos de Pasa / Falla para cada sistema y los medios para controlar dichos criterios deben ser establecidos por el aspirante y aprobados por la autoridad de certificación de aviación competente.

15.3.2-PERTURBACIONES EN SISTEMAS AVIÓNICOS DEBIDOS A HIRF.

- La onda electromagnética en el espacio, puede perturbar de dos maneras a los sistemas aviónicos de una aeronave. En una de ellas, imprime una fuerza indeseada sobre el haz de electrones que se mueve dentro de un CRT, y en otro caso, dicha fuerza indeseada la imprime sobre los electrones que se mueven en los conductores.

Dependiendo de la intensidad de la onda HIRF, en el caso de los sistemas de representación como el EFIS, el efecto de la perturbación, se manifiesta mediante la representación desenfocada de la información que aparece sobre la pantalla. En el caso de los conductores, la HIRF induce tensiones y corrientes en el cableado de los circuitos, haciendo que dichos circuitos respondan de una u otra manera a la señal externa. El nivel de tensión que se genera en los cables, depende de la longitud de los mismos y de la separación de la fuente de radiación. Para el caso de circuitos digitales, se han medido variaciones de tensión desde 0,4 a 1,2 V en el rango de radiofrecuencia, variación que puede resultar en un cambio de estado lógico, variando por lo tanto la información digital. La experiencia en la realización de test de HIRF en sistemas aviónicos de aeronaves comerciales indica que las fuentes de alimentación, circuitos de monitores y circuitos de detección de error de datos digitales son mucho más sensibles a las perturbaciones que las computadoras digitales internas.

El fallo más común es la desconexión automática de la fuente de alimentación. Los circuitos de detección de sobrevoltaje detectan voltajes inducidos por RF y desconectan la fuente. Otro fallo es la inducción de voltajes en elementos detectores e indicadores de error de un sistema determinado. Cuando existe inducción por HIRF, el elemento sensor de fallos, muestra la correspondiente bandera y frecuentemente desconecta parte del sistema implicado al que protege.

Es importante que cuando los sistemas son certificados para HIRF, estos sean sometidos a un nivel de test que realmente cause tales perturbaciones, que el operador o el instalador tengan conocimiento de la naturaleza de una perturbación HIRF, el aspecto, el impacto en las operaciones y la forma de recuperación de la perturbación.

EFFECTOS DE HIRF EN LAS COMUNICACIONES VHF.

- Las antenas de las aeronaves están diseñadas para recibir señales de RF dentro de unos rangos específicos de frecuencias y transportar esta energía de RF a los receptores de radio o radar dentro de la aeronave. Las radios de las aeronaves están diseñadas para operar en frecuencias asignadas de acuerdo con localizaciones del espectro de RF nacionales e internacionales. Estas localizaciones se desarrollan para asegurar que fuentes de RF de alta energía autorizadas no interfieran con los equipos de radar y radio dentro de la aeronave.

15.4-RAYOS (LIGHTNING).

15.4.1-INTRODUCCIÓN.

ESTRUCTURA ELÉCTRICA DE LA TIERRA.

- La tierra está cargada eléctricamente y funciona como un condensador esférico. Tiene una carga neta negativa compensada por una carga exactamente igual pero positiva, de la atmósfera.
- En el aire existen diversas clases de iones: los debidos a partículas de polvo y polución que son los llamados iones grandes, e iones creados por los rayos cósmicos, llamados iones pequeños. Los iones más grandes y pesados, se mueven más lentamente (más resistividad en el aire) que los iones pequeños. También los iones grandes, captan cargas de los iones pequeños reduciendo la conductividad total.

La conductividad del aire es por lo tanto, muy variable, dependiendo de la “suciedad” que haya en él. En la superficie de la tierra, varia de lugar a lugar y de momento en momento.

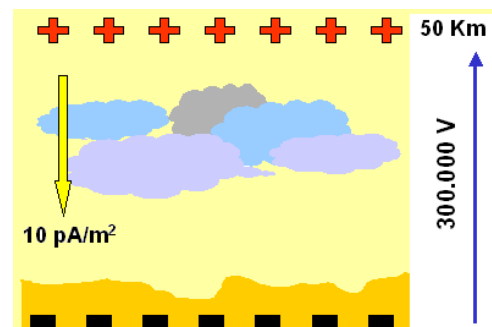
La resistividad eléctrica de la atmósfera va descendiendo rápidamente con la altura hasta aproximadamente los 50 Km., desde donde dicha resistividad se mantiene más o menos constante. Las razones para este descenso (o aumento de la conductividad) son, por un lado, la ionización producida por los rayos cósmicos y por otro la densidad del aire. La ionización aumenta con la altura, más iones, menos resistividad. La densidad del aire disminuye con la altura, el camino libre de los iones aumenta, de esta manera los iones pueden recorrer un camino mas largo en el campo eléctrico, antes de sufrir un choque.

A partir de los 50 Km., el aire se comporta como un conductor, aunque no tan bueno como en la ionosfera debido a la gran cantidad de iones presentes en esta, generados fotoeléctricamente por el sol.

CORRIENTES ELÉCTRICAS EN LA ATMÓSFERA.

- La diferencia de potencial entre la superficie de la tierra y la capa conductora a 50 Km, es de aproximadamente 300.000 voltios (y se mantiene mas o menos constante a partir de esta altura), con lo que tenemos un campo eléctrico medio a través de la atmósfera de 6 V/m. Cerca de la superficie de la tierra, este campo eléctrico tiene un valor de alrededor de los 100 V/m.

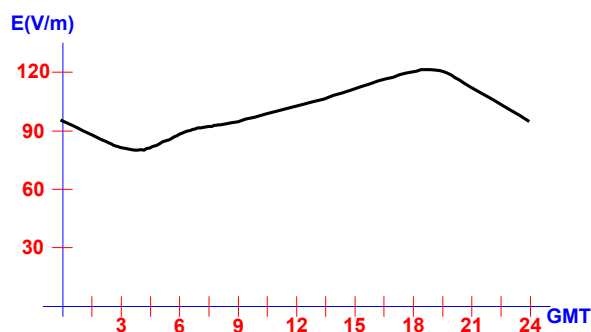
Debido al campo eléctrico y a la conductividad del aire, existe una corriente procedente de la capa conductora a través de la atmósfera que lleva cargas positivas mediante los iones del aire (iones creados por los rayos cósmicos, por el mar y por las actividades humanas) hacia la tierra. La densidad de esta corriente es de unos 10 pA/m^2 (superficie paralela a la tierra). La corriente total que atraviesa la atmósfera hacia la tierra es de cerca de 1800 A. Con esta corriente, en media hora las cargas negativas de la tierra quedarían compensadas, es decir, la tierra se descargaría. ¡Pero esto no ocurre!



Debido al campo eléctrico y a la conductividad del aire, existe una corriente procedente de la capa conductora a través de la atmósfera que lleva cargas positivas mediante los iones del aire (iones creados por los rayos cósmicos, por el mar y por las actividades humanas) hacia la tierra. La densidad de esta corriente es de unos 10 pA/m^2 (superficie paralela a la tierra). La corriente total que atraviesa la atmósfera hacia la tierra es de cerca de 1800 A. Con esta corriente, en media hora las cargas negativas de la tierra quedarían compensadas, es decir, la tierra se descargaría. ¡Pero esto no ocurre!

Haciendo una serie de mediciones de la corriente sobre cualquier océano en periodos temporales de un día y promediando estas medidas en cada uno de los puntos donde se han realizado, se obtiene una grafica como la de la figura siguiente. En ella se representa la variación del gradiente de potencial equivalente a la variación de corriente en un día despejado, sobre cualquier océano y referido a la hora GMT.

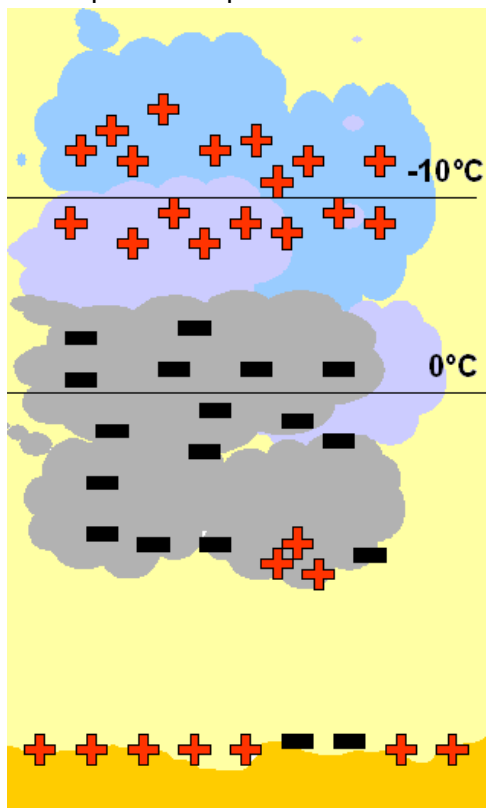
Se puede observar en la figura, que la corriente es máxima sobre las 19 h (GMT) sobre cualquier punto de medición en la tierra, y mínima sobre las 4 h (GMT). Esto tiene su explicación en la alta conductividad de la parte alta de la atmósfera que impide que la d.d.p. entre dicha capa y el suelo pueda variar localmente, aunque como se puede observar, si lo hace de forma global a lo largo del día.



La manera en que se compensa la pérdida de cargas eléctricas negativas de la tierra debido a las corrientes en la atmósfera, es mediante la inyección hacia la misma del mismo número de cargas que la tierra pierde por el efecto de las citadas corrientes. El mecanismo que lleva estas cargas negativas a tierra es la tormenta eléctrica, con sus rayos. Cuando cae un rayo a tierra se produce la descarga de gran cantidad de cargas eléctricas negativas. Según estimaciones, cada segundo "caen" a la tierra una media de 100 rayos, con un pico de la actividad a las 19 h (GMT).

CARGAS ELÉCTRICAS EN UNA TORMENTA ELÉCTRICA.

- Una tormenta eléctrica ordinaria está compuesta por regiones espaciales que tienen una superficie límite en dirección horizontal y en las cuales ocurren todos los procesos que originan una tormenta eléctrica. Cada una de estas regiones espaciales se llama célula. Las células están juntas unas a otras pero independientes entre sí. En cada célula ocurre el mismo proceso para la formación de una tormenta eléctrica, aunque en instantes diferentes.
- La distribución de cargas en una célula madura (tormenta ya formada) de tormenta eléctrica, es tal que, en la parte superior, a 6 o 7 Km. de altura con una temperatura de alrededor de -20°C , tiene carga positiva (zona P), y la parte inferior, a 3 o 4 Km. donde la temperatura aproximada está entre 0 y -10°C , tiene carga negativa (zona N), a excepción de una pequeña zona (zona p) de esta parte inferior en la cual hay cargas positivas. La zona N y la P tiene aproximadamente la misma carga, creando un dipolo. Medidas realizadas en un determinado entorno y condiciones (Malan, en 1963) dieron como valores de carga $+10\text{C}$ (culombios) a 2Km de altura, -40C a 5 Km y $+40\text{C}$ a 10 Km.



Cuando la carga negativa formada en la parte inferior de la nube (cúmulo nimbo), es tal que, genera un campo eléctrico a la altura de la región de carga negativa, de magnitudes típicas de 3-4 KV/cm., se produce la ruptura eléctrica del aire, y consecuentemente, el rayo. El rayo transporta las cargas negativas de la parte inferior de la nube hacia la tierra, que ahora está a un potencial relativo más positivo que la parte inferior de la nube (a excepción del área correspondiente a la zona p). Por otro lado, las cargas positivas en la parte superior de la nube, dejan esta y entran en la capa superior conductora de la atmósfera, extendiéndose mediante esta capa por toda la tierra. Después como ya se ha indicado, en las zonas de tiempo despejado estas cargas regresarán a la tierra creando la corriente atmosférica.

- ❑ Los rayos no solamente ocurren entre nube y tierra (*nube a tierra*), pueden formarse también entre puntos de la misma nube (*intranube*), de una nube a otra nube (*nube a nube*) y entre una nube y el aire despejado (*nube a aire*). Las descargas pueden originarse en una nube desde un centro de carga positiva (descarga positiva), o desde un centro de carga negativa (descarga negativa), cada una con sus características propias. Las descargas intranube, redistribuyen la carga dentro de la nube. Representan el 50% de los rayos producidos en latitudes norte. Las descargas nube a nube y nube a aire son menos comunes. Estos tres tipos de descarga son los que más pueden afectar al entorno de vuelo.
- ❑ Cada vez que se produce un rayo, la célula tarda alrededor de 5 segundos en volver a las condiciones iniciales de carga. Esto no quiere decir que inmediatamente después de esos 5 segundos, se va a producir otro rayo, pues cambian condiciones como la geometría, etc.
- ❑ La forma en la que las cargas son generadas o separadas en la nube no se conoce todavía con exactitud. Existen dos teorías que intentan explicar el fenómeno. Por un lado, definida de forma básica, la teoría de la convección propone que los iones libres en la atmósfera son capturados por gotas de nube y desplazados por convección en la propia nube para producir las regiones cargadas. Por otro lado, la debida a C. T. R. Wilson parece ser la más aceptada. Resumiendo, esta teoría establece que una gota de agua que cae en un campo eléctrico, hacia la tierra cargada negativamente, tendrá un momento dipolar inducido, siendo positiva la parte inferior de la gota y negativa la parte superior. En su descenso, los iones grandes negativos serán atraídos por la parte inferior de la gota y se adhieren a ella. Los iones grandes positivos serán repelidos y se alejarán de dicha gota hacia la parte superior de la nube arrastrados por las diversas corrientes ascendentes. La gota en su descenso adquiere una carga negativa.

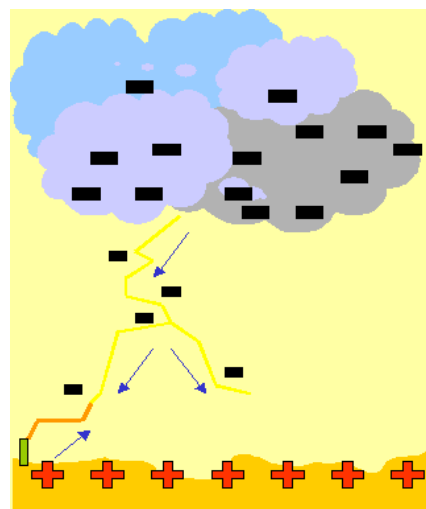
GENERACIÓN DEL RAYO.

Para el caso de un rayo nube a tierra, el fenómeno que se produce es el siguiente (para los otros tipos de rayos, el proceso es análogo)

- ❑ Cuando el campo creado por la carga negativa de la parte inferior de la nube es tal que produce la ruptura eléctrica del aire, se genera lo que se llama *descarga guía escalonada*. Esta descarga se produce por pasos, de nube a tierra, por el camino de menos resistencia del aire. Cada paso consiste en unas decenas de metros de distancia y una pausa de unos 50 μ s. En su camino hacia el suelo la guía va dejando una senda de gas ionizado. Algunas veces, esta guía se divide en varios caminos en su descenso hacia la tierra.

Cuando la guía toca el suelo, tenemos un “cable” conductor desde ese punto de contacto, hasta la nube. Los electrones de la parte inferior de la guía van al suelo, dejando tras de sí, cargas positivas que a su vez, atraen cargas negativas de más arriba, que van al suelo dejando cargas positivas más arriba que atraen mas cargas negativas de la nube Este fenómeno es el llamado rayo principal o rayo de retorno, y se desplaza hacia arriba. Es el que produce la luz tan brillante, y el calor que, originando una expansión rápida del aire, produce el trueno.

Después de que el rayo de retorno desaparece, el fenómeno puede finalizar, o si en la nube hay suficiente carga otra guía vuelve a bajar de forma continuada, sin escalones por el mismo camino que inicio la guía escalonada. Esta guía se llama *guía oscura* y cuando toca el suelo produce otro rayo de retorno que sube por el mismo camino.



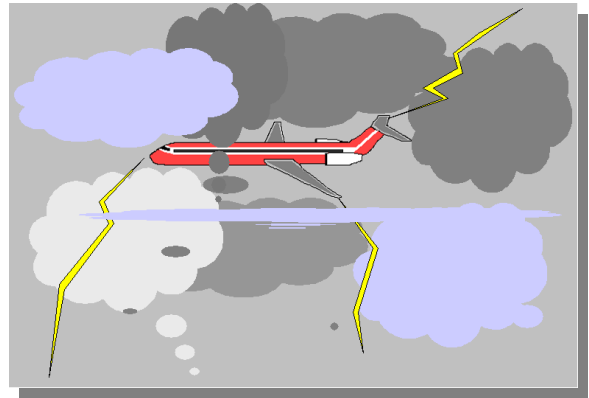
El promedio de rayos de retorno en cada proceso descrito es de 3 o 4, cada uno de ellos separado generalmente en el tiempo por 40 a 80 ms.

- Hay evidencia de que cuando la guía escalonada llega más o menos a un centenar de metros del suelo, dependiendo del campo eléctrico que se genere, sube una descarga desde el suelo hacia la guía para encontrarse con ella. Cuando esto ocurre se producen los fenómenos explicados más arriba.

En presencia de un objeto puntiagudo, tal como un edificio con una punta en el tejado, cuando la guía se acerca, los campos son tan intensos que se inicia una descarga desde la punta, la cual sube hasta la guía. El rayo tiende a caer en esa punta.

15.4.2-EL FENÓMENO DEL RAYO EN LAS AERONAVES.

- Se estima que, por ejemplo en USA, cada aeronave es alcanzada por un rayo una vez cada año. En experimentos en vuelo, se ha demostrado que existen dos tipos de acción de los rayos sobre una aeronave. El caso más frecuente ($\approx 90\%$) se trata de rayos generados por la intrusión de la aeronave en una región o nube con un campo electrostático intenso; en este caso, es la aeronave quien dispara el inicio de la guía escalonada entre centros de carga, saliendo de la aeronave y alejándose de esta. Parece ser que esto se debe a la

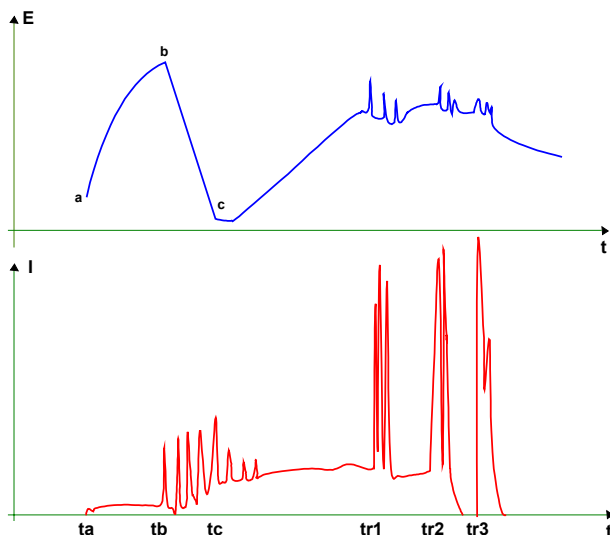


formación de guías de diferente carga sobre la superficie de la aeronave. El otro caso, es la intercepción por parte de la aeronave, de una rama de un rayo generado de forma natural. También se ha comprobado que la aeronave es raramente impactada por un rayo en medio de fuertes turbulencias y precipitaciones. La intensidad de estos dos factores no guarda relación con el número de impactos directos por un rayo. Además, cuando una aeronave vuela por la parte superior de la zona de tormenta, zona eléctricamente más activa, hay más probabilidad de ser impactada por un rayo cuando la temperatura baja de los -40°C , aunque esto no elimina el riesgo de que una aeronave pueda ser impactada a cualquier altitud o temperatura.

- Hoy en día, gracias a las técnicas de protección contra fenómenos como el rayo, el hecho de que un rayo impacte en una aeronave puede pasar, en muchos casos, desapercibido incluso para la tripulación de dicha aeronave. En otros casos, el hecho se puede percibir mediante un destello y un ruido fuerte, pero, aún así, nada serio le sucederá a la aeronave. Los últimos accidentes de aviación civil confirmados, que fueron directamente atribuidos a un rayo, datan, uno, del 8 de diciembre de 1963. En este fatal accidente, un rayo causó la explosión de un tanque de combustible del ala izquierda de un B707 de la Pan-Am. El otro de 1976, un B747 se estrelló después de que fuera impactado por un rayo y el tanque de combustible de su ala izquierda explotara.
- Generalmente, cuando un rayo impacta en una aeronave, este lo hace en un extremo de la misma, tal como el morro, las puntas de las alas o puntas de las palas. La corriente provocada por el rayo, viaja entonces por la superficie y estructuras conductoras de la aeronave y sale por cualquier otro extremo, como por ejemplo la cola. Puede haber más de dos zonas de impacto simultáneamente. Algunos pilotos han indicado que cuando esto ocurre, se detecta un parpadeo temporal de las luces y una interferencia de corta duración en los instrumentos.

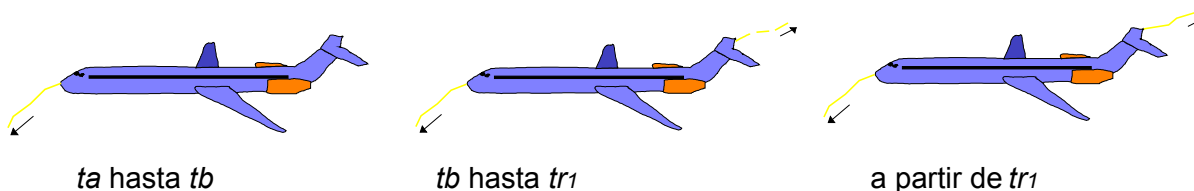
- Para el caso de rayos generados por la intrusión de una aeronave en un campo eléctrico muy intenso, experimentalmente se demuestra que existen dos fases en el proceso de interacción rayo y aeronave.

La primera fase se caracteriza por el inicio y desarrollo de la guía bidireccional. Esta fase comienza cuando una aeronave vuela en una región de tormenta en la cual el campo electrostático llega a valores críticos en el rango de 50-100 KV/m. En este caso, una descarga eléctrica (guía positiva) se inicia en la aeronave. Durante el desarrollo de esta guía (desde t_a hasta t_b), una carga eléctrica negativa se inyecta en la aeronave, produciendo una variación positiva del campo eléctrico en la superficie de la misma. Pocos milisegundos después, las condiciones eléctricas para el inicio de una guía escalonada negativa se alcanzan. Esta se desarrolla en la aeronave y se propaga en dirección opuesta a la anterior. Esta guía negativa, inyecta carga positiva en la aeronave, con lo cual reduce la carga negativa neta de la misma, produciendo una variación



negativa del campo eléctrico (desde t_b hasta t_c). Inmediatamente después, la guía positiva se acelera y ramifica, produciendo un aumento del campo eléctrico.

La segunda fase comienza (t_{r1}), con la propagación de la guía bidireccional por la evolución de caminos de retroceso. Esta fase se caracteriza por grandes impulsos de corriente llamados ráfagas, separados por unas pocas decenas de milisegundos.



EFFECTOS.

- Los efectos producidos por los rayos se dividen en dos categorías:

EFFECTOS DIRECTOS.

Daños físicos producidos por el acoplamiento del canal del rayo debido a las altísimas temperaturas que produce la alta corriente (entre 10 y 200 KA) que porta. Estos daños principalmente son la quemadura en los puntos de entrada y salida del rayo y la fatiga y deformación de los materiales afectados por el paso de la corriente.

EFFECTOS INDIRECTOS.

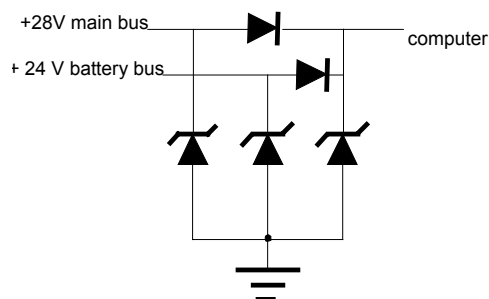
Peligros causados por el acoplamiento electromagnético asociado con el rayo. Cuando un rayo impacta en una aeronave, puede transcurrir más de un segundo antes de que los extremadamente rápidos e interactivos efectos electromagnéticos desaparezcan. Por otro lado, el efecto del campo magnético asociado, puede hacerse notar hasta distancias de 1 Km. del lugar del impacto del rayo.

Los efectos indirectos son principalmente transitorios de voltajes y corrientes inducidos en cableados de sistemas interconectados, antenas e interior de los equipos aviónicos.

PROTECCIÓN.

- La mayor parte de la superficie externa de las aeronaves está construida fundamentalmente de aluminio, el cual es un buen conductor de la electricidad. Si en esta superficie de aluminio de la aeronave, se asegura que no existen discontinuidades para la corriente, la mayor parte de la misma permanecerá en dicha superficie exterior. Para las aeronaves construidas con materiales compuestos, que son menos conductores que el aluminio (o nada conductores), se provee en la construcción capas o fibras conductoras insertadas en el material compuesto, para de esta manera obtener un camino fácil para la corriente del rayo.

Por otro lado, el rayo en su viaje por la capa externa de la aeronave, puede inducir transitorios en el cableado y en los equipos como computadores y elementos de control internos. Estos problemas causados por efectos indirectos, son evitados por un lado, mediante el apantallado de los cables y una correcta puesta a tierra de los mismos y por otro lado, con circuitos de protección contra sobre voltajes de las fuentes de alimentación, circuitos a base de diodos zener de potencia. En la figura se muestra uno de estos circuitos donde los diodos zener mantienen la tensión estable hacia el computador aunque existan transitorios en el lado de las fuentes de alimentación. Los dos diodos rectificadores realizan la función de desacoplo de fuentes y computador.



Cada circuito, o parte de un equipo que es esencial o crítico para la seguridad del vuelo y aterrizaje de una aeronave, debe ser verificada por el fabricante para ser protegida contra los efectos de los rayos de acuerdo con las regulaciones de las autoridades competentes de aviación.

El sistema de combustible es un punto importante a tener en cuenta, donde como ya sabemos, una simple chispa puede causar una catástrofe. La capa externa en la parte de los depósitos de combustible es lo suficientemente gruesa como para soportar un sobrecalentamiento por rayo. Todas las juntas estructurales y sujeciones son diseñadas para prevenir cualquier chispa cuando el rayo viaja de una sección a otra, además, cada vez se utilizan combustibles que producen vapores menos explosivos.

Otro punto importante en el que se requiere un esfuerzo especial en cuanto al diseño es el cono de morro (Radome) de las aeronaves. Este recinto, contiene el radar y otros instrumentos de vuelo. Para que el radar funcione correctamente, el volumen que lo contiene no puede ser conductor. Para proteger este espacio de los rayos, y a la vez que esta protección no interfiera con el radar, se colocan barras o tiras conductoras de desviación de la corriente del rayo, de tamaños y situación cuidadosamente estudiados, adheridas en la superficie externa.

En cualquier caso, diferentes zonas de la aeronave deben poder tolerar diferentes niveles de corriente. La aeronave se divide en zonas caracterizadas por posibilidad de impacto de un rayo sobre ellas, estas zonas ordenadas de mayor a menor según el nivel de esa probabilidad son *Zona 1A*, *Zona 1B*, *Zona 2A*, *Zona 2B* y *Zona 3*. La siguiente figura muestra la situación de dichas zonas.



SIMULACIÓN.

- Cualquier valoración sobre los efectos que los rayos pueden producir en una aeronave, lleva consigo tres aspectos: (1) pruebas para identificar las zonas de la aeronave que de forma preferente puedan ser susceptibles de impacto por los rayos, (2) pruebas que simulen un impacto directo de un rayo para ayudar a valorar los efectos directos, y (3) pruebas para simular los efectos indirectos y tener una estimación de los voltajes inducidos.

Estas pruebas realizadas mediante modelos computacionales, están clasificadas como test de alto voltaje y test de alta corriente. Los primeros, simulan el alto campo eléctrico que existe alrededor de la aeronave justo antes del ataque del rayo. Los segundos, simulan las corrientes que fluyen por la aeronave cuando esta forma parte del circuito del rayo.



Automatic Maintenance

16

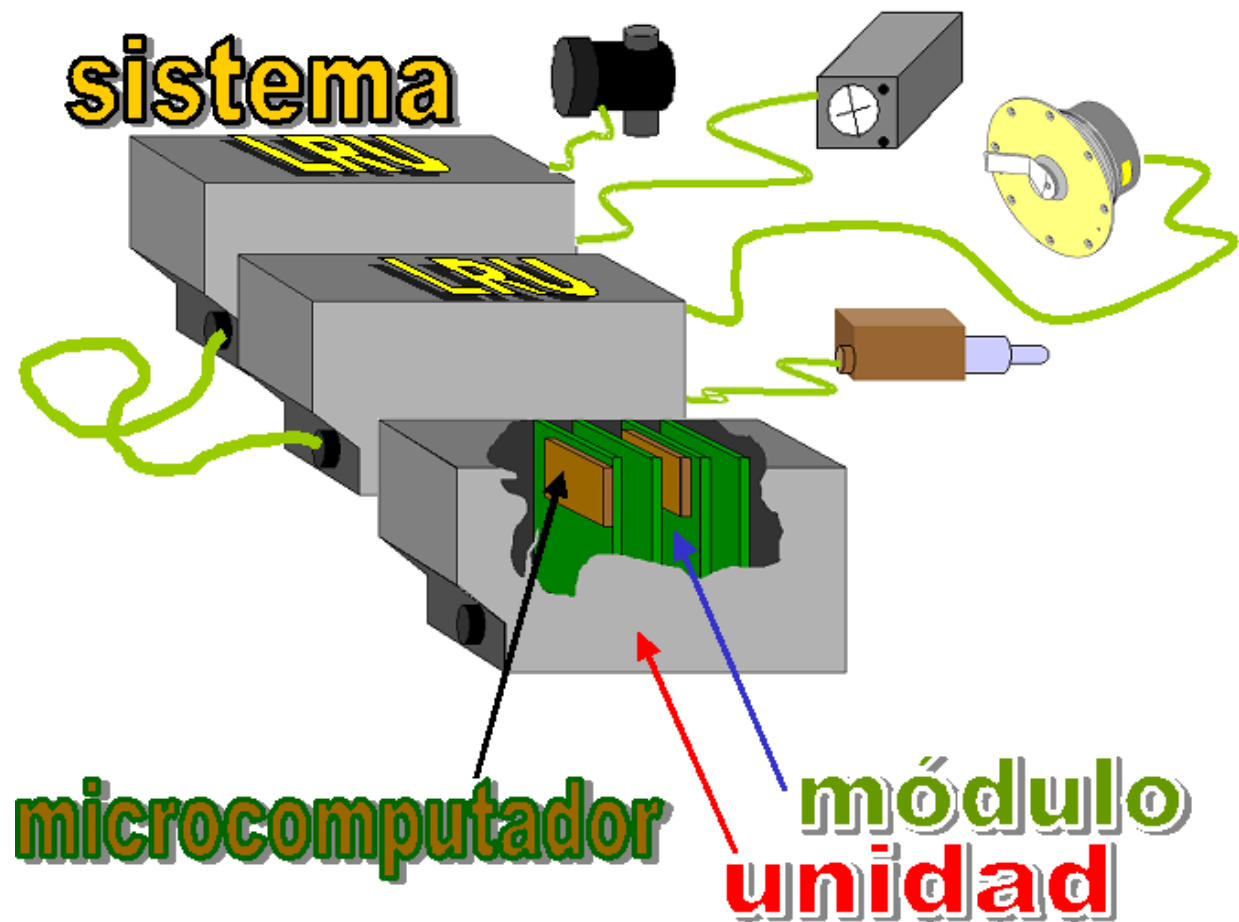
16.-MANTENIMIENTO AUTOMÁTICO.

16.1-AUTO PRUEBAS.

Los métodos de auto prueba son utilizados para asegurar que un computador, la unidad a la que pertenece, o incluso el sistema al que pertenece, está trabajando de forma correcta, conforme al propósito para el cual fue diseñado. A la vez, provee mecanismos para en caso de que un fallo sea detectado, realice la función correctora apropiada.

16.1.1-CATEGORÍAS DE AUTO PRUEBA.

Una auto prueba puede consistir en un simple chequeo, o un conjunto de chequeos con mayor o menor complejidad; Esto va a depender de factores como la aplicación, el grado de seguridad implicado y los costes.



Normalmente, las auto pruebas van a ser realizadas a través de la CPU, según rutinas programadas para este fin. Toda la diversidad de auto pruebas que pueden ser realizadas, van a caer dentro de una, o una combinación de alguna de las siguientes categorías: (1) Circuitos que componen el microcomputador, (2) circuitos a nivel de módulo y (3) circuitos a nivel de sistema.

1. CIRCUITOS QUE COMPONEN EL MICROCOMPUTADOR.

- En estos casos, la rutina de auto prueba está enfocada al análisis de los circuitos íntimamente relacionados con el microprocesador (incluido el mismo) y su forma de operación. La forma de auto prueba puede ser la de (a) conversión, la de (b) temporización o la de (c) validación de datos.

a.- Mediante el método de *conversión*, se realiza un análisis de la memoria de programa ROM, EPROM o EEPROM para determinar si el programa ha sufrido cualquier tipo de alteración. Este análisis se puede realizar efectuando el checksum al programa y comparándolo con valores guardados previamente. Según esto, un código especial obtenido mediante un algoritmo matemático, se coloca en determinadas posiciones de la memoria de programa cuando esta es programada. Cuando se comprueba la integridad de los datos, se genera un código basado en el mismo algoritmo. Si el código obtenido en el test, no coincide con el almacenado en la memoria, indica que ha habido modificación.



b.- El control mediante la *temporización*, consiste en el uso de un circuito temporizador independiente (Watchdog) que supervise que las operaciones tanto de flech, como de movimiento de datos, se realizan dentro de los tiempos previstos según el diseño del microcomputador. En caso de que exista anomalía en la ejecución, el watchdog avisa de la misma al μ procesador, generalmente mediante una línea de interrupción o similar.

c.- Con el método de *validación de datos*, se comprueban elementos como la memoria RAM, los registros del μ P, puertos de I/O, etc. Mediante la escritura de diferentes patrones de palabras tales como, todo "unos", todo "ceros", "unos" y "ceros" alternados, etc., en la memoria o registros, y posterior lectura de esos elementos, se comprueba que efectivamente lo que se escribe es lo que se guarda, es decir no existe cambio en ninguno de los bits, y por lo tanto, cada celda de la memoria o registros, tanto del μ procesador como de las unidades I/O están operando correctamente.

2. CIRCUITOS A NIVEL DE MÓDULO.

- La auto prueba en estos circuitos es realizada también a través de la CPU, pero en este caso, los circuitos sobre los que se efectúa dicha prueba, no son los componentes de lo que es la estructura básica de sistema microcomputador (μ procesador, memoria, unidad I/O y buses). Las formas de auto prueba de los módulos puede realizarse mediante (a) realimentación (feedback), (b) retorno (wrap-around) y (c) comparación.

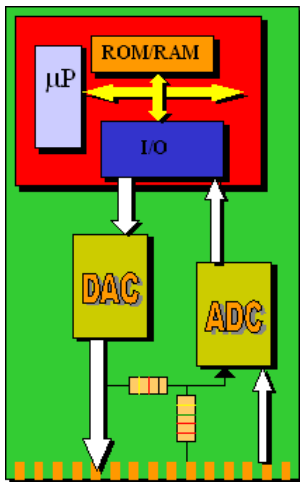
a.- *Realimentación* es el método más utilizado. Parte de la información o señal de salida es incorporada de nuevo como entrada para que mediante el análisis de esta muestra se pueda valorar si la salida ha sido la apropiada o no. El análisis se realiza para cada operación de salida.

b.- *Retorno* es otra forma de realimentación de la información de salida a la entrada, pero en este caso se trata de realimentar toda la salida, no solamente una porción de la misma para que sea analizada. A menudo, cuando la información es enviada al exterior del μ computador, este envío se realiza a través de registros que actúan como buffers, es decir, elementos que sirven de acople entre la circuitería del computador y la circuitería de destino. Este acople se refiere principalmente al ajuste de tiempos entre ambos circuitos. Cuando la salida es enviada al registro, el computador realiza una lectura del mismo para comprobar la información que ha sido depositada en el mismo.

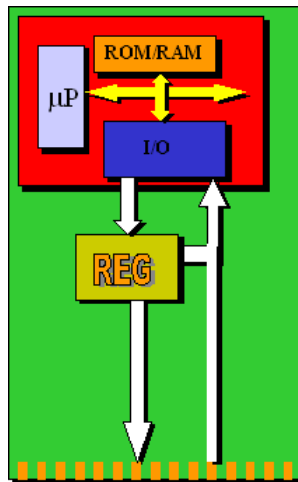


En el caso de salida analógica, el proceso es análogo al descrito, simplemente que en este caso, se realiza la conversión primero de digital a analógico, obteniéndose la salida correspondiente, y posteriormente, de analógico a digital para que pueda ser tratada por el μ computador.

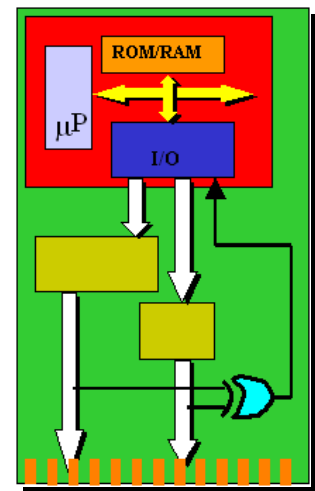
c.- *Comparación.* Ciertos circuitos que realizan la misma función dentro de un módulo, son comparados en el sentido de comprobar que sus salidas están dentro de los márgenes previsibles. Si el resultado de esta comparación no es satisfactorio, origina una situación de fallo indicada con el correspondiente aviso y tratada convenientemente para ser corregida.



a.- realimentación



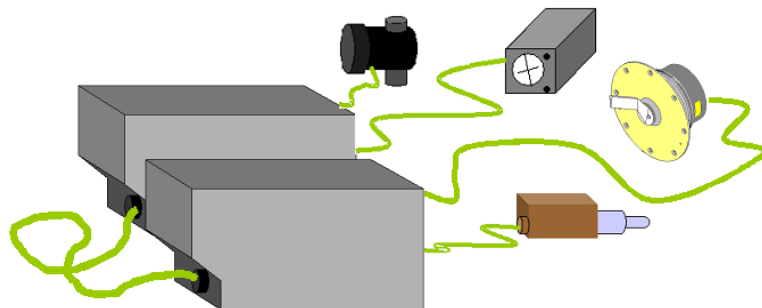
b.- retorno



c.- comparación

3. CIRCUITOS A NIVEL DE SISTEMA.

- Las auto pruebas a nivel de sistema llevan consigo métodos más largos y complejos que los realizados a los anteriores niveles, debido a la heterogeneidad que el propio sistema lleva implícito, esto es partes eléctricas combinadas con partes mecánicas, transductores, actuadores etc. Habitualmente se usa una acción mecánica en combinación con estímulos eléctricos para comprobar la integridad del sistema. Esta comprobación lleva asociada un incremento en el hardware (relés, transductores, conectores, etc.) que se necesita para la realización de la propia prueba con el consiguiente incremento del peso de abordó, lo que hace que la tendencia sea a la realización de la auto prueba a nivel de módulo.



Los métodos para la realización de la auto prueba a un sistema, son los mismos que los utilizados a nivel de módulo; Mediante la realimentación se pueden monitorizar la posición de un elemento dentro del sistema. El retorno puede ser también utilizado para comprobar la salida digital enviada. El método de comparación puede ser utilizado por ejemplo para comparar señales de gobierno de superficies de control, de las que se conoce la relación entre ellas.

16.1.2-MÉTODOS DE AUTOPRUEBA.

Los métodos de auto prueba se determinan durante el proceso de diseño del hardware y software del sistema. Los más comunes son el método de (1) energización (Power-Up) y el (2) continuo.

1. ENERGIZACIÓN.

- Durante el proceso de energización (conexión del sistema a la alimentación después de cortes de alimentación prolongados de mas de 0,2 s) y antes de que se inicie la ejecución del programa principal según la función preestablecida para ese sistema, el μ computador realiza un auto-chequeo en sus propios componentes, es decir, las memorias de datos y de programa, unidades de entrada y salida, circuitos de temporización y μ procesador. Además, mediante la aplicación de salidas específicas, chequea las salidas y los actuadores conectados al sistema, buses de datos, así como también, las entradas de información al mismo mediante la aplicación de niveles de prueba para comprobar su comportamiento.

Una vez realizado el chequeo en la energización, si no se detecta ningún fallo, el sistema da paso al programa principal. Si hay fallos, el programa de chequeo puede reiniciar el sistema y registrar el fallo en la memoria de mantenimiento.

2. CONTINUO.

- El chequeo se realiza durante la ejecución del programa principal, en momentos específicos previamente establecidos durante el diseño. Unas rutinas de chequeo son intercaladas en el programa principal en lugares estratégicos del mismo, donde un fallo en la operación del sistema pudiera ser de consecuencias fatales. Si durante el chequeo se encuentra un funcionamiento erróneo del sistema comprobado, la propia rutina desconectaría este sistema de la ejecución de la tarea y conectaría el correspondiente redundante para la realización de la misma.

El método continuo, realiza muchas de las comprobaciones que se hacen también durante el método de energización (chequeo del μ computador, unidades I/O, entradas, salidas, etc.), aunque es más corto que este, debido a que tiene que compartir tiempo con la ejecución del programa principal.

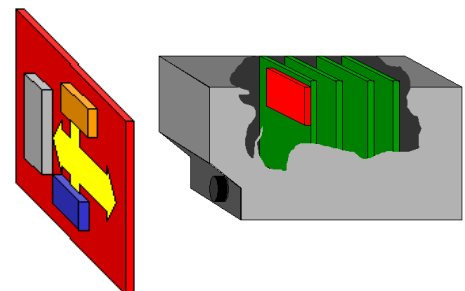
16.1.3-TIPOS DE AUTOPRUEBA DE COMPUTADORES.

Los computadores utilizados en las aeronaves, pueden utilizar internamente uno o varios μ computadores, cada uno de los mismos procesando la misma, similar o diferente información. Además, estos μ computadores pueden realizar auto-chequeos, o chequear los otros dentro del mismo computador.

La autoprueba de computadores se puede clasificar en tres tipos: (1)individual, (2) interactivo y (3) vigilancia.

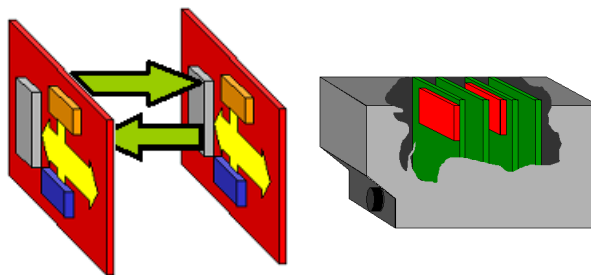
1. INDIVIDUAL.

- Cuando el computador solamente contiene un μ computador o unidad de procesamiento (μ procesador, RAM, ROM, I/O) el auto chequeo se limita al tipo de μ computador básico descrito.



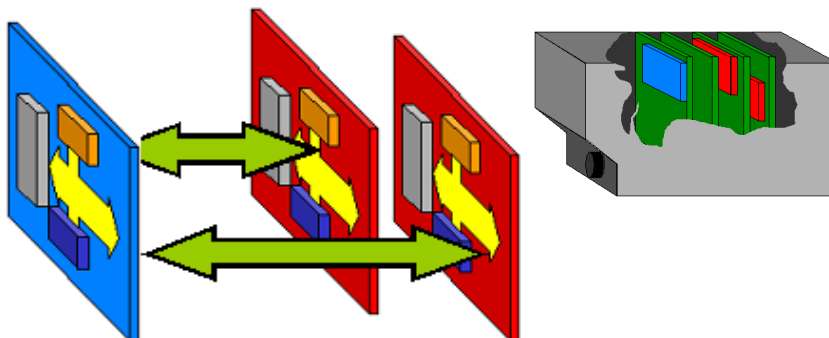
2. INTERACTIVO.

- Cuando el computador contiene más de una unidad de procesamiento, se puede establecer, además de los auto chequeos de los μ computadores básicos para cada una de ellas, uno interactivo entre las unidades de procesamiento existentes, de tal manera que se realice un chequeo entre ellas.



3. VIGILANCIA.

- Un computador puede tener una unidad de procesamiento separada, con el único propósito de vigilar el funcionamiento de otras unidades dentro del computador, incluso si estas otras unidades procesan información diferente. El μ computador vigilante, en su función, puede procesar exactamente la misma información que el μ computador principal, para realizar una comparación de resultados entre ambos.



16.1.4-FALLOS EN LA AUTOPRUEBA. ACCIONES.

Las formas más comunes de alertar sobre un fallo como resultado de la ejecución de una auto prueba son mediante una señal luminosa o bandera de fallo en cabina y mediante mensajes en los displays. Otros indicadores situados directamente en las LRU-s, pero visibles desde el exterior de las mismas, son las bolas de fallo (fault balls) y los diodos LED. Las bolas de fallo son elementos electromecánicos que se ponen en una posición determinada para indicar fallo. Estos indicadores avisan al personal de mantenimiento de fallos en las LRU-s.

Las acciones a tomar por la detección de un fallo como resultado de la ejecución de una auto prueba, van a depender de lo crítico que sea el circuito o sistema afectado, realizando el mismo una o más de las siguientes acciones:

- Continúa funcionando en un modo distinto.
- Comienza la ejecución del programa desde el principio.
- Comienza el programa desde el principio un número determinado de veces antes de terminar con la operación.
- Transfiere el control a un sistema redundante.

16.2-BUILT-IN TEST EQUIPMENT (BITE).

El propósito principal del sistema BITE es facilitar la labor de mantenimiento de los sistemas de las aeronaves en cuanto a la detección y aislamiento de fallos, y a la verificación de la operación de los sistemas. Los beneficios que puede aportar este sistema son: mejora en la calidad del mantenimiento, reducción en costes de mantenimiento, simplificación de los procedimientos de resolución de problemas y por lo tanto disminución de retardos, reducción del mantenimiento directamente sobre la aeronave y asistencia en el mantenimiento en taller.

El BITE es la parte de un computador, sensor, o actuador, que supervisa continuamente el correcto funcionamiento del elemento del cual forma parte. Puede, por lo tanto, detectar una anomalía en tiempo real.

El sistema BITE es una herramienta que sirve principalmente para realizar un mantenimiento en línea. Esta herramienta proporciona al técnico de mantenimiento la información necesaria para que este pueda determinar la operatividad de un sistema, y obrar en consecuencia.

16.2.1-CLASIFICACIÓN DE FALLOS.

Generalmente, la acción para realizar un mantenimiento no programado, será como respuesta a una petición, por parte de la tripulación, referente a un determinado sistema. En esta petición, el técnico de mantenimiento, será informado mediante una identificación de la anomalía o problema encontrado en el sistema por dicha tripulación. Previamente, la tripulación habrá tenido conocimiento de dicha anomalía a través de los indicadores de advertencias de fallos situados en el cockpit.

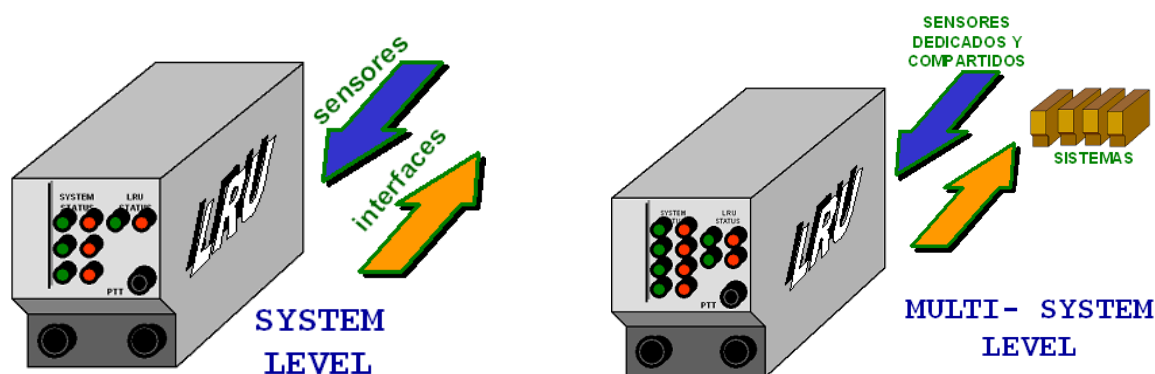
En cuanto al nivel de los fallos, estos se clasifican en **fallos graves** y **fallos fuera de tolerancia**. En el primer caso, el fallo es extremo, haciendo que la unidad afectada no realice la función para la que fue diseñada. En el segundo caso, la unidad realiza su función sobrepasando ciertos límites, o no realiza su función bajo todas las condiciones, pero funcionando de forma aceptable bajo la mayor parte de ellas.

En cuanto al tipo de fallos, estos se clasifican en **activos** y **pasivos**. Los fallos pasivos no son normalmente indicados a la tripulación, y es el propio sistema el que se encarga de eliminarlos. Los fallos activos son indicados a la tripulación o al equipo de tierra. Estos últimos se pueden subdividir a su vez en fallos **“hard”** y fallos **“soft”**. Los primeros cambian el comportamiento en cuanto a la operación del sistema. Generalmente se deben a un fallo de algún componente de la LRU, cableado o fallo en algún sistema relacionado. Los segundos no cambian de forma inmediata la forma de operación del sistema y pueden ser debidos al fallo en algún componente no esencial.

16.2.2-CATEGORÍAS BITE.

Los niveles BITE pueden ser agrupados en tres categorías: (1) *BITE a nivel de componente*, donde las pruebas BITE se limitan a una LRU específica. (2) *BITE a nivel de sistema*, donde las pruebas se limitan al sistema donde el BITE está implementado y (3) *BITE a nivel de multi-sistema* donde el BITE implementado en un sistema es utilizado para chequear otro sistema independiente o relacionado. Un ejemplo clásico de esta última categoría es el sistema BITE del sistema de piloto automático, donde el BITE se responsabiliza de muchos sistemas relacionados con él en la aeronave.





16.2.3-TIPOS DE BITE.

Según el grado en el cual el técnico de mantenimiento en línea participa en el BITE, este se puede clasificar en tres tipos: (1) *BITE manual*; En este tipo, el técnico es requerido durante la prueba para operar manualmente ciertos elementos como switches, botones o realizar acciones según requerimiento. La forma más simple y mejor entendida de resultados de salida es mediante la indicación de “GO” o “NO GO”, según el resultado del test. (2) *BITE semiautomático*. En este tipo, la participación del técnico queda prácticamente limitada a la inicialización del test. Se utilizan técnicas donde los fallos pueden ser memorizados para posterior tratamiento de esa información. (3) *BITE automático*. En este caso, el BITE proporciona una continua vigilancia de la actividad de un sistema o LRU. El propio BITE inicia el autotest y evalúa los resultados, almacenando los fallos para posterior mantenimiento en taller.

16.2.4-CONTROLES, INDICACIONES Y SALIDAS EN EL BITE.

DISEÑO BITE.

En el diseño estándar de BITE, el panel frontal de la LRU tiene el pulsador de test y una indicación de GO / NO GO que se muestra solamente si la función “push to test” es activada. La indicación GO afecta a todo el sistema, mientras que la indicación de NO GO es indicada para cada componente de dicho sistema (LRU, CDU, antena, etc.), siempre que la cantidad de componentes lo haga viable. Hoy en día el diseño de BITE abarca muchas variedades. Podemos encontrar BITE cuyos tests son realizados desde el compartimiento de vuelo, y cuyos resultados se muestran en pantallas o por impresora.

PRECONDICIONAMIENTOS PARA EL BITE.

Muchos diseños BITE requieren de condiciones especiales (por ejemplo “Airplane on GND”, entradas de sensores relacionados, operación o posicionamiento de superficies de control, etc.) para una realización correcta y precisa que dé resultados válidos. Las precondiciones pueden incluir la necesidad de personal adicional para activar y monitorizar los resultados BITE cuando el control de este está situado en el panel frontal de una LRU y los resultados son mostrados en una posición remota. También puede ser necesario personal adicional para asegurar que fuera de la aeronave existen los márgenes de seguridad apropiados en el espacio donde van a moverse superficies de control.

SALIDAS EN EL BITE.

Las salidas BITE normalmente consisten en algún tipo de indicador o indicadores visuales localizados en el panel frontal de una o más LRU-s, además de una señal eléctrica para uso con indicadores remotos, pantallas o AIDS (Airborne Integrated Data System).

Una señal tipo discreta es normalmente generada como salida cuando el BITE detecta una condición de “NO GO”. La señal es utilizada para activar dispositivos remotos tales como Warning Lights, flags (banderas) electromecánicas, mensajes por pantalla, sonidos, etc.

DISPLAYS BITE.

Los displays BITE están generalmente localizados en la LRU “principal” montada en el rack de equipos. Esto particularmente se cumple cuando algunas de las LRU-s de un sistema están montadas en lugares menos accesibles de la aeronave. Por ejemplo, algunos sistemas de radio-altímetro, tienen tres displays, uno para indicar antena defectuosa, otro para los indicadores y el tercero para la unidad transmisora-receptora, localizados en el panel frontal de esta última unidad. Por otro lado, algunos displays BITE son de algún tipo de consola de control/display de BITE centralizado, situada en el compartimiento de vuelo u otro lugar conveniente.

La presentación de la información varía de unos diseños a otros. En unos casos las indicaciones se dan mediante indicadores de “GO” “NO GO” tales como LED-s o bolas de fallo.

En otros casos son utilizados códigos de diagnóstico expresados en decimal, binario, hexadecimal o código alfanumérico representados mediante displays LED de siete segmentos, matrices LED, LCD o CRT.

Cuando son utilizados indicadores de tipo mecánico o magnético, los colores utilizados para indicar una condición de “GO” o “NO GO” son los siguientes:

GO: negro o el mismo color del panel donde está montado el indicador.

NO GO: amarillo brillante.

Cuando son LED-s los utilizados para la indicación, los colores estándar son los siguientes:

GO: verde, blanco o apagado.

NO GO: rojo, ámbar o luz encendida

MEMORIA DE FALLOS.

Cada fallo identificado en vuelo mediante el monitorizado continuado del sistema, o identificado en tierra mediante el iniciado manual (push to test), es guardado en una memoria no volátil. Esta información es guardada en la memoria para la propia LRU y para los periféricos que no tengan la capacidad de guardar sus propios fallos. Los fallos en un vuelo determinado, o en vuelos anteriores pueden ser anunciados automáticamente por el BITE o bajo solicitud.

CONTROLES DE RESET.

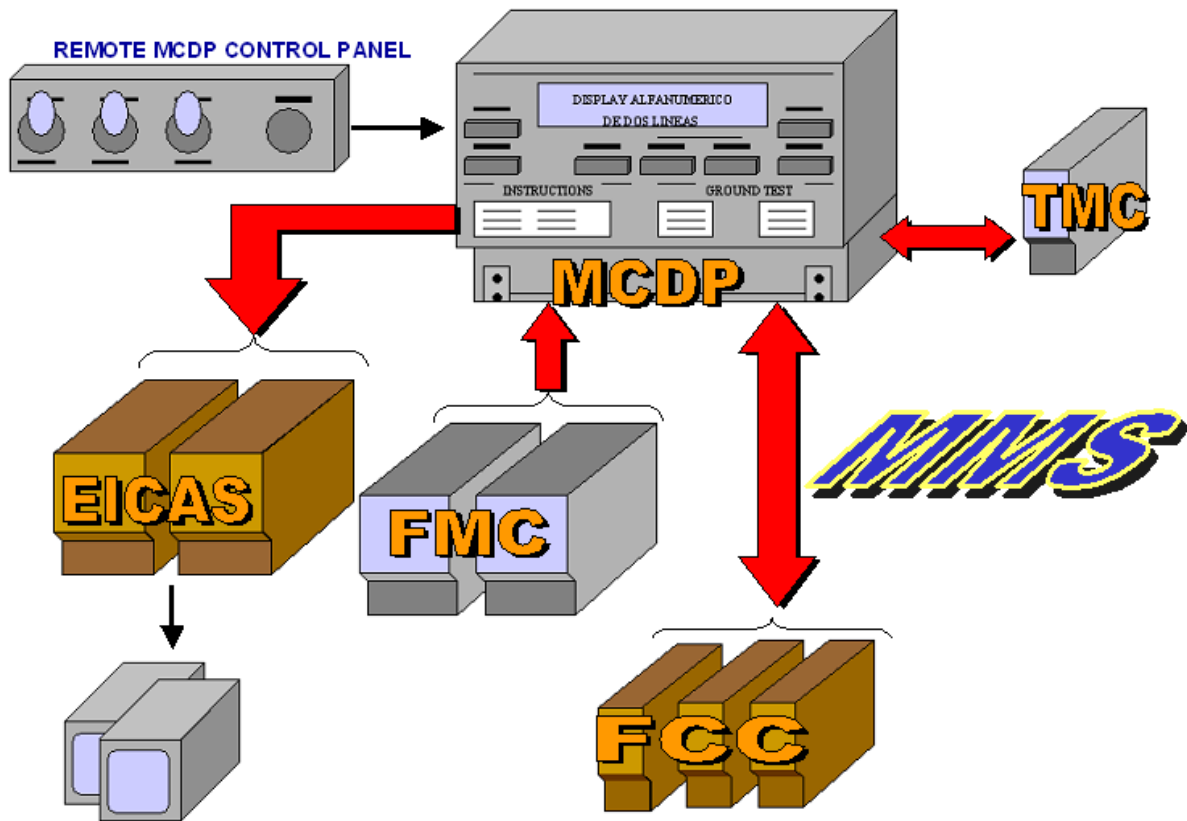
El propósito de los controles de reset es permitir al técnico de mantenimiento de línea eliminar la indicación de fallo después de que la LRU que ha dado dicho fallo haya sido remplazada o se haya realizado la acción correctiva apropiada.

Los indicadores de BITE en una LRU, pantalla o panel de control/display para indicar fallo en alguna LRU o sistema, deben ser reseteables por el técnico. El control de reset está generalmente localizado en la LRU individual o tiene una función remota desde el compartimiento de vuelo. El control de reset de mantenimiento en línea, está diseñado para que no pueda ser reseteada cualquier memoria que guarde información para el mantenimiento en taller.

16.2.5-EJEMPLOS BITE.

MAINTENANCE MONITOR SYSTEM. MMS.

Para entender un poco mejor la filosofía BITE veamos un ejemplo en la forma del **MMS** (Maintenance Monitor System), ATA 22-41-00

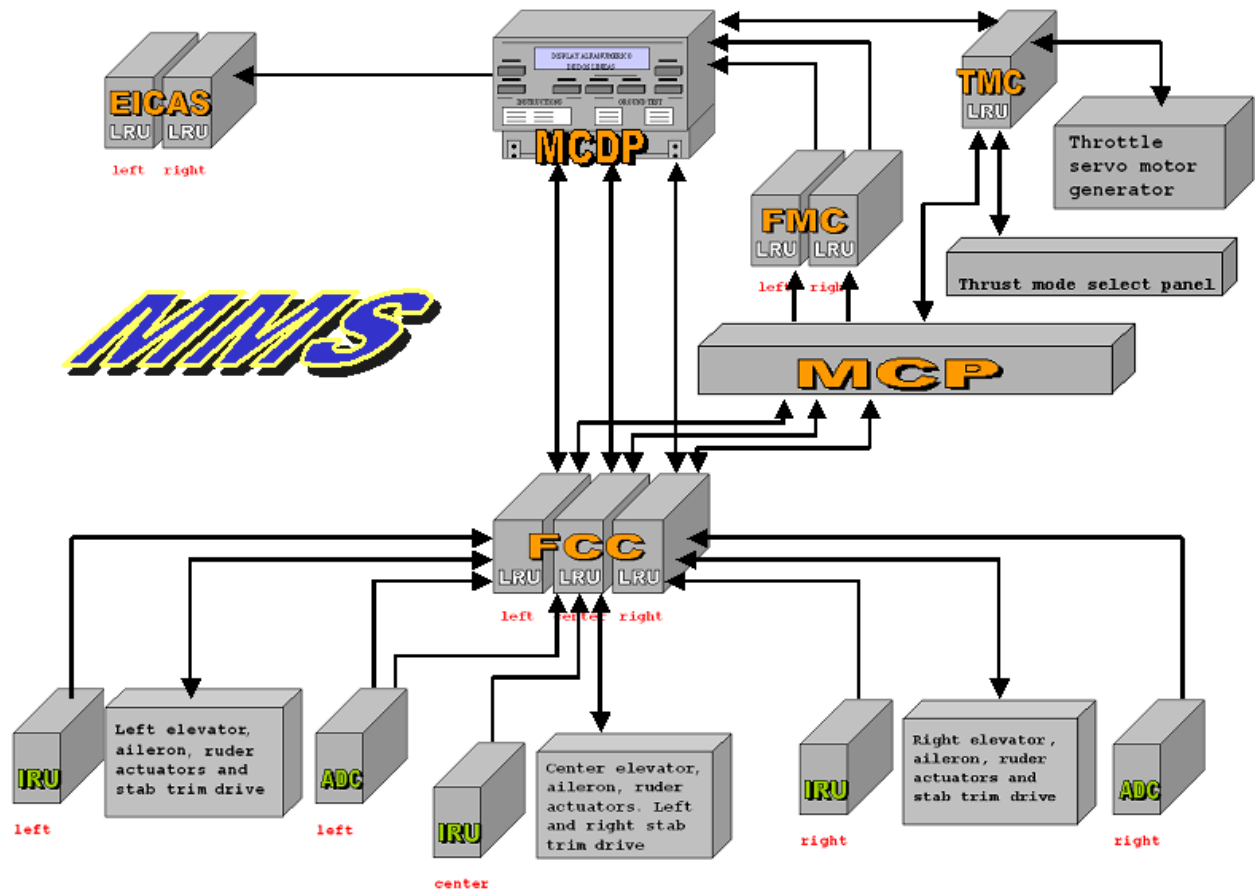


El MMS consiste en un Maintenance Control Display Panel (**MCDP**) que monitoriza el estado de los Flight Control Computers (**FCC**), Flight Management Computers (**FMC**), Thrust Management Computers (**TMC**) y los sensores relacionados con ellos. El MCDP interroga a los computadores después de cada aterrizaje y guarda hasta cinco fallos de vuelo por computador en una memoria no volátil para posterior interrogación por parte del personal de mantenimiento.

Una CPU en el MCDP controla y procesa órdenes y funciones de test en tierra, y fallos en tierra y en-aire. El MCDP muestra en una pantalla situada en el frente de la unidad fallos de vuelo y tierra, número de test, nombre o estado e instrucciones del operario.

El test en tierra puede realizarse desde la propia unidad y visualizar los resultados en la misma, o puede ser llevado a cabo desde el compartimiento de vuelo a través del Remote MCDP Control Panel y ser visualizado a través del EICAS (Engine Indication and Crew Alerting System).

Las interfaces primarias con el MCDP son cada uno de los FCC, FMC, y TMC. Los secundarios son los sensores individuales de cada uno de los anteriores sistemas. Ambos, las interfaces primarias y secundarios son monitorizados por el MCDP durante las funciones de test en tierra.



-Interfaces primarias:

Los FCC (izquierdo, derecho y centro) transmiten datos de fallos al MCDP, y reciben datos de test en tierra a través de un bus ARINC 429. Líneas analógicas de discretos son utilizadas para el control del test en tierra.

Los FMC (izquierdo y derecho) transmiten datos de fallos al MCDP a través de un bus ARINC 429. Los tests en tierra del MCDP no afectan a los sistemas FMC.

El TMC utiliza también un bus ARINC 429 para transmitir datos de fallos al MCDP y recibir datos del test en tierra de este. Una línea analógica de discreto es utilizada para el control del test en tierra del TMC por parte del MCDP.

- Interfaces secundarias:

El MCP suministra señales de control y recibe datos de estado de cada FCC, FMC y TMC a través de un bus ARINC 429.

Cada FCC suministra señales analógicas de control a sus servos de control y recibe señales analógicas de posición del servo. Cada FCC recibe datos a través de un bus ARINC 429 de la IRU y ADC asociadas a él.

El TMC recibe señales de control a través de un bus ARINC 429 procedentes del Thrust Mode Select Panel (TMSP). El TMC también transmite datos de estado al TMSP, por un bus ARINC 429. El TMC proporciona señales analógicas de control y recibe señales analógicas de posición del Autothrottle Servomotor Generator.

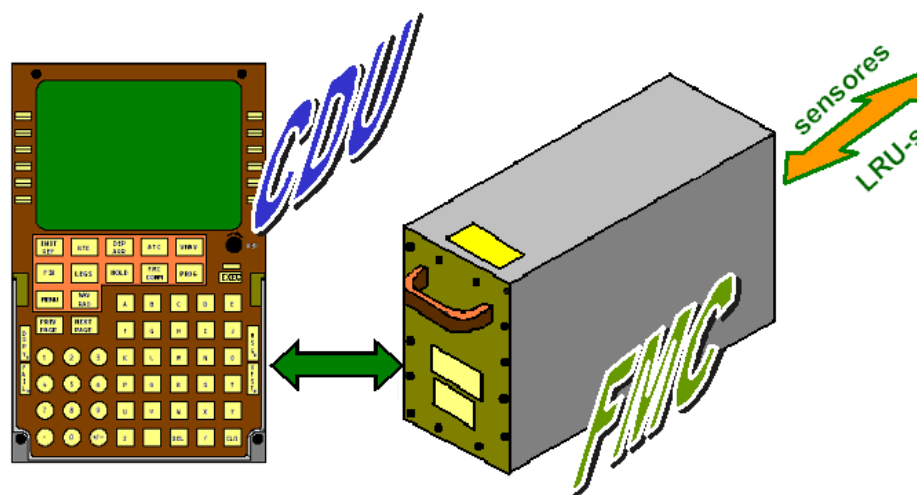
El MCDP obtiene y guarda de forma automática (en una memoria EAROM) datos de fallo procedentes de cada FCC, FMC y TMC, después de cada aterrizaje. La CPU en el MCDP inicia un energizado automático y una rutina de auto prueba después de cada aterrizaje, cuando una condición de “en-tierra” es captada por cualquier relé sensor aire/tierra (el MCDP no está normalmente energizado en vuelo, a menos que sea energizado por la tripulación). Si la auto prueba finaliza de forma positiva, se inicia la rutina de fallos de vuelo. Mediante esta rutina se solicita datos de fallos a los seis computadores primarios (3 FCC, 2 FMC, 1 TMC). La CPU normalmente desenergiza el MCDP después de que todos los fallos de vuelo han sido recibidos o después de tres minutos. Si la auto prueba falla, la CPU desenergiza el MCDP y por lo tanto no se transfiere ningún dato de fallos de vuelo.

El personal de mantenimiento puede realizar el proceso anterior de forma manual presionando de forma momentánea el switch ON/OFF localizado en el panel frontal del MCDP. De esta manera puede tener acceso a los datos de fallos del último vuelo, y vuelos anteriores. También pueden llevar a cabo un test en tierra, para comprobar, por ejemplo, la operatividad de una LRU que ha sido remplazada o realizar una auto prueba a cada uno de los seis computadores primarios.

FLIGHT MANAGEMENT SYSTEM. FMS.

Otro sistema que nos ayudará también a entender la filosofía BITE es el BITE del Flight Management Control System (**FMCS**), ATA **34-61-00**.

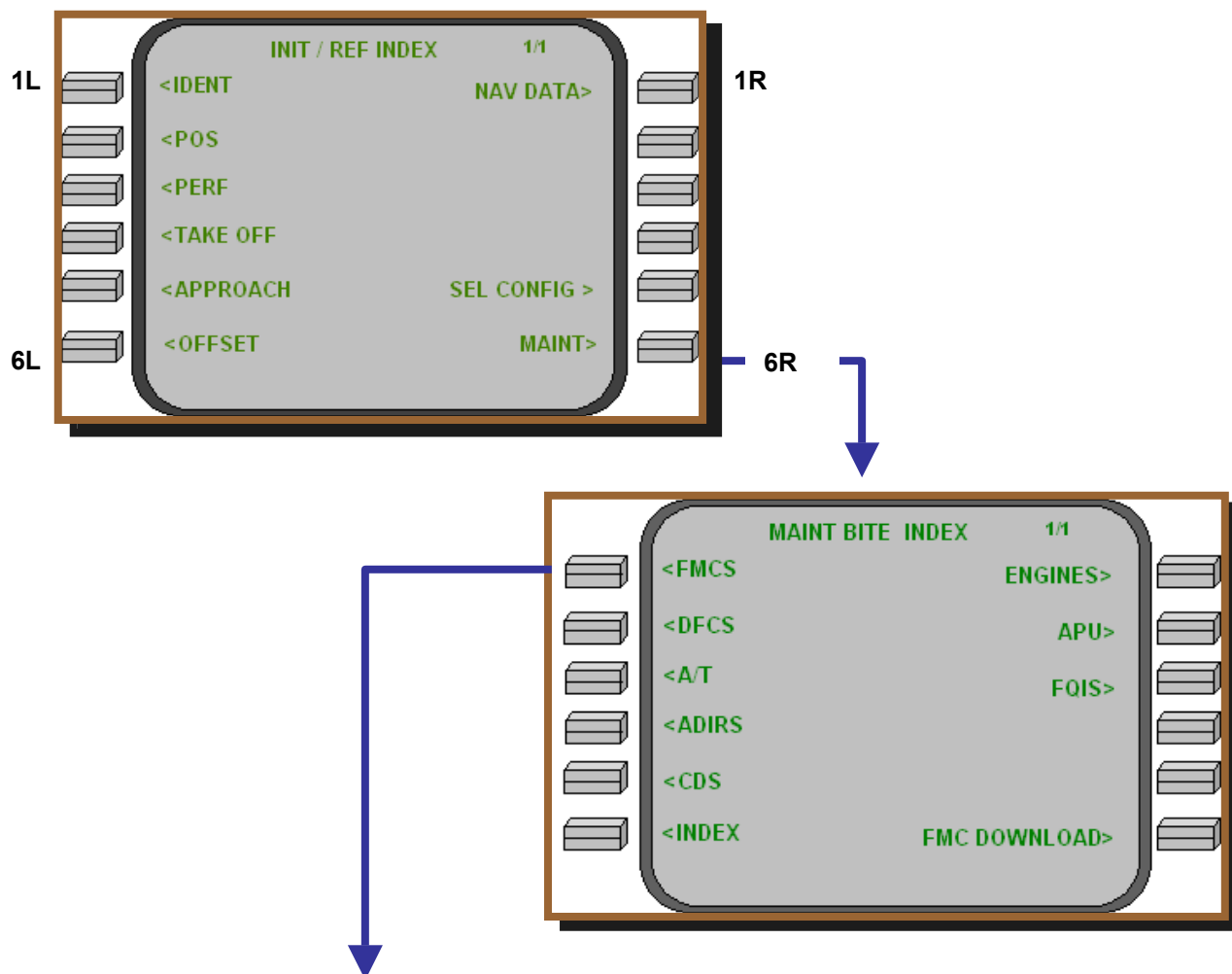
El FMCS está formado por las Control Display Units (**CDUs**) situadas en el compartimiento de vuelo y el Flight Management Computer (**FMC**) situado en el compartimiento de aviónica. El BITE está en el FMC y es operado a través de las CDUs.



Las funciones BITE monitorizan continuamente el estado del FMC, de las CDU-s y los sensores de los sistemas que interaccionan directamente con el FMCS. Esta monitorización se realiza en todo momento.

Al mantenimiento BITE se accede a través de una serie de menús de pantalla que van apareciendo en la CDU. El técnico va moviéndose a través de estos menús y realizando las oportunas selecciones de los mismos mediante una serie de teclas de selección de línea, **LSK** (*Line Select Keys*), situadas al lado derecho (R) e izquierdo (L) de la pantalla. Estas teclas están identificadas mediante un número del 1 al 6, correspondiendo el 1 a las teclas superiores y el 6 a las inferiores. Para identificar una tecla en concreto, el número de la misma va acompañado de la letra, R si dicha tecla está en el lado derecho de la pantalla, o la letra L, si la misma está en el lado izquierdo.

Si el avión está en tierra y la velocidad respecto al suelo es menor de 20 knots, en la página INIT/REF INDEX aparecerá un menú identificado como MAINT>, al cual se accede mediante la tecla LSK 6R. (Si el avión estuviera en el aire, en la misma posición del menú MAINT>, aparecería ahora el menú NAV STATUS>). Accionando la tecla LSK 6R pasamos a la página de MAIN BITE INDEX donde se muestra la serie de subsistemas del avión siguiente: FMCS, DFCS, A/T, ADIRS, CDS, ENGINE, APU Y FQIS. Para acceder al BITE del FMCS el técnico deberá pulsar la tecla LSK 1L. Ver la figura siguiente.



Una vez en la página de FMCS BITE encontramos las siguientes categorías de mantenimiento FMC:

INFLT FAULT. El FMC registra sus fallos en vuelo y los fallos en vuelo de las CDUs y las LRUs con las que interactúa. Tiene una capacidad de almacenamiento para nueve vuelos y veinte fallos por vuelo. Además de lo anterior, también registra fallos que ocurren en tierra.

CDU TEST. Test de las pantallas y teclado de las CDUs.

SENSORS. En esta página se muestra el estado de las interfaces de las LRUs y la validez de sus entradas.

DISCRETES. En esta página se muestra el estado de las señales discretas analógicas.

FIXED OUTPUTS. Muestra los valores de salida que son enviados a los componentes seleccionados en la pantalla de la CDU.

MODEL/ENG. Esta página muestra el estado de pines programables referentes a la configuración modelo/motor de avión.

SW OPTIONS. En esta página se muestra el estado de la opción de software de la base de datos.

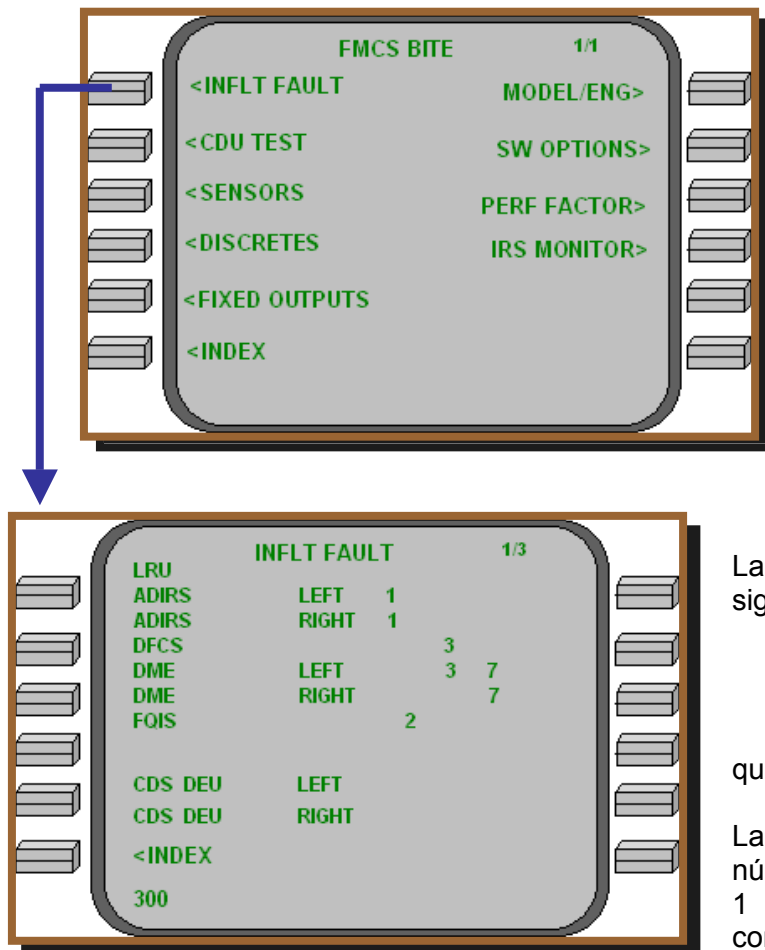
PERF FACTR. En esta página se muestran los códigos y factores de funcionamiento o ejecución (performance).

IRS MONITOR. Esta página muestra el régimen de error de posición ADIRU calculado por el FMC.

**INFLT FAULT.*

Para el registro de fallos, un nuevo vuelo comienza cuando la velocidad del avión respecto a tierra (groundspeed) es superior a 20 knots y el avión se configura en modo aire. El vuelo finaliza cuando el avión toca tierra y su groundspeed es menor de 20 knots. Los fallos

que suceden en tierra son guardados como fallos de tierra. Estos últimos fallos, son borrados de la memoria del FMC cuando comienza un nuevo vuelo. Si el fallo está aún presente en esta situación, este fallo se convierte en "fallo en vuelo".



La página de fallos en vuelo muestra la siguiente información:

Nombre de la LRU/subsistema
El vuelo donde sucedió el fallo (1 a 9).
Donde sucedió el fallo. Una G indica que el fallo sucedió en tierra.

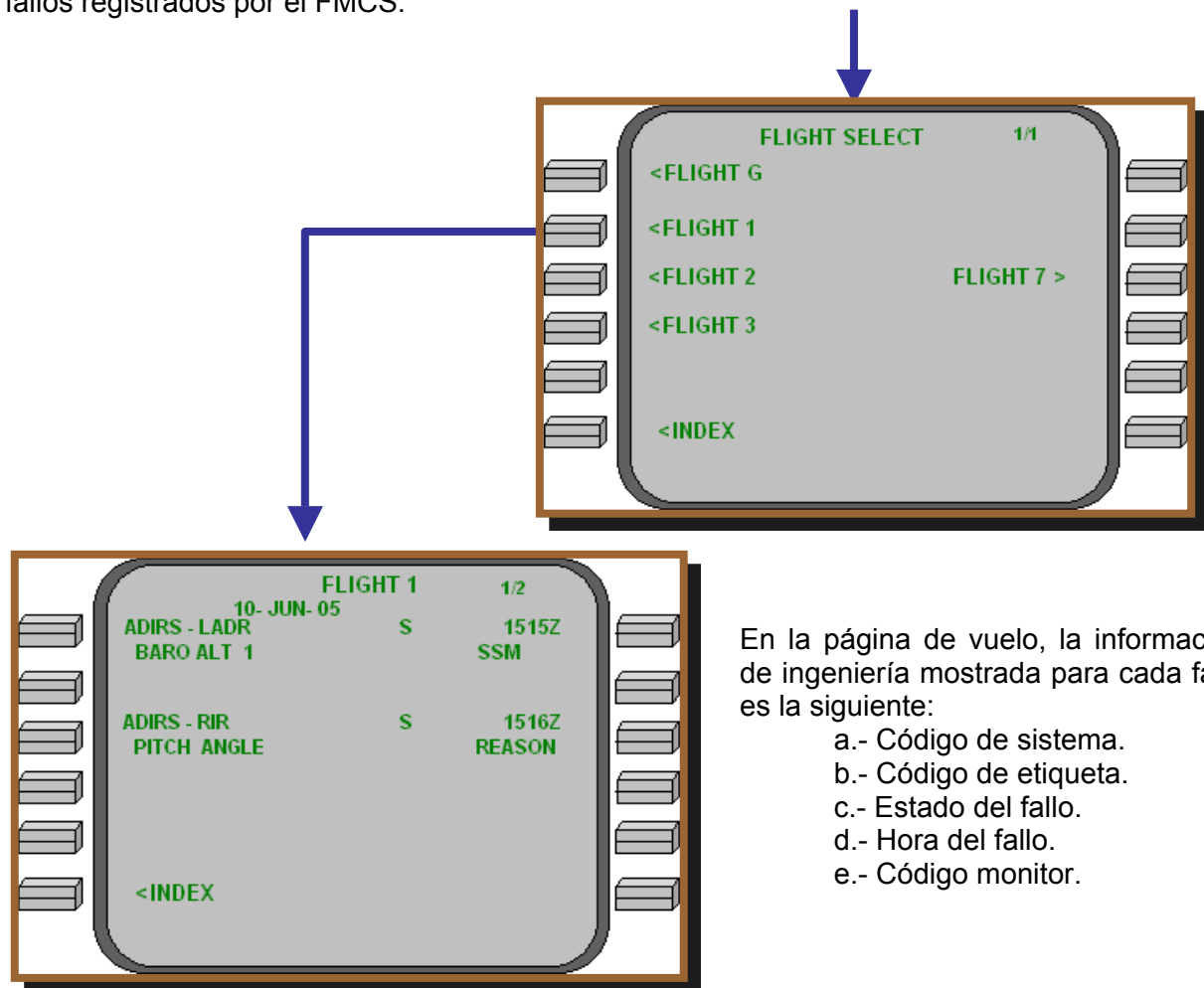
La LRU/subsistema con fallos, muestra el número del vuelo donde este ocurrió. El 1 corresponde al último vuelo y el 9 corresponde a hace 9 vuelos.

Todos las LRU/subsistemas que interaccionan con el FMCS se muestran en las páginas de fallos en vuelo (in-flight faults). Como se ve en la figura, en la esquina superior derecha aparece el número 1/3, indicando que estamos en la página 1 de 3 páginas que conforman el menú de in-flight faults. Las LRU/subsistemas que aparecen teniendo en cuenta las tres páginas son:

Air Data Inertial Reference System (**ADIRS**)
Digital Flight Control System (**DFCS**)

Distance Measuring Equipment (**DME**)
 Fuel Quantity Indicating System (**FQIS**)
 Common Display System (**CDS**)
 VHF Omni directional Range (**VOR**)
 Multy-Mode Receiver (**MMR**)
 Clock
 Flight Management Computer (**FMC**)
 Control Display Unit (**CDU**)

Si queremos ver mas detalles acerca de los posibles fallos, datos de fallo a nivel de ingeniería, se introduce el código de ingeniería 300 en la zona de trabajo (scratch pad) de la CDU y se pulsa la tecla LSK 6R. Estas acciones nos llevan a la página de selección de vuelo (FLIGHT SELECT). Esta página solamente muestra los números de vuelo donde ha habido fallos registrados por el FMCS.



En la página de vuelo, la información de ingeniería mostrada para cada fallo es la siguiente:

- a.- Código de sistema.
- b.- Código de etiqueta.
- c.- Estado del fallo.
- d.- Hora del fallo.
- e.- Código monitor.

a.- El Código del sistema, muestra la LRU o subsistema que ha fallado.

b.- Para un fallo del FMCS el código de etiqueta muestra la circuitería que ha fallado o un fallo de programa en el FMC. Para un fallo de entrada de un sensor, el código de etiqueta identifica la palabra ARINC con el fallo.

c.- El estado de fallo muestra si el fallo fue permanente (S) o intermitente (I). Todos los fallos son inicialmente registrados como permanentes (steady-S). Si posteriormente, el fallo desaparece, la indicación cambia a intermitente (I).

d.- La hora del fallo muestra el tiempo GMT en el que sucedió la primera ocurrencia de fallo. Este tiempo no cambia si el fallo desaparece y posteriormente vuelve a aparecer.

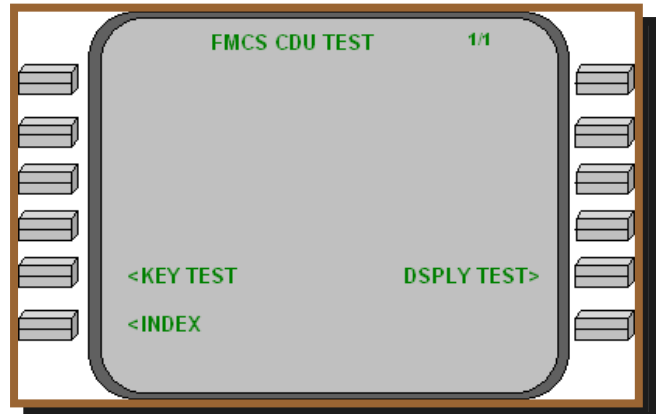
e.- Para un fallo de FMCS, el código monitor muestra el tipo de fallo de programa en el FMC. Para un fallo de entrada de sensor, el código monitor muestra el tipo de fallo específico ARINC 429. Estos pueden ser:

Rate (fallo de velocidad de recepción), Parity (fallo de paridad), SSM (fallo de sign/status matrix), Reason (fallo razonable).

*CDU TEST.

Para comprobar las CDU-s, se selecciona CDU TEST mediante la tecla LSK 2L desde la página FMCS BITE. Este test, es un autotest interactivo que muestra los caracteres de la CDU y comprueba los anunciadores y pulsadores de la misma.

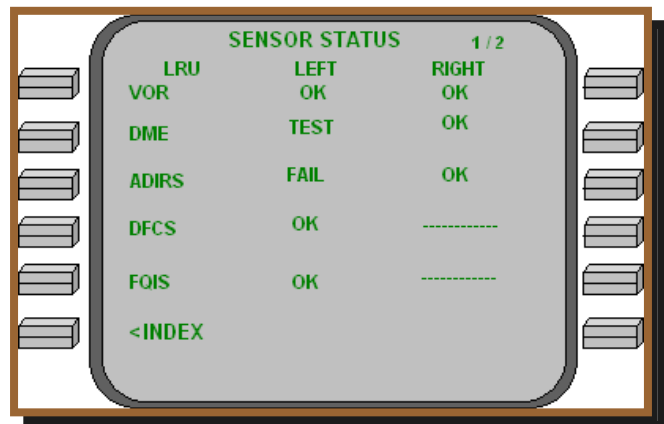
Una vez en la página CDU TEST, se puede seleccionar CDU DSPLY TEST (LSK 5R) o CDU KEY TEST (LSK 5L). Mediante la primera opción, el test muestra el set completo de caracteres con el que trabaja la CDU. Con la segunda opción, el test muestra el panel frontal de la CDU. Cuando se pulsa una tecla, la CDU resalta en pantalla la tecla pulsada mientras que en la zona de trabajo aparece el nombre de la tecla pulsada.



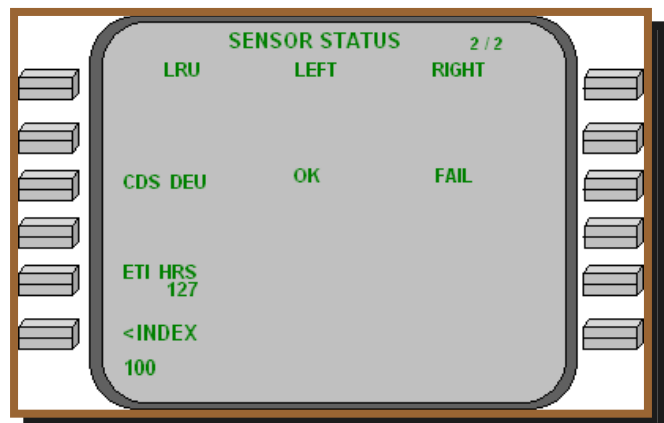
*SENSORS.

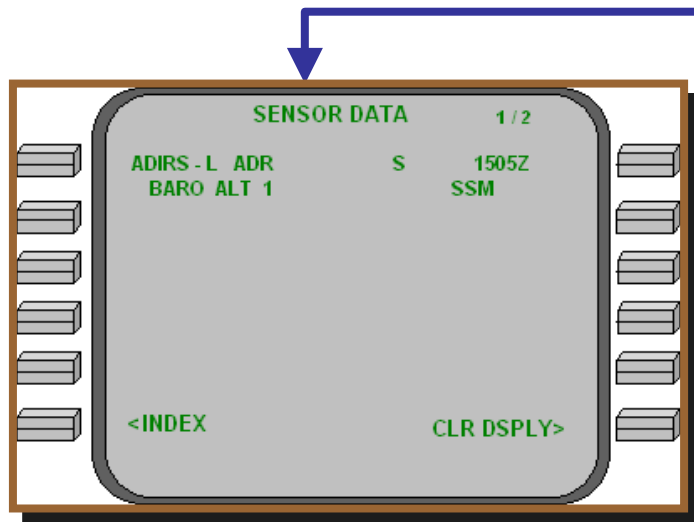
Para comprobar el estado actual de los sensores que envían datos al FMC se selecciona SENSORS mediante la tecla LSK 3L desde la página FMCS BITE. Esto nos introduce en la página SENSOR STATUS.

Cuando aparecen guiones en la línea del campo de estado, indica que la LRU no existe o que el FMC no interactúa con la LRU.



Desde la página de SENSOR STATUS se puede acceder a datos de ingeniería en la página SENSOR DATA, introduciendo el código de ingeniería 100 en la zona de trabajo (scratch pad) y validando esta entrada con la tecla LSK 6R. Esta página muestra datos de ingeniería para los sensores que aparecen con un estado de fallo en la página SENSOR STATUS. Los fallos se muestran en orden cronológico.





En la página de datos de ingeniería aparecen datos adicionales referentes a los fallos detectados. En este caso, en la pagina 1/2 aparece la LRU que falló, (ADIRS-L ADR), una descripción del fallo (BARO ALT 1), el fallo fue permanente (S), la hora del fallo (15:05), el código monitor (SSM).

**DISCRETES.*

Para realizar un chequeo del estado de los discretos del FMCS se selecciona DISCRETES mediante la tecla LSK 4L desde la página FMCS BITE. Esto nos introduce en las páginas (cuatro) de DISCRETES. Estas páginas muestran el estado actual de los discretos analógicos que envían datos al FMC.

En la página 1 tenemos:

ECS PACK. Se muestra la posición del control de flujo del ECS y la válvula de cierre en cada pack.

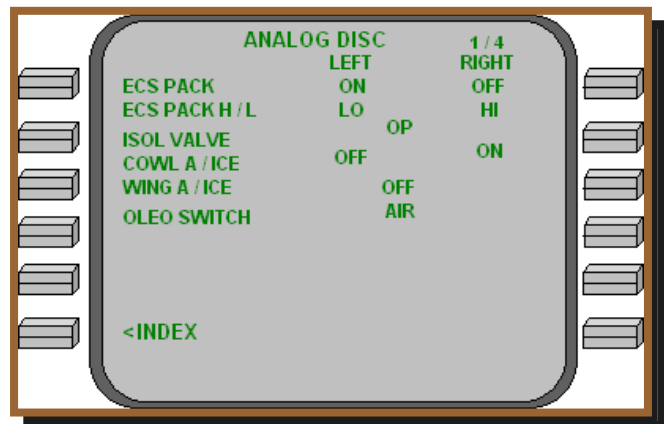
ECS PACK H/L. Muestra si el pack está en flujo normal o flujo alto.

ISOL VALVE. Muestra la posición de la válvula de aislamiento de aire de sangrado.

COWL A/ICE. Muestra la posición de los switches anti-hielo del capó del motor.

WING A/ICE. Muestra la posición de los switches anti-hielo del ala.

OLEO SWITCH. Muestra la aeronave en modo tierra o aire. Esta entrada proviene de la unidad electrónica del switch de proximidad (Proximity Switch Electronic Unit-PSEU)



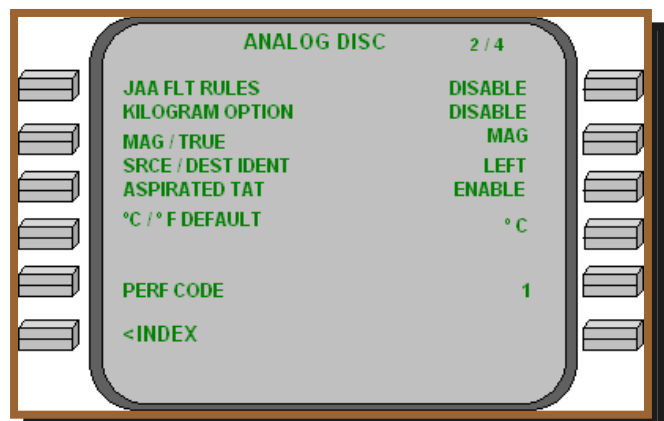
En la página 2 tenemos:

FAA/JAA FLIGHT RULES. Con esta opción se establece el CG para las reglas de vuelo apropiadas.

KILOGRAM OPCION. Establece esta opción como entradas de peso en la CDU.

MAG/TRUE. Esta opción no se utiliza.

SRCE/DEST IDENT. Esta opción se utiliza para identificar el FMC. Siempre está configurado como izquierdo.



ASPIRATED TAT. Esta opción se habilita cuando una sonda de TAT aspirado esta en la aeronave.

DEGREE C/DEGREE F DEFAULT. Establece la unidad de temperatura para entradas en la CDU.

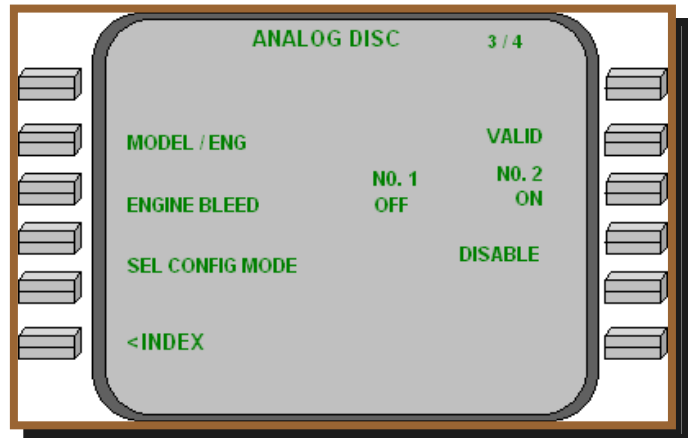
PERF CODE. Establece el código de performance para el FMC.

En la página 3 tenemos:

MODEL/ENGINE. Esta opción establece los datos de aero / propulsión para el FMC.

ENGINE BLEED. Muestra la posición de los switches del aire de sangrado del motor.

SEL CONFIG MODE. Habilita la configuración del software cargable.



En la página 4 tenemos:

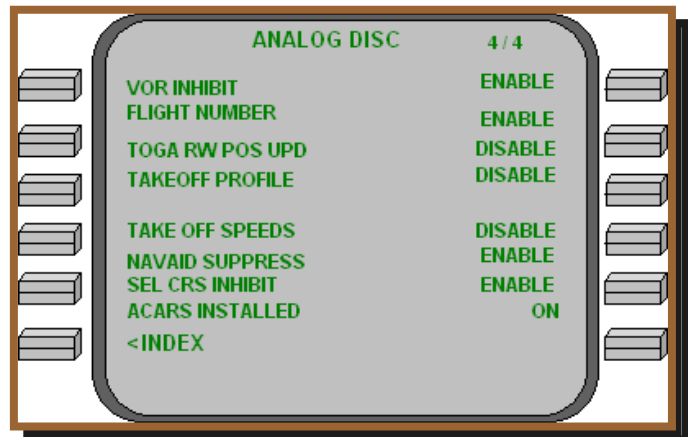
VOR INHIBIT. Inhabilita las entradas de VOR para la función de navegación.

FLIGHT NUMBER. Permite a la tripulación introducir en número de vuelo en la página RTE.

TOGA RW POS UPD. Permite al FMC realizar una actualización de la posición cuando los switches de TOGA están empujados en tierra.

TAKEOFF PROFILE. Proporciona al FMC control de la altitud a la cual el empuje de despegue se reduce al de ascenso.

TAKEOFF SPEEDS. Permite al FMC calcular las velocidades de despegue basándose en parámetros de la sección QRH (Quick Reference Handbook) y las muestra en la página de despegue.



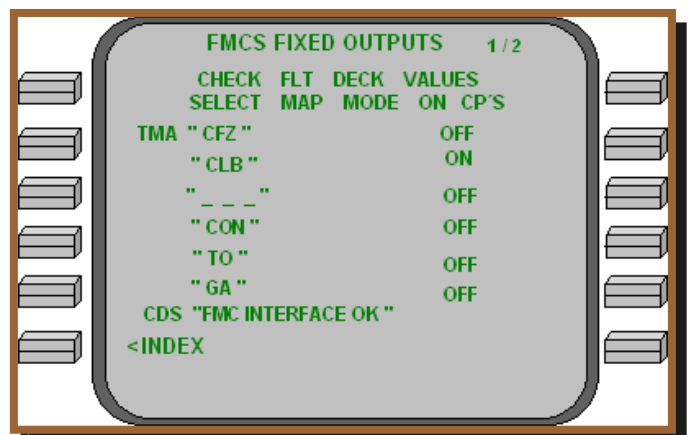
NAVAID SUPPRESS. Suprime las navais de auto sintonía en la pantalla de mapa del CDS.

SEL CRS INHIBIT. Esta opción suprime radiales de curso seleccionados por navaid sintonizadas manualmente en la pantalla de mapa del CDS.

ACARS INSTALLED. Permite la interface ACARS /FMC.

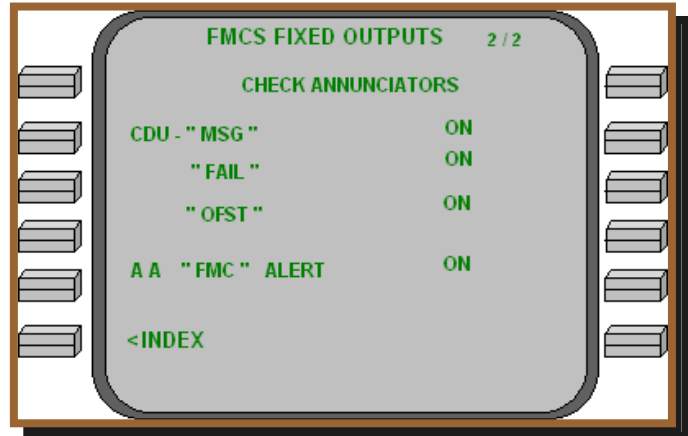
*FMCS-FIXED OUTPUTS.

Para mirar las salidas fijas del FMCS se selecciona FIXED OUTPUTS mediante la tecla LSK 5L desde la página FMCS BITE. Los selectores de modo en ambos paneles de control del EFIS deben colocarse en modo MAP.



La página 1 / 2 muestra datos del FMCS que van al Thrust Mode Annunciation (TMA) en la CDS (Common Display System) de motor. Los modos de empuje (Thrust Modes) se muestran en secuencia en el TMA. En la CDS de navegación, se muestra el mensaje FMC INTERFACE OK

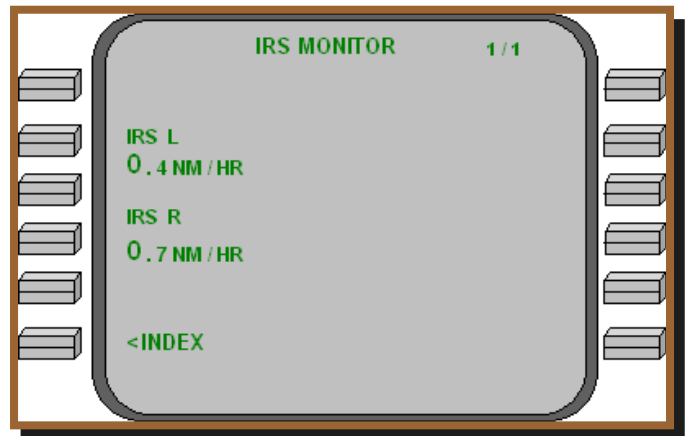
En la página 2 / 2 los siguientes anunciadores de la CDU: MSG, FAIL y OFST están en ON durante el test de FIXED OUTPUTS.



**IRS MONITOR.*

Para chequear la tasa de error de posición del IRS se selecciona IRS MONITOR mediante la tecla LSK 5R desde la página FMCS BITE.

En la página IRS MONITOR se muestra la tasa de error de posición (NM / HR) para cada IRS. El FMCS calcula la tasa de error de posición para cada vuelo al finalizar estos; para ello, divide la diferencia entre la posición calculada por el y la calculada por el IRS, entre el tiempo total de vuelo.



**PERFORMANCE FACTORS.*



Para chequear factores de funcionamiento del FMCS se selecciona PERF FACTR mediante la tecla LSK 4R desde la página FMCS BITE.

Los factores de funcionamiento son: DRAG FACTOR cuyo rango es $\pm 9,9\%$. F-F (Fuel flow factor) cuyo rango es $\pm 9,9\%$. MNVR MARGIN (maneuver margin). El rango está establecido en la base de datos de funcionamiento. MIN CRZ TIME (minimun cruise time) cuyo rango

es desde 1 hasta 20 minutos). MIN R/C (minimun rate-of-climb). El rango es desde 0 hasta 999 para ascenso y crucero y desde 0 hasta 500 para engine out.

***SOFTWARE OPTIONS.**

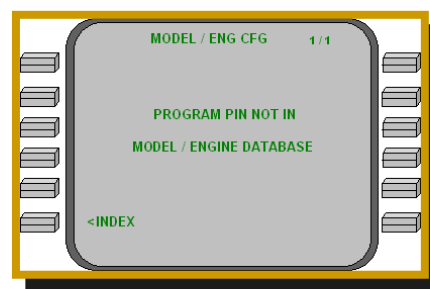
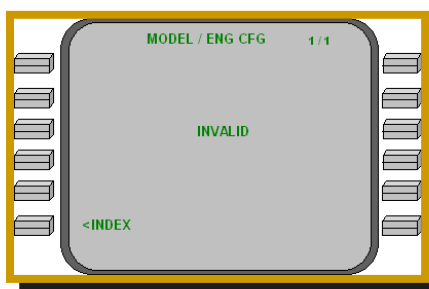
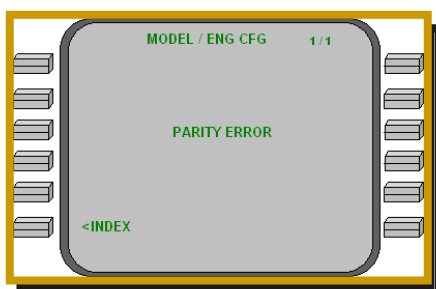
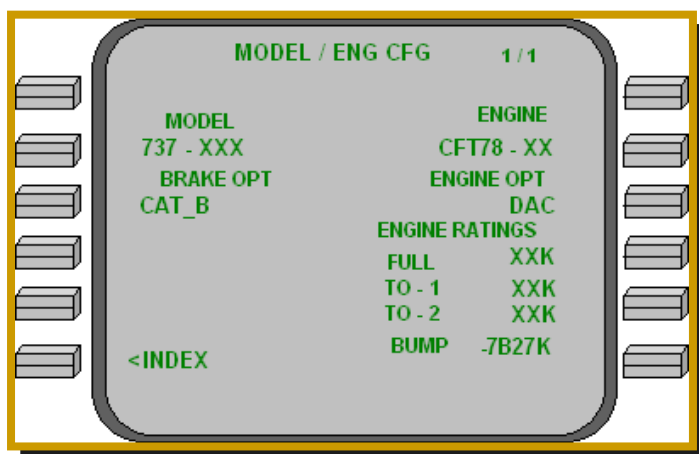
Para ver las opciones de software del FMCS se selecciona SW OPTIONS mediante la tecla LSK 3R desde la página FMCS BITE. Las páginas de opción de software muestran el estado de las opciones que se encuentran en la base de datos de opciones de software.

El código de la opción de software consiste en un número hexadecimal de 12 dígitos.

***AIRPLANE MODEL AND ENGINE CONFIGURATION.**

Para ver la página de MODEL/ENG CFG se selecciona MODEL/ENG mediante la tecla LSK 1R desde la página FMCS BITE. Esta página muestra la configuración de los motores y de la aeronave. También muestra el tipo de cámara de combustión del motor y las opciones de freno seleccionadas.

Durante la energización, el FMC monitoriza los pines programables de configuración de célula / motor para asegurar que la configuración es válida. Si existe una condición anormal, se muestra uno de los siguientes mensajes:



PARITY ERROR. Se muestra si los pines programables de configuración célula / motor no están configurados para paridad impar. También se muestra para una configuración discreta de SDI inválida.

INVALID. Se muestra si la base de datos de performance de modelo / motor no está cargada o la cargada MEDB ha dado fallo en su chequeo de validez.

PROGRAM PIN NOT IN MODEL/ENGINE DATA BASE. Se muestra si los pines programables de la configuración célula / motor no concuerdan con los datos de la base de datos de performance de modelo /motor.

16.3-AUTOMATIC TEST EQUIPMENT (ATE).

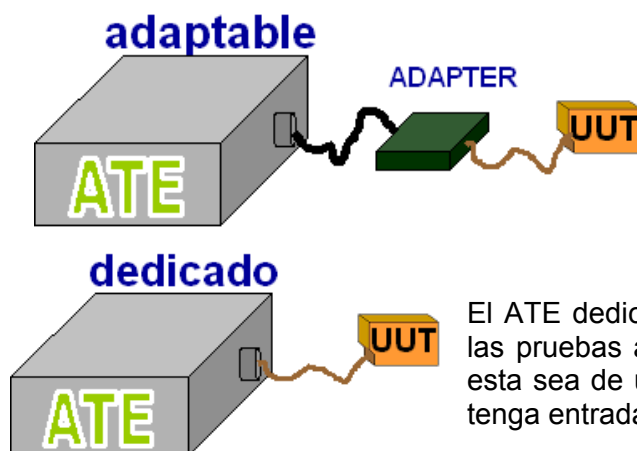
Las unidades individuales de los sistemas aviónicos existentes hoy en día, con su complejidad y características particulares de cada uno de ellos, no pueden ser probados y reparados de forma manual, en un banco de trabajo convencional, con una serie de instrumentación común de reparación y test, sin el inconveniente de invertir un tiempo relativamente elevado, y una pérdida de eficiencia, en muchos casos significativa.

Por otro lado, para poder realizar las pruebas pertinentes, destinadas a una comprobación y reparación, el técnico, debe tener la habilidad y conocimientos necesarios para poder operar una variedad diferente de equipos de test y dispositivos de medida y a la vez, ser capaz de interpretar los resultados mientras se siguen los procedimientos de realización de un test de comprobación. Normalmente, se deben realizar muchas tareas repetitivas y seguir una serie de procedimientos largos en la comprobación de la unidad, procedimientos que pueden durar muchas horas, e incluso días.

La forma de reducir el tiempo que estos tests y comprobaciones suponen, es mediante el uso de los equipos automáticos de test (ATE), equipos estos que realizan las funciones de test y comprobación bajo la supervisión de un sistema controlado por un computador.

16.3.1-TIPOS DE ATE.

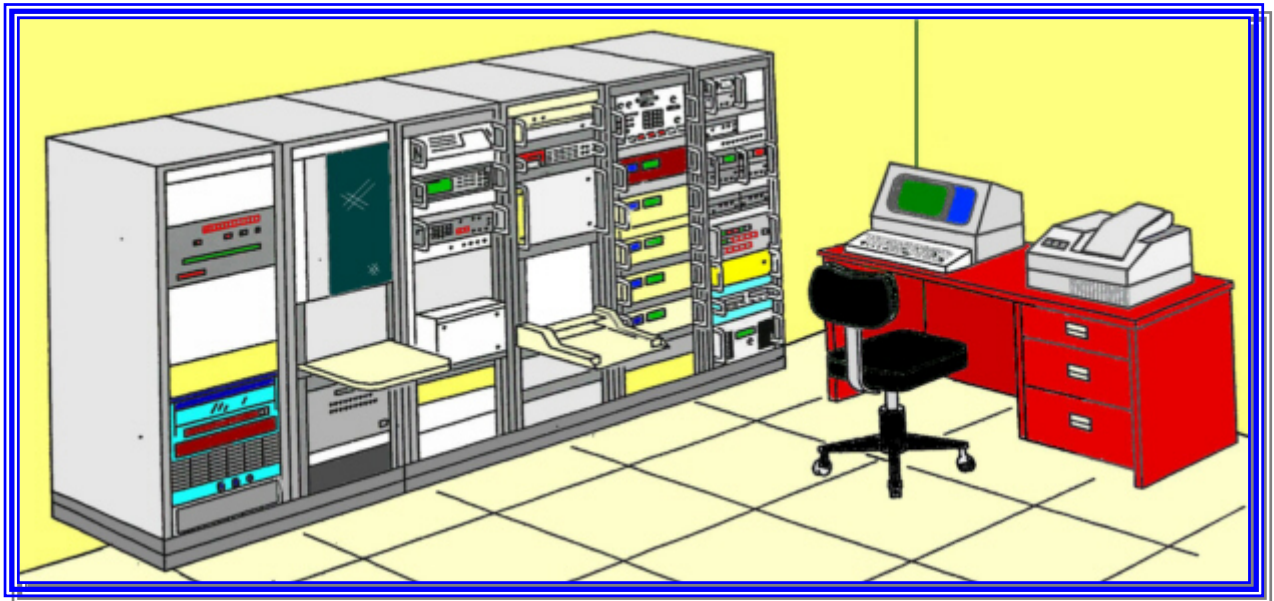
- La forma que puede tener un sistema ATE, va desde un simple computador preparado para realizar unas determinadas funciones con el propósito de realizar una serie de test y pruebas a un equipo concreto y relativamente simple, hasta una estación automática de test para realizar test a equipos aviónicos complejos. Sea cual sea la forma de los ATE, estos pertenecen a uno de los siguientes tipos: ATE adaptable o ATE dedicado. El tipo más común es el tipo adaptable. Estos,



común es el tipo adaptable. Estos, tienen una variedad de test instalados que pueden ser usados según requerimientos de diferentes tipos de unidades, (UUT-Unit Under Test-) las cuales se adaptan a la estación ATE mediante un sistema de adaptadores.

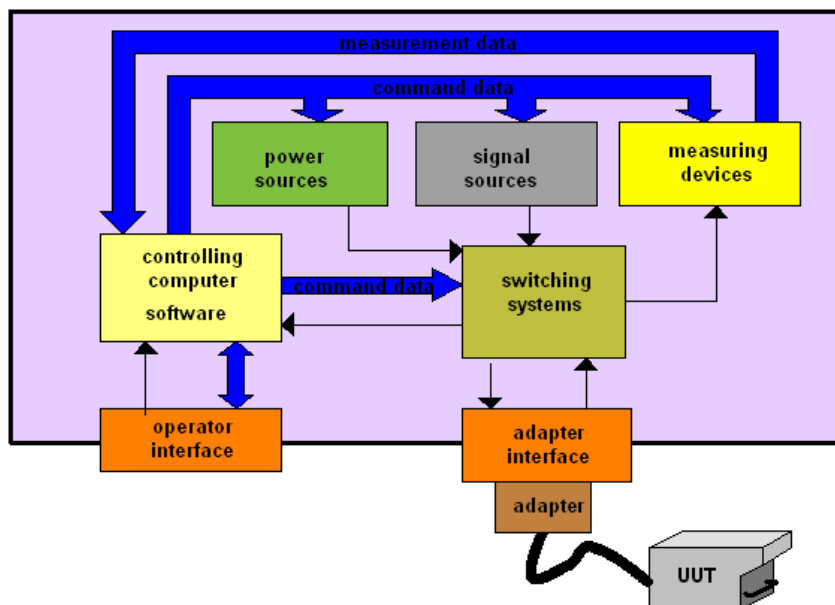
El ATE dedicado, es normalmente utilizado para realizar las pruebas a un tipo de UUT determinado, bien porque esta sea de una complejidad excesiva, o bien porque no tenga entradas y salidas comunes a otras unidades.

16.3.2-AUTOMATIC TEST STATION (ATS).



- Una vez que se ha evaluado el equipamiento necesario para realizar el estímulo correcto de la UUT, el ATE es ensamblado tanto física como eléctricamente en un sistema llamado *Automatic Test Station* (ATS). La ATS generalmente está formada por diversas secciones, cada una de las cuales se encarga de realizar una función o tarea específica. Las funciones básicas comunes a la mayoría de las ATS son: la interfaz para el operario, el computador controlador, software, fuentes de alimentación, fuentes de señal, sistema de switches mecánicos o electrónicos, equipos de medida, interfaz para el adaptador.

OPERATOR INTERFACE. Esta interfaz es usada para la comunicación bidireccional entre el operador y el computador controlador. La interfaz puede ser parte del computador o puede ser una consola de control y puede tener elementos tales como switches, luces indicadoras, teclado y displays. Normalmente, el operador introduce la información para el



computador controlador a través del teclado y /o switches y el computador, devuelve información de resultados o petición de acciones a realizar por parte del operario a través de LED-s, CRT, LCD, un conjunto de luces o una impresora.

La interface del operador, puede estar situada en la misma ATS o puede ser una interfaz remota, situada incluso en otro edificio diferente al de la localización de la estación ATE.

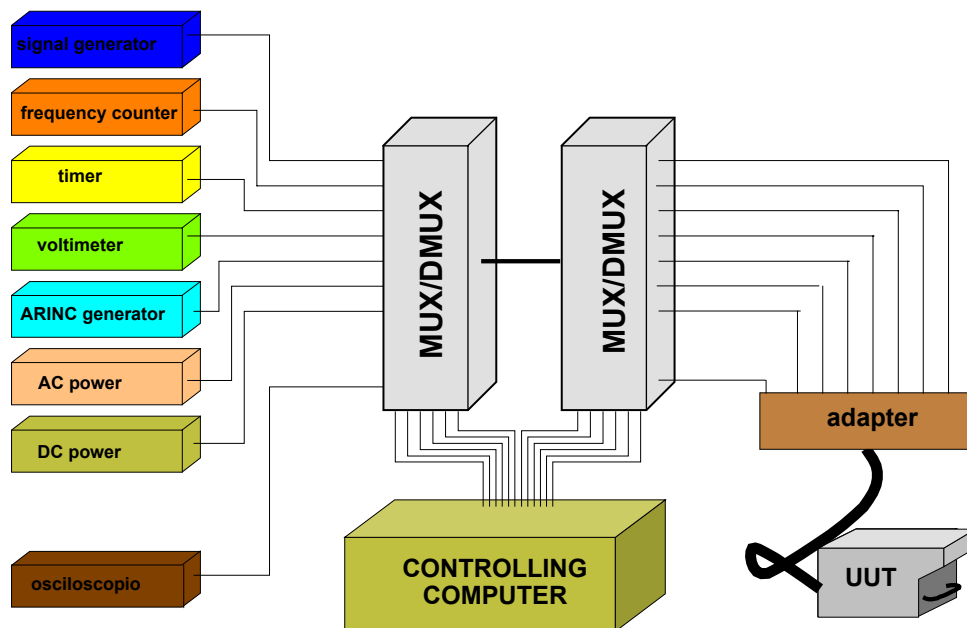
SOFTWARE. El software guardado en el computador controlador, contiene diversos programas en los cuales están contenidos los pasos y procedimientos que el sistema debe realizar para efectuar los tests y comprobaciones necesarias para diferentes UUT.

POWER SOURCE. La fuente de alimentación proporciona los voltajes y corrientes necesarios, tanto de AC a 400 Hz (monofásicos, bifásicos y trifásicos) como de DC, para la realización de los tests.

SIGNAL SOURCE. Esta sección proporciona las señales con las formas de onda concretas, tales como impulsos, ondas senoidales, ondas rectangulares, ondas cuadradas, ondas en diente de sierra, rampas, señales digitales ARINC, señales moduladoras, etc. que son requeridas en los tests de las UUT.

MEASURING DEVICES. Estos dispositivos (medidores de frecuencia, espectrómetros, voltímetros, osciloscopios, etc.) de alta resolución, son los encargados de realizar las medidas tanto de las salidas dadas por las UUT-s como de las entradas aplicadas a las mismas. Estos aparatos de medida, pueden trabajar de forma automática en la ATS o manipulados por el operador a requerimiento del programa que se está ejecutando.

SWITCHING SYSTEM. El programa, encargado de realizar el test a una UUT determinada, va a requerir que de forma automática, algunas señales, en tiempos determinados, sean acopladas a ciertos pines de la UUT. En otros momentos va a requerir que sean acoplados unos dispositivos de medida, incluso en el mismo pin que anteriormente le fue acoplada una función de entrada, para medir las respuestas a una excitación determinada, etc. La sección encargada de realizar estas conexiones y desconexiones automáticas, según requerimiento del programa, es el sistema de conmutación. (ver figura siguiente)



ADAPTER INTERFACE. Esta unidad se encarga de proveer un medio rápido de conexión y desconexión eléctrica de los circuitos de la UUT, a la vez de ser un dispositivo de conexión común para los adaptadores de las UUT.

Para poder comprobar una gran variedad de UUT-s en la ATS, se debe disponer de un sistema de adaptadores tal que podamos conectar las UUT con su gran variedad de conectores y /o conexiones, a la interfaz adaptadora. Aunque existen cientos de sistemas de adaptadores, la mayoría de ellos son combinaciones o variaciones de tres categorías básicas: el conector intercambiable, el ATE suplementario y electrónica adicional.

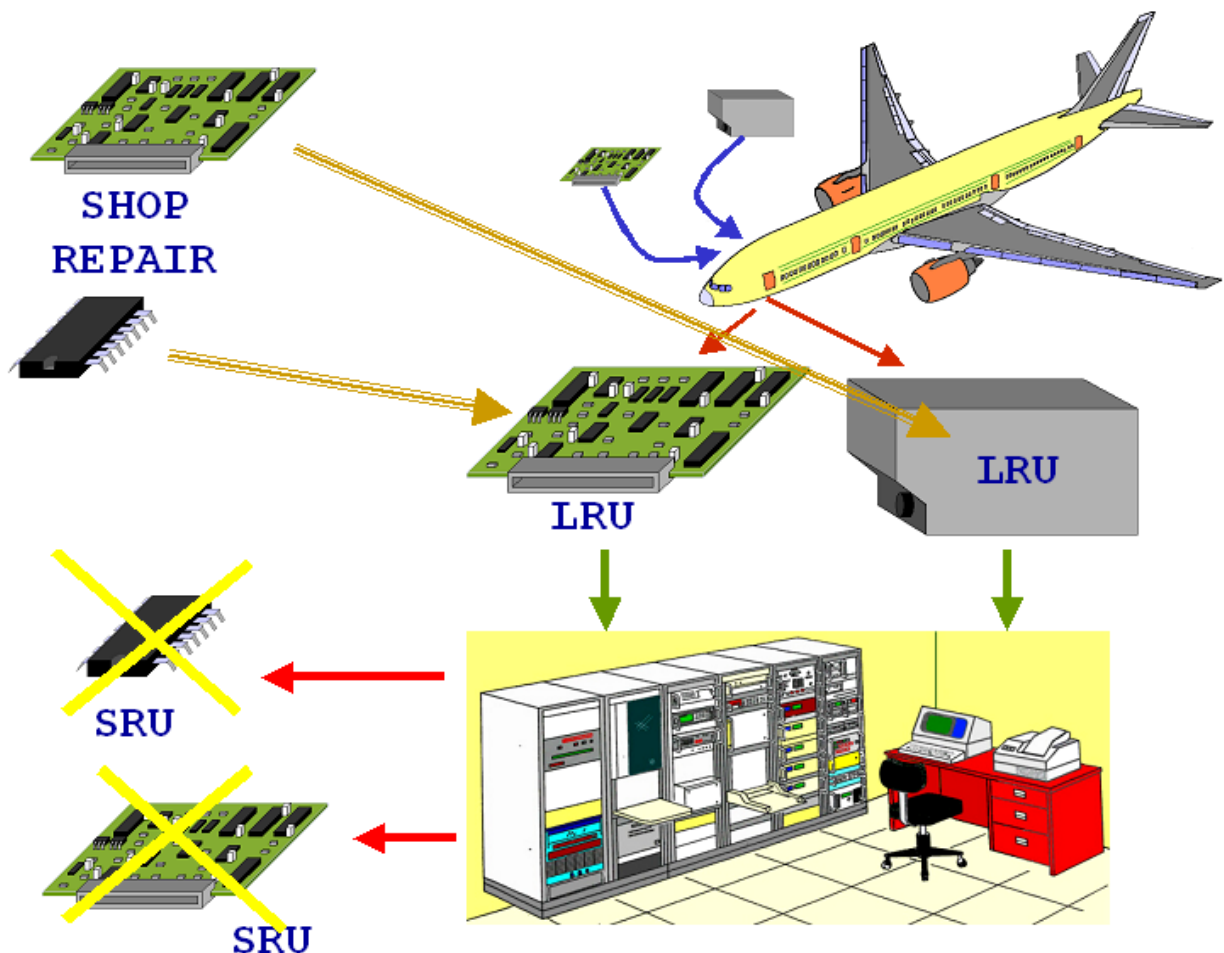
***CONECTOR INTERCAMBIABLE:** Un conjunto de conectores que se adaptan a los conectores que poseen las UUT, se van acoplando a los cables que han de unir estas con la ATS.

***ATE SUPLEMENTARIO:** En alguna ocasión se puede dar el caso de que la ATS no tenga instalado el equipo ATE necesario para realizar las pruebas a una UUT particular, bien porque esta sea única, o porque sea un modelo de nueva aparición. Dependiendo del diseño del ATS, el ATE necesario puede estar integrado en el sistema de adaptadores.

***ELECTRÓNICA ADICIONAL:** La adaptación se realiza mediante circuitos electrónicos adicionales que se enchufan en lugares determinados del adaptador para este fin. Dependiendo de la particularidad de la UUT, se conectará un circuito u otro para poder adaptar dicha UUT a la ATS.

16.3.3-PROCESO DE MANTENIMIENTO.

- Cuando se detecta una avería en un sistema, bien mediante el equipo BITE, o bien mediante pruebas funcionales, y esa avería supuestamente es debida a un problema electrónico, el elemento defectuoso es sacado de su emplazamiento en la aeronave, para ser comprobado en la **ATS** (*Automatic Test Station*). Este elemento se llama Line Replaceable Unit (LRU). Normalmente, una LRU en buenas condiciones de trabajo es instalada en el lugar de donde se tomó la otra, y de esta manera, la aeronave puede volver al servicio sin más demora. Una LRU, puede ser una unidad electrónica con más o menos capacidad de funciones dentro de un sistema, o ser simplemente una tarjeta electrónica de circuito impreso.



Después de que una LRU ha sido comprobada en la ATS mediante la serie de pruebas establecidas para la misma, y algún elemento (una tarjeta de circuito impreso o un componente específico) de la misma ha resultado estar defectuoso, este es retirado de la LRU. Este elemento defectuoso, se llama *Shop Replaceable/repairable Unit (SRU)*. Un técnico, o equipo de técnicos, procederá entonces a realizar la reparación de la SRU en el laboratorio o en taller, según sea el elemento defectuoso. Una vez reparada, es colocada de nuevo en la LRU, a la que se le realiza de nuevo la prueba en el ATE o ATS para verificar su correcto funcionamiento y de esta manera, poder volver a instalarla en una aeronave que la requiera.



Avionics Systems

17

17.-AVIONICS SYSTEMS.

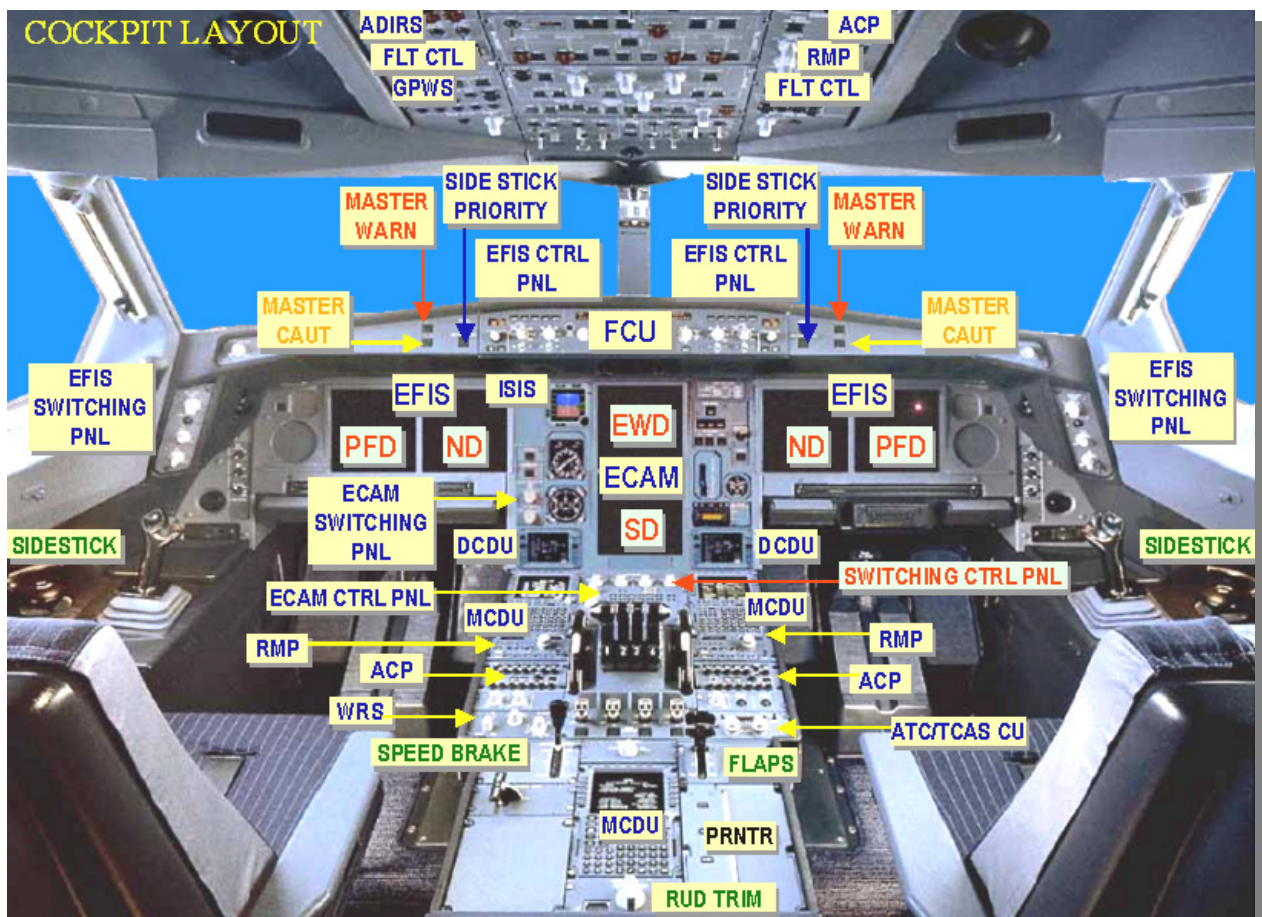
17.1-COCKPIT LAYOUT.

Los sistemas aviónicos que forman parte de una aeronave actual, hacen que esta pueda ser gobernada, sin mayor problema, por una tripulación compuesta por dos personas. Esto se debe en un alto grado, a que el trabajo de verdadera responsabilidad, crítico para el gobierno y seguridad de la aeronave, ha sido puesto en manos de unos sistemas electrónicos capaces de tomar decisiones ante entornos predecibles y responder con gran eficacia ante situaciones de adversidad.

Al igual que la tripulación de vuelo debe seguir una serie de procedimientos bien definidos como parte de su labor al mando de la aeronave, los sistemas aviónicos también deben seguir una serie de procedimientos como parte integrante de gobierno de la aeronave. Estos procedimientos están establecidos mediante una serie de órdenes que componen el software de programas que los computadores digitales componentes de los actuales sistemas aviónicos, almacenan en una memoria interna.

La correcta coordinación entre los sistemas electrónicos y la tripulación (partiendo de la premisa de un correcto mantenimiento de los sistemas aviónicos), hace que el vuelo de una aeronave halla alcanzado el grado de fiabilidad y seguridad que actualmente caracteriza el transporte aéreo.

En la siguiente figura, se muestra la interfaz entre algunos sistemas aviónicos de una aeronave moderna y la tripulación de vuelo.



En posteriores capítulos se analizan algunos de los sistemas que interaccionan con la tripulación de vuelo en el cockpit.

Los acrónimos a los que se hace referencia en la figura son los siguientes:

ACP. *Audio Control Panel.*

ADIRS. *Air Data Inertial Reference System.*

ATC / TCAS CU. *Air Traffic Control / Traffic Alert and Collision Avoidance System Control Unit.*

DCDU. *Datalink Control and Display Units.*

ECAM. *Electronic Centralized Aircraft Monitoring.*

EFIS CTRL PNL. *EFIS Control Panel.*

EFIS. *Electronic Flight Instruments System.*

EWD. *Engine Warning Display.*

FCU. *Flight Control Unit.*

FLT PNL. *Flight Panel.*

GPWS. *Ground Proximity Warning System.*

ISIS. *Integrated Standby Instruments System.*

MCDU. *Multipurpose Control Display Unit.*

ND. *Navigation Display.*

PFD. *Primary Flight Display.*

PRNTR. *Printer.*

RMP. *Radio Management Panel.*

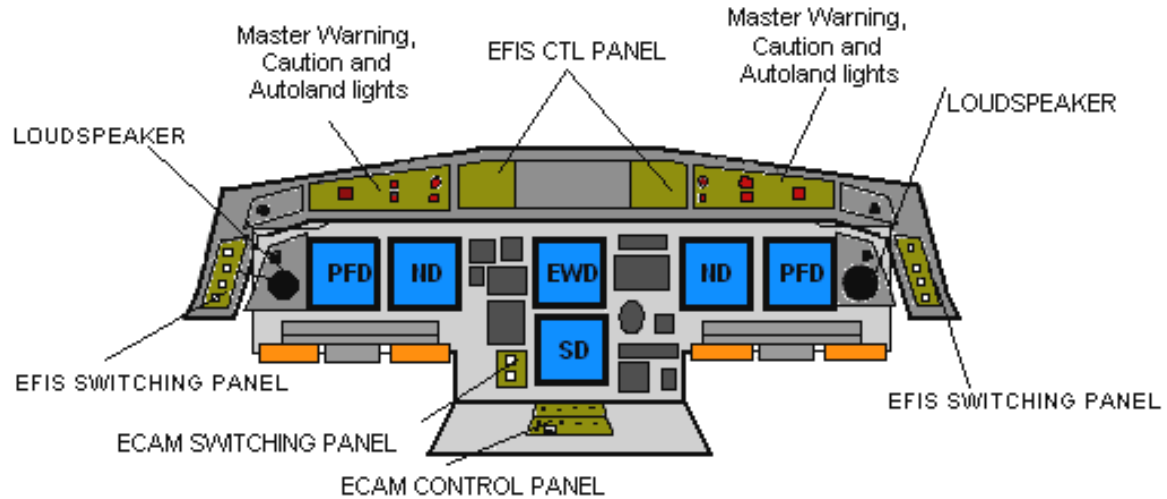
RUD TRIM. *Rudder Trim.*

SD. *System Display.*

WRS. *Weather Radar System.*

Otras interfaces en el cockpit son los paneles de luces, tanto internas como externas, panel de arranque de motores, panel de aire, panel de sistema eléctrico, panel de fuel, panel de fuego de motor, panel de hidráulico, panel de anti-hielo, panel de presión de cabina, panel de emergencia eléctrica, etc.

El EIS, bajo el control de los **DMC-s**, presenta a través de las seis unidades de display (DU-s), información de parámetros de vuelo, datos de navegación, etc. a través del **EFIS** e indicaciones de parámetros primarios de motor, diagramas sinópticos de la aeronave, etc. a través del **ECAM**. Además, bajo el control de los **FWC-s**, presenta a través de luces y altavoces, señales visuales y acústicas de aviso y precaución a la tripulación.



El EFIS lo componen cuatro DU-s; dos frente al capitán (**CAPT**) y las otras dos, frente al primer oficial (**F/O**). En configuración normal, la función de cada una de las DU-s es de *Primary Flight Display (PFD)* en las DU-s externas, y la de *Navigation Display (ND)* en las DU-s internas, tal y como aparecen mostradas en la figura superior.

Las PFDs presentan información a corto plazo, necesaria para el vuelo:

- ángulos de actitud: Pitch, Roll, Heading.
- indicaciones de velocidad respecto del aire: velocidad computada y velocidades límites o singulares.
- altitud, velocidad vertical.
- desviación de trayectoria (ILS)
- anuncios en modo de vuelo automático.
- alertas TCAS

Las dos ND presentan información de vuelo a medio plazo necesaria para la navegación:

- radionavegación (VOR, DME, ILS, MLS, ADF)
- rutas de vuelo
- mapa del radar meteorológico.
- informes TCAS, estado del GPS

Las DU-s internas pueden mostrar a petición de los pilotos, información ECAM en lugar de información de navegación, para cubrir el caso de que las ECAM DU-s fallen. También pueden mostrar información del radar meteorológico (**WXR**).

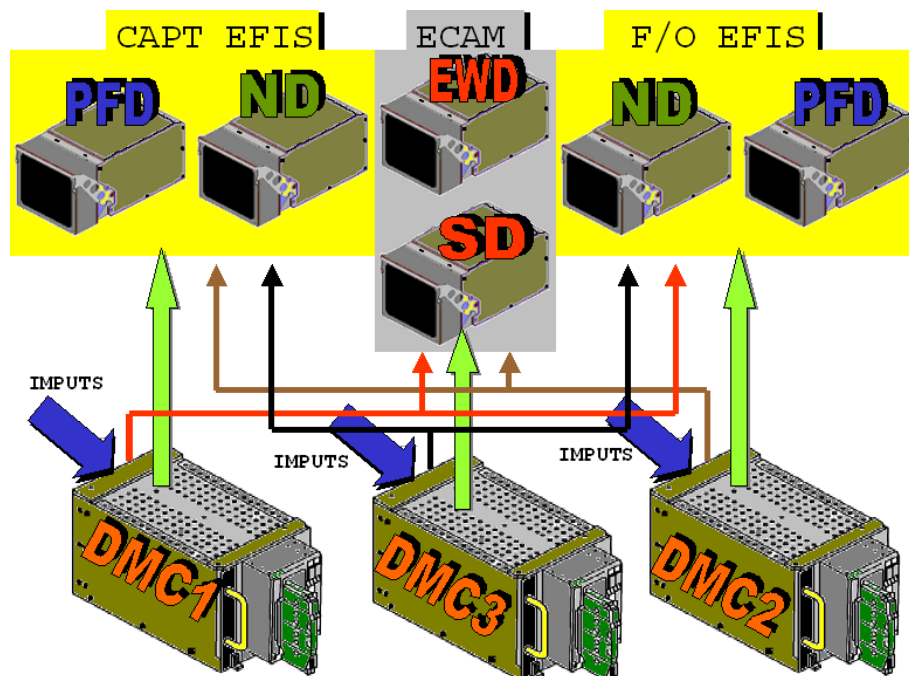
El ECAM lo componen dos DU-s, una sobre la otra, situadas en el panel central de instrumentos y es función única para los dos pilotos. La DU superior, llamada *Engine/Warning Display (EWD)* presenta información compuesta por indicaciones primarias de motor, configuración de las superficies de control y mensajes de aviso/precaución (warning/caution) en caso de que se produzca algún fallo. La DU inferior, llamada *System Display (SD)*, presenta

mensajes sobre el estado de la aeronave y diagramas sinópticos de diferentes sistemas, de acuerdo a la configuración que la aeronave tenga. Los sistemas que pueden ser representados mediante diagramas sinópticos son:

- ENGINE. Parámetros secundarios de motor.
- BLEED. Aire de sangrado.
- CAB PRESS. Presurización de la cabina.
- ELEC/AC. Alimentación eléctrica AC.
- ELEC/DC. Alimentación eléctrica DC
- HYD. Potencia hidráulica.
- C/B. Estado de los circuit breakers.
- FUEL. Combustible.
- APU. Unidad de potencia auxiliar.
- COND. Aire acondicionado.
- DOOR/OXI. Puertas y oxígeno.
- WHEEL. Tren de aterrizaje, ruedas, frenos.
- F/CTL. Controles de vuelo.
- CRUISE. Parámetros en la fase de crucero.
- STATUS. Estado operacional.

Los DMC-s, tres unidades idénticas (DMC1, DMC2, DMC3) situadas en el compartimiento de aviónica, se encargan de la adquisición de los datos y parámetros de diferentes sistemas de la aeronave para procesar esta información y de acuerdo a los resultados, generar las presentaciones que se muestran en las DU-s situadas en el cockpit.

En condiciones normales de operación, los tres DMC reciben exactamente la misma información proveniente de los sistemas de la aeronave, para generar las presentaciones en sus displays asociados, los cuales para dichas condiciones normales de operación son: CAPT EFIS (PFD y ND) para el DMC1; F/O EFIS (PFD y ND) para DMC2 y finalmente EWD y SD para el DMC3, según se muestra en la siguiente figura.



Los DMC-s están divididos en dos partes, una parte para las funciones de EFIS y la otra para las de ECAM. Cada una de las partes es totalmente independiente de la otra, hasta el punto de que cada una de ellas posee su propia fuente de alimentación, sus propios módulos de adquisición y su propio canal de procesamiento de información. En condiciones normales, la parte EFIS del DMC 1 y DMC 2 suministra información a su EFIS correspondiente, y la parte

ECAM del DMC 3, suministra información a los displays correspondientes del ECAM. La parte EFIS del DMC 3 permanece en Standby.

En condiciones de emergencia, cada DMC es capaz de generar simultáneamente 4 representaciones en los displays:

Por un lado 1 PFD y 1 ND manejados por la parte EFIS y por otro lado el EWD y el SD manejados por la parte ECAM.

Más aún, si existe doble fallo de DMC, el DMC restante puede manejar las 6 pantallas:

Por un lado, la EWD y la SD manejadas por la parte ECAM, y por otro lado, 2 PFD y 2 ND manejados por la parte EFIS con la misma información (modo copia).

Los dos *System Data Acquisition Concentrator (SDAC)*, realizan las funciones de adquisición, concentración y digitalización de datos provenientes de los diferentes sistemas de la aeronave, para ser enviados a la parte ECAM de los DMC-s. Con estos datos, los circuitos ECAM de los DMC-s, generan información para ser mostrada en el SD. (Los DMC-s reciben directamente datos de los sistemas de la aeronave para ser mostrados en el EWD).

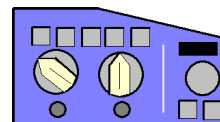
Los datos de precaución (ámbar) provenientes de los sistemas de la aeronave, son recibidos y digitalizados por los SDAC-s para ser enviados a los *Flight Warning Computers (FWC-s)*

Los dos FWC-s realizan las siguientes funciones principales: adquisición de datos de algunos sistemas de la aeronave, además de los datos provenientes de los SDAC, procesamiento de datos de aviso (rojo) correspondientes a situaciones que los requieran y procesamiento de la fase de vuelo de la aeronave. La salida en forma de mensajes digitalizados, es enviada a la parte ECAM de los DMC-s.

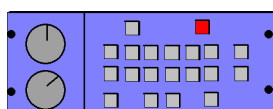
Los FWC-s también generan señales de atención compuestas por un lado por las luces de *Master Warning*, *Master Caution* y *Auto Land*, y por otro lado por señales de audio (sonidos) y *Auto Call Out* (voces sintéticas) a través de la *Audio Management Unit (AMU)*, que son emitidas por los altavoces. Estas señales carecen de la posibilidad de ser reguladas en volumen.

La función de reconfiguración del EIS se lleva a cabo mediante un conjunto de switches (interruptores, conmutadores, pulsadores) y software implementado en los DMC-s. La reconfiguración de los DMC-s y de las DU-s, puede ser realizada de forma automática o de forma manual. Los paneles dedicados a la reconfiguración, son los siguientes:

EFIS CONTROL PANEL. Este panel, que es parte de la *Flight Control Unit (FCU)*, está dividido en dos partes, una para controlar el PFD (parte derecha) y la otra para controlar el ND.

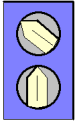


EFIS SWITCHING PANEL. Mediante este panel, se puede controlar tanto el encendido de las pantallas, como su brillo. También se puede seleccionar otro DMC cuando el correspondiente a la configuración normal falla y mediante un pulsador (P/B - pushbutton) intercambiar la información entre el PFD y el ND.



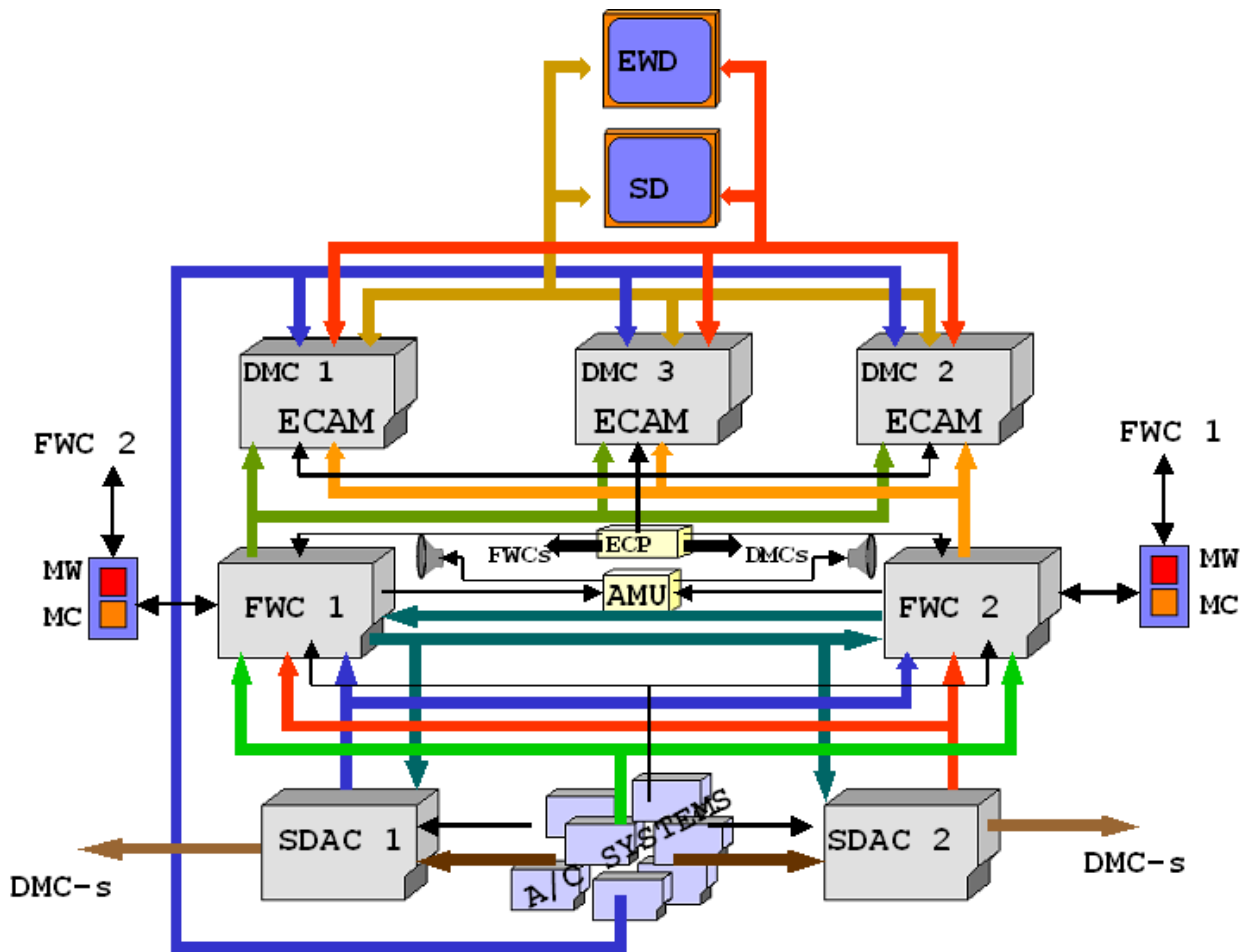
ECAM CONTROL PANEL (ECP). Este panel contiene potenciómetros de brillo y los controles necesarios para el control del ECAM.

ECAM SWITCHING PANEL (ESP). Mediante este panel se selecciona que DMC aporta información ECAM (en la posición automática, es el DMC 3 quien lo aporta). También permite que la información ECAM, sea representada en una de las dos pantallas de ND.



El sistema de llamada de atención esta compuesto por las luces de *Master Warning* (**MW**), las luces de *Master Caution* (**MC**), las luces de *Auto Land* (**AL**) y por los altavoces.

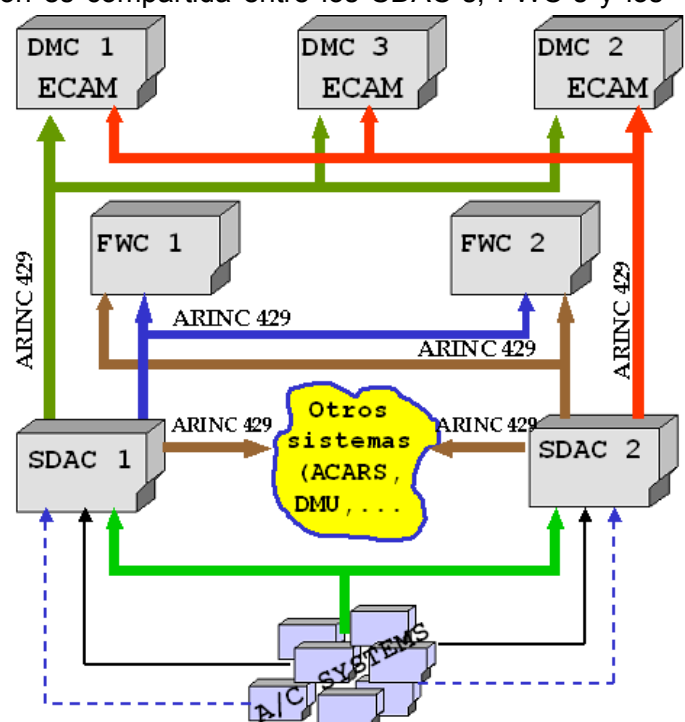
17.2.2-ELECTRONIC CENTRALIZED AIRCRAFT MONITORING (ECAM).



El ECAM realiza principalmente dos funciones: La adquisición de datos y su concentración y el procesamiento de datos de avisos. La adquisición es compartida entre los SDAC-s, FWC-s y los DMC-s. El procesamiento de datos de avisos e información memo, es realizado por los FWC-s.

SDAC.

Los dos SDAC-s son idénticos e intercambiables. Realizan principalmente tres funciones. Adquisición, concentración y digitalización de datos. Además, recogen de los sistemas de la aeronave los datos de fallos o disfunciones correspondientes a situaciones de precaución y los envían a los FWC-s para generar las correspondientes alertas y mensajes de procedimientos. Los dos SDAC-s, recogen y envían a los tres DMC-s todas las señales provenientes de los sistemas de la aeronave necesarias



para ser representadas en las pantallas. Además, son también usados como concentradores de datos para otros sistemas como la FDIU, ACARS, DMU, etc.

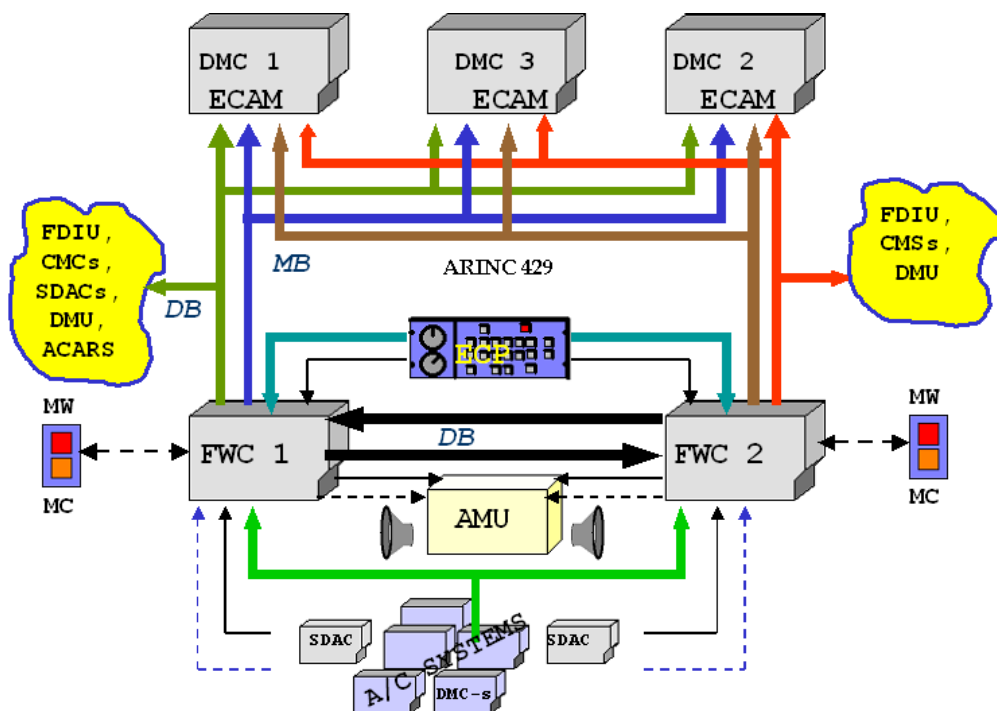
Todas las señales, discretas, analógicas y digitales que se aplican a los SDAC-s, son concentradas y convertidas a formato digital. Las secciones de salida de los SDAC-s reparten las señales digitales en formato **ARINC 429** de alta velocidad (HS).

El software operacional de los SDAC-s está contenido en unos dispositivos llamados *On Board Replaceable Module (OBRM)* situado en el frontal de la LRU, para poder facilitar cualquier modificación del software sin tener que desinstalar la LRU. Cada SDAC tiene un OBRM.

FWC.

Los dos FWC-s son idénticos e intercambiables. Realizan principalmente tres funciones: Adquisición de datos, procesamiento de datos de avisos correspondientes a situaciones de aviso, y procesamiento de los datos de la fase de vuelo. Además, genera todas las alertas y voces sintéticas para radio altura, Automatic Call Out y otros anuncios. Estas señales acústicas son enviadas a la AMU (Audio Management Unit) para ser escuchadas a través de los altavoces.

Los FWC-s proporcionan información visual y acústica para poder determinar en tiempo real, todos los posibles fallos en los sistemas y las configuraciones peligrosas. Cada FWC genera códigos alfanuméricos correspondientes a todos los textos y mensajes que deben ser representados en las pantallas del ECAM. Estos son: títulos de sistemas y avisos, procedimientos asociados a los avisos y estatus y memos.



Los FWC-s adquieren información directamente de sistemas que no están interconectados con los SDAC-s y de los buses de datos de los SDAC-s. Cada FWC también envía una copia de los datos que recibe, al otro FWC a través de un bus ARINC 429.

Las salidas de los FWC-s son de cuatro tipos: (1) discretos para llamada de atención visual, (2) discretos y analógicas para sonidos, (3) ARINC 429 de alta velocidad (HS) llamado FWC Data Bus (DB) que está conectado al FWC opuesto, los DMC-s, FDIU, DMU, CMS-s y

ACARS MU para transmitir varios datos tales como códigos de fallo y a los SDAC-s para propósitos de mantenimiento. (4)ARINC 429 HS, llamado FWC Message Bus (MB) conectado a los DMC-s para transmitir todos los mensajes de aviso.

El software operacional de los FWC-s está dividido en dos partes, una para la función de adquisición de datos, y la otra para el procesamiento. Cada parte de software está contenida en un OBRM, por lo tanto, cada FWC contiene dos OBRM.

Cada FWC centraliza la información BITE procedente de los SDAC-s, del ECP y de su propio BITE. Cada FWC está unido al OMS (Onboard Maintenance Systems) pero solamente el FWC 1 envía la información BITE a ambos CMC-s (Central Maintenance Computers) vía bus ARINC 429 HS.

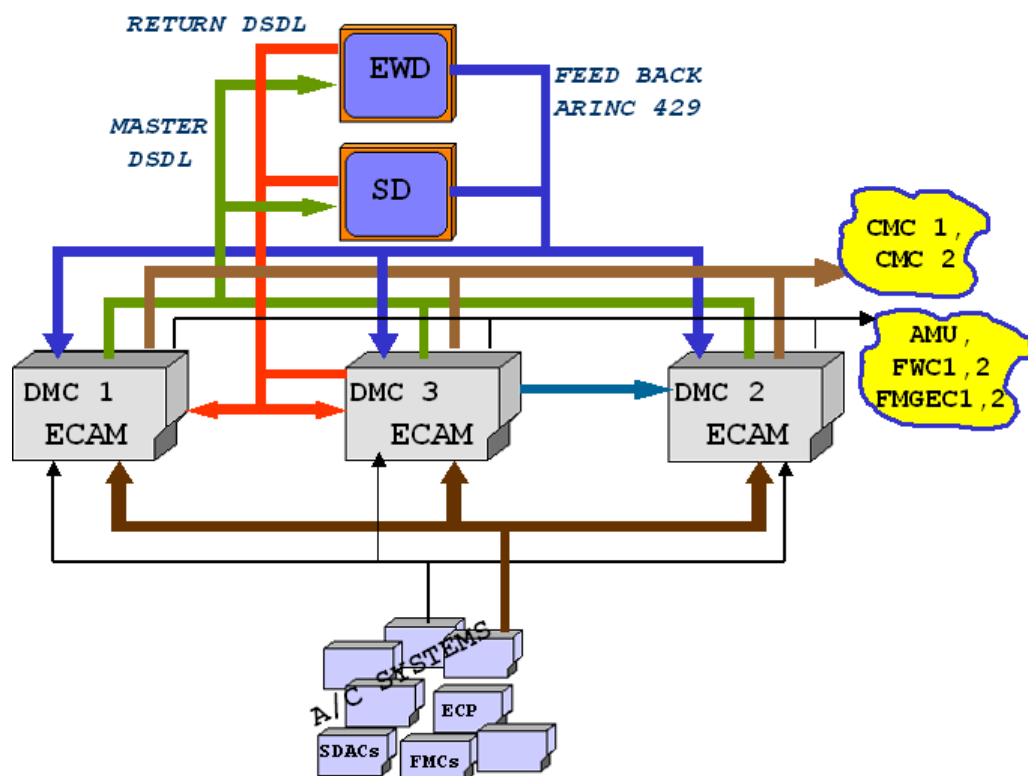
Para cada uno de los FWC un OBRM es dedicado al software del microcomputador 1 (adquisición + BITE) y un OBRM es dedicado al software del microcomputador 2 (lógica de avisos + BITE). El software de audio, está implementado en la placa de salida de audio y discretos, ya que no es susceptible de cambios frecuentes.

DMC.

Los tres DMC-s son idénticos e intercambiables. Reciben datos directamente de los sistemas de la aeronave o de los SDAC-s y FWC-s. Los datos que reciben, los decodifican y procesan para ser representados en las DU-s.

Las dos DU-s son idénticas e intercambiables. La EWD y la SD reciben señales digitales de su correspondiente DMC a través de un **Master DSDL** (*Dedicated Serial Data Link*).

Los DMC reciben datos de los sistemas en formato digital **ARINC 429**, datos del radar meteorológico en formato **ARINC 453** (bus de alta velocidad) y discretos como la reconfiguración de pantallas. Los displays además reciben también datos digitales del radar meteorológico procesados por los DMC-s, a través de un bus particular denominado **RS 422**.



Las DU-s envían a los DMC-s por un lado, señales de reconocimiento a través de un bus DSDL de retorno y por otro lado, parámetros críticos detectados en los displays como pueden ser parámetros primarios de motor, a través de un bus ARINC 429.

Cada DMC lleva a cabo las funciones BITE para sí mismo y para las DU-s a las que está conectado.

ECP.

El *ECAM Control Panel (ECP)* transmite las selecciones de los pilotos a los DMC-s y a los FWC-s. Este panel, recibe datos procedentes de los DMC-s para iluminar las luces correspondientes a las teclas.

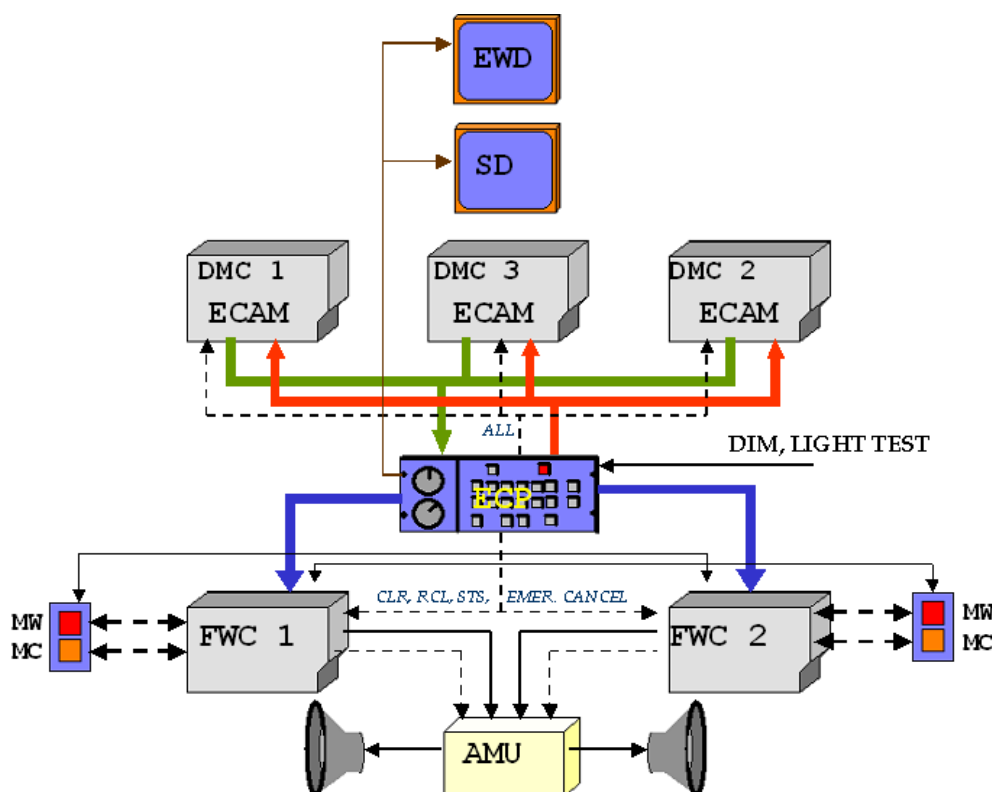
Las funciones posibles a través del ECP son:

- gestión manual del Display de mensajes de aviso en el EWD (funciones Clear, Recall, T. O. configuration y Emergency cancel)
- control manual del display de mensaje de STATUS en el SD.
- selección manual del display de página de sistema en el SD.
- ajuste de brillo de EWD y del SD.

El ECP presenta salidas en forma de discretos para las teclas CLR, RCL, STS y EMER. CANCEL cableadas a los FWC-s, y para la tecla ALL cableada a los DMC-s. A través de este panel, se realiza el control de brillo de los displays del EWD y del SD.

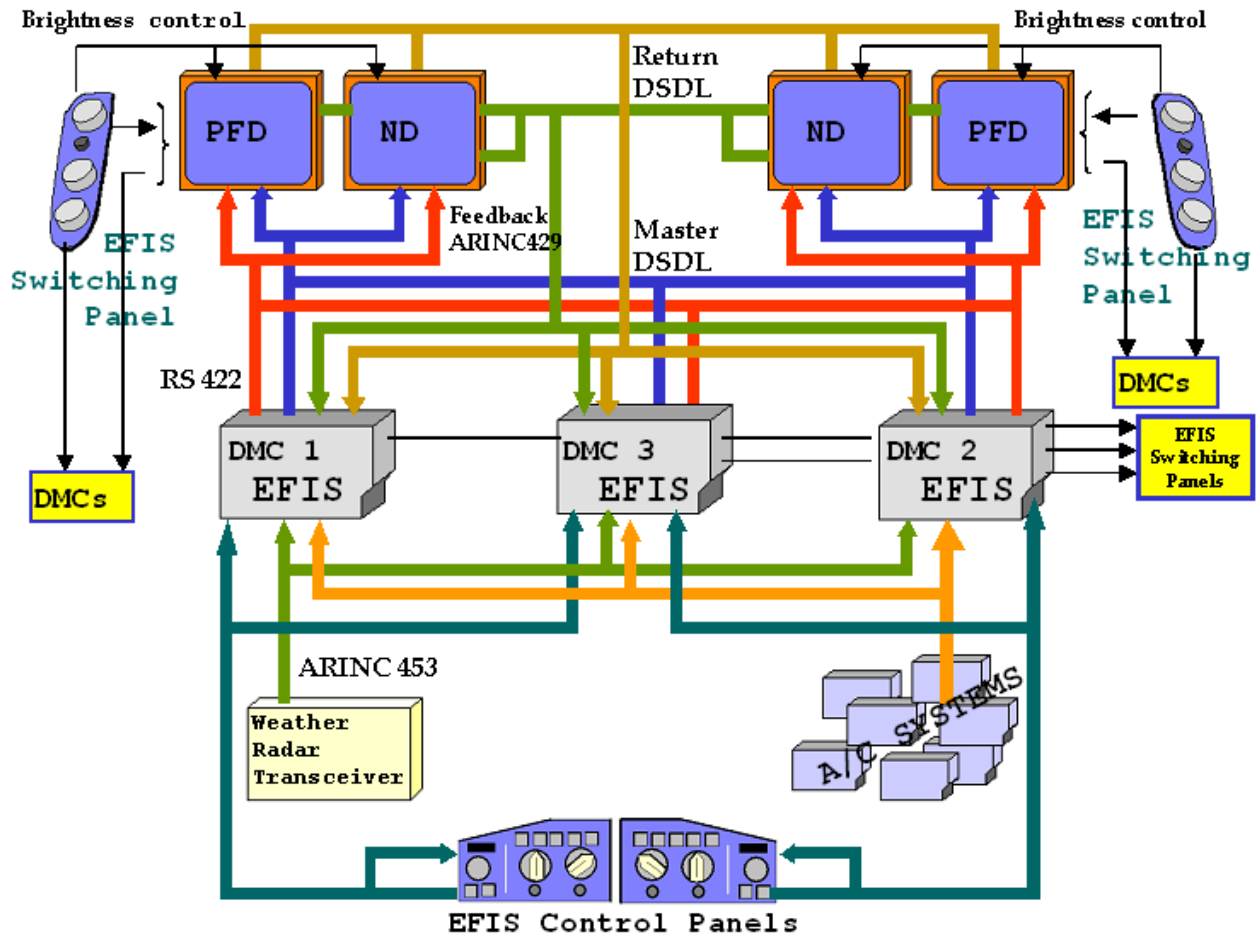
ATTENTION GETTERS.

Las luces de Master Warning (MW) y Master Caution (MC) son controladas por cada FWC-s. Los avisos auditivos y voces sintéticas generadas en los FWC-s, son enviadas a la Audio Management Unit (AMU) junto con señales discretas de sincronización.



17.2.3-ELECTRONIC FLIGHT INSTRUMENTS SYSTEM (EFIS).

El EFIS realiza principalmente tres funciones: La adquisición de datos, su procesamiento (llevado a cabo por los DMC-s) y la representación de los mismos a través de los displays del CPT y F/O.



Los tres DMC-s son idénticos e intercambiables. Reciben datos directamente de los sistemas de la aeronave los decodifican y procesan para ser representados en las DU-s.

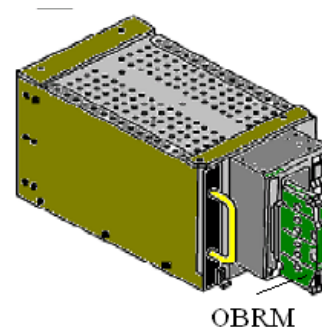
Las tres partes EFIS de los DMC-s reciben datos de los sistemas de la aeronave a través de buses **ARINC 429**, e información del radar meteorológico a través de buses de alta velocidad **ARINC 453**. también reciben algunos discretos como los de reconfiguración de las DU-s. Los datos procesados, son enviados a los PFD-s y a los ND-s a través del Master **DSDL**. La información del radar meteorológico es enviada a través de buses específicos **RS 422**. Las DU-s también reciben información en forma analógica y discretos de los EFIS Control Panels.

Las DU-s envían información de vuelta a los DMC-s. Esta información se envía por un lado, a través de un DSDL como reconocimiento de las señales y para información de fallo de DU, y por otro lado a través de un ARINC 429 para parámetros críticos: ángulos de pitch y roll, altitud, rumbo, posición de la aeronave.

Los EFIS Control Panels (que forman parte de la Flight Control Unit - FCU), están unidos a los DMC-s mediante buses ARINC 429.

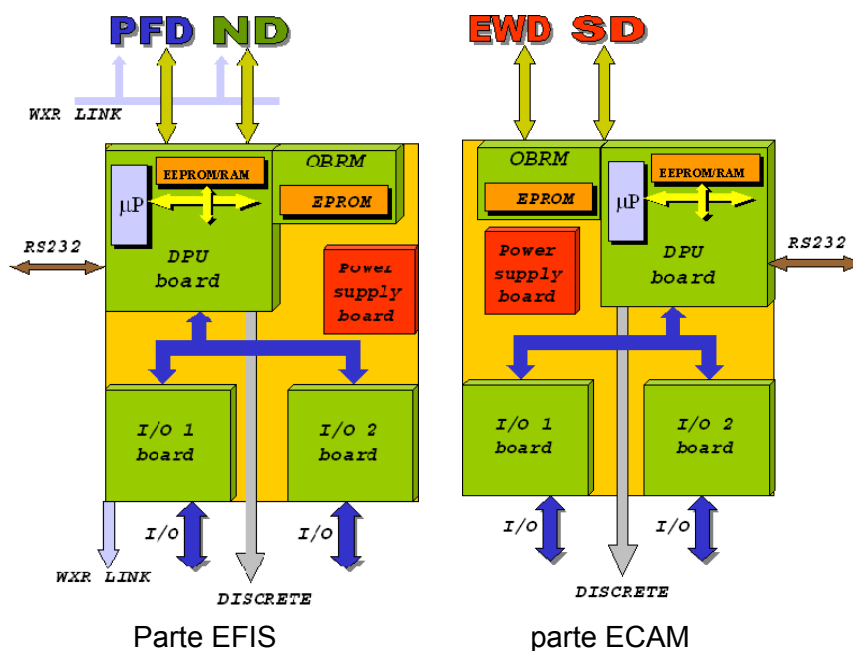
DESCRIPCIÓN DEL DMC.

Desde el punto de vista operativo, cada una de las partes en las que se divide el DMC (la parte EFIS y la parte ECAM) se dividen en un canal de procesamiento compuesto por la **DPU** (*Data Processing Unit*) y el **OBRM** (*On Board Replaceable Module*), varias placas que forman la interfaz I/O, y el módulo de alimentación. El hardware de las dos partes es idéntico. Cada DMC, incluye también dos BITE.



En el canal de procesamiento, cada placa DPU contiene un sistema microcomputador basado en un microprocesador RISC (*Reduced Instruction Set Computers*) de 32 bits. El programa que debe ejecutar este microcomputador, está almacenado en el OBRM asociado. Este programa está guardado en memoria EPROM y ocupa 512K32 para el ECAM y la misma cantidad para el EFIS. La memoria RAM de trabajo y la memoria EEPROM (EAROM) para el BITE asociado, se encuentran en las respectivas placas DPU.

En cuanto a la interfaz de I/O, cada parte del DMC tiene dos placas, I/O 1 y I/O 2. Las señales de I/O comprenden: señales discretas del tipo OPEN/GROUND (en este caso solo de entrada. Las salidas son procesadas por las DPUs), señales ARINC 429 y ARINC 453 para datos del WXR (Solo entradas. Las señales de datos del WXR son procesadas solamente por la placa I/O 1 del EFIS).



Cada una de las partes del DMC tiene asignada información específica de diferentes sistemas aviónicos de la aeronave.

Para la parte EFIS, estos sistemas son:

- WXR** (Weather Radar Transceivers)
- FWC** (Flight Warning Computer)
- DME** (Distance Measurement Equipment)
- FMGEC** (Flight Management Guidance and Envelope Computer)
- ADF** (Automatic Direction Finder)

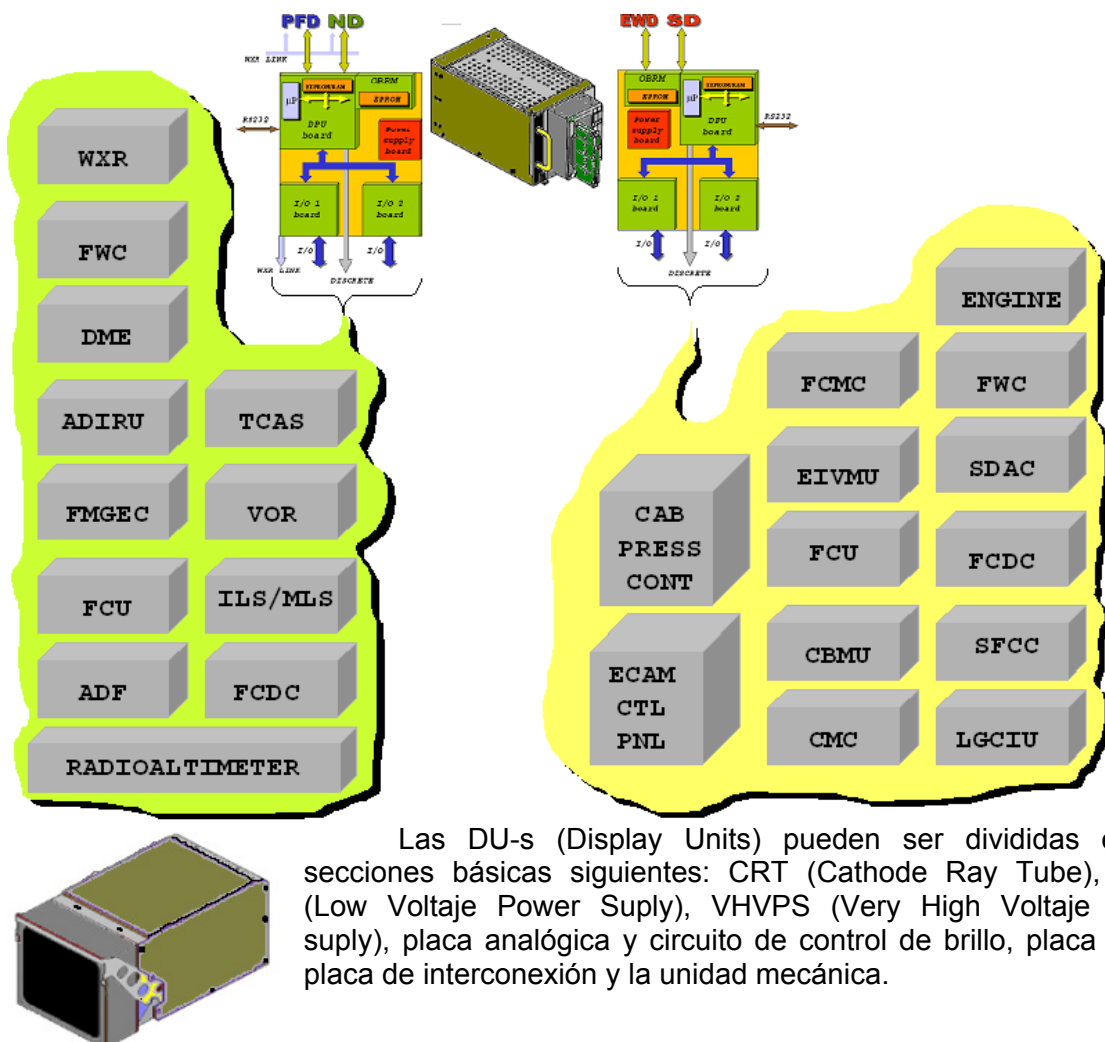
-RADIOALTIMETER

- ADIRU** (Air Data/Inertial Reference Unit)
- FCU** (Flight Control Unit)
- TCAS** (Traffic Alert and Collision Avoidance System)
- VOR** (VHF Omnidirectional Range)
- ILS/MLS** (Instrument Landing System/Microwave Landing System)
- FCDC** (Flight Control Data Concentrator)

Para la parte ECAM, los sistemas son:

-ENGINE

- FWC** (Flight Warning Computer)
- SDAC** (System Data Acquisition Concentrator)
- FCDC** (Flight Control Data Concentrator)
- SFCC** (Slat Flap Control Computer)
- LGCIU** (Landing Gear Control and Interface Unit)
- FCMC** (Fuel Control and Monitoring Computer)
- EIVMU** (Engine Interface and Vibration Monitoring Unit)
- FCU** (Flight Control Unit)
- CBMU** (Circuit Breaker Monitoring System)
- CMC** (Central Maintenance Computer)
- CAB PRESS CONT**
- ECAM CTL PNL**

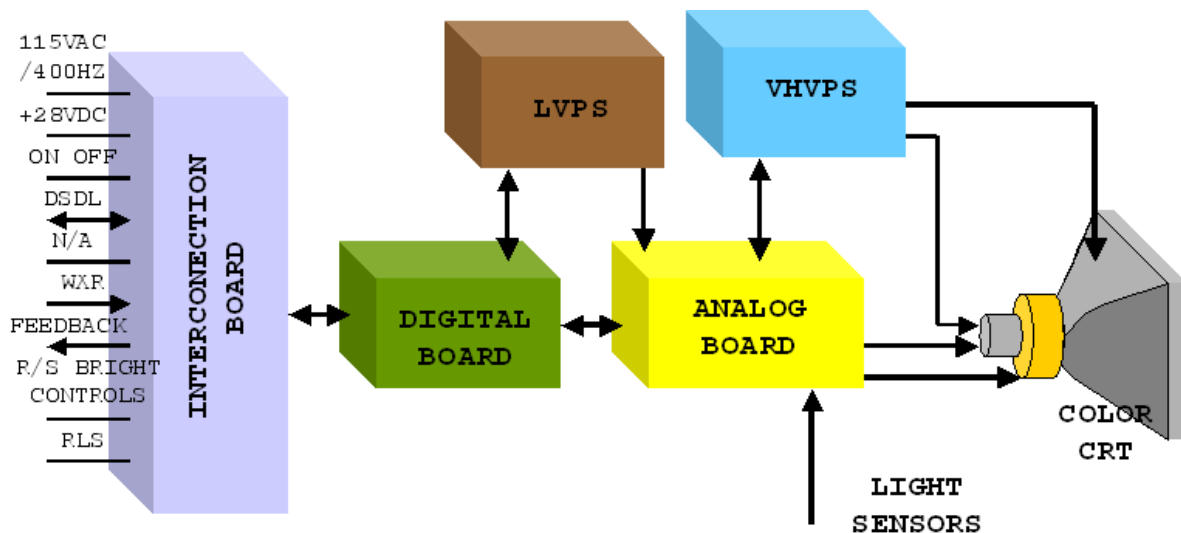


Las DU-s (Display Units) pueden ser divididas en las secciones básicas siguientes: CRT (Cathode Ray Tube), LVPS (Low Voltaje Power Suply), VHVPS (Very High Voltaje Power suply), placa analógica y circuito de control de brillo, placa digital, placa de interconexión y la unidad mecánica.

En la parte frontal del CRT se encuentran dos sensores de luz (RLS- *Remote Light Sensor*-) cuya señal una vez procesada sirve para adaptar la luminosidad del tubo de rayos catódicos a la de ambiente detectada en cabina (Dimming).

La sección de LVPS genera a partir de la tensión de red de la aeronave 115V/400Hz, los voltajes de DC necesarios para la alimentación de los los circuitos electrónicos. Entre estos voltajes se encuentra el valor regulado de +5V para la operación de circuitos lógicos.

La sección de VHVPS genera todos los voltajes necesarios para la operación del CRT: voltaje de ánodo, voltaje para el enfoque y voltaje de preaceleración.



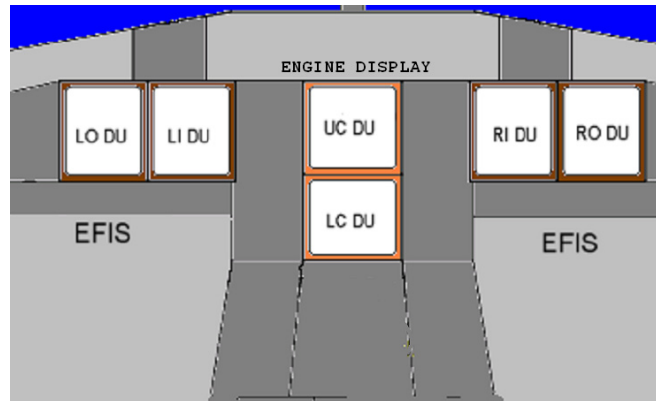
La placa analógica contiene todos los circuitos para el control del CRT tales como el control de brillo del mismo, la desviación X e Y, corrección de la convergencia y circuito de protección.

La placa digital, genera las señales de deflexión y las señales de R, G, B de video para la composición de la imagen en la pantalla. Esta placa recibe a través del enlace **DSDL** (*Dedicated Serial Data Link*) algún programa generador de símbolos, modos de órdenes y palabras de control y lo procesa mediante el programa cargado inicialmente. Además, a través de dos enlaces digitales (**WXR** links) recibe información del radar meteorológico.

La placa de interconexión asegura la conexión eléctrica entre los distintos conectores portadores de alimentación y señales de información.

17.3-CDS (COMMON DISPLAY SYSTEM).

Al igual que el EIS, este sistema tiene como fin proporcionar información de navegación y de motor a la tripulación.



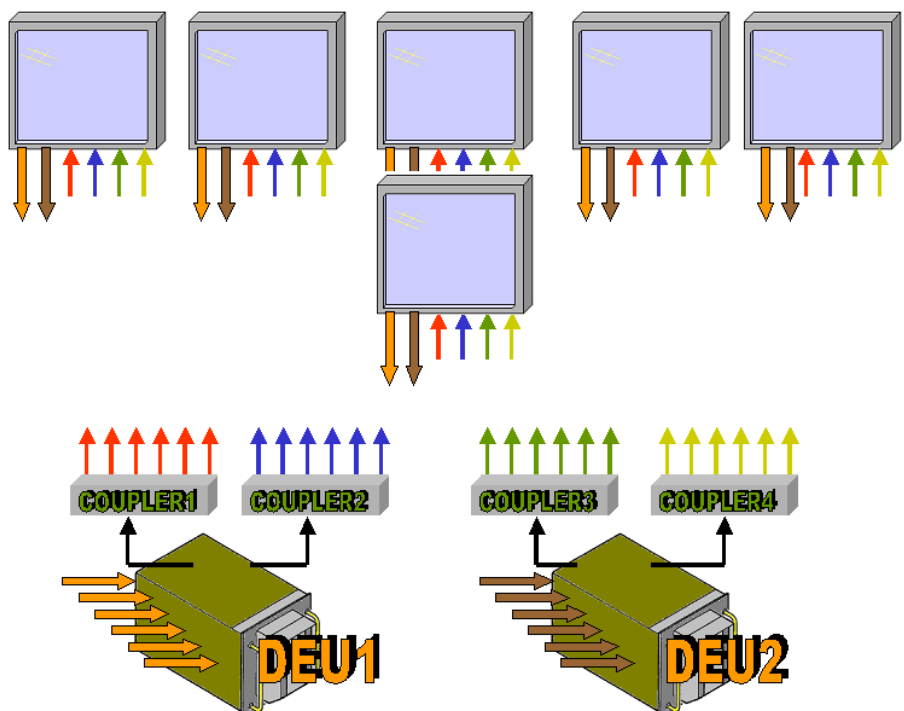
17.3.1-DESCRIPCIÓN GENERAL.

Los componentes del **CDS** (*Common Display System*) son:

2 Display Select Panels, 1 Engine Display Control Panel, 2 EFIS Control Panel, 2 Display Source Selectors, 2 Displays Electronics Units, 4 acopladores coaxiales, 6 Displays Units, 2 Paneles de control de brillo, 2 sensores remotos de luz.

La información que va a aparecer en los displays va a depender de factores como el estado de las DU-s, de su localización y de la selección hecha en los paneles de control. Los formatos que se presentan en las pantallas son display primario EFIS, display secundario EFIS, display compactado y display de motor.

El computador para el CDS es la **DEU** (*Display Electronic Unit*) que como se ha indicado, tiene dos. Estos computadores recogen datos de diversos sistemas aviónicos, transforman estos datos a una señal de video y la envían a través de un cable coaxial. Un acoplador coaxial, separa la señal de video y la envía a las 6 displays units (DU-s). Ambos computadores, envían datos a las 6 DU-s.

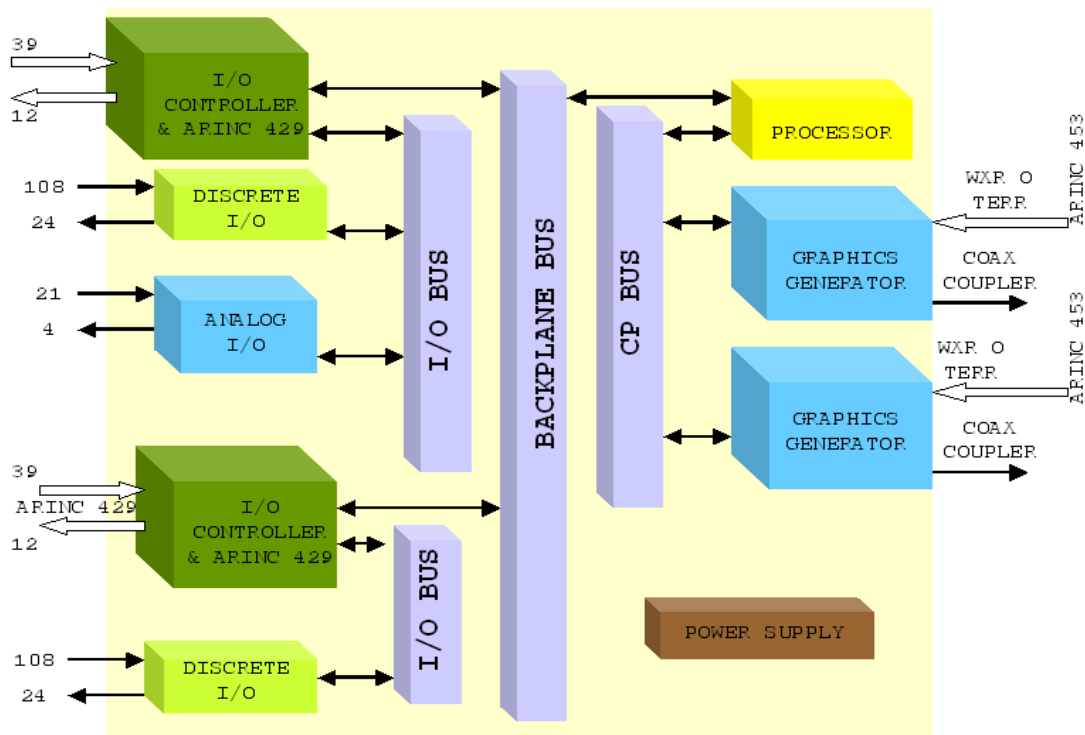


Los DEUs también trabajan como interfaz entre sistemas aviónicos. Por ejemplo, reciben datos BITE de la *Electronic Engine Controller (EEC)* y de la *Auxiliar Power Unit (APU)* y los envían al *Flight Management Computer (FMC)*.

Cada DEU posee dos conjuntos de tarjetas de circuitos generadores de gráficos. (**GG CCA-Graphics Generator Circuit Card Assemblies**). Cada uno de los GG CCA manda datos a un acoplador coaxial el cual separa la señal y manda los datos gráficos a cada una de las DU-s. Por lo tanto, cada uno de los GG CCA puede mandar información a las 6 DU-s. Estas DU-s son automonitorizadas para detectar fallos, en el caso de detección de uno, se lo hacen saber a cada DEU a través de un bus ARINC 429.

Cada DEU envía datos a la otra DEU a través de dos buses de canales cruzados. La salida de los dos buses, contiene información diferente.

Internamente, las DEUs están formadas por nueve tarjetas electrónicas: I/O Controller (IOC) y ARINC 429 I/O (2), discretos I/O (2), analog I/O (1), processor (1), graphics generators (2), power supply (1).



* **I/O Controller (IOC)** y ARINC 429 I/O. Cada IOC y ARINC 429 I/O tiene dos funciones. La función I/O ARINC 429 tiene 39 receptores y 12 transmisores para sus buses. La función IOC, da formato a todos los datos de entrada para mandarlos a través del bus del plano posterior (*Backplane*) al procesador.

* **discretos I/O.** Hay dos I/O de discretos. Cada I/O de discretos, suministra una interfaz para 108 entradas y 24 salidas de discretos. Las entradas son digitalizadas y enviadas a través del bus I/O hacia la función IOC. La función IOC devuelve datos en el bus I/O para que los discretos I/O envíen salidas.

* **analog I/O.** Esta I/O proporciona una interfaz para 21 entradas y 4 salidas analógicas. Las entradas son digitalizadas y enviadas a través del bus I/O hacia la función IOC. La función IOC devuelve datos en el bus I/O para que I/O analógica envíe salidas.

* processor. El procesador recibe y envía datos a la función IOC a través del bus backplane. En el procesador, existen dos microprocesadores idénticos que realizan la misma operación y utilizan el mismo software. El procesador compara las salidas de ambos microprocesadores para comprobar que la información es la misma. El procesador, también compara datos críticos que vienen de las dos funciones IOC. Si los datos no son iguales, existe un fallo en una de las IOC y ARINC 429 I/O. El procesador, envía datos al generador de gráficos a través del Core Processor (CP) bus.

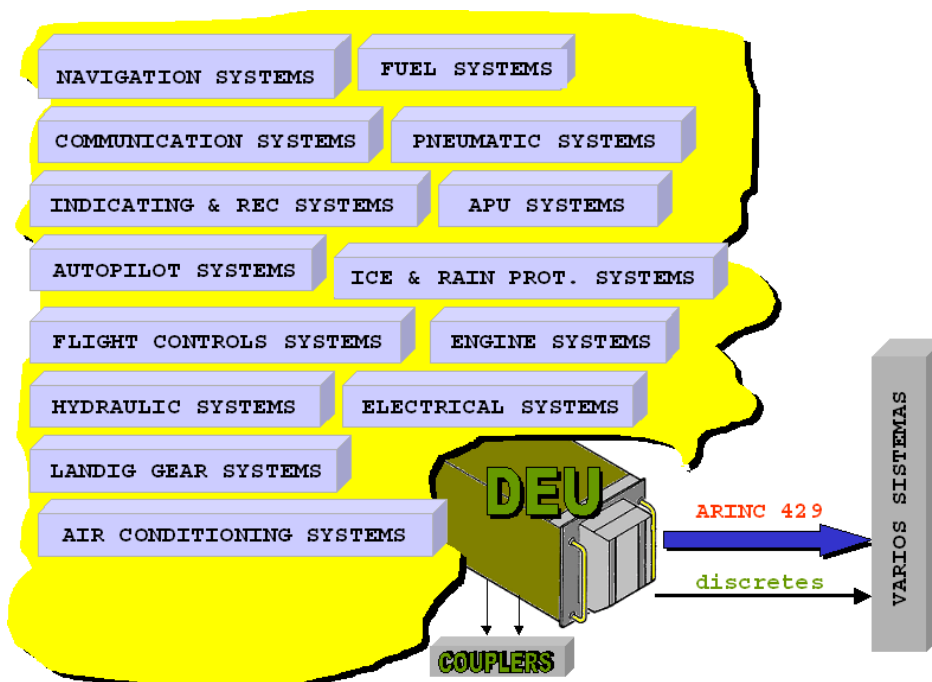
* *Graphics Generators (GGs)*. Hay dos generadores de gráficos idénticos, como se ha indicado anteriormente, estos reciben datos del procesador a través del CP bus para realizar las representaciones.

Los GGs, también reciben datos del radar meteorológico (**WXR**) y del *Enhanced Ground Proximity Warning System (EGPWS)* a través de un bus **ARINC 453**. Las presentaciones, y los datos ARINC 453 son comprimidos y mandados hacia las DU-s a través de cable coaxial.

* Power Suplpy. Suministra energía a todas las placas a partir de los +28 V DC, realizando las conversiones necesarias a voltajes que se requieran.

Muchos sistemas de la aeronave envían datos a las DEUs. En algunos casos, los datos son transformados, convirtiéndose en señales de video para ser mostrados en los displays. En otros casos, las DEUs envían datos a otros sistemas mediante discretos (TCAS Computer, APU ECU, Fuel System Module, Engine Star Switch, etc.) o mediante buses ARINC 429 (ADIRUs, FMC, FDAU, etc.).

Como ejemplo de uno de los sistemas que interactúa con las DEUs, sistema de



navegación, veamos las LRU que envían o reciben datos a o desde las DEUs. Estas son:

-**ADIRU 1** (*Air Data Inertial Reference Unit 1*) y **ADIRU 2**. Envían y reciben en ARINC 429.

-**GPWC** (*Ground Proximity Warning Computer*). Envía y recibe en ARINC 429.

-**TCAS** (*Traffic Alert and Collision Avoidance System*) computer. Envía en ARINC 429. Recibe en discretos.

-**WXR 1** (Weather Radar Transceiver 1). Envía en ARINC 453.

-**FMC 1** (Flight Management Computer 1). Envía y recibe en ARINC 429.

-**FMC** transfer relay 1. Envía en discreto y en ARINC 429.

-**FMC** transfer relay 2. Envía en discreto y en ARINC 429.

- Data Loader Control Panel. Envía y recibe a través del Data Loader Bus.

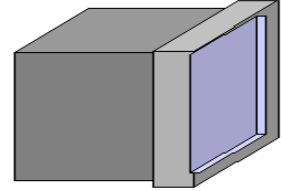
-**MMR 1**(Multi-Mode Receiver 1) y **MMR 2**. Envía en ARINC 429.

-**VOR** (VHF OmnidirectionalRange) / Marker Beacon Receiver 1. Envía en ARINC 429.

-**VOR** (VHF OmnidirectionalRange) / Marker Beacon Receiver 2. Envía en ARINC 429.

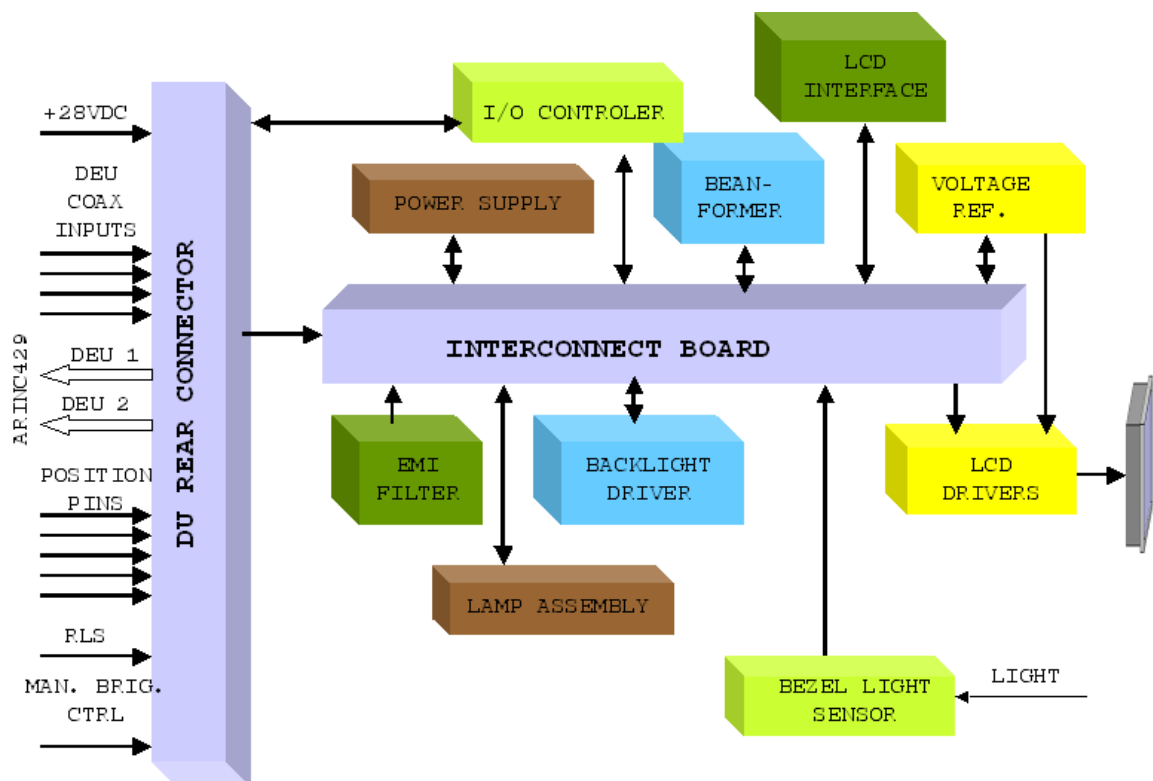
- ADF 1** (Automatic Direction Finder) receiver 1. Envía en ARINC 429.
- DME** (Distance meassurent Equipment) Interrogator 1. Envía en ARINC 429.
- DME** (Distance meassurent Equipment) Interrogator 2. Envía en ARINC 429.
- RA** (Radio Altimeter) Transceiver 1 y -**RA** Transceiver 2. Envía en ARINC 429.
- Captain Navigation Control Panel. Envía en un discreto.
- F/O Navigation Control Panel. Envía en un discreto.

Las DU-s físicamente están formadas por pantallas LCD cuadradas de 20,3 cm de lado y 24,8 cm de profundidad. Cada una de las DU-s tiene sensor de temperatura y un sensor de luz BLS (Bezel Light Sensor).



Todas las conexiones eléctricas a las DU-s van a través de conectores posteriores con protección HIRF. Los generadores de gráficos de las DEUs, envían datos a las DU-s, y estas, envían datos de temperatura, intensidad luminosa y BITE a aquellas. En cada DU hay una placa de interconexión que reparte todas las tensiones, masas, y señales necesarias para la transmisión entre funciones. Una fuente de alimentación se encarga de convertir los en voltajes de aplicación interna, +28 V de alimentación a la DU en +5 V, ± 15 V, +30 V y +375 V.

Un controlador de I/O (IOC) controla todos los datos digitales tanto de entrada (imagen gráfica procedente de la DEU, discretos para la identificación de la DU) como de salida (ARINC 429). Este controlador, también realiza las funciones BITE de la DU.



La imagen gráfica que llega de la DEU, es descomprimida por la unidad de Beamformer y transformada en datos de imagen para la LCD. La interfaz LCD cambia los datos de imagen en voltajes digitales. Estos voltajes, son convertidos a voltajes analógicos de referencia (VREF). Los drivers del LCD utilizan estos voltajes para el control de los 1152 X 1152 píxeles.

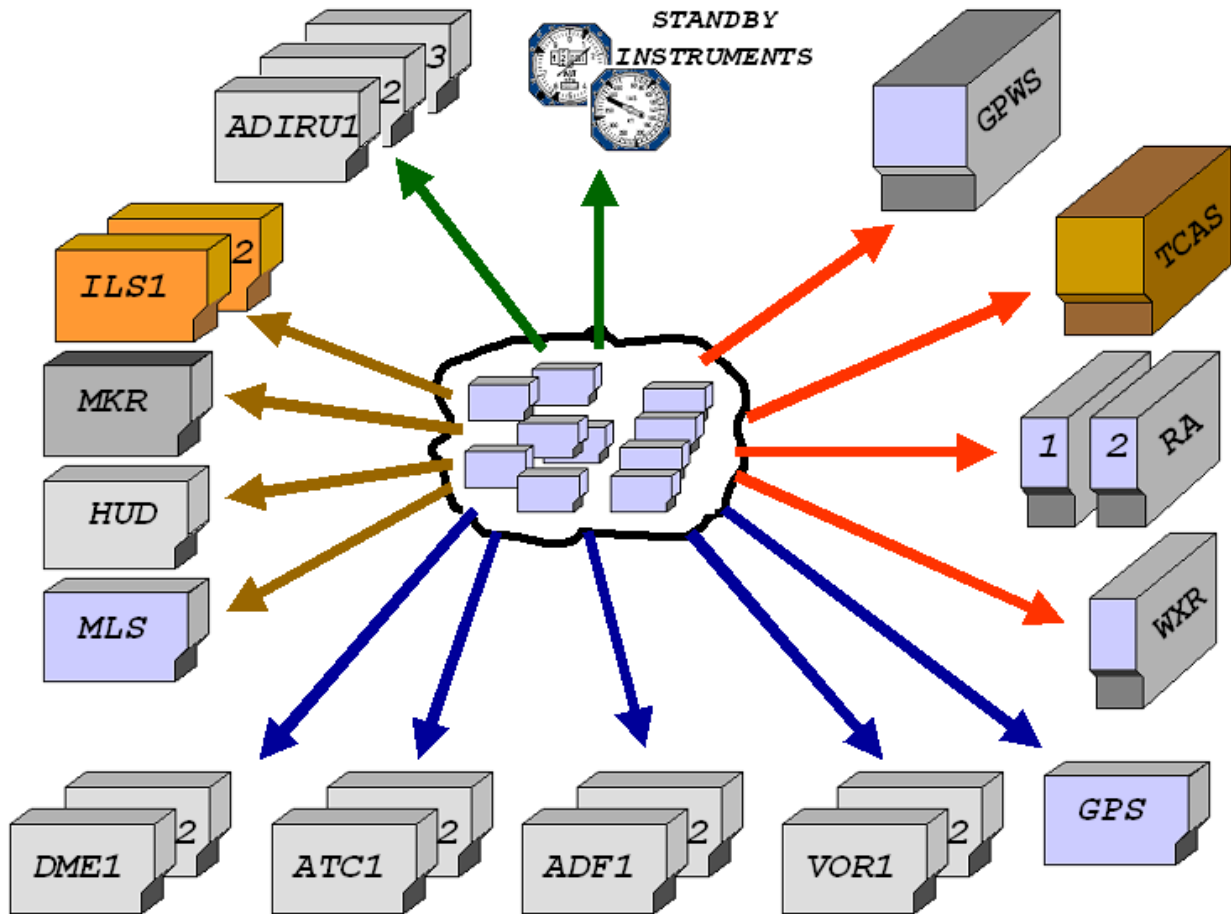
Una unidad (*backlight driver assembly*) controla la luz posterior consistente en un único fluorescente. También controla los calentadores de la lámpara y del LCD, manteniendo la

primera a 60° C y el LCD a 25° C. La unidad de lamp assembly tiene una superficie posterior para reflejar la luz del fluorescente y así incrementar la luz de salida. También posee los sensores de temperatura.

El sensor de luz (BLS) monitoriza la luz incidente, y la placa de interconexión cambia la señal procedente del BLS a formato digital para ser enviado mediante un bus ARINC 429 a la DEU para el control de la intensidad luminosa.

17.4-SISTEMA DE NAVEGACIÓN.

17.4.1-DESCRIPCIÓN GENERAL.



Sistema de datos de aire y referencia inercial:

ADIRU (*Air Data / Inertial Reference Unit*). Tres unidades que proporcionan los parámetros principales de navegación, tales como altitud barométrica, temperatura total del aire, velocidad respecto al aire, ángulo de ataque, altitud inercial, rumbo, posición, etc.

Sistemas de navegación standby:

El sistema de navegación de standby, sirve como respaldo a las ADIRU-s. Está formado por los siguientes instrumentos: una brújula, un altímetro, un indicador de velocidad respecto al aire y un horizonte artificial.

Sistemas de aproximación y ayudas al aterrizaje:

ILS (*Instrument Landing System*). Dos unidades que procesan los datos durante las fases de aproximación y aterrizaje para mostrar la posición de la aeronave respecto a una senda de descenso predeterminada.

MKR (*Marker Receiver*). Se utiliza junto con el ILS durante el aterrizaje. Proporciona indicaciones visuales y auditivas mientras se sobrevuelan cada una de las tres balizas transmisoras situadas a distancias concretas de la cabecera de pista.

HUD (*Head Up Display*). Proporciona imágenes en una pantalla transparente, que permite al piloto ver a través de ella. La información que presenta es la de guiado durante las fases de despegue, aproximación y aterrizaje.

MLS (*Microwave landing System*). La función es la misma que la del ILS. Se puede decir que este sistema es un ILS mejorado.

Sistemas de determinación de la posición dependientes:

DME (*Distance Measuring Equipment*). Proporciona la distancia entre la aeronave y la estación predeterminada de DME en tierra.

ATC (*Air Traffic Control*). Permite que la aeronave sea localizada e identificada por el control de tráfico aéreo en tierra o por el equipo TCAS de otra aeronave.

ADF (*Automatic Direction Finder*). Muestra la orientación hacia un radiofaro NDB en tierra con respecto al rumbo de la aeronave.

VOR (*Very high frequency Omni directional Range*). Muestra la orientación hacia una estación de tierra respecto al rumbo de la aeronave. También indica la desviación de la aeronave respecto a un curso preseleccionado.

GPS (*Global Positioning System*). Proporciona la posición de la aeronave mediante el procesamiento de señales procedentes de una red de satélites.

Sistemas de determinación de la posición independientes:

WXR (*Weather Radar*). Permite la detección y localización de perturbaciones atmosféricas y proporciona a través de un display una imagen de la intensidad de dicha perturbación.

RA (*Radio Altimeter*). Muestra de forma precisa la altura de la aeronave, respecto a tierra hasta una altura de 2500 pies.

GPWS (*Ground Proximity Warning System*). Proporciona avisos auditivos y visuales si la aeronave está en una configuración peligrosa próxima al suelo.

TCAS (*Traffic Collision Avoidance System*). Proporciona avisos auditivos y visuales si la aeronave está en una configuración peligrosa respecto a otro tráfico aéreo, durante el vuelo.

COCKPIT LAYOUT.

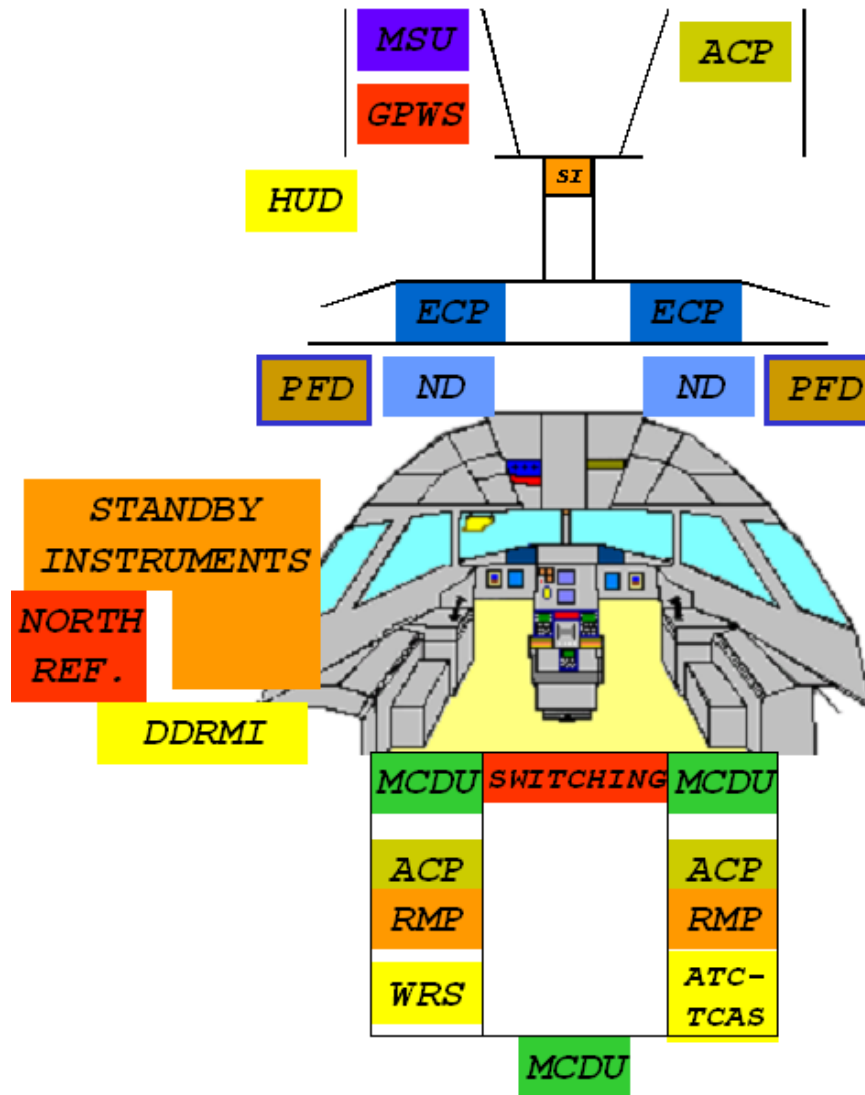
PFD. (*Primary Flight Display*). Muestra datos inmediatos para el control de la aeronave.

ND. (*Navigation Display*). Muestra datos de medio y largo plazo necesarios para la navegación.

ECP. (EFIS Control Panel). Permite a la tripulación el control de PFD y ND. El ECP forma parte de la *Flight Control Unit* (**FCU**).

MSU. (Mode Selector Unit). Permite a la tripulación energizar las ADIRU-s y monitorizar sus estados y fallos.

MCDU. (Multipurpose Control/Display Unit). Permite que los parámetros de las ADIRU-s sean inicializados y representados. También, controla el sistema de radio-navegación (VOR, ADF, ILS, DME).



SWITCHING. En caso de fallo, mediante este panel (parte de él), se puede seleccionar las ADIRU-s que serán fuente de datos a representar en los displays.

NORTH REF. P/B. Sirve para conmutar desde norte magnético a norte verdadero para la navegación polar.

El sistema **STANDBY INSTRUMENTS** lo forman una brújula magnética, un indicador de velocidad respecto al aire (IAS) basado en la diferencia de presiones entre la estática y la PITOT, un altímetro alimentado con presión estática por el sistema standby de datos de aire y un horizonte artificial.

ACP (Audio Control Panel). Permite el control de audio y de las señales de radio-

navegación para identificación.

RMP (Radio Management Panel). Permite un control de respaldo de la radio-navegación. (Modo Backup)

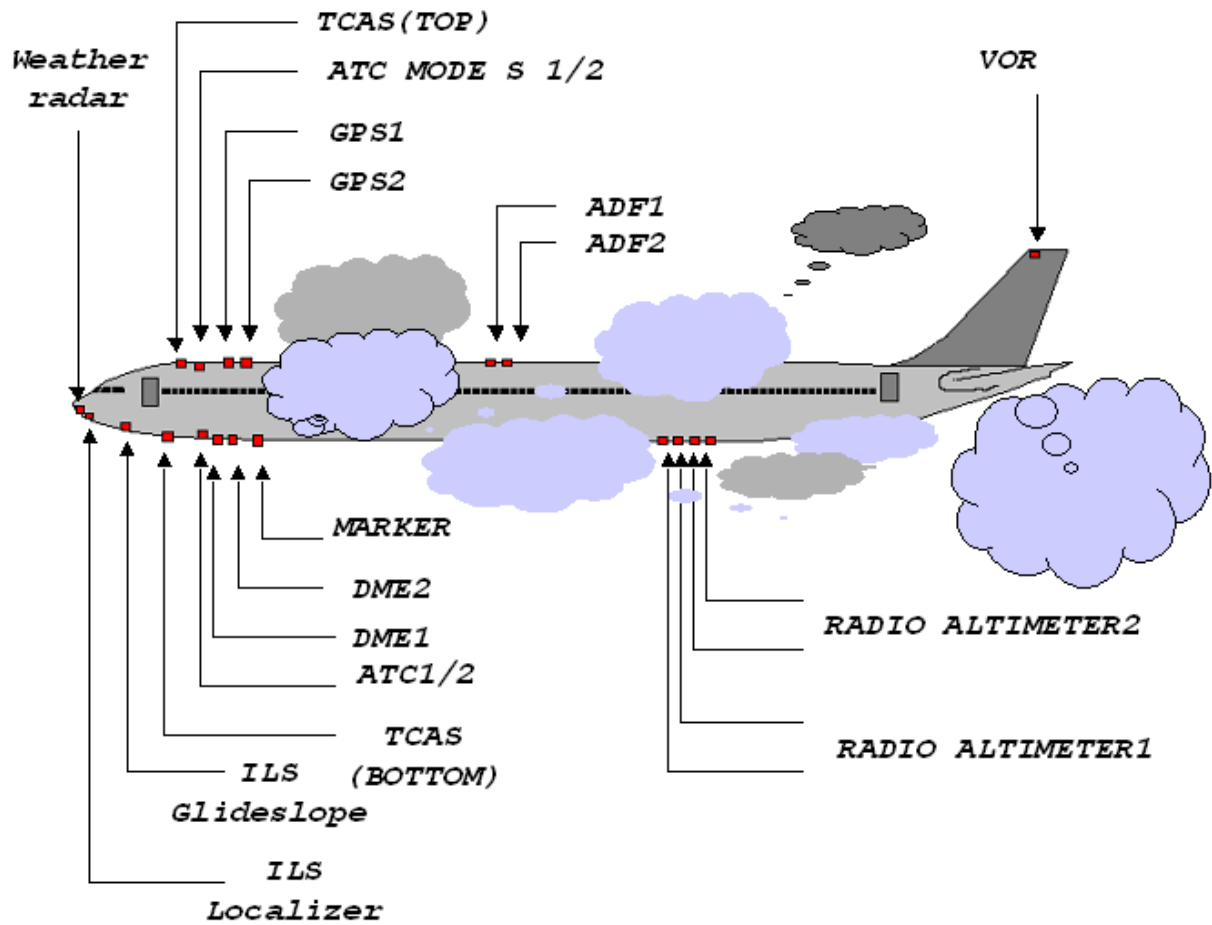
DDRMI (Digital Distance Radio Magnetic Indicator). Muestra las indicaciones de VOR, ADF y DME.

WRS (Weather Radar System). Sirve para controlar el sistema de radar meteorológico.

ATC-TCAS (Air Traffic Control- Traffic Alert and Collision Avoidance System). Sirve para controlar los diferentes modos de operación del sistema de ATC y TCAS.

GPWS (Ground Proximity Warning System). Sirve para controlar el GPWS.

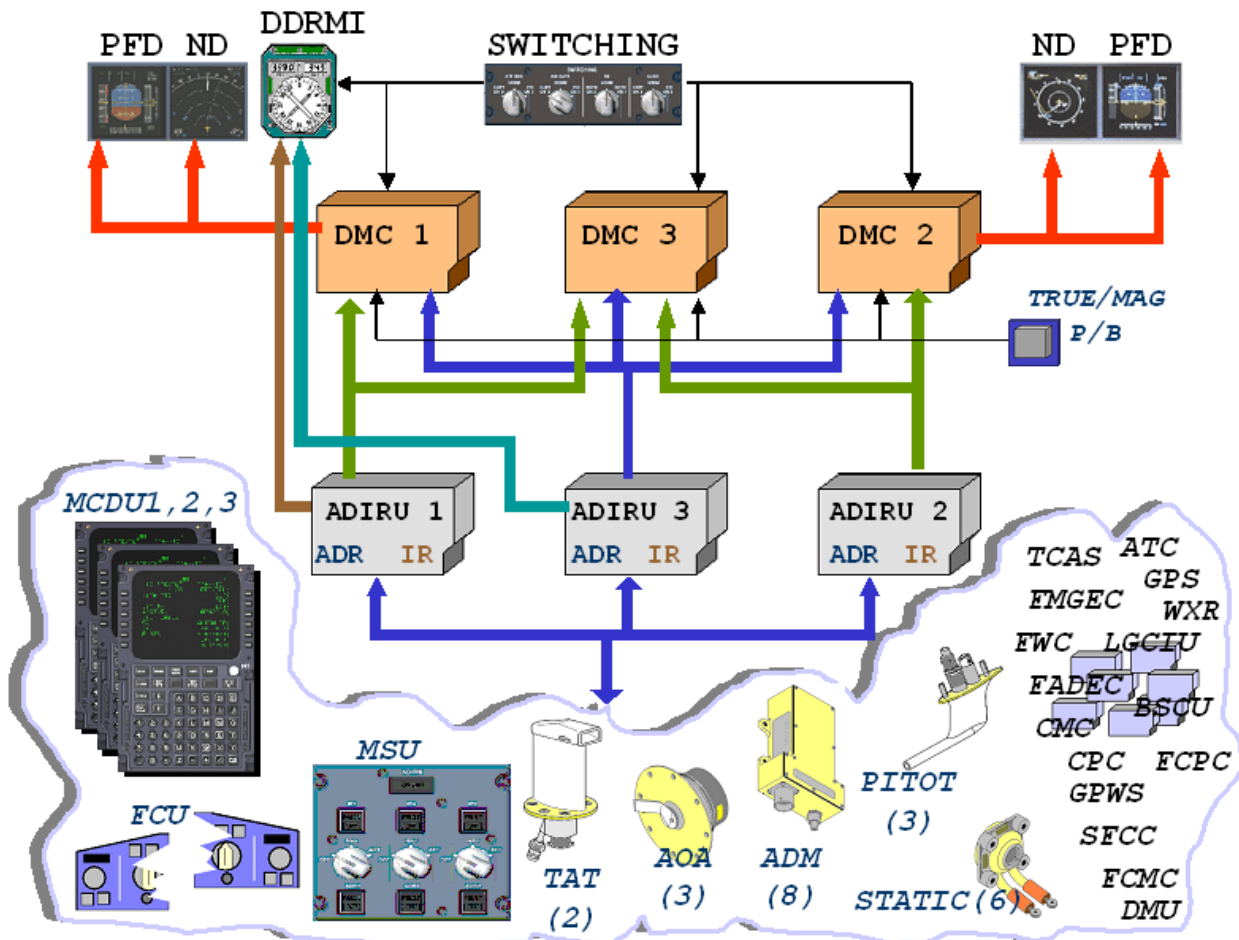
HUD (Head Up Display). Proporciona información de guiado.

LOCALIZACIÓN DE LAS ANTENAS DE NAVEGACIÓN.

17.5-ADIRS (AIR DATA INERTIAL REFERENCE SYSTEM).

17.5.1-DESCRIPCIÓN GENERAL.

El ADIRS pertenece al sistema de navegación. Sus principales componentes son los tres **ADIRU**-s, los cuales proporcionan datos (ARINC 429) de navegación a la tripulación, de acuerdo a varias entradas y controles. Cada ADIRU se compone de una combinación de dos bloques funcionales: referencia de datos de aire (**ADR**) que proporciona la función de datos de aire y referencia inercial (**IR**) que proporciona las funciones de actitud, rumbo, velocidad respecto a tierra y posición actual.



Las tres **ADIRU**-s proporcionan señales a los PFD-s (Primary Flight Displays) y ND-s (Navigation Displays) a través de los DMC-s (Display Management Computers).

Las partes IR de las ADIRU-s 1 y 3 proporcionan datos al *Digital Distance and Radio Magnetic Indicator* (**DDMRI**).

Dependiendo de cómo esté el pulsador True/Mag P/B (pulsador Verdadero/Magnético), los datos de rumbo presentados en el PFD o ND, están referidos al norte geográfico (True) o magnético (Mag).

Las **MCDU**-s (*Multipurpose Control Display Unit*) permiten a la tripulación realizar alineamiento, controlar y representar parámetros del ADIRS.

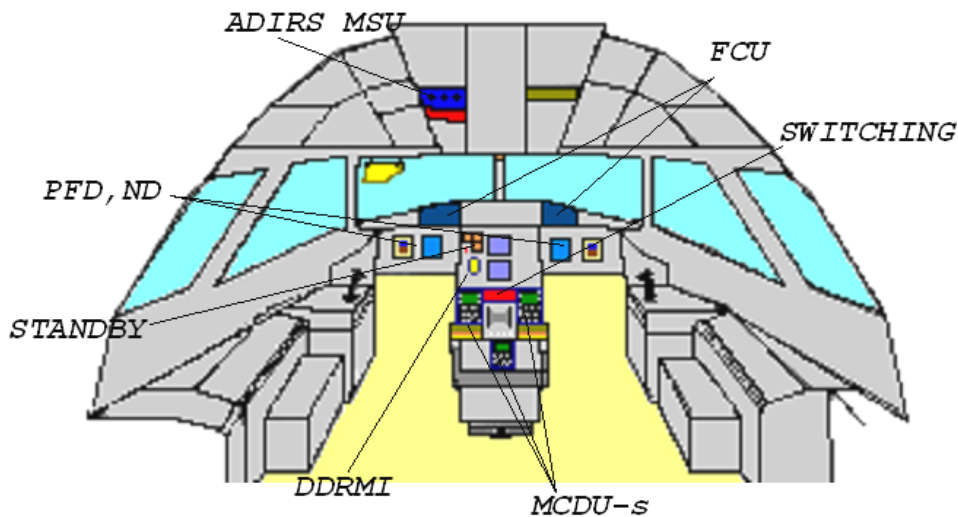
La **MSU** (*Mode Selector Unit*) sirve para controlar el modo de operación e indicar los fallos, si existen, en las tres ADIRU-s.

Una serie de sondas y sensores (dos *Total Air Temperature-TAT*, tres *Angle of Attack-AOA*, tres sondas *PITOT* y seis *Static Ports*), se encargan de suministrar al sistema, datos de aire. Las sondas *PITOT* y los *Static Ports*, están conectados a los *Air Data Module (ADM)* los cuales integran transductores de presión para enviar datos de presión estática o total a los ADIRU-s a través de un bus ARINC.

En la *Flight Control Unit (FCU)* dos botones independientes de ajuste barométrico se usan para proporcionar correcciones barométricas a las ADIRU-s asociadas.

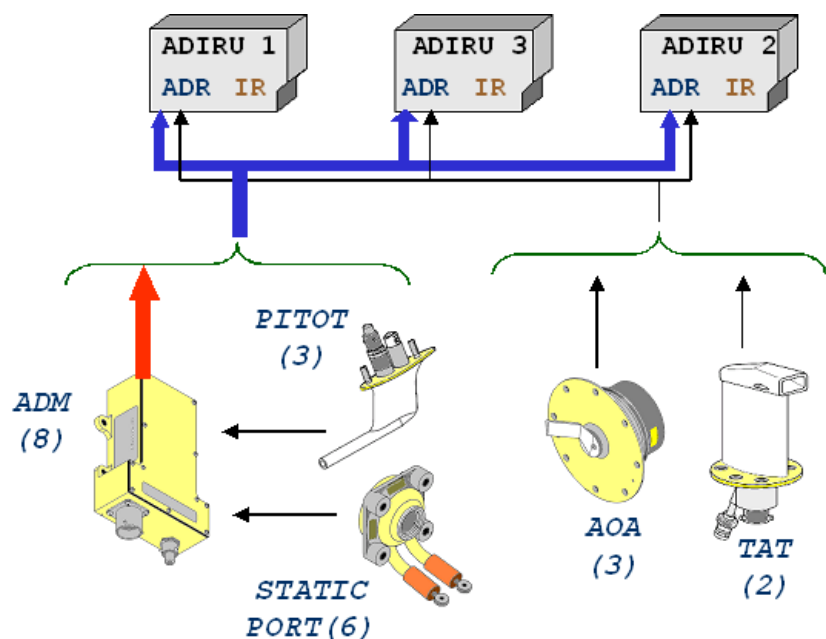
En la configuración básica, la ADIRU 1 está asociada a los instrumentos del CAPT, la ADIRU2 a los instrumentos del F/O y la ADIRU 3 está en Standby.

Hay dos switches selectores para referencia de datos de aire, y para referencia inercial. En caso de fallo de la función de referencia de datos aire (ADR), del ADIRU 1 o 2, los instrumentos o displays afectados pueden ser manualmente conmutados al ADIRU 3. Esta posibilidad también existe para la función de referencia inercial (IR).



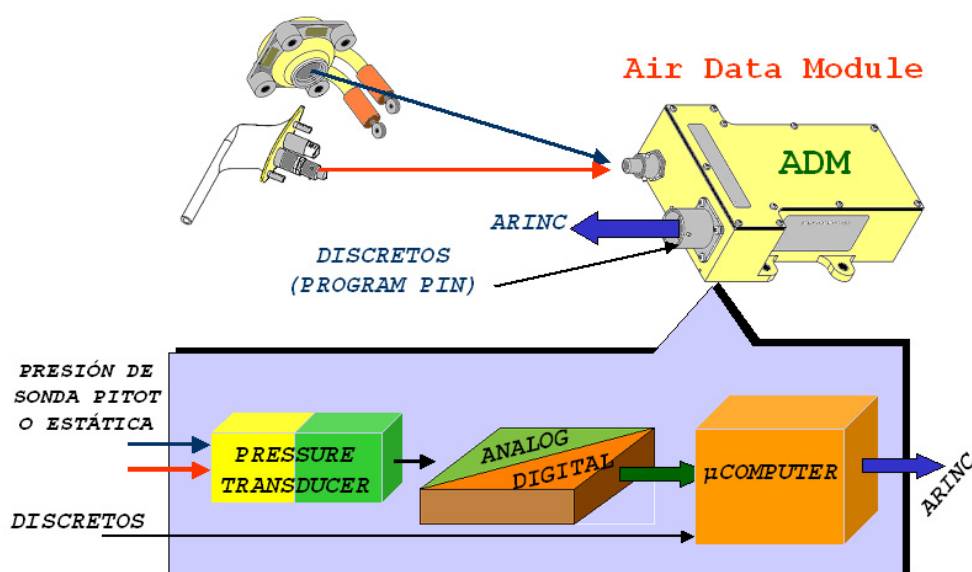
FUNDAMENTOS BÁSICOS.

Las sondas de estática y las de *PITOT*, están conectadas a la ADIRU a través del *Air Data Module (ADM)*, es decir, la información que las sondas entregan, es procesada por el ADM y este entrega dicha información procesada a la función ADR de las ADIRU-s. La información dada por los sensores de ángulo de ataque (*AOA*) y de temperatura total del aire (*TAT*) es entregada directamente a la parte ADR de las ADIRU-s.



ADM.

Todos los ADM son idénticos. Las entradas a esta LRU son de tipo discretos (*Program Pins*) y presión debida a las sondas PITOT o a las tomas de estática.



Los discretos determinan de forma codificada a través de los pines programables, la instalación donde está localizado el ADM mediante el SDI, tipo de datos de presión (PITOT o STATIC) y de que lado provienen, y la paridad, para entregar a la función ADR del ADIRU. Esta última, proporciona $\pm 13,5$ V DC para alimentar cada uno de los ADM que tiene asociado.

Cada uno de los ADM comprende básicamente las siguientes partes:

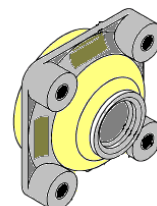
- *una placa transductora que proporciona dos señales de impulsos, una en función de la presión y otra en función de la temperatura.
- *un μ computador que procesa las señales de impulsos para calcular la presión.
- *una placa para la protección contra rayos y **EMI** (*Electromagnetic Interference*) a base de filtros y una FA de 13,5 V AC.

El microcomputador procesa las señales de discretos y las señales eléctricas procedentes de la presión de entrada al ADM del elemento que tenga conectado (PITOT o STATIC PORT). La salida se realiza a través de un transmisor **ARINC 429 LS** (*low speed*), que suministra información digital de la presión, tipo de presión e identificación del ADM al ADIRU.

SONDAS Y SENSORES DE AIRE.**STATIC PORTS.**

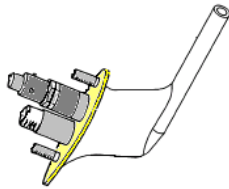
.- Los STATIC PORTS del CAPT y F/O suministran directamente presión estática para cuatro ADM-s que convierten esta presión en una señal digital. Cada uno de los tres sistemas (CAPT, F/O y STBY) comprenden dos puertos de estática, los cuales, están unidos a cada parte ADR de los ADIRU-s a través de cinco ADM (las dos sondas de estática de STBY, están unidas a un solo ADM).

Estos sensores están protegidos contra la congelación, mediante un elemento calefactor alimentado a 28 VDC.



Los STBY STATIC PORT suministran una presión promedio, directamente a los instrumentos de standby (indicador de velocidad de aire y altímetro) y a un ADM conectado a la ADIRU 3.

PITOT.



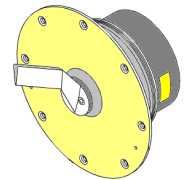
.- Cada uno de los tres sistemas (CAPT, F/O y STBY) tiene asociado una sonda PITOT, la cual está unida a cada ADR de las ADIRU-s a través de un ADM. Las tres sondas PITOT, suministran directamente la presión total a su correspondiente ADM.

Estas sondas, están protegidas contra la congelación, mediante un elemento calefactor alimentado a 115 VAC.

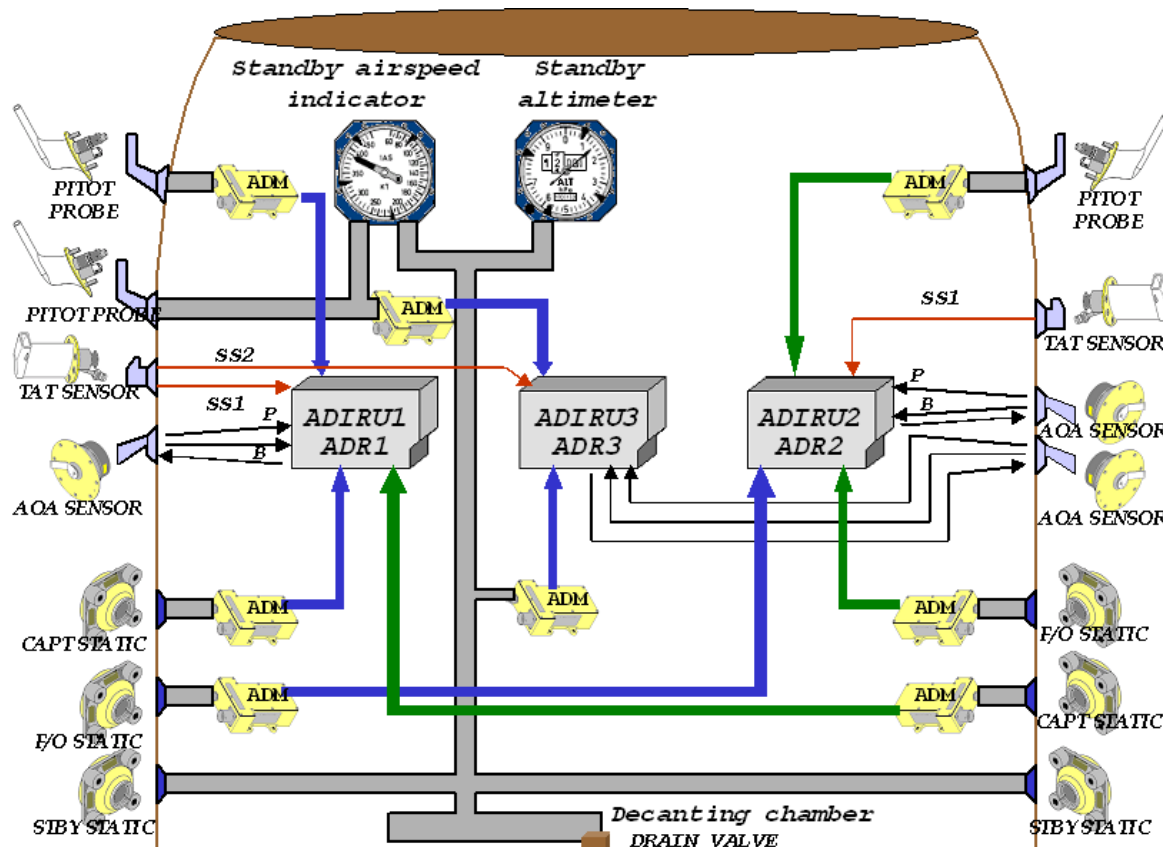
AOA.

.- Cada sensor de ángulo de ataque (AOA) suministra señal primaria (P) y de backup (B) a cada ADIRU. En caso de fallo, la ADIRU realiza la selección de la señal de forma automática.

El sensor es del tipo álabe el cual esta posicionado en la dirección de la corriente del aire. El mecanismo del alabe se encuentra estabilizado alrededor del eje de rotación y mueve los dispositivos encargados de transmitir la señal del ángulo de ataque actual. Estos dispositivos transmisores están formados por resolvers alimentados a 26 VAC/400Hz, que convierten la información angular en señales (seno y coseno) eléctricas proporcionales al ángulo. Un dispositivo de amortiguamiento filtra las oscilaciones mecánicas.

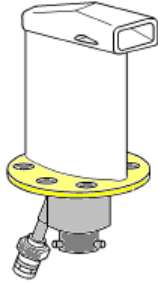


Estas sondas, están protegidas contra la congelación, mediante un elemento calefactor alimentado a 115 VAC.



TAT

.- Cada sensor de temperatura total del aire (TAT) tiene dos elementos sensores independientes. Para el sensor de TAT del CAPT, los dos elementos sensores están conectados. Uno al ADIRU 1 y el otro al ADIRU 3. Para el sensor de TAT del F/O, solamente está conectado un elemento sensor al ADIRU 2.



El elemento básico sensor es un hilo conductor de platino. La resistencia, de este hilo, variará en función de la temperatura del aire. Procesando estas variaciones, se obtiene la temperatura.

Estas sondas, están protegidas contra la congelación, mediante un elemento calefactor alimentado a 115 VAC.

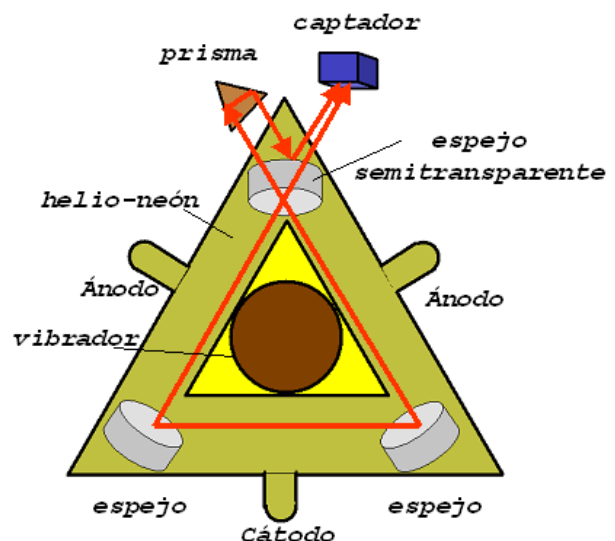
Las sondas se encuentran instaladas en el exterior de la aeronave, de tal manera que las líneas de presión no necesitan un drenaje de agua, a excepción de los STBY STATIC PORTS.

Los datos que se pueden obtener de la sección de ADR de la ADIRU son velocidad de aire, número de Mach, temperatura estática, temperatura total, altitud barométrica.

GIRÓSCOPOS LASER.

Cada eje de rotación de la aeronave tiene asociado un giróscopo láser que proporciona datos de rotación inercial de la misma en cada uno de los ejes.

Este giróscopo se compone básicamente de dos haces láser con direcciones opuestas dentro de un anillo. Mientras el giróscopo está en reposo, los dos haces láser llegan al captador con la misma frecuencia. Cuando existe rotación sobre cualquiera de los tres ejes de la aeronave, en el/los giróscopos correspondientes, los dos haces láser llegarán con diferente frecuencia a los captadores. Esa diferencia en frecuencia es función directa de la magnitud y sentido del giro realizado.



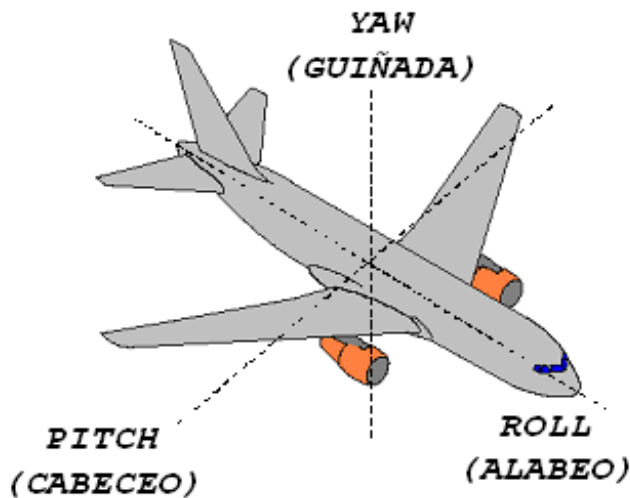
La diferencia en frecuencia, medida por medios ópticos, es computada para obtener una señal digital que una vez procesada ofrece datos de velocidad de rotación (yaw, roll y pitch rate).

ACELERÓMETROS.

Cada eje de rotación de la aeronave tiene asociado un acelerómetro, cada uno de los cuales proporciona aceleración lineal en su eje respectivo.

La parte de referencia inercial del ADIRS, proporciona referencia de actitud, rumbo (magnético o verdadero), aceleraciones y variaciones angulares. A través del software de la unidad, se puede calcular también la posición inercial, las velocidades respecto al suelo, velocidad inercial vertical, ángulo de deriva, datos del viento y datos de la ruta de vuelo.

Una vez procesada la información procedente de los giróscopos y de los acelerómetros, cada ADIRU proporciona datos de referencia inercial a los sistemas usuarios.



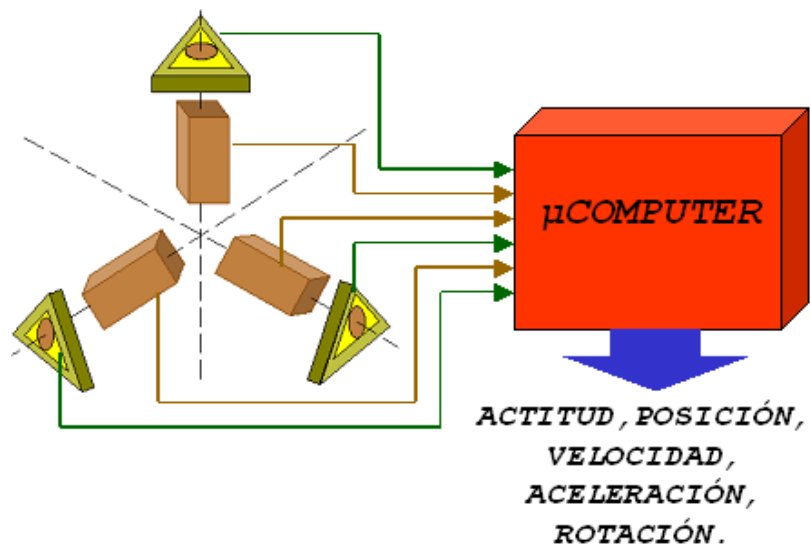
ALINEAMIENTO.

El alineamiento es el primer modo de inicialización para la sección de la referencia inercial (IR) de la ADIRU. Su función es inicializar las funciones de integración de la actitud, velocidad y posición, implementadas en el modo de navegación.

El alineamiento de la IR se divide en tres partes: procesamiento a nivel aproximado, procesamiento de girobrújula (o azimut) y nivel, y procesamiento de datos de entrada.

.- En procesamiento a nivel aproximado, estima la vertical local usando los tres acelerómetros y la medida de la aceleración de la gravedad. Este proceso está en marcha durante los primeros 30 s del modo de alineación de IR.

En este tiempo el software calcula las coordenadas de nivel, el cabeceo, el alabeo, y las aceleraciones asociadas.



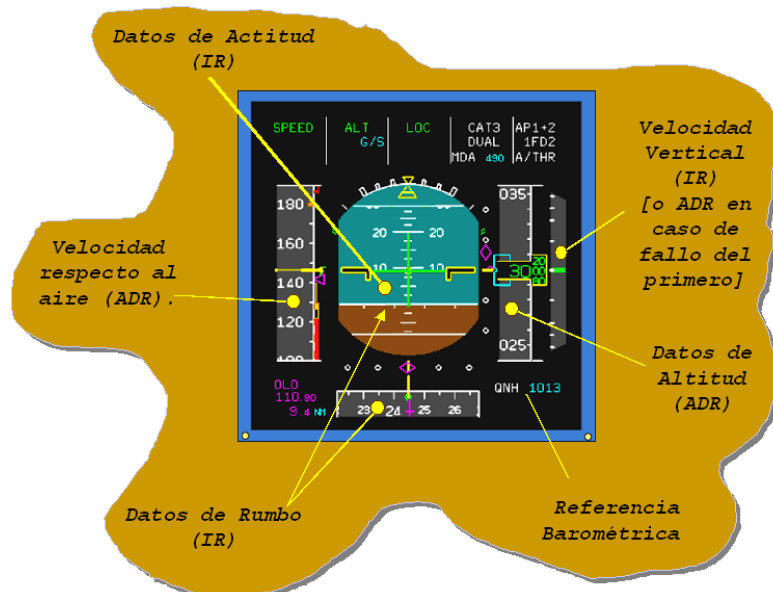
.- La segunda parte entra después de los primeros 30 s (cuando la parte primera ha finalizado) y dura al menos 9,5 minutos en función de la latitud. El procesamiento de azimut se utiliza para orientar el cuerpo de la estructura al norte (usando la rotación de la tierra detectada por los giróscopos). Durante esta parte, se procesa una latitud estimada, calculada usando la componente vertical de la rotación de la tierra, y los resultados de la primera parte se agudizan para obtener mejor conocimiento de las coordenadas geográficas.

.- La tercera parte puede tener lugar en la primera o segunda parte. El par longitud y latitud procesado para inicialización de la posición se recibe de la misma fuente (FMGEC1 o 2). Tan pronto como se recibe el par de inicialización de posición válida, el software realiza tres tests BITE para comprobar si la latitud y longitud están dentro de unos límites comparado con la posición grabada al final del último ciclo de energización.

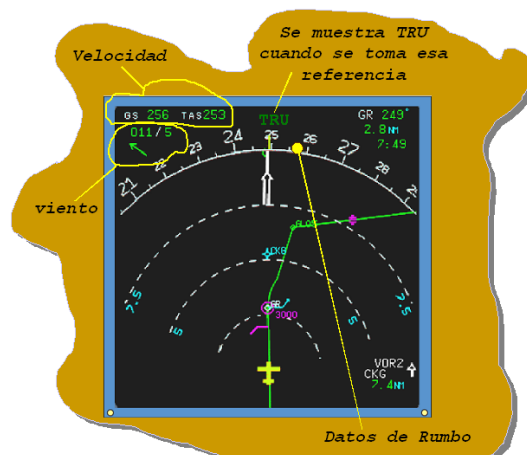
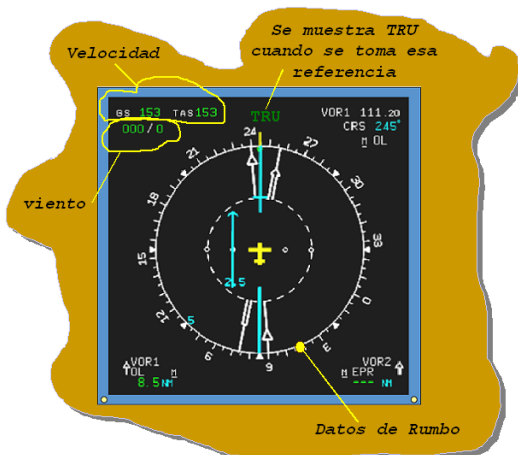
El alineamiento puede quedar completo después de 10 minutos si se han recibido unos datos válidos de posición y han sido verificados por la IR.

17.5.2-PRESENTACIÓN EN DISPLAYS.

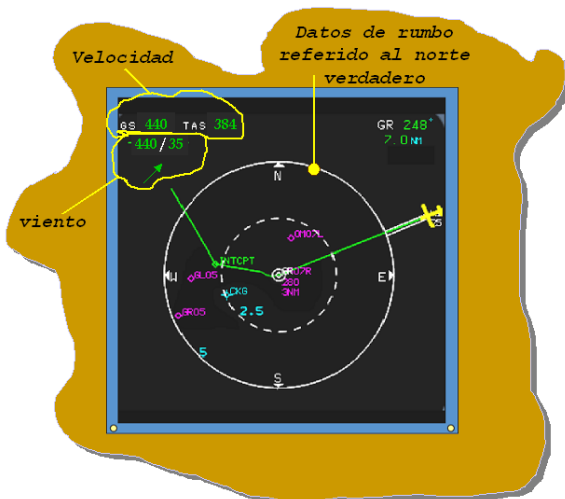
PFD.



ND.



MODO ROSA.



MODO PLANO

MODO MOTOR

ECAM.



17.6-ACARS (ARINC COMMUNICATION ADDRESSING AND REPORTING SYSTEM).

17.6.1-DESCRIPCIÓN GENERAL.

El **ACARS** es una red de comunicación aire-tierra que permite a la aeronave funcionar como un terminal móvil asociado a sistemas de control, de gestión y de órdenes de las actuales aerolíneas.

El ACARS es utilizado para transmitir o recibir informes o mensajes generados de forma automática o manual, desde o hacia una estación de tierra. Puede manejar la transmisión y la recepción de datos. Los mensajes digitales tierra-aire, o aire-tierra son transmitidos o recibidos a través del transceptor VHF3. Este transceptor es utilizado principalmente para el sistema de datos ACARS, pero puede ser utilizado como sistema de respaldo para comunicaciones de voz.

La información transmitida, es llevada a través de una red de estaciones de tierra dedicada exclusivamente a la comunidad aeronáutica, a un computador central, donde los datos son convertidos en mensajes particulares de la aerolínea. A través de esta red, los datos son transmitidos desde el receptor de tierra hasta la base principal de la aerolínea. Existen cuatro redes en el mundo: **ARINC** en EEUU, **CANADIAN** en Canadá, **JAPANESE** en Japón y **SITA** en otras regiones. Las cuatro redes están interconectadas por lo que los datos son transmitidos por cualquiera de ellas.

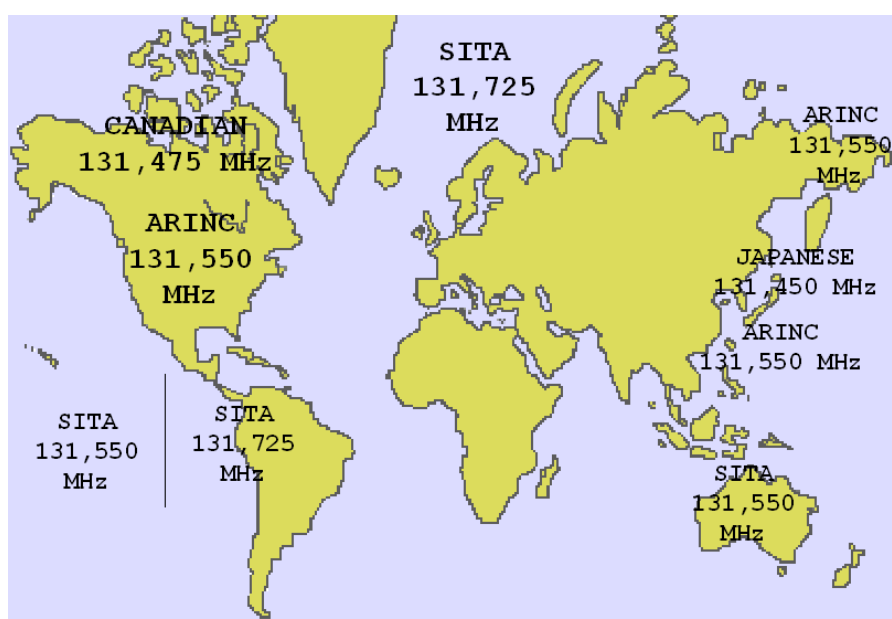
En la red, es posible comunicarse a través de dos tipos de transmisión:

CATEGORÍA A. La aeronave, puede transmitir o recibir datos a o desde todos los transceptores VHF, que están en la línea visual de la aeronave.

CATEGORÍA B. La aeronave, solamente puede transmitir o recibir a o desde un solo transceptor VHF seleccionado.

En tierra, cada proveedor del servicio trabaja en una frecuencia especial:

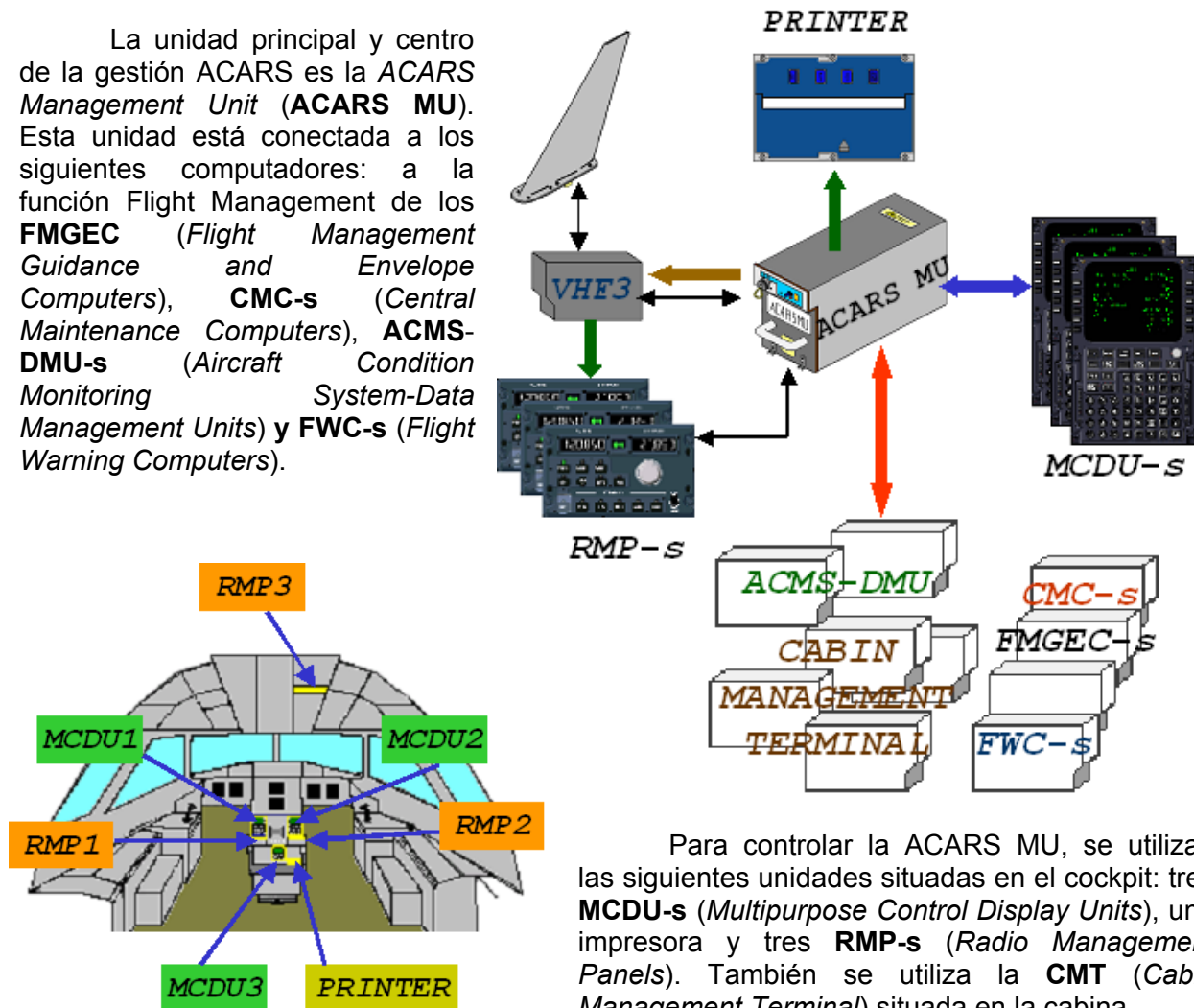
ARINC: **131,550** MHz, CANADIAN **131,475** MHz, JAPANESE **131,450** MHz, SITA **131,725** MHz.





Los componentes principales del sistema son los siguientes:

La unidad principal y centro de la gestión ACARS es la **ACARS Management Unit (ACARS MU)**. Esta unidad está conectada a los siguientes computadores: a la función Flight Management de los **FMGEC** (*Flight Management Guidance and Envelope Computers*), **CMC-s** (*Central Maintenance Computers*), **ACMS-DMU-s** (*Aircraft Condition Monitoring System-Data Management Units*) y **FWC-s** (*Flight Warning Computers*).



Para controlar la ACARS MU, se utilizan las siguientes unidades situadas en el cockpit: tres **MCDU-s** (*Multipurpose Control Display Units*), una impresora y tres **RMP-s** (*Radio Management Panels*). También se utiliza la **CMT** (*Cabin Management Terminal*) situada en la cabina.

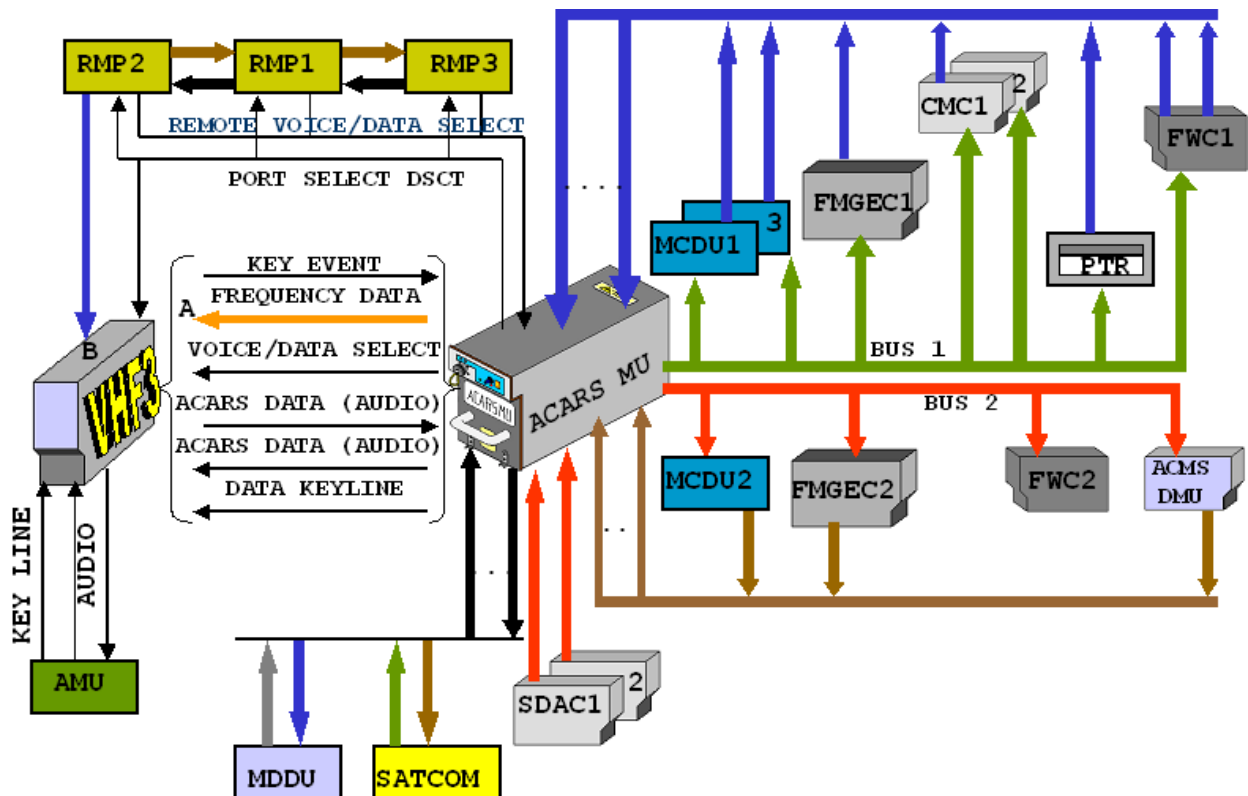
Las funciones principales del ACARS son:

En modo transmisión,

- gestión de los datos entregados por los SDAC, los FWC, la DMU, los CMC y los FMGEC.
- transmisión a la aerolínea del tiempo OOOI (Out, Off, On, In).
- actualización del reloj de la MU.
- transmisión de texto libre a tierra.

En modo recepción,

- gestión de la recepción, presentación e impresión de los mensajes de tierra.
- recepción de datos meteorológicos.

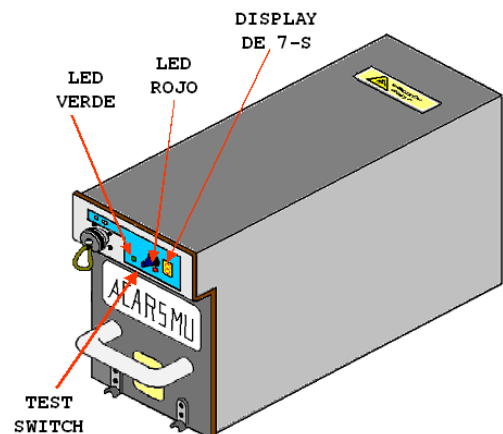
OPERACIÓN.**ACARS MU.**

La ACARS MU controla todas las operaciones relativas al ACARS. Controla la emisión y recepción de datos a través del transceptor VHF3. Para el intercambio de datos con los computadores periféricos, utiliza buses ARINC 429 en HS y LS.

La transferencia de ficheros de datos utiliza el System Address Label para identificar el sistema receptor ya que todos los computadores periféricos están conectados a los dos buses de salida generales (bus 1 y bus 2) de la ACARS MU.

Los datos procedentes de los subsistemas periféricos, son recopilados y formateados por la ACARS MU para la transmisión hacia los sistemas computadores basados en tierra a través del transceptor VHF 3. Los datos transmitidos hacia tierra, son guardados por la ACARS MU y transmitidos de nuevo a intervalos hasta que hay reconocimiento por parte de tierra o bien se alcanza el límite de retransmisión. En este último caso, la tripulación es avisada de que se ha perdido la comunicación con el enlace de tierra. Los datos recogidos para la transmisión son retenidos en la memoria del sistema para una posterior transmisión.

Los datos recibidos desde tierra, destinados a la aeronave, son distribuidos a los subsistemas periféricos o guardados en la memoria, según el software de aplicación de la aerolínea.



DMU.

La ACARS MU transmite a y recibe datos de la DMU en forma de ficheros. La DMU transmite catorce informes básicos y tres informes programables a la ACARS MU. Los informes básicos son los siguientes: engine cruise report (01), cruise performance report (02), engine take-off report (04), engine report on request (05), engine gas path advisory report (06), engine mechanical advisory report (07) engine divergence report (09), engine start report (10), engine run up report (11), APU MES/IDLE report (13), APU shut down report (14), loader report (15), ECS report (16), RAT report (20). Los tres informes programables llevan la numeración 16 para el nº 1, 17 para el nº 2 y 18 para el nº 3. Cada informe generado por la DMU puede ser programado individualmente para la transmisión al ACARS MU de forma manual o automática.

MCDU.

El dialogo entre una MCDU y la ACARS MU comienza cuando se selecciona ACARS en el menú de la MCDU. Las funciones que permite la MCDU son:

- .- mostrar en el display los datos transmitidos por la ACARS MU, la estación de tierra o por computadores periféricos.
- .- seleccionar funciones de la ACARS MU.
- .- realizar comprobaciones e introducir datos por parte de la tripulación.

Las MCDU-s 1 y 3 están conectadas a la ACARS MU a través del bus de salida general 1 y la MCDU 2 está conectada a través del bus de salida general 2. De las tres MCDU-s, la 3 es la que tiene la prioridad más alta para la ACARS MU.

FMGEC.

Los FMGEC-s informan a la aerolínea a través del ACARS, de la configuración de la aeronave. La aerolínea puede inicializar y actualizar el plan de vuelo en los FMGEC-s a través del ACARS.

El FMGEC envía informes de pre-vuelo y post-vuelo a través de la ACARS MU mediante solicitud a través de las MCDU-s. Además, envía informes a petición de tierra por la ACARS MU y de forma automática envía el informe de vuelo a esta, después del despegue.

El intercambio de datos entre los FMGEC-s y la ACARS MU se realiza a través de buses ARINC 429 LS.

CMC.

Todos los mensajes de fallos y todos los avisos grabados por el CMC, son transmitidos automáticamente a la ACARS MU tan pronto como el CMC los recibe de un sistema o del FWC.

El CMC transmite de forma manual o automática al ACARS MU el informe de post-vuelo al final del mismo.

Cualquier informe mostrado por el CMC en la MCDU, para el cual la orden "SEND" está disponible, puede ser transmitido de forma manual a la ACARS MU.

El CMC proporciona a la ACARS MU (y a todos los sistemas enlazados a él) la siguiente información:

- .- identificación de la aeronave.
- .- número y fase del vuelo.
- .- identificación de los aeropuertos de salida y llegada.
- .- fecha, hora y sistemas opcionales que se han instalado.

FWC/SDAC.

La ACARS MU recibe datos enviados por los SDAC 1, 2 y por los FWC 1, 2 vía buses ARINC 429 HS. La ACARS MU envía parámetros de estado a los FWC-s, a baja velocidad (LS).

Los FWC-s 1, 2, muestran en la Engine Warning Display (EWD) una de las cinco configuraciones del ACARS proporcionadas por la ACARS MU. Estas configuraciones son:

- *ACARS MSG*: indica que un mensaje ACARS ha sido recibido por la aeronave.
- *ACARS STBY*: indica pérdida de comunicación entre la aeronave y tierra.
- *VHF3 VOICE*: indica que VHF3 opera en modo voz (la comunicación de datos está interrumpida).
- *ACARS CALL*: indica que se ha recibido desde tierra, un mensaje demandando una conversación mediante voz.
- *ACARS ALERT*: indica que mensaje ACARS de nivel (por ejemplo ATC) ha sido recibido por la aeronave.

Cuando la ACARS MU no proporciona estos datos a los FWC-s o estos no los reciben en condiciones normales (validación o refresco incorrectos), la ACARS MU aparece en la lista de sistemas con fallo. (página de ECAM STATUS en el SD) y el aviso en color ámbar de ACARS FAULT aparece en el EWD.

Los parámetros transmitidos por los SDAC1, SDAC2, y el FWC1 permite a la ACARS MU establecer el tiempo OOOI.

MDDU.

La *Multipurpose Disk Drive Unit* permite a través de un disquete, cargar en la ACARS MU el software operacional que requiere. Esta unidad está conectada al ACARS MU por medio del conmutador selector DATA LOADER, situado en el panel del techo.

VHF 3.

El transceptor VHF 3 puede ser sintonizado automáticamente mediante la ACARS MU o manualmente usando las MCDU-s, o vía los RMP-s, dependiendo del estado del PORT SELECT DISCRETE.

El transceptor VHF 3 tiene dos interfaces para control de frecuencia; una es una entrada digital procedente de la ACARS MU (entrada A) y la otra una entrada digital procedente de los RMP-s (entrada B). El estado del PORT SELECT DISCRETE puede ser controlado manualmente a través de los RMP-s, o automáticamente a través de la ACARS MU. Cuando el discreto está a tierra, la ACARS MU sintoniza el VHF 3 a través de su entrada A, con los datos de frecuencia que le llegan por la línea FREQUENCY DATA. Si el discreto está en circuito abierto, las RMP-s sintonizan el VHF 3 a través de su entrada B.

La ACARS MU puede conmutar VHF 3 de modo datos a modo voz mediante un comando a través de las RMP-s, o automáticamente al recibir la demanda desde tierra.

También puede conmutar de voz a datos mediante comando a través de las RMP-s o una MCDU, o automáticamente mediante la conexión de la ACARS MU y el discreto KEY EVENT. La ACARS MU está conectada al discreto KEY EVENT del transceptor VHF 3, el cual indica que la entrada PTT (push-to-talk) del VHF 3 está activada. Aprovechando este discreto, la ACARS MU puede conmutar después de un tiempo de la activación del PTT (conversación) al modo datos.

La ACARS MU controla el modo de operación a través del discreto VOICE / DATA SELECT. Cuando este está a tierra, VHF 3 opera en modo datos. Si está en circuito abierto, VHF 3 opera en modo voz. En este último modo, VHF 3 activa su interfaz tipo audio con la AMU y desactiva su interfaz de audio con la ACARS MU.

La frecuencia controlada mediante las MCDU-s, se usa para establecer una comunicación telefónica entre la aeronave y tierra o para forzar a la MCDU a trabajar con otra frecuencia para datos.

El transceptor VHF 3 no es de uso exclusivo del ACARS, es usado también como sistema de respaldo de los VHF 1 y VHF 2.

RMP.

Las RMP pueden demandar a la ACARS MU una transferencia en el modo datos o voz a través del discreto REMOTE VOICE / DATA SELECT.

Cada una de las tres RMP-s reciben los discretos PORT SELECT. Cuando estos discretos están a masa, (la frecuencia del VHF 3 es controlada a través de la ACARS MU) cada una de las tres RMP-s muestran el mismo tipo de información en el modo VHF 3: ACARS en la ventana activa y una frecuencia en la ventana standby. Esta frecuencia puede ser modificada por medio del botón de selección situado debajo de la pantalla de STBY. La frecuencia puede ser diferente para cada una de los RMP-s.

Cuando los discretos están abiertos (la frecuencia del VHF 3 es controlada a través de los RMP-s), cada RMP muestra el mismo tipo de información en modo VHF 3: la misma frecuencia en la ventana activa y ACARS en la ventana STBY.

Mediante este discreto, las RMP-s son informadas de cuando la ACARS MU toma o cede el control del transceptor VHF 3.

El discreto PORT SELEC puede ser cambiado automáticamente o manualmente mediante la ACARS MU o manualmente a través de una RMP. Cada vez que se selecciona la tecla TRANSFER en una RMP, el estado del discreto cambiará momentáneamente, forzando a la ACARS MU a cambiar el estado de los discretos de PORT SELECT y VOICE SELECT. VHF 3 conmuta al modo de voz a datos o de datos a voz y los RMP-s cambian los displays entre la ventana activa y la STBY.

SATCOM.

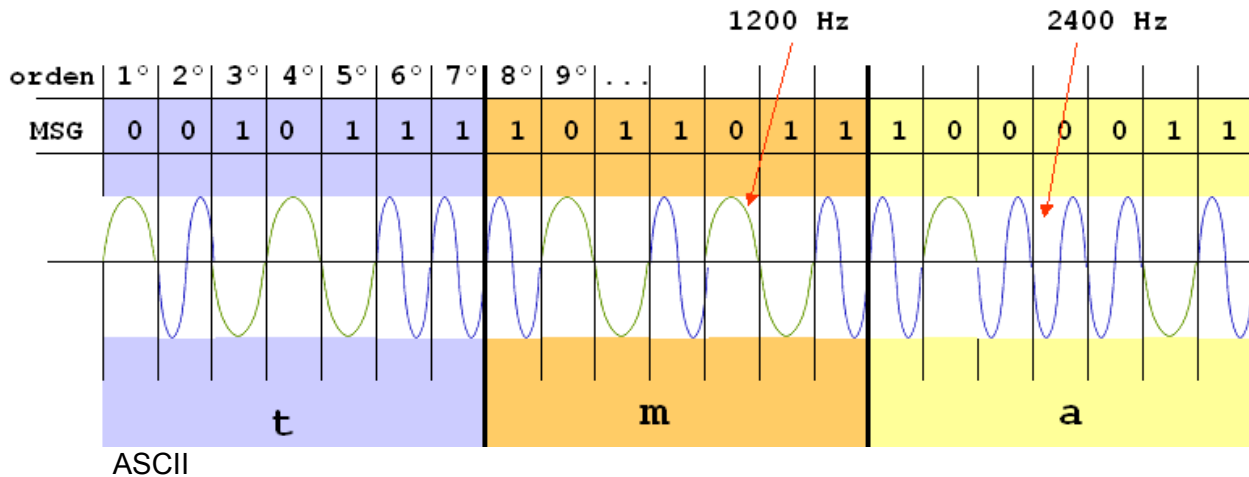
Cuando el enlace de datos con tierra VHF 3, no puede establecerse, la ACARS MU transmite y recibe datos a través del SATCOM (*Satellite Communications*). En operación normal, el transceptor VHF 3 tiene prioridad sobre el SATCOM.

IMPRESORA.

Permite imprimir los datos generados por la ACARS MU a partir de los datos transmitidos desde tierra y por los computadores periféricos. También permite imprimir datos introducidos por la tripulación a través de las MCDU-s.

CODIFICACIÓN.

Los datos intercambiados entre la ACARS MU y el transceptor VHF 3 están codificados por dos tipos de modulación en frecuencia, 1200 Hz y 2400 Hz. En la forma de onda, la parte final del periodo determina la naturaleza del BIT codificado de tal manera que una pendiente positiva codifica un 1 binario y una pendiente negativa, codifica un 0 binario.



Como se aprecia en la figura anterior, un BIT con un valor de 0 se codifica por medio periodo positivo de 1200 Hz si es el primero de una cadena y por 2400 Hz comenzando por el semiperiodo negativo cuando este BIT está precedido por uno o varios ceros. Un BIT con valor 1 se codifica por un semiperiodo negativo de 1200 Hz si es el primero de una cadena y por un periodo de 2400 Hz comenzando por el semiperiodo positivo cuando está precedido por uno o varios unos. También se puede decir que la modulación corresponde a 1200 Hz cuando existe una serie de bits diferentes y a 2400 Hz cuando existe una serie de bits iguales.

Los datos transmitidos son caracteres del código ISO ALPHABET N° 5 (ASCII), a excepción del campo de secuencia de chequeo de bloque al final de la transmisión. El orden de transmisión de los bits, es comenzando por el LSB de cada carácter.

Esta señal transmitida desde la ACARS MU al transceptor VHF 3 modula la señal de radio que se envía a tierra. Al contrario, la señal transmitida desde el VHF 3 a la ACARS MU, se obtiene después de demodular la señal procedente de tierra.

Como resultado de esta codificación, la velocidad de la transmisión es por lo tanto de 2400 bps desde la aeronave a tierra y viceversa.

El formato de los mensajes intercambiados entre tierra y la aeronave es como sigue, según el orden de transmisión:

Pre-Key (16 caracteres). Todos los bits están a 1. esta fase permite el ajuste de los niveles de transmisión y recepción VHF.

Bit synchro (2 caracteres). Permite el chequeo cruzado del valor del BIT.

Character synchro (2 caracteres). Permite establecer la sincronización de los caracteres.

Start of heading (1 carácter). Indica el comienzo de la transmisión de datos.

Mode (1 carácter). Permite la selección de una estación de tierra o un grupo de ellas para la comunicación con la aeronave.

Address (7 caracteres). Contiene la identificación de la aeronave que transmite o recibe el mensaje.

Technical acknowledgement (1 carácter). Reconocimiento positivo o negativo relacionado con el último mensaje recibido.

Label (2 caracteres). Identifica un tipo específico de mensaje.

Uplink block identifier (1 carácter). Permite a la aeronave detectar mensajes duplicados de tierra.

Downlink block identifier (1 carácter). Permite a tierra detectar mensajes duplicados de una aeronave.

Start of text (1 carácter). Indica el comienzo de la transmisión de datos útiles.

Text (220 caracteres max.). Solamente se pueden transmitir ciertos caracteres del código ISO N° 5 en este campo.

Suffix (1 carácter). Indica el final de la transmisión de datos útiles.

Block check sequence (BCS) (16 bits). Es el resultado de un cálculo realizado antes de la transmisión en el set de bits anteriores. En la recepción se realiza el mismo cálculo y se comparan los dos resultados. Si son iguales, la transmisión es correcta.

BCS suffix (1 carácter). Indica el final de BCS.

17.6.2-TEST BITE.

El estado del ACARS se determina monitorizando los intercambios de información que puede haber con la tarjeta asociada de procesamiento.

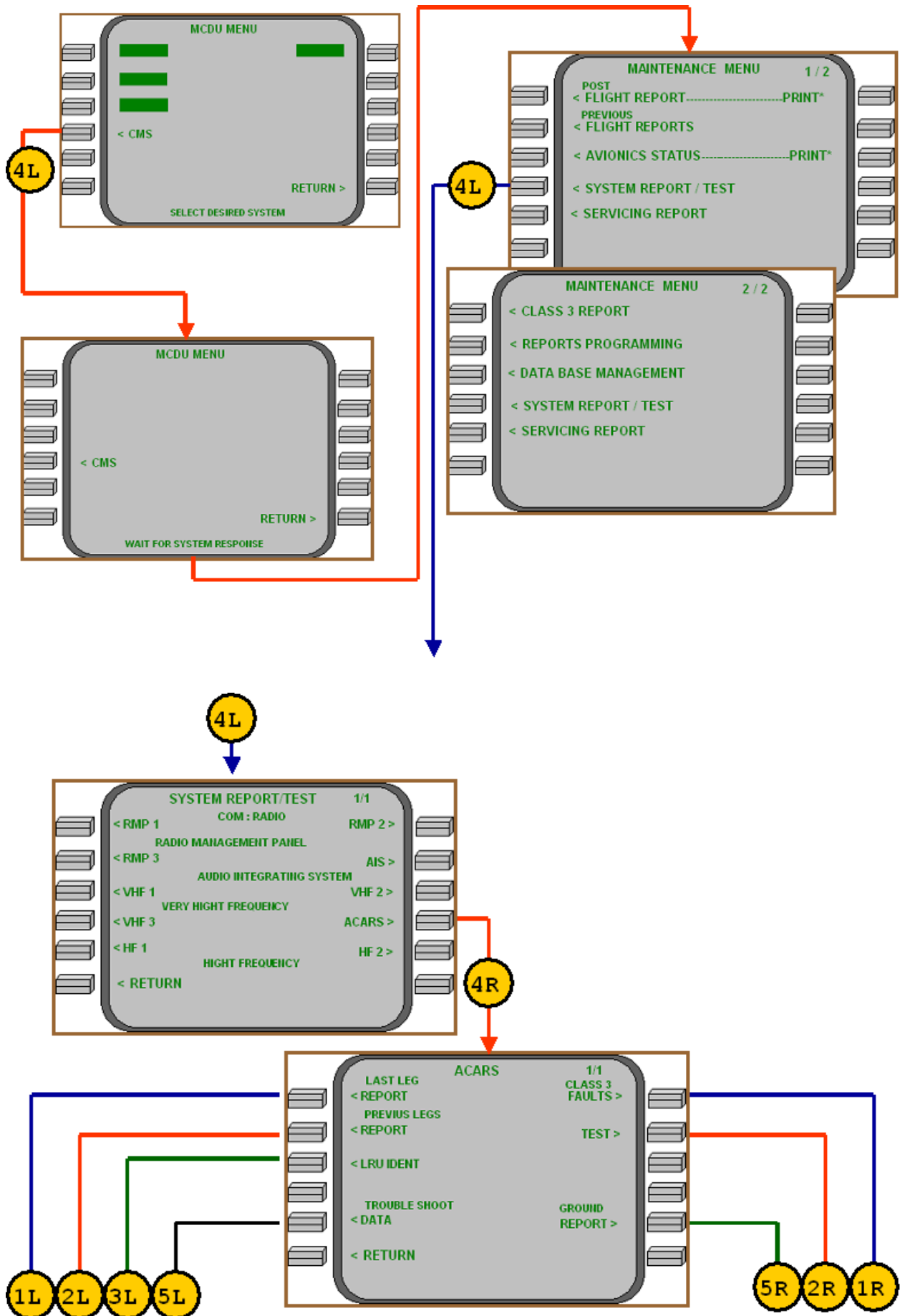
Si el protocolo de intercambio es correcto, los datos transmitidos están en el formato correcto y el ACARS no manifiesta que tenga un defecto mas allá de su test interno, entonces se dice que el ACARS está operacional. Si no es así, se declara defectuoso. El estado del ACARS se determina monitorizando los intercambios de información que cada una de las tarjetas de procesamiento pueden tener con su ACARS correspondiente y analizando los resultados de los tests internos de cada una de las tarjetas de procesamiento.

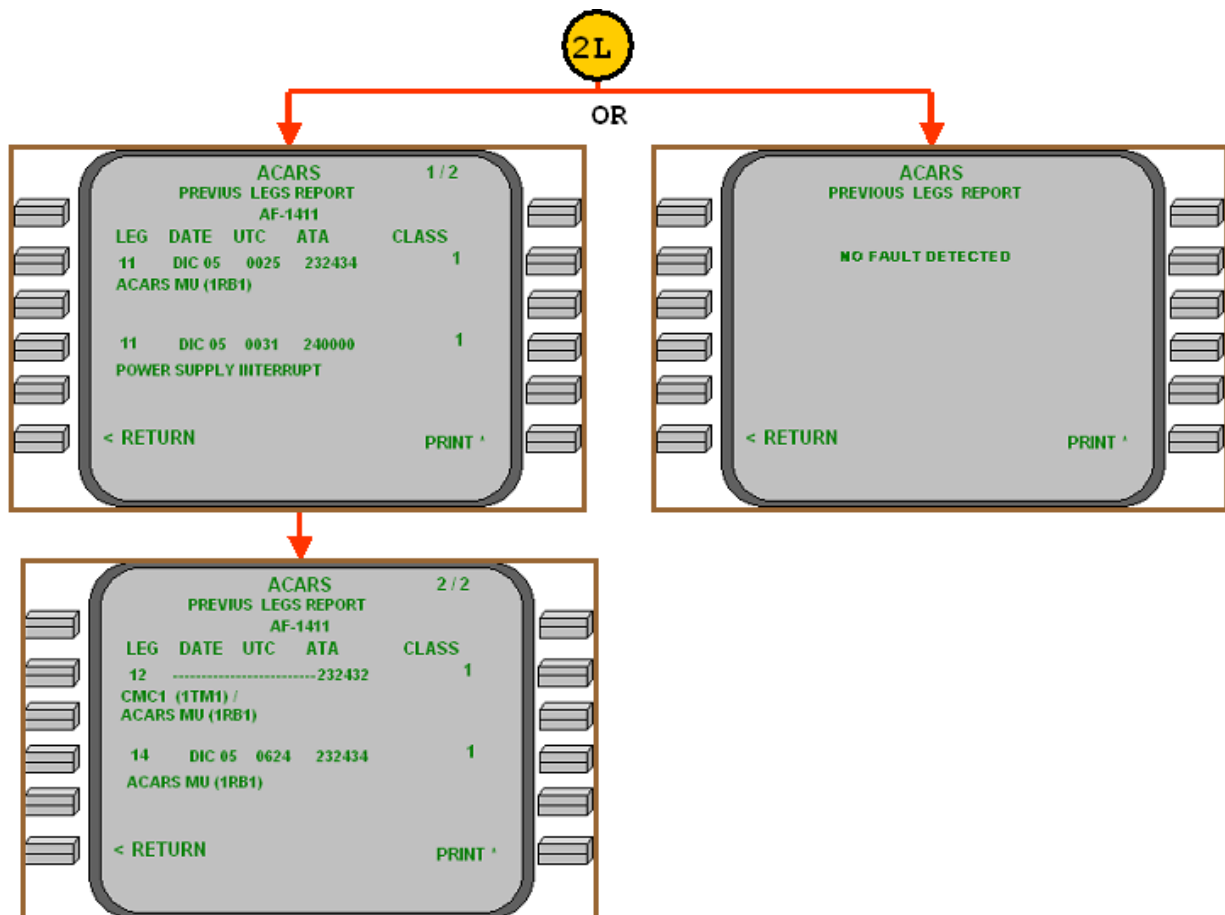
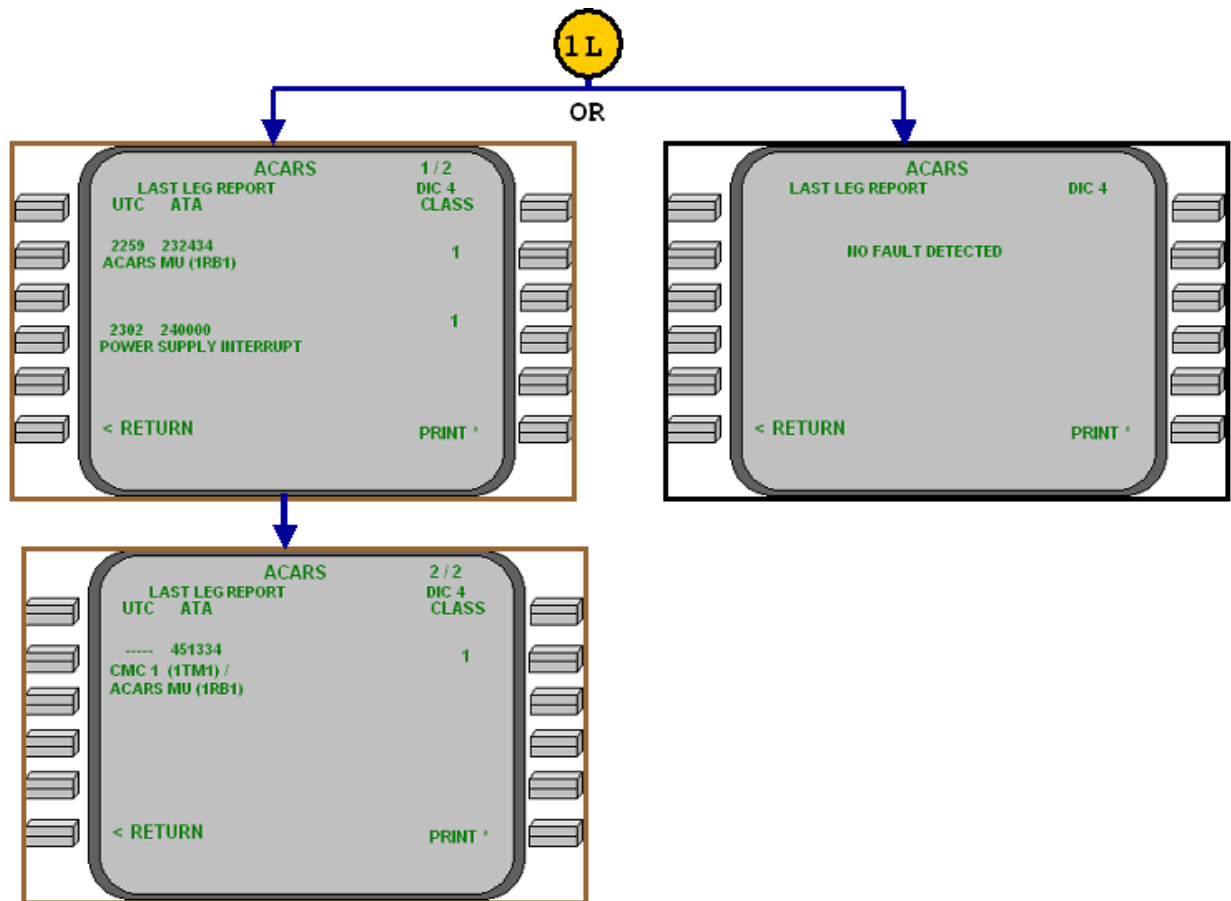
Hay dos modos de operación:

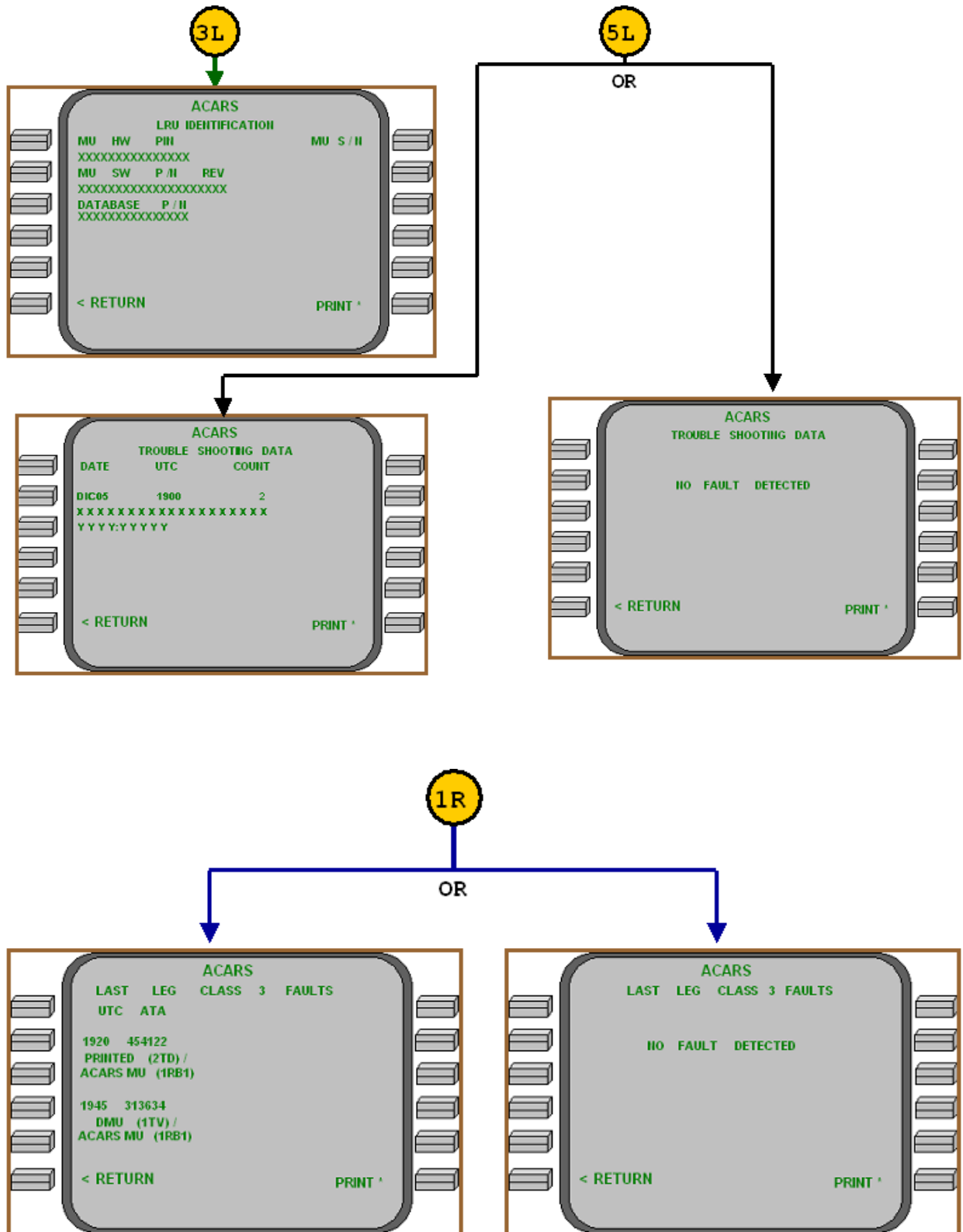
Modo normal: El BITE monitoriza cíclicamente el estado de las tarjetas del ACARS y transmite estos datos al CMC durante el vuelo correspondiente, a través de un bus ARINC 429, con la etiqueta 356. En caso de detección de fallo, el BITE guarda la información en la memoria de fallos.

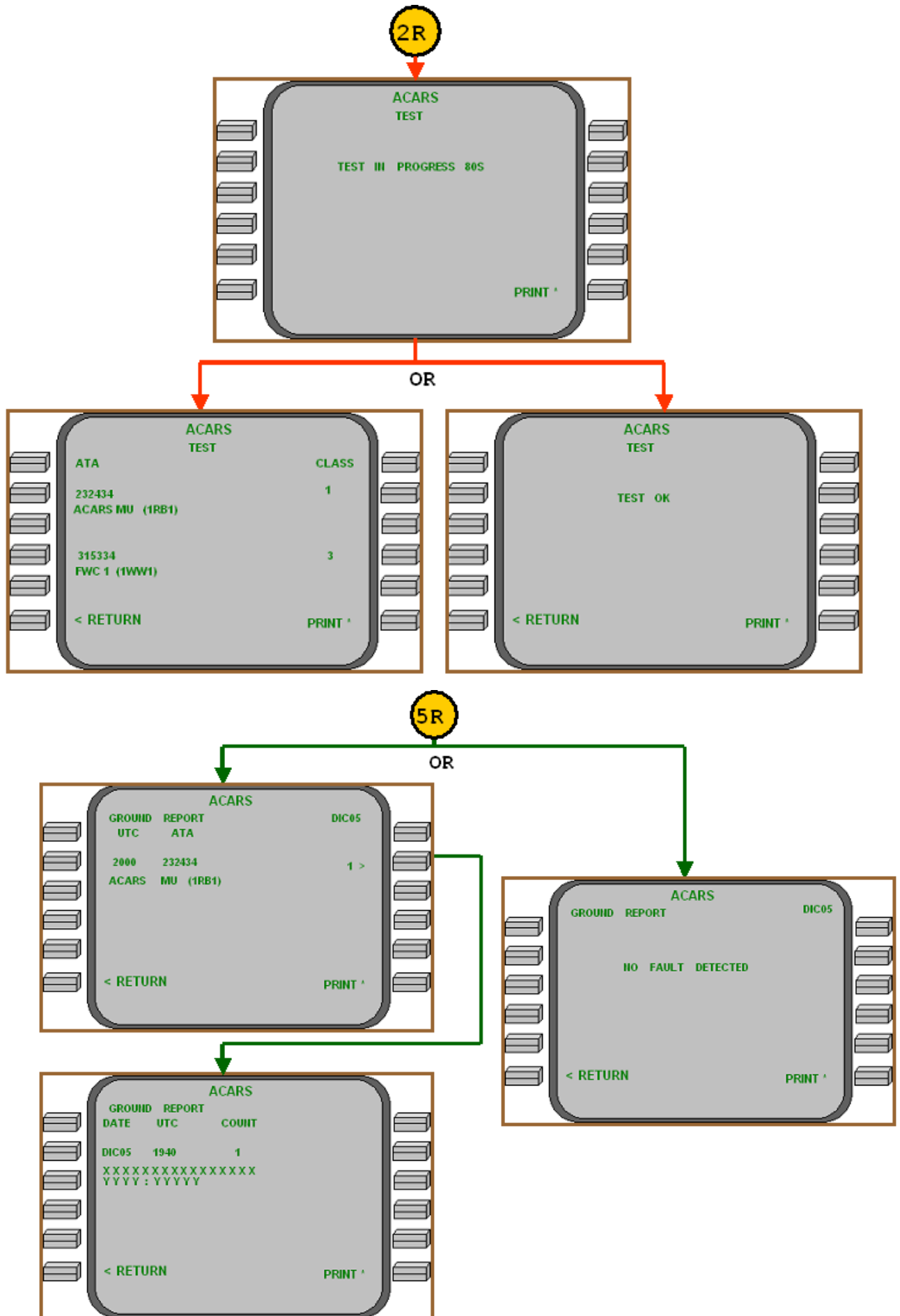
Modo menú: Este modo solo puede ser activado con la aeronave en tierra y la orden es dada por el CMC vía la MCDU. Estos datos de tierra son suministrados por la CMC, si no, pueden ser dados por el discreto de Flight-Ground proporcionado por el ACARS (etiqueta 126 BIT 28).

MODO MENÚ.









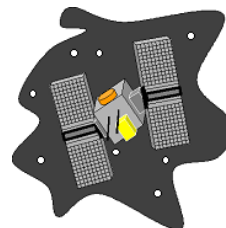
17.7-GPS (GLOBAL POSITIONING SYSTEM).

17.7.1-DESCRIPCIÓN GENERAL.

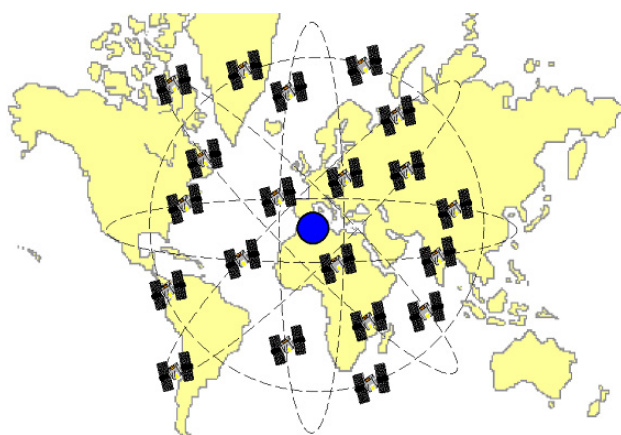
El **GPS** es una radio ayuda para la navegación a nivel mundial, que usa señales procedentes de satélites. El principio de procesamiento de la posición GPS, está basado en la medida del tiempo de transmisión de señales GPS transmitidas por 3 satélites cuya posición es conocida (los datos de la posición de los satélites, están incluidos en la señal radiada por estos). Las medidas de tiempos, son convertidas en valores de distancias. Para sincronizar los relojes de la aeronave y de los satélites, se utiliza un cuarto satélite.

Midiendo los desplazamientos Doppler de fase de las señales GPS, se puede determinar también la velocidad del receptor.

La arquitectura del sistema la forman tres bloques principales: Bloque Inercial, Bloque de Control y Bloque de Usuario.



- El bloque inercial está compuesto por una constelación de 24 satélites de 815 Kgs. de masa, 21 de los cuales están siempre disponibles. Estos satélites tienen una vida de siete años y medio, trabajan con relojes de cesio y rubidio y operan a una frecuencia de **1575,42 MHz**.



Los satélites están distribuidos en seis órbitas circulares, situadas a 20.231 Km. de altitud MSL (Mean Sea Level), con cuatro satélites por cada una de ellas, con una recurrencia orbital de 12 horas siderales (1 día sideral son 23 horas 56 min 4 s y 1/10).

- El bloque de control en tierra, está compuesto por cuatro *Monitor Stations* y una *Master Control Station*. Las cuatro estaciones encargadas del monitorizado están situadas en Kwajeleín, Hawái, Isla Ascensión y Diego García. La estación maestra está situada en Colorado Springs.



Estas estaciones, hacen el seguimiento de los satélites, procesan las efemérides, realizan las correcciones de los relojes atómicos y transmiten a intervalos regulares un mensaje de información para los usuarios GPS. Más concretamente, las Estaciones Monitor, acometen la solución a la navegación estacionaria, llevan a cabo auto pruebas y funciones de comprobación del buen estado del sistema y cargan en los satélites, palabras de verificación y el alanaque. La estación maestra, analiza los datos del enlace hacia tierra, procesa efemérides y errores en los relojes y realiza la manipulación de datos, carga de mensajes de navegación, mantenimiento y tareas de sistema.

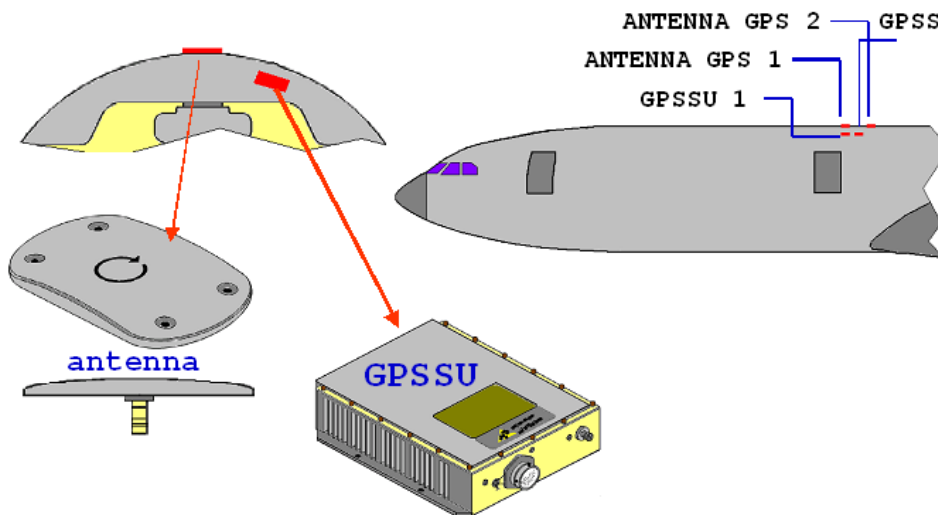
- El bloque de usuario, está compuesto básicamente por tres secciones: una sección de recepción UHF de la señal procedente de los satélites, formada por la antena, y las funciones de amplificación y filtrado. Una sección de procesamiento de la señal, donde se obtiene ya una pseudo distancia, y por último, una sección de procesamiento de navegación donde se desarrollan funciones tales como elección de satélites de la constelación, correcciones de propagación ionosfera-troposfera, procesado de datos de posicionamiento y navegación.

COMPONENTES.

El sistema GPS en la aeronave, está constituido por dos antenas y dos unidades procesadoras de la señal recibida por las antenas.

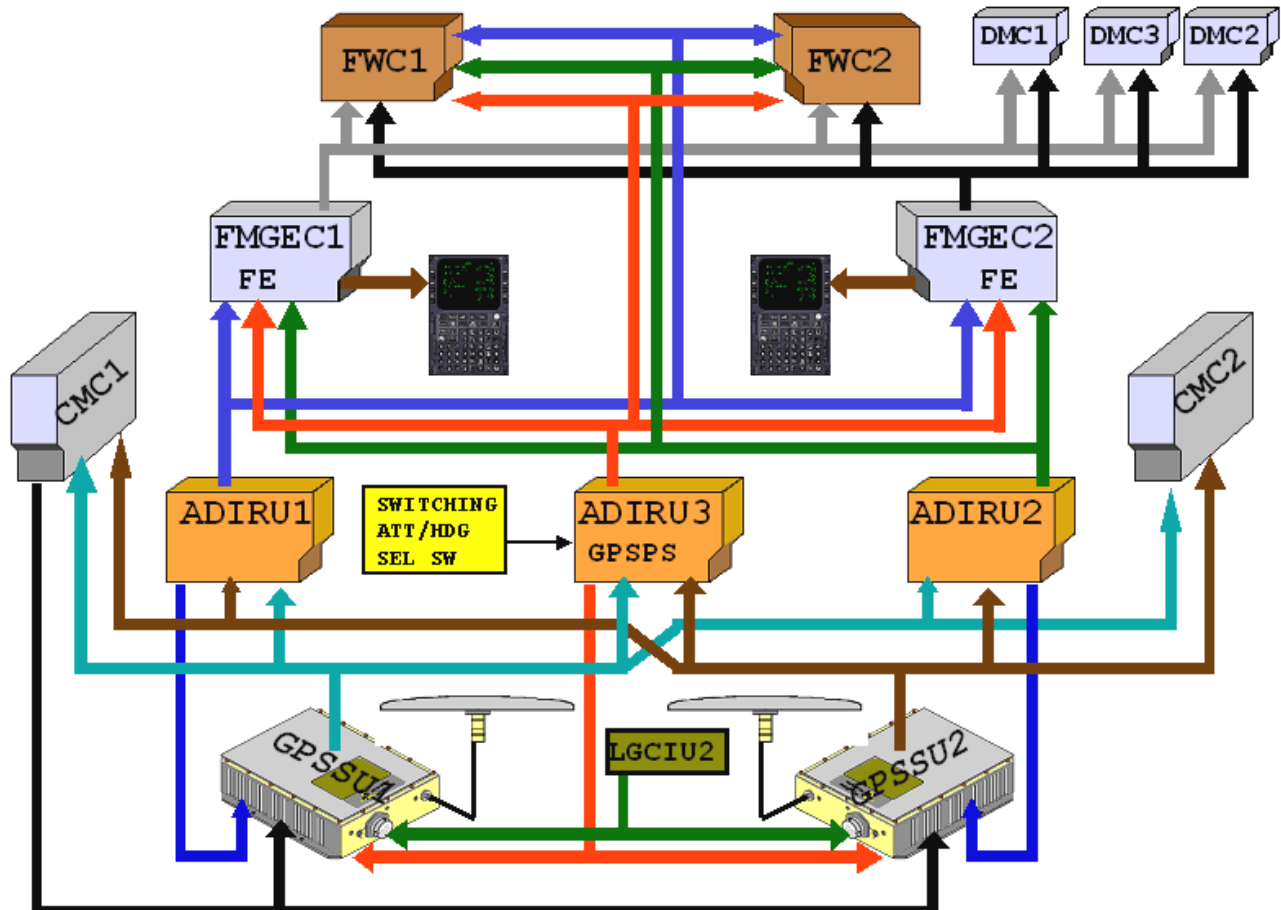
Las antenas GPS, están situadas en el exterior de la aeronave, en la parte superior del fuselaje.

Las GPS Sensor Units (GPSSU-s), están situadas en el interior de la aeronave, encima del panel de



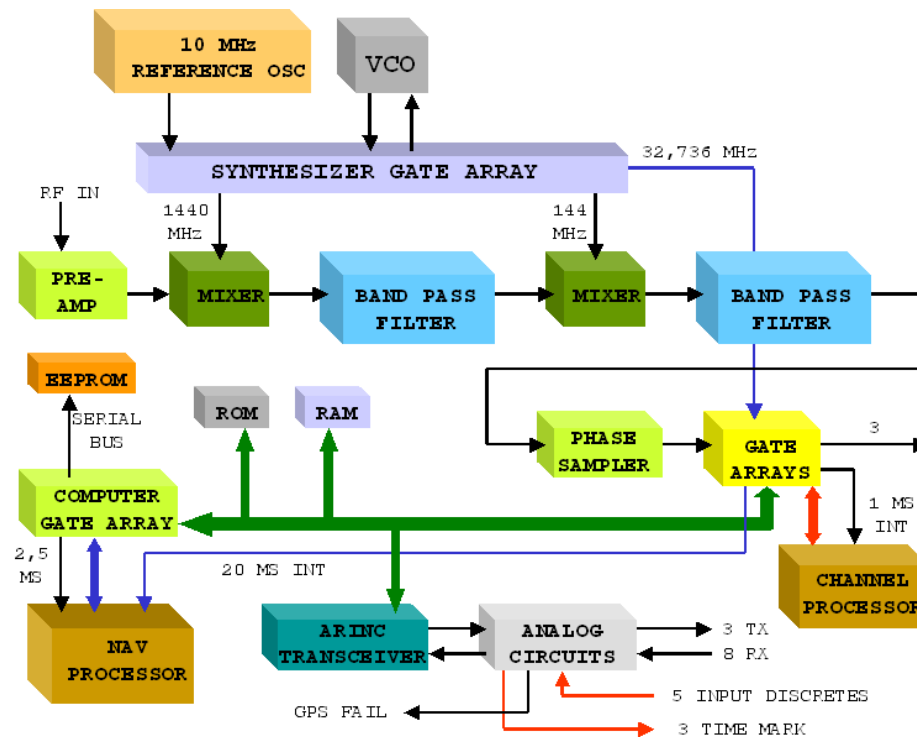
del panel de techo de cabina. En condiciones normales de operación, los datos de la GPSSU 1 son utilizados por las ADIRU-s 1 y 3, y los datos de la GPSSU 2, son utilizados por la ADIRU 2.

FUNCIONES.



GPSSU.

Las GPSSU-s son sensores autónomos de navegación por satélite que operan en tres modos diferentes: modo de Inicialización, modo de Adquisición y modo de Navegación. En el modo de Inicialización, en el momento en que estas unidades son energizadas, realizan un auto-test de sus componentes. Al finalizar el test, activan sus receptores para iniciar comunicación con la parte IR de las ADIRU-s. Cuando se transfieren desde las IR-s a las GPSSU-s datos válidos de posición inicial, altitud, fecha y hora, comienza la adquisición normal de los satélites GPS. Si no hay datos válidos disponibles, las GPSSU entran en modo de búsqueda (Search the Sky mode) para localizar cualquier satélite. Estas unidades, pueden llegar a realizar el seguimiento de hasta ocho satélites de forma simultánea.



El modo de Adquisición, es utilizado para transferir las GPSSU-s desde el modo de Inicialización o cuando existen tiempos largos sin señal de los satélites, al modo de Navegación. Cuando los satélites seleccionados han sido capturados (cuatro a ocho), las GPSSU-s, pasan al modo de Navegación. Durante este periodo, las IR pueden estar transmitiendo datos a las GPSSU-s.

En el modo de Navegación, las GPSSU-s capturan hasta ocho satélites en la línea de visión y mantienen un estado actual de cada uno para transmisión a las IR-s. Una vez procesada la señal de RF procedente de la antena y realizados los cálculos oportunos, cada GPSSU proporciona a las tres ADIRU-s datos de:

- .- UTC (Universal Coordinated Time), fecha.
- .- posición, altitud.
- .- velocidad respecto a tierra, ángulo de deriva.
- .- velocidad N/S, velocidad E/W, velocidad vertical.
- .- Dilución de precisión horizontal y vertical, precisión.
- .- posición de los satélites (efemérides)
- .- medición de los satélites (pseudo distancia, corrección de pseudo distancia, distancia, variación de distancia, tiempo de medida UTC)
- .- estado de medida del GPS en función del S/N ratio de los satélites captados.

ADIRU / FMGEC / DMC / FWC.

En operación normal, los datos de la GPSSU 1 son utilizados por las ADIRU-s 1 y 3, y los datos de la GPSSU 2 son utilizados por la ADIRU 2.

Para reducir el tiempo de inicialización de la GPSSU, las ADIRU-s 1 y 2 envían datos (posición inercial, altitud, fecha, UTC) a las GPSSU-s 1 y 2 respectivamente, tan pronto como el alineamiento de la referencia inercial está en proceso.

La parte de referencia inercial (IR) de las ADIRU-s 1 y 2, proporcionan a los FMGEC-s 1 y 2 respectivamente, datos puros de referencia inercial, datos híbridos de GPIR (Global Positioning Inertial Reference) utilizados por FMGEC para fijar la posición, y datos puros de GPS, los cuales son mostrados en las MCDU-s 1 y 2. La función de hibridación de datos dentro de las ADIRU-s realiza las siguientes funciones:

- monitorización de la GPSSU utilizando la palabra de estado del GPS y el ADIRU BITE.
- generación de mensaje para el Display ECAM.
- utilización de los datos pseudo distancia/delta distancia para calcular la posición GPS.
- utilización de datos inerciales para suavizar la posición / velocidad GPS.
- Utilización de filtros para estimar y minimizar errores
- transmisión de datos GPS y GPIR al FMGEC para fijar la posición y para propósitos de presentación en display.

En caso de que un GPS falle, las dos ADIRU-s automáticamente seleccionan el GPS operativo para procesar los datos híbridos GPIR. En este caso, en la parte inferior del EWD (Engine / Warning Display), aparece el mensaje NAVGPS 1 (o 2) FAULT. Este mensaje se acompaña con la activación de las luces MASTER CAUTION y el sonido Single Chime (SC).

En caso de fallo de la ADIRU 1, el FMGEC 1 utiliza los datos GPS 1 de la ADIRU 3. En caso de fallo de la ADIRU 2, FMGEC 2 utiliza los datos GPS 2 de la ADIRU 3. En caso de fallo de dos ADIRU-s, los dos FMGEC-s utilizan la ADIRU operativa. Esta ADIRU, recibe datos del GPS de su propio lado (1 o 2).

Para preservar la segregación entre lado 1 y lado 2 (arquitectura GPS 1/ADIRU 1/FMGEC 1 y GPS 2/ADIRU 2/FMGEC 2), siendo la fuente primaria de la ADIRU 3 el GPS 1, es necesario seleccionar el puerto de entrada secundario de la ADIRU 3 (GPS 2) por medio del selector SWITCHING ATT / HDG.

Las GPSSU-s están monitorizadas por las tres ADIRU-s utilizando la palabra de estatus enviada por cada GPSSU y el BITE de la parte de referencia inercial de las ADIRU-s.

ANTENA.

La antena GPS es una antena pasiva (sin pre-amplificación) de Banda L, diseñada para operar a **1575, 42** MHz con una polarización circular dextrógira para proporcionar una cobertura hemisférica superior omnidireccional.

CMC.

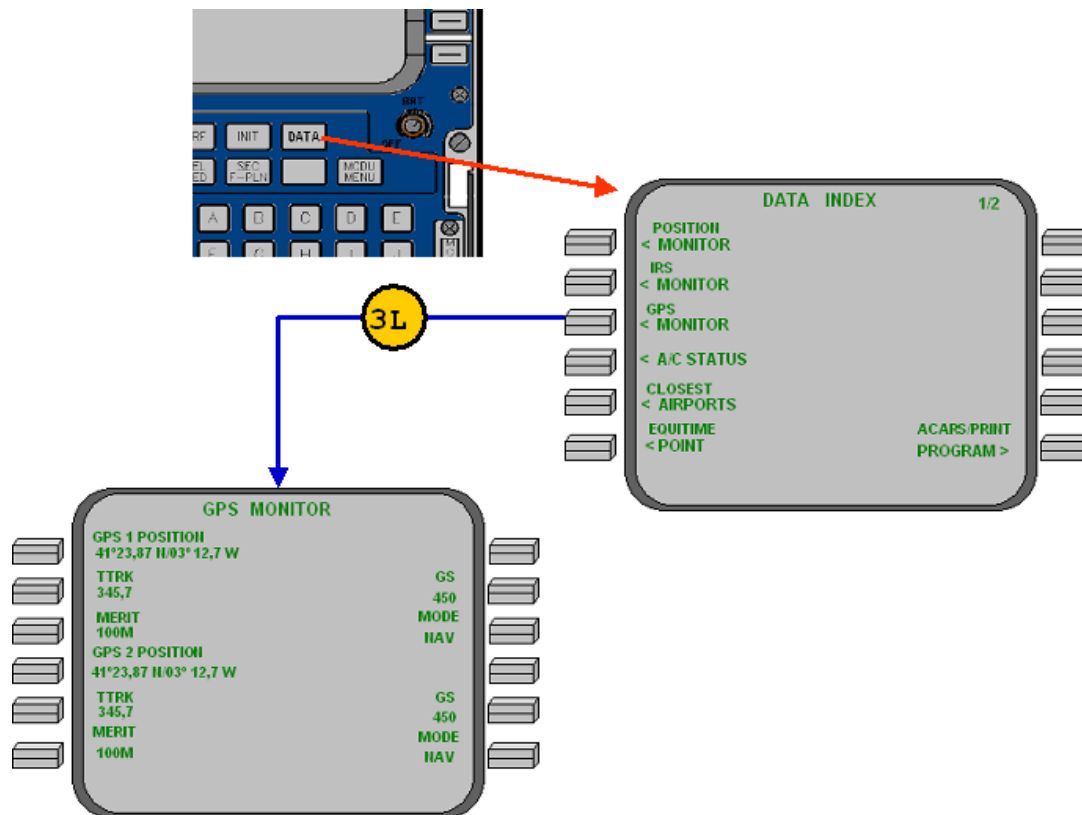
Los circuitos BITE de las GPSSU están conectados a los *Central Maintenance Computers* (CMC-s). Los elementos comprobados son los GPSSU-s, las antenas GPS y los cables coaxiales que unen los anteriores elementos.

LGCIU 2.

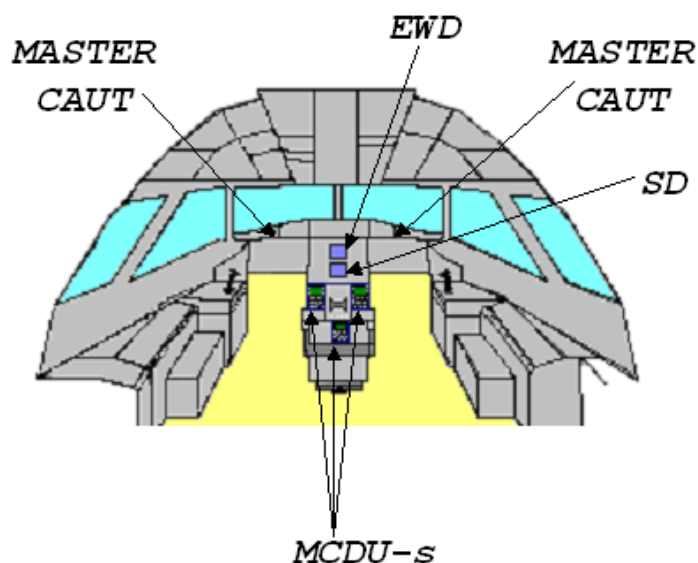
La información de tierra/vuelo proporcionada por la *Landing Gear Control Interface Unit* (LGCIU), es utilizada por los módulos BITE de las GPSSU para contar los tramos de vuelo.

17.7.2-INDICACIÓN Y CONTROL.

Los datos GPS se muestran en la página GPS MONITOR, de la MCDU. Las indicaciones mostradas son las siguientes: posición GPS (lat / long), True Track (TTRK), Precisión Tridimensional (FOM) expresado en metros, Ground Speed (GS) y Mode.



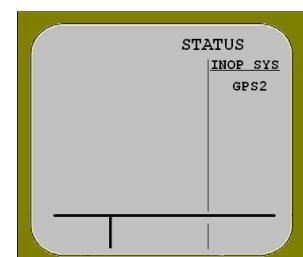
Las GPSSU-s están monitorizadas por las tres ADIRU-s, utilizando la palabra de estado enviada por cada GPSSU y el BITE de la parte IR. En caso de fallo del GPS, el mensaje NAV GPS 1 (2) FAULT es mostrado en la parte inferior del la EWD (Engine / Warning Display). Este mensaje va acompañado de la activación de las luces MASTER CAUT y de un aviso auditivo de campanilla simple (single chime). El fallo es recordado en la página INOP SYSTEM del SD (System Display)



EWD



SD



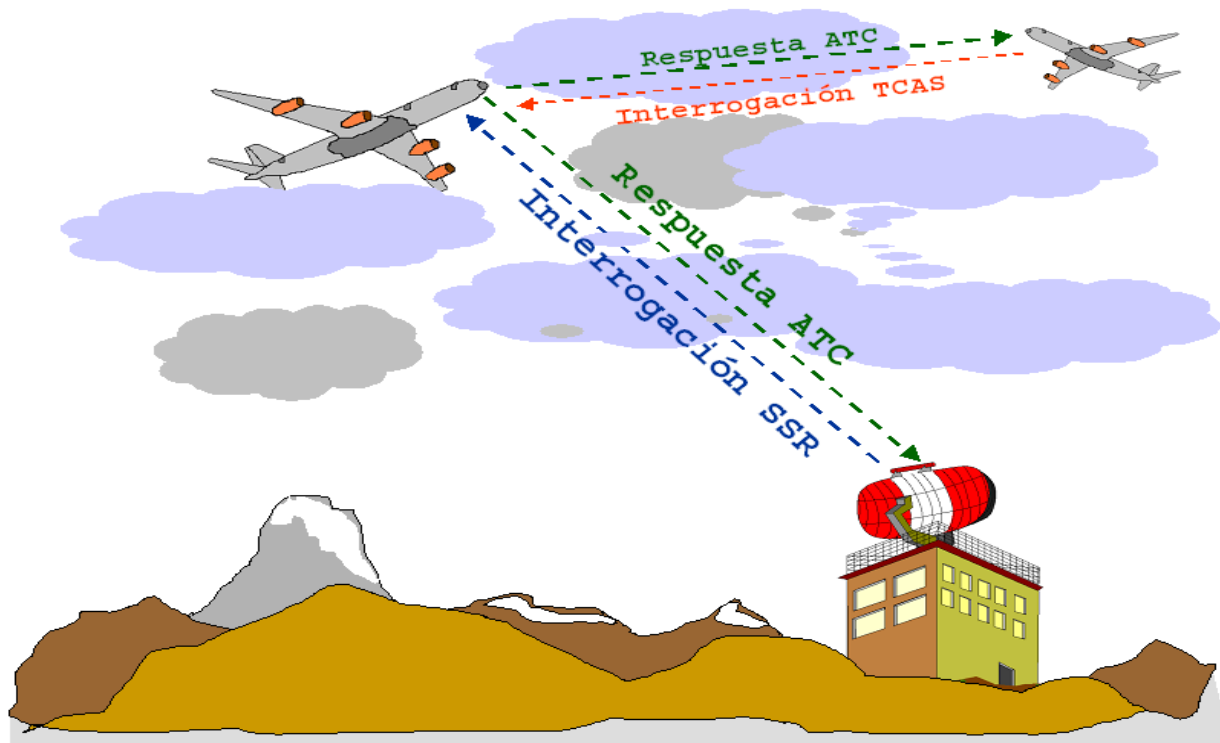
17.8-ATC / TCAS (AIR TRAFFIC CONTROL / TRAFFIC ALERT AND COLLISION AVOIDANCE SYSTEM).

17.8.1-ATC. DESCRIPCIÓN GENERAL.

El *Air Traffic Control System (ATC)*, implementa la función circulación aérea y sirve para localizar las aeronaves y mantener una vigilancia efectiva del tráfico aéreo, desde tierra. Este sistema está basado en las respuestas proporcionadas por los transpondedores de bordo mediante una serie de impulsos eléctricos, en respuesta a las interrogaciones (también mediante una serie de impulsos eléctricos) realizadas desde tierra por el *ATC Secondary Surveillance Radar (SSR)*.

El Radar ATC consta de dos partes: el *Primary Surveillance Radar (PSR)* y el SSR. El PSR es un sistema pasivo de respuesta, destinado a la localización de blancos, proporcionando parámetros de distancia y azimut. El SSR, es un sistema activo de respuesta, esto quiere decir que el blanco interviene añadiendo información a la señal "eco". La filosofía es la de equipo interrogador (radar) en tierra y equipo respondedor (Transponder) a bordo de la aeronave.

La información que los controladores aéreos obtienen del sistema, una vez decodificados los impulsos, es función del modo operativo de los transpondedores. En el **Modo A**, la información que se obtiene acerca de la aeronave es un *código* de identificación tabulado. En el **Modo C** la información que se obtiene acerca de la aeronave es la *altitud barométrica*. Y en el **Modo S**, el sistema realiza interrogaciones *Selectivas* desde el equipo de tierra para transmisión de datos de vuelo específicos.



En tierra, el radar Modo S opera tres tipos de interrogaciones: (1) llamada general en Modos A o C, (2) llamada general en Modos A, C y S (*All Call interrogations*), (3) llamada selectiva en modo S. En el primer caso (1), responden solamente los transpondedores convencionales. Para el caso 2, las aeronaves con transponder activado en modo A o C, se

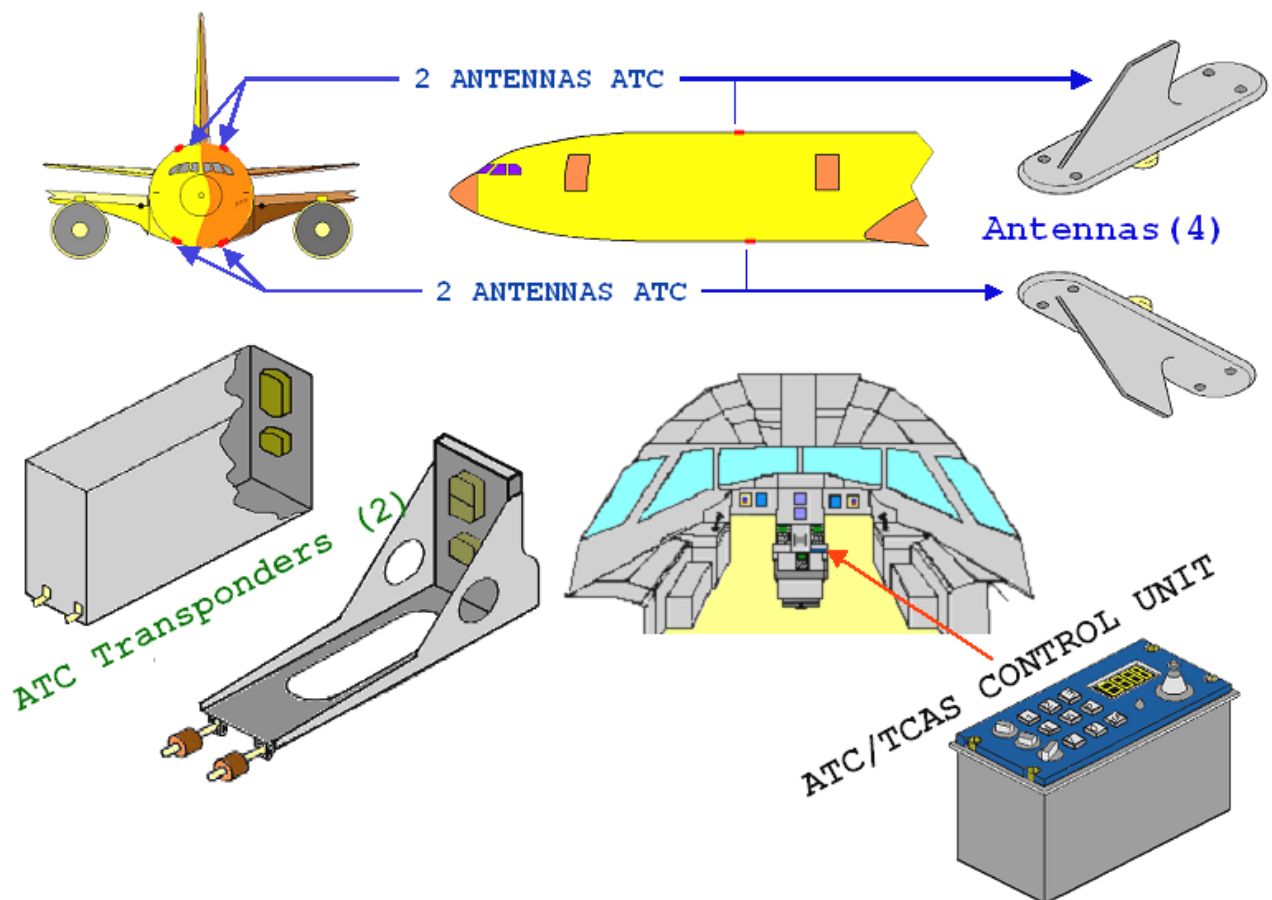
interpreta como una interrogación convencional, y la respuesta se da como tal. Para aeronaves con transponder activado en modo S, se interpreta como que el ATC de tierra está intentando conocer el código de dirección de las aeronaves que se encuentran en su zona de cobertura. En el tercer caso, responden solamente las aeronaves específicamente preguntadas por el ATC de tierra, a partir de la dirección conocida de aquellas (IP-24).

El transponder ATC Modo S, recibe interrogación de estaciones de tierra SSR y Modo S y de aeronaves equipadas con el sistema **TCAS** (el transponder Modo S transmite datos TCAS de otras aeronaves equipadas con TCAS, a su propio computador TCAS). Cada aeronave equipada con ATC Modo S tiene su propia dirección de Modo S. Esta dirección (definida por 24 bits), está incluida en todas las transmisiones Modo S, por lo tanto cada interrogación puede ser dirigida a una aeronave particular, y de esta manera evitar respuestas múltiples. Además, las respuestas vienen inequívocamente de una aeronave específica.

Hay que indicar que, el modo S ha sido diseñado como una adicción evolucionada al sistema ATC convencional (Modos A y C), para proporcionar la vigilancia y comunicación mejoradas que requiere la automatización del control del tráfico aéreo.

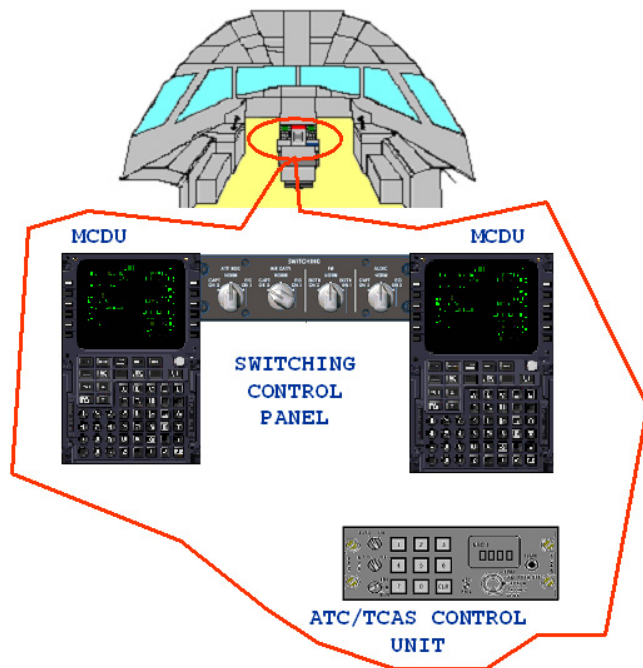
COMPONENTES ATC.

El ATC está compuesto por dos sistemas independientes, consistente cada uno de ellos en un transponder situado en el compartimiento de aviónica, dos antenas, una en la parte inferior del fuselaje y otra en la superior y una unidad de control ATC/TCAS, situada en el cockpit.

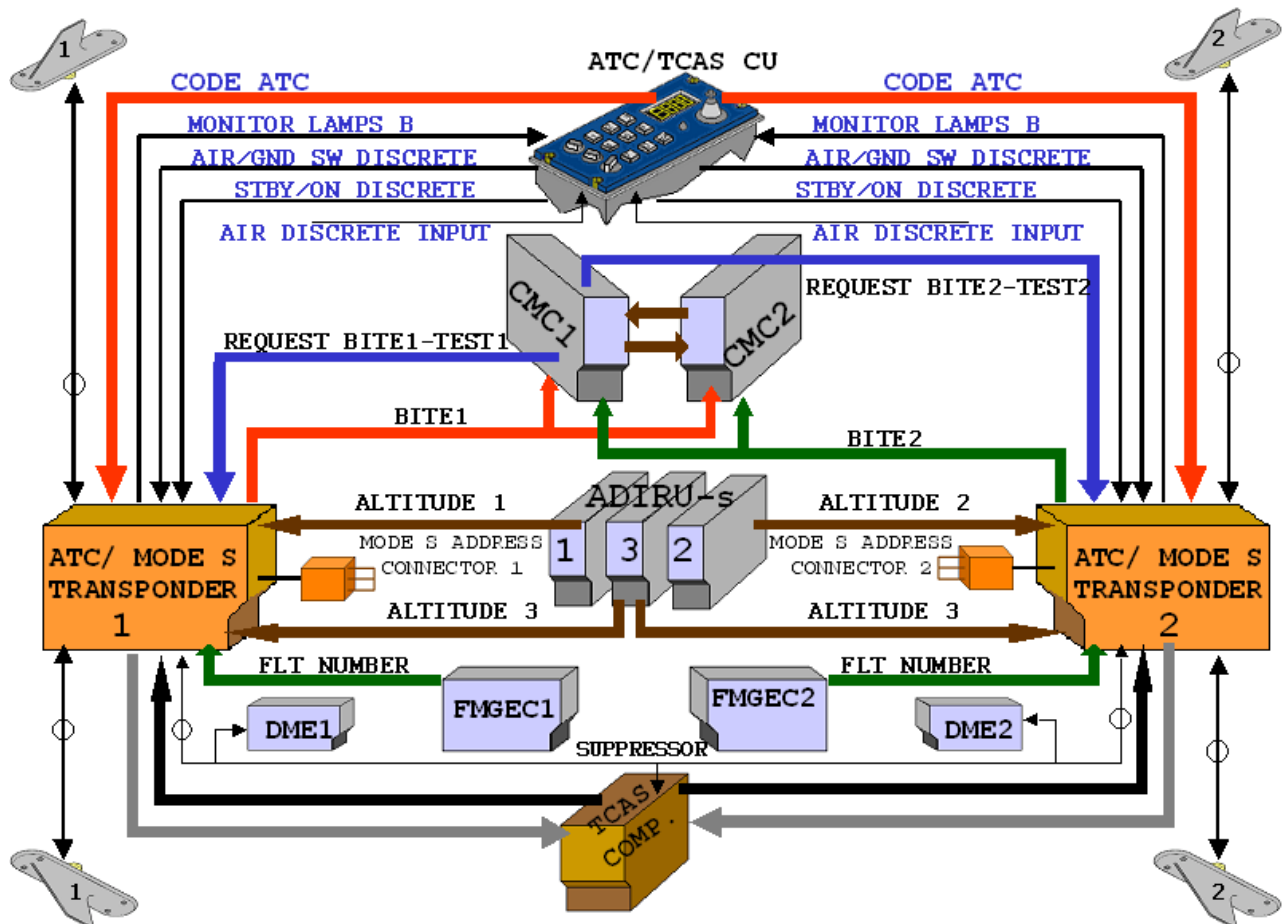


17.8.2-INDICACIÓN Y CONTROL ATC.

Los elementos que sirven de control e indicación del sistema ATC son las MCDU-s delanteras, una parte del Switching Control Panel y la unidad de control ATC/TCAS. Todos estos elementos están situados en el pedestal del cockpit.



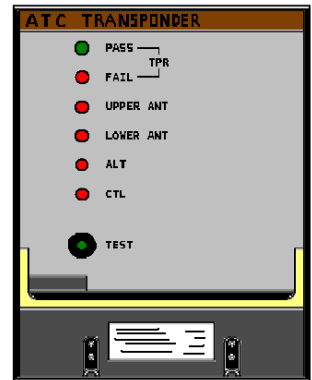
ARQUITECTURA ATC.



ATC TRANSPONDER.

En configuración normal, cada transponder recibe la información de la altitud, de su correspondiente ADIRU 1 o 2. En caso de que exista un fallo en estas ADIRU-s, el piloto puede seleccionar la información de altitud del ADIRU 3. Esta selección se realiza a través del AIR DATA selector switch en el Switching Control Panel situado en el pedestal central.

Externamente en su frontal aparecen un pulsador de TEST y seis diodos LED. Cuando se oprime y se mantiene el pulsador de test, se inicia una prueba de los diodos. Después de esta prueba la unidad lee los datos de fallo de la memoria no volátil correspondientes a los últimos cuatro vuelos, activando los LED-s pertinentes según el estado operacional de la unidad:

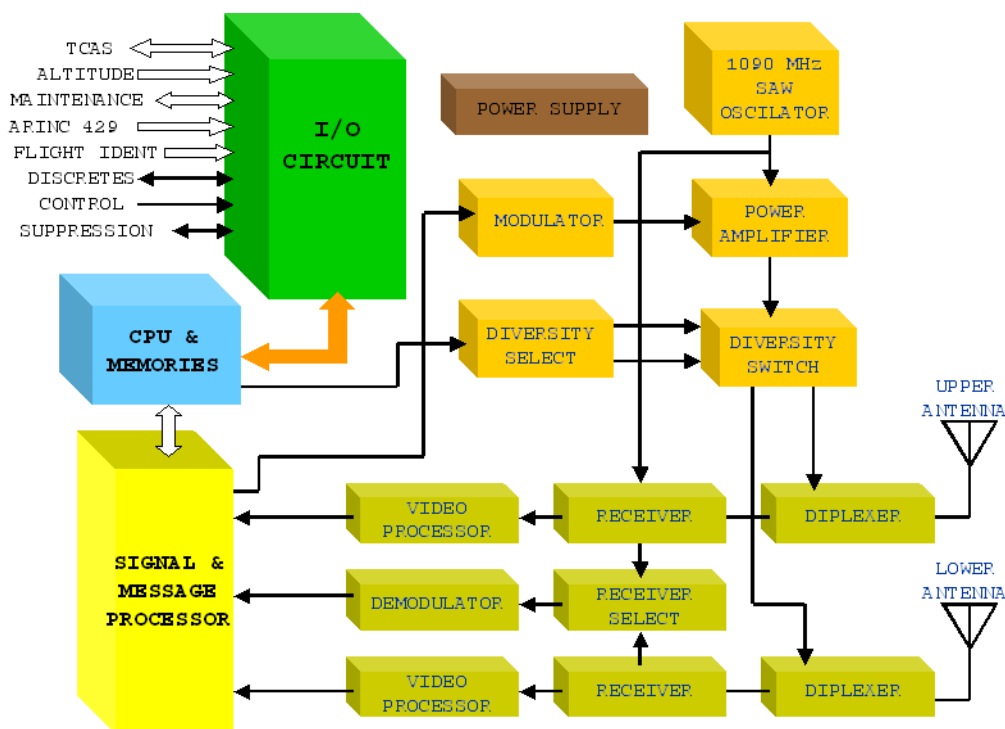


En la parte posterior están situados los conectores (ARINC 600) ordenados en tres grupos: (1) interfaz ATE, (2) interconexiones con el

LED-S	FUNCIÓN
TPR PASS (verde)	Encendido después de que el test indique operación normal
TPR FAIL (rojo)	Encendido después de que el test indique disfunción.
UPPER ANT (rojo)	Encendido después de que el test indique operación o conexión anormal en las antenas superiores.
LOWER ANT (rojo)	Encendido después de que el test indique operación o conexión anormal en las antenas inferiores.
ALT (rojo)	Encendido después de que el test indique formato anormal de los datos de altitud.
CTL (rojo)	Encendido después de que el test indique palabra de control anormal (disfunción en la unidad de control)

sistema y (3) alimentación y masa.

Las señales de interrogación de 1030 MHz procedentes de tierra, son recibidas por una o ambas antenas. La señal recibida por cada antena es demodulada y enviada a los circuitos de procesamiento donde se realiza la selección de diversidad. Los circuitos DIVERSITY SELECT, determinan qué receptor proporciona la señal más potente (o primera). La señal recibida seleccionada, es entonces aplicada a través de un circuito demodulador a los circuitos de procesamiento.



Los circuitos de procesamiento determinan si la interrogación es una interrogación ATCRBS, una interrogación Modo S destinada a la aeronave en particular, o una interrogación Modo S destinada a otra aeronave. Dependiendo del tipo de interrogación, los circuitos de procesamiento determinan la respuesta apropiada (si se requiere alguna), la envían a través de un circuito procesador de mensajes a los circuitos DIVERSITY SWITCHING. Estos circuitos seleccionan la antena a través de la cual se transmitirá la respuesta (esta es transmitida a través de la misma antena desde la cual se origina la señal de recepción seleccionada).

El circuito de I/O, es un circuito interfaz para recibir datos ARINC 429, la entrada de la dirección de 24 bits, el código de altitud y varios discretos. La CPU proporciona el enlace entre los circuitos I/O y los circuitos procesadores de señal.

ATC / TCAS CONTROL UNIT.

A través del panel frontal se puede realizar las siguientes funciones:

- seleccionar mediante teclado el código octal de cuatro dígitos (desde 0000 a 7777) a transmitir por el transponder en modo A.

- seleccionar mediante el switch **SYS 1/2**, que sistema ATC (el transponder 1 o el transponder 2) se hace operativo.

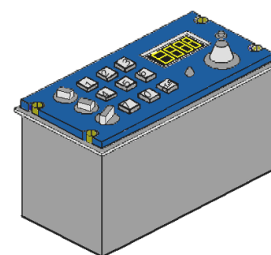
- mediante el switch selector **AUTO/ON**, en la posición ON, el ATC transponder seleccionado opera, y en la posición AUTO, el ATC transponder seleccionado opera en vuelo, y en tierra opera solamente en Modo S.

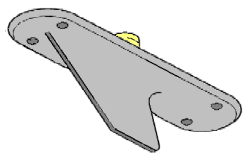
- mediante el switch selector **ABV/N/BLW** (Above/Normal/Below) se puede seleccionar la zona de vigilancia TCAS ampliada por encima (se establece un rango de altitud de 9900 ft por encima y 2700 ft por debajo) o ampliada por debajo (se establece un rango de altitud de 9900 ft por debajo y 2700 ft por encima) de la aeronave equipada con TCAS, o normal, en el cual todos los intrusos son mostrados sin condición, cuando son detectados.

- mediante un selector **ATC/TCAS** (de cinco posiciones), se puede llevar al sistema a las siguientes formas de funcionamiento:

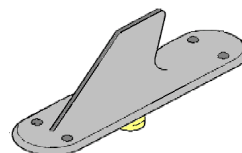
- * posición **STBY**: todas las funciones del ATC y TCAS están inhabilitadas.
- * posición **ALT RPTG OFF**: el ATC opera en Modo A (código) y el TCAS permanece inhabilitado.
- * posición **XPNDR**: ATC responde en Modo A o C y el TCAS está inhabilitado.
- * posición **TA ONLY**: el TCAS está operativo, pero solamente realiza Traffic Advisories (TA). Los intrusos TA son mostrados en los ND-s.
- * posición **TA/RA**: el TCAS realiza Traffic Advisories y Resolution Advisories. Los intrusos TA y RA son mostrados en los ND-s. Resoluciones preventivas o correctivas de RA son mostradas en los PFD-s y se oyen mensajes asociados por los altavoces del cockpit.

Además, mediante un Display LCD se puede visualizar el código ATC seleccionado (7700 o 7777 para emergencia, 7600 para fallo total del sistema de radiocomunicación, ...); mediante un indicador **ATC FAIL** se indica a la tripulación de vuelo sobre un fallo de transponder, y mediante un pulsador **IDENT** se puede generar un impulso adicional de identificación en la señal de respuesta.



ANTENA.

La antena de aluminio, es de tipo cuchilla, con una cavidad radiante gobernada por acoplamiento capacitivo. Opera dentro de la banda L desde 960 MHz hasta 1250 MHz. Utiliza polarización vertical, tiene una impedancia de $50\ \Omega$, y una VSWR de 1,5:1. La conexión al sistema se realiza mediante un conector coaxial tipo C.



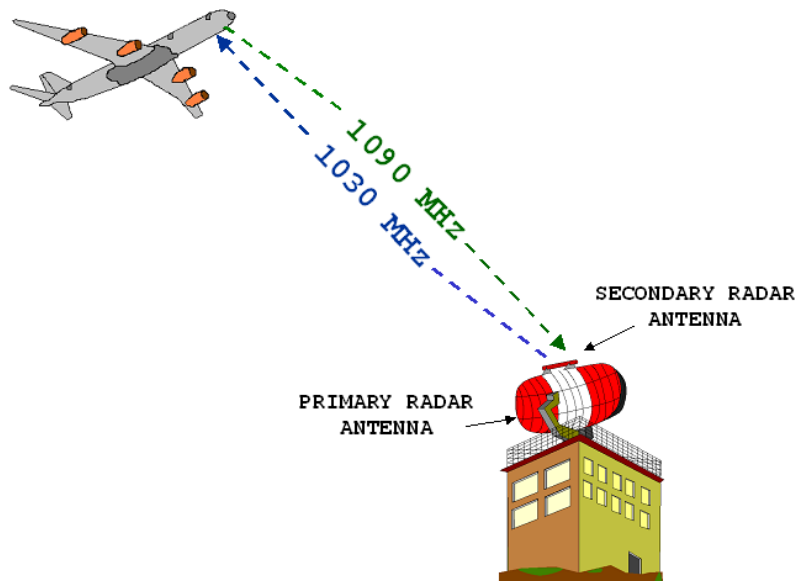
17.8.2-OPERACIÓN ATC.

Un transponder de a bordo proporciona señales codificadas en respuesta a las señales de interrogación procedentes del radar secundario de tierra y de aeronaves que estén equipadas con TCAS. El sistema ATC/Modo S, tiene la capacidad de trabajo en el Modo A (identificación), modo C (altitud barométrica) y modo S. En este último Modo, el transponder responde a las interrogaciones en formato **ATCRBS** (*Air Traffic Control Radar Beacon System*) con respuestas también en formato ATCRBS, pero, además, el transponder puede ser discretamente direccionado directamente por interrogadores Modo S, esto significa que solamente el transponder Modo S específicamente direccionado, será el que responda.

El transponder Modo S, se utiliza también para evitar colisiones. En esta función, el transponder funciona como un ATCRBS y transponder Modo S para necesidades ATC, mientras que simultáneamente sirve como parte integral del TCAS. Las funciones básicas que realiza, relacionadas directamente con el TCAS son las dos siguientes:

- 1. Aproximadamente cada segundo, el transponder Modo S realiza una transmisión. Estas transmisiones se alternan entre las antenas Modo S ventral y dorsal. La transmisión, está codificada con la dirección Modo S de la aeronave y las capacidades del equipamiento utilizado. Una estación de tierra y una aeronave equipada con TCAS escuchan la transmisión como indicación inicial de que una aeronave equipada con Modo S ha entrado en el área de vigilancia o cobertura TCAS.

- 2. La comunicación entre dos aeronaves equipadas con TCAS ocurre desde la antena TCAS de una aeronave, a la antena del transponder Modo S de la otra aeronave. A bordo de esta última aeronave los datos recibidos por el transponder Modo S pasan al computador TCAS y éste determina el contenido de los mensajes de respuesta.



El sistema ATC, opera con dos frecuencias: cada interrogador de tierra transmite sus interrogaciones a la frecuencia portadora de **1030 MHz**, las respuestas desde los transpondedores se realizan a una frecuencia portadora de **1090 MHz**.

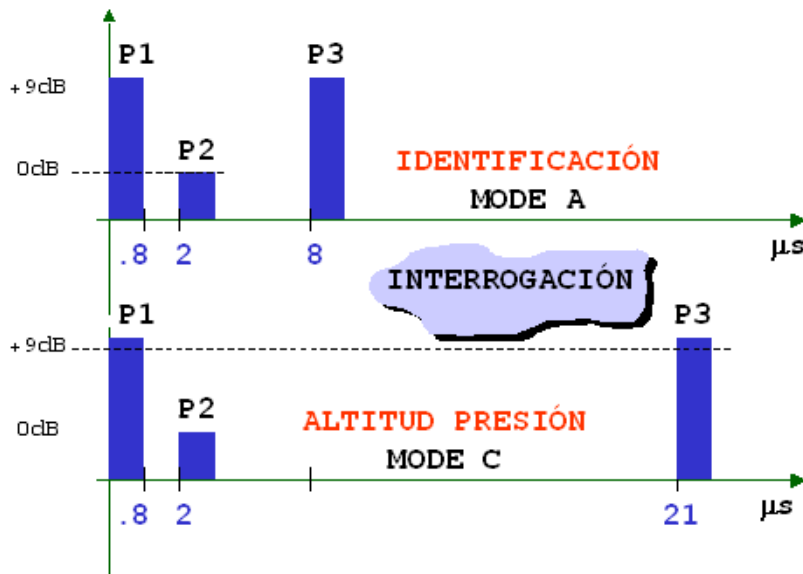
Las interrogaciones básicas, se realizan en la forma de series de dos impulsos. Según el número e intervalo entre impulsos, se definen tres modos de

interrogación diferentes. El transponder, después de recibir los impulsos, identifica y decodifica las interrogaciones. Dependiendo del modo de interrogación detectado, el transponder transmite la identificación de la aeronave, o su altitud barométrica o datos de vuelo. Para realizar estas funciones, el transponder está asociado con una unidad de control que define el código de identificación de la aeronave y a las ADIRU-s.

INTERROGACIONES.

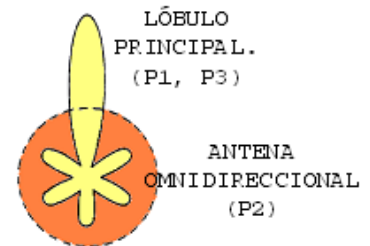
La estación de tierra transmite tres clases de interrogación. (1) Modo A o C, (2) Modo S general (All Call), y (3) Modo S. El transponder, responde de acuerdo a estas interrogaciones:

-* Una interrogación Modo A o C consiste en dos impulsos llamados **P1** y **P3**. Un impulso de control de transmisión **P2**, se transmite después del primer impulso P1. P1 y P3 son radiados en el haz principal por una antena de haz estrecho rotatorio. P2 llamado también impulso *Side Lobe Suppression (SLS)*, es radiado por una antena omnidireccional,



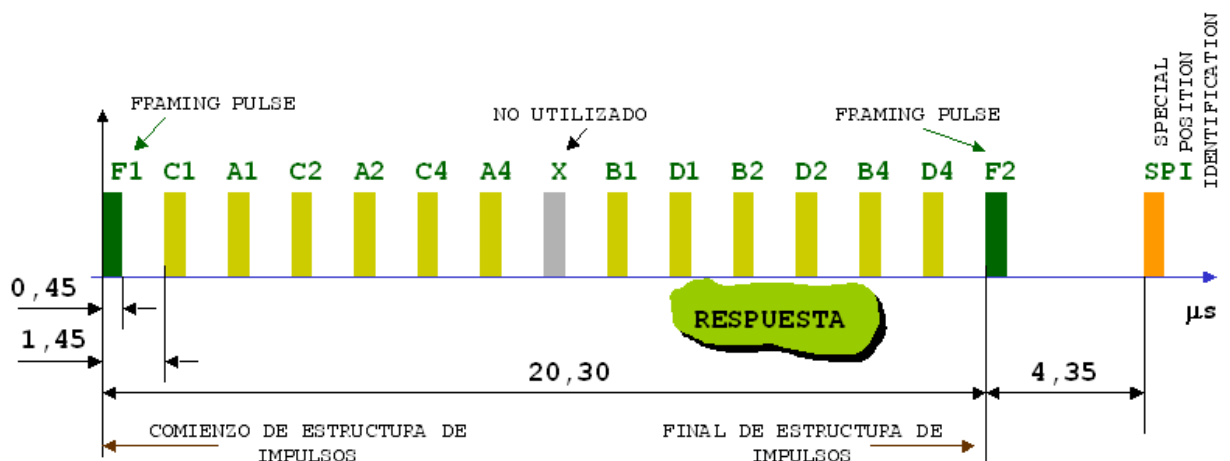
a una amplitud tal que es igual o superior a la potencia de la señal de la transmisión del mayor lóbulo lateral de la antena radiante de P1-P3. Además, la amplitud de P2 esta dispuesta para tener un nivel inferior a **+9dB** respecto a la amplitud de P1, en la dirección del eje principal del lóbulo principal. En el patrón de impulsos radiados por el transmisor, el intervalo entre P1 y P2 es de **2 μs**,

pero la posición de P3 es variable, siendo este el método por el cual el tipo de información (Modo) requerido del transponder, es indicado. Para el Modo A, el espacio entre P1 y P3 es de **8 μs**, y para el Modo C, el espacio entre P1 y P3 es de **21 μs**.

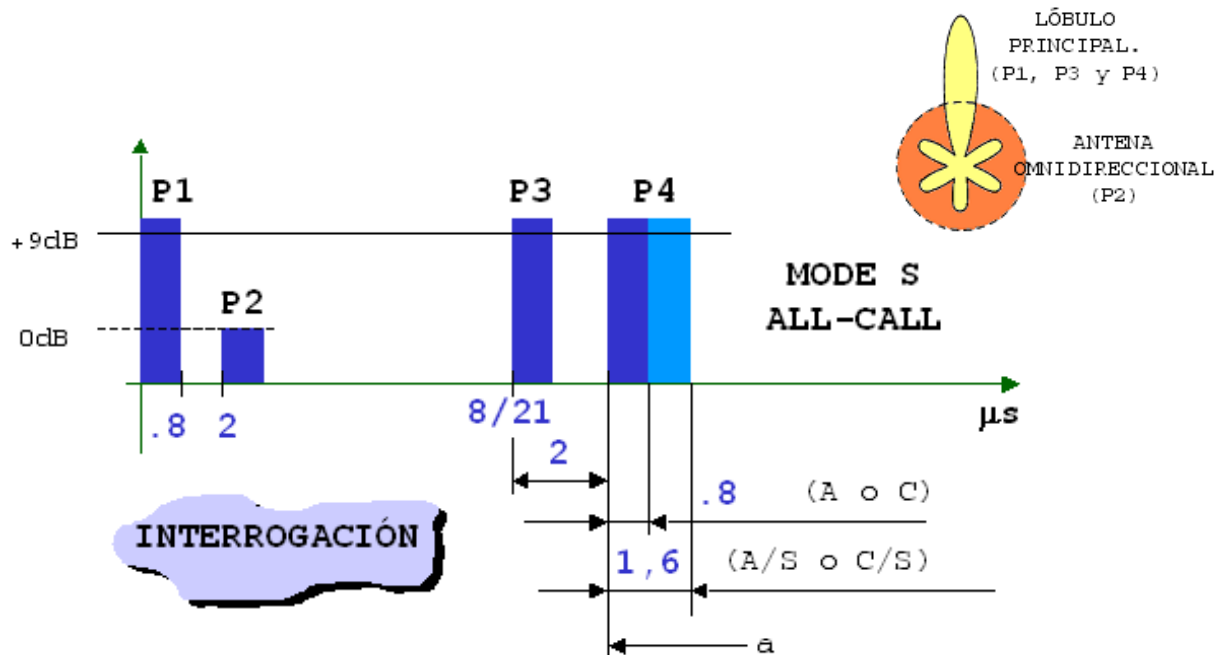


Al recibir una interrogación válida, el transponder de la aeronave radiará dos impulsos (FRAMING PULSES), espaciados **20,3 μs**, entre los cuales, a incrementos de **1,45 μs** después del primer impulso F1, están los impulsos de información (código o altitud en cientos de pies).

Entre los *framing pulses*, pueden existir hasta 13 impulsos de información. Estos impulsos están divididos en cuatro grupos de 3, mas una posición adicional designada como X, reservada para futuros usos y que de momento no es utilizada. Los impulsos en cada grupo, tienen los pesos de **1, 2 y 4**, pudiendo codificar cada grupo en octal desde el número 0 (000) hasta el número 7 (111). Así, con cuatro grupos, los números 0000 hasta 7777, pueden ser codificados en octal (4096 combinaciones).

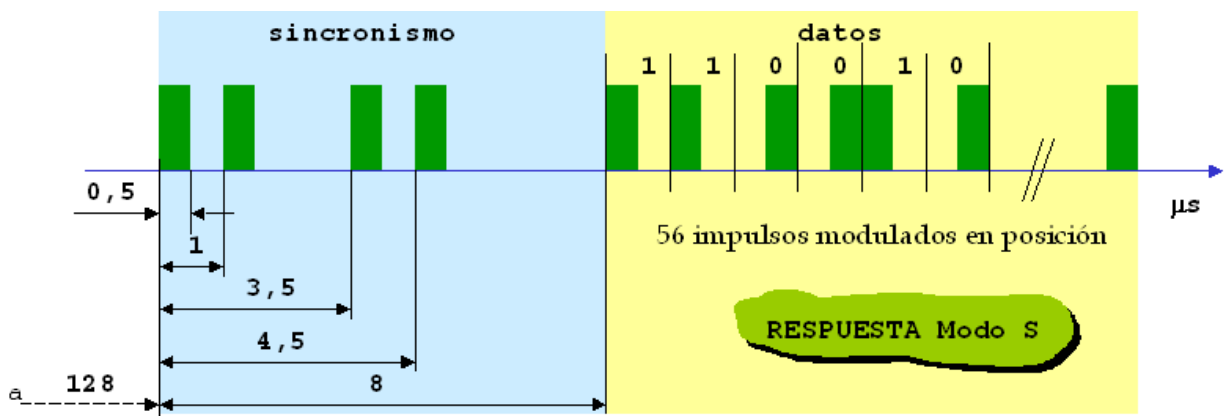


-* Una interrogación Modo S general consiste en tres impulsos llamados **P1**, **P3** y **P4**. Un impulso de control de transmisión **P2** (SLS pulse), se transmite después del primer impulso P1. Los valores de amplitud se mantienen similares a los de la interrogación Modo A o C. P4 se transmite 2 μs después de P3, y su duración depende del modo de interrogación. Si P4 tiene una duración de 0,8 μs , quiere decir que la interrogación solamente concierne a ATC convencional (transpondedores en Modo A o Modo C, dependiendo de cuando se ha transmitido P3) y consecuentemente, el transponder Modo S no debería responder. Si la duración de P4 es de 1,6 μs , quiere decir que es el transponder en Modo S el que debe responder, con información Modo S (dirección IP-24).



La respuesta en Modo A o C es la convencional.

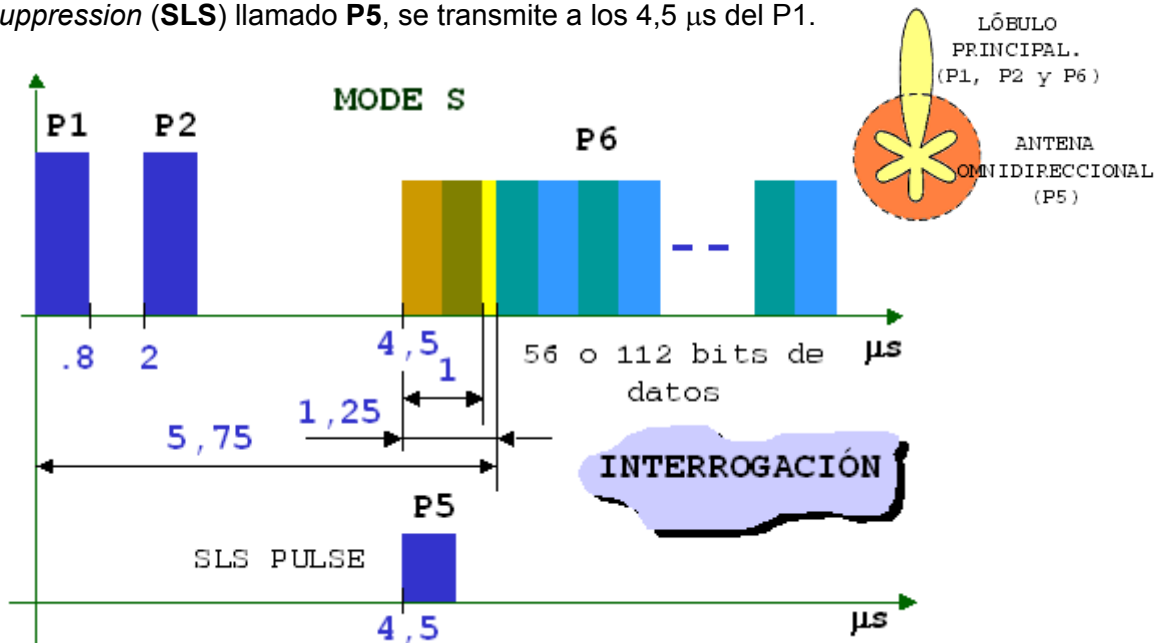
La respuesta en modo S consiste en un preámbulo de sincronización basado en dos pares de impulsos de 0,5 μs , seguido de un bloque de datos que contiene 56 impulsos modulados en posición (PPM). Cada BIT, dura 1 μs y está dividido en dos intervalos de 0,5 μs . Si el impulso correspondiente al BIT ocurre en el primer intervalo, se indica un "1" binario. Si el impulso ocurre en el segundo intervalo, indica un "0" binario. (ver figura siguiente)



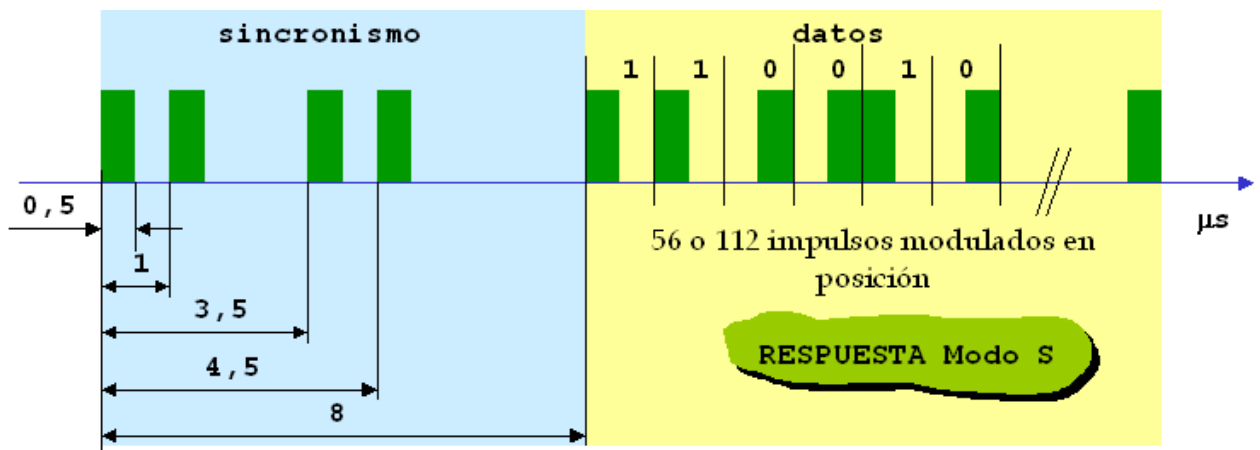
-* Una interrogación Modo S consiste en un preámbulo de sincronismo de dos bits, seguido por un bloque de datos. El preámbulo consiste en dos impulsos llamados **P1** y **P2**. Las características de estos impulsos son las mismas que para la interrogación Modo S

general, a excepción de que la amplitud del impulso P2 es igual que la del impulso P1. A continuación, hay una señal de sincronismo de duración $1\ \mu\text{s}$ seguida de un espacio de $0,25\ \mu\text{s}$. El bloque de datos, consiste en un simple impulso **P6** con una modulación **DPSK** (*Differential Phase Shift Keying*). La duración de este impulso es de dos tipos: duración corta de **$15,5\ \mu\text{s}$** y duración larga de **$29,5\ \mu\text{s}$** . El bloque de datos, está formado por 56 o 112 bits. Como el bloque de datos contiene la dirección de la aeronave, solamente la aeronave direccionada responderá.

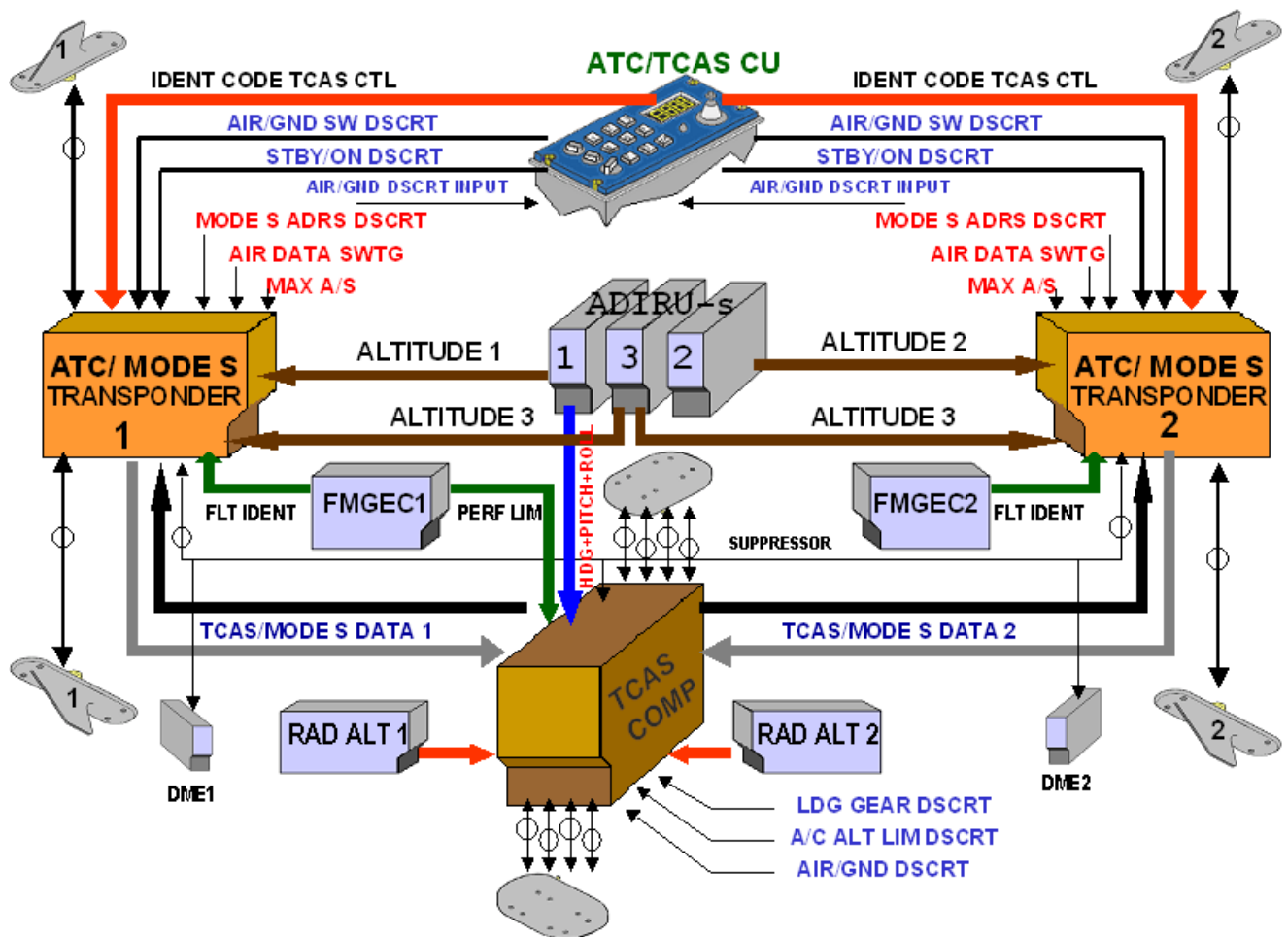
Un impulso de control de transmisión omnidireccional o impulso *Side Lobe Suppression* (**SLS**) llamado **P5**, se transmite a los $4,5\ \mu\text{s}$ del P1.



La respuesta es la misma que para el caso anterior de respuesta en Modo S, a excepción de que en este caso, el bloque de datos puede tener 56 o 112 impulsos.



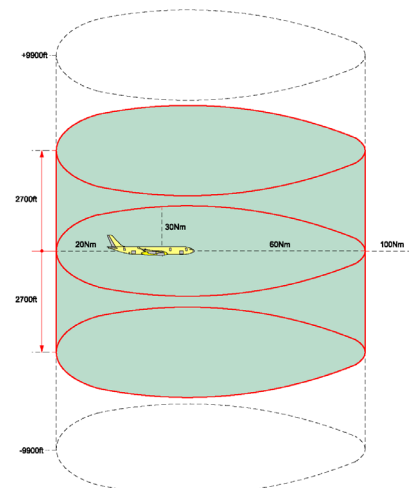
17.8.3-TCAS. DESCRIPCIÓN GENERAL.



La función del TCAS es la de detectar y mostrar aeronaves en el entorno espacial próximo y proporcionar a la tripulación de vuelo las indicaciones necesarias para evitar a esos intrusos cambiando la trayectoria de vuelo al menos en el plano vertical (TCAS II). El TCAS periódicamente interroga a los transpondedores sobre los posibles intrusos, procesa sus trayectorias y determina la amenaza potencial. El sistema también puede establecer comunicación individualizada con cada aeronave a través de los transpondedores ATC/Modo S.

El TCAS tiene capacidad para comunicarse con estaciones de tierra **ATCRBS** (*Air Traffic Control Radar Beacon System*) equipadas con el sistema Modo S para indicarles las órdenes de maniobra vertical presentadas a los pilotos de la aeronave. Esta información puede facilitar la tarea del controlador de la estación en tierra, el cual puede modificar el modo de operación del TCAS si lo cree conveniente por seguridad.

El sistema mantiene vigilancia dentro de una esfera determinada por la potencia de transmisión y la sensibilidad del receptor del computador TCAS. Como ejemplo, el sistema puede detectar hasta 40 intrusos al mismo tiempo, su capacidad de detección cubre una distancia de 60 NM por delante (incluso hasta 100 NM, según el nivel de vuelo y cantidad de tráficos), 20 NM por detrás, 30 NM en los cuadrantes laterales y ± 9900 ft. en altitud.



Dependiendo de la localización y equipamiento de la aeronave intrusa, se pueden dar los casos que se representan en la siguiente tabla:

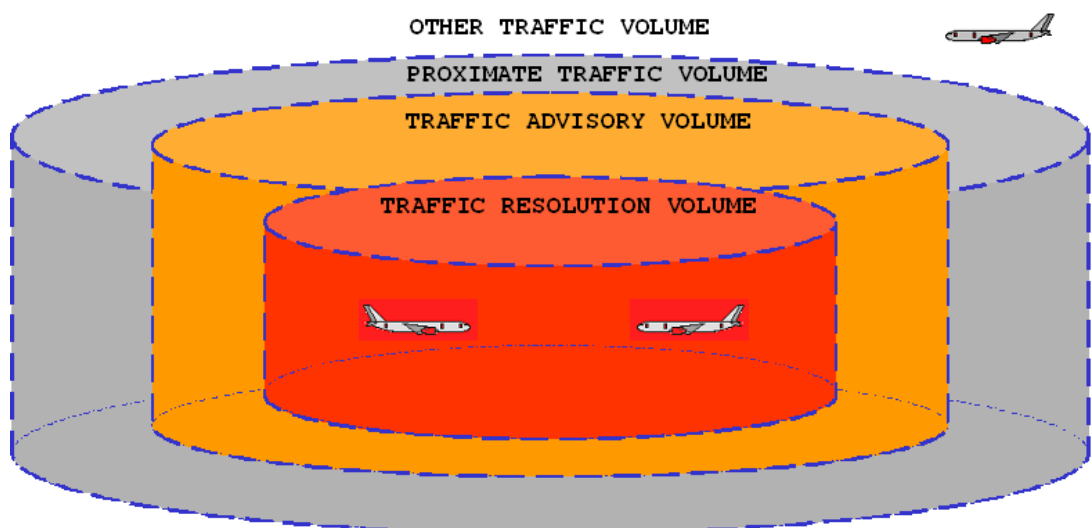
LOCALIZACIÓN (aeronave intrusa) EQUIPAMIENTO	RESOLUTION ADVISORY VOLUME ¹	TRAFFIC ADVISORY VOLUME ²	PROXIMATE TRAFFIC VOLUME ³	OTHER TRAFFIC VOLUME ⁴	OUT OF DETECTION.
Con ATC no operativo	NO DETECTION	NO DETECTION	NO DETECTION		NO DETECTION
Con ATC Modo A	TA without intruder relative level	TA without intruder relative level	Traffic information without intruder relative level		
Con ATC Modo C o S	RA without A/C maneuver coordination.	TA with intruder relative level	Traffic information with intruder relative level		
Con TCAS II	RA with A/C maneuver coordination (in vertical plane only).				

¹ Cuando el intruso representa una amenaza de colisión, el TCAS dispara alarmas visuales y auditivas llamadas *Resolution Advisory (RA)*.

² Cuando el intruso está relativamente cerca, pero no representa una amenaza inmediata, el TCAS proporciona información auditiva y visual conocida como *Traffic Advisory (TA)*.

³ Cuando un intruso es detectado dentro de un cierto rango, el TCAS lo muestra en el ND para informar a la tripulación.

⁴ Other traffic volume se refiere al volumen donde se encuentra un tráfico con un transponder respondiendo dentro del rango de display y que no está incluido en las otras categorías de localización.

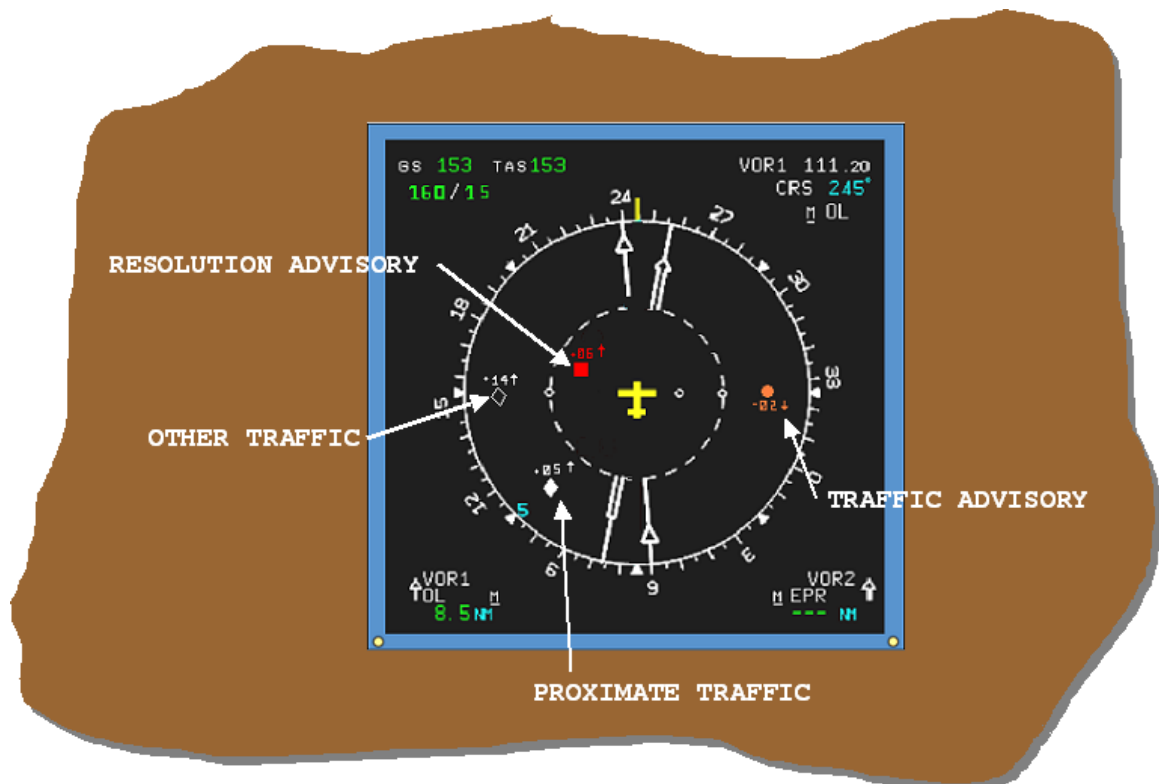


El computador TCAS, puede almacenar un número determinado de los más recientes RA-s y TA-s en la memoria, con información relativa a la aeronave implicada (distancia, altitud, rumbo, dirección Modo S, etc.).

INDICACIÓN.

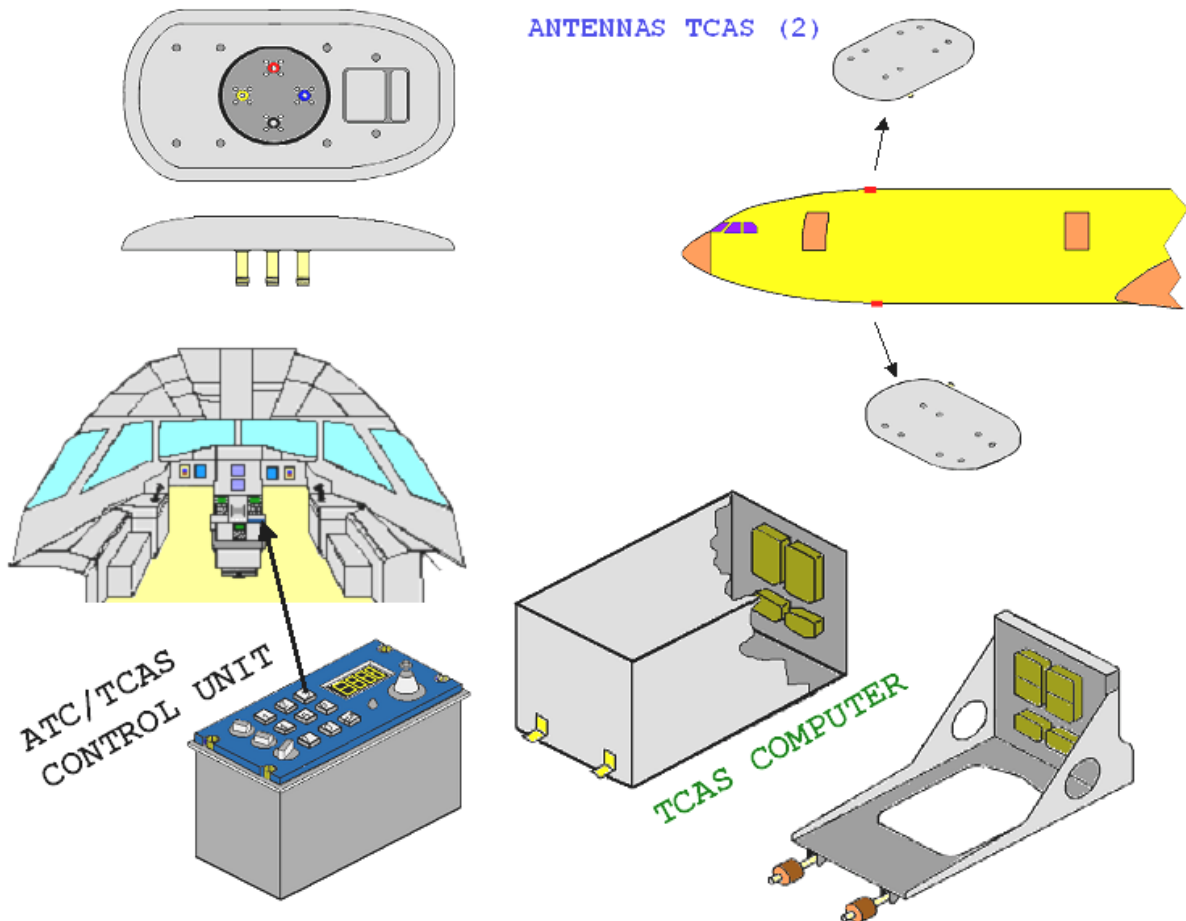
Las indicaciones TCAS aparecen habitualmente en el PFD y en el ND. En el PFD aparece una indicación en la zona correspondiente a la velocidad vertical, limitando esta velocidad. En el ND, en los modos ROSE o ARC, se presentan las indicaciones para cada uno de los diferentes tráficos que se pueden dar. Estas indicaciones muestran la situación en la zona de vigilancia. Las aeronaves presentes en esta zona, están representadas por símbolos cuya forma y color corresponden a los tipos de intrusos definidos en el TCAS: rombo blanco hueco para OTHER TRAFFIC, rombo blanco para PROXIMATE TRAFFIC, círculo ámbar para TRAFFIC ADVISORY y cuadrado rojo para RESOLUTION ADVISORY.

Junto con cada uno de los símbolos, aparecen dos dígitos que indican la altitud relativa entre la aeronave y las aeronaves intrusas, dada esta diferencia en cientos de pies. También aparece un signo, que indica si la aeronave intrusa está por encima (+) o por debajo (-) y una flecha a la derecha del símbolo que indica la tendencia vertical (ascendente o descendente) de la aeronave intrusa. Las indicaciones visuales de Resolution Advisory y Traffic Advisory están asociadas a indicaciones auditivas tales como "TRAFFIC, TRAFFIC", "CLIMB, CLIMB", ...



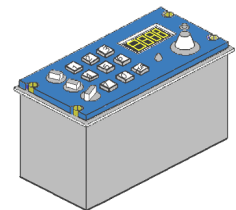
COMPONENTES TCAS.

Los componentes del sistema TCAS son, dos antenas transmisoras-receptoras, una situada en la parte superior de la aeronave y la otra en la parte inferior, un computador TCAS situado en el compartimiento de aviónica y una unidad de control ATC/TCAS situada en el cockpit.

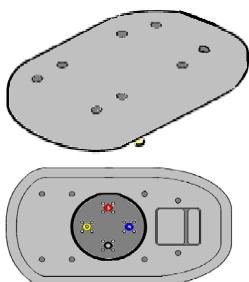


ATC / TCAS CONTROL UNIT.

Realiza las funciones explicadas en la anterior sección referente al ATC.



ANTENAS TCAS.

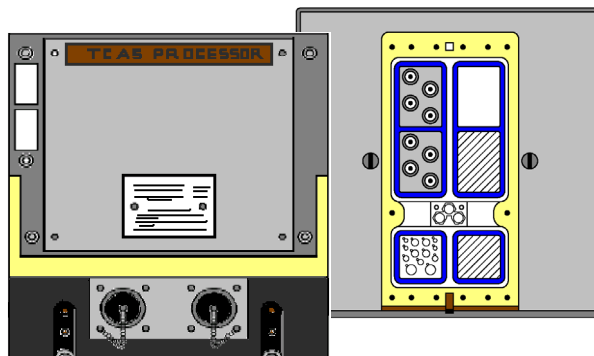


Cada antena direccional del tipo emisora /receptora, está compuesta por cuatro elementos pasivos polarizados verticalmente. La antena, en su parte inferior posee cuatro conectores codificados con cuatro colores para conectar los cuatro elementos de la antena al computador TCAS mediante cable coaxial. Las antenas se utilizan para recibir y proporcionar información direccional para Modo S y respuestas ATCRBS a la frecuencia de **1090 MHz**. La fase apropiada de los cuatro elementos de la antena, permite omni o direccional transmisión a **1030 MHz**, o mensajes de coordinación y ATCRBS o interrogaciones Modo S.

COMPUTADOR TCAS.

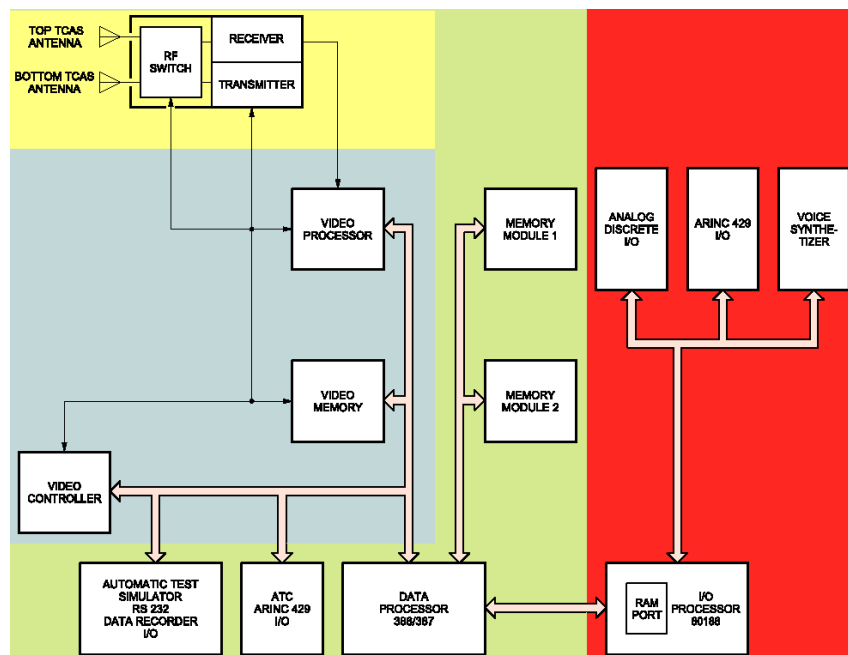
El computador TCAS es el corazón del sistema. Lleva a cabo dos funciones principales: función de transmisión y recepción en la banda L para la adquisición de datos de intrusos y una función de procesamiento para asegurar un total control de la operación.

Este computador está dividido en cuatro secciones principales:



- Radio Frequency (RF), Video Processor (VP), Data Processor (DP) y I/O Processor (IOP).

-La sección de RF, consiste en el transmisor, receptor y conmutación. Esta sección se encarga de procesar los datos de la respuesta ATCRBS o intruso Modo S, recibidos por cualquiera de las dos antenas (superior o inferior, seleccionada mediante la función RF Switch). También, se encarga de interrogar intrusos una vez por segundo, a través de cualquiera de las dos antenas.



ser enviadas al procesador de vídeo.

El transmisor genera una portadora de **1030 MHz**. Para la interrogación Modo S, esta frecuencia es modulada en **DSPK (Data Phase Shift Keyed)** por el módulo de controlador de vídeo. La frecuencia modulada, es dividida en cuatro partes y transmitida por los cuatro canales que gobiernan los elementos de la antena.

El circuito receptor consiste en tres canales. Trabaja con una frecuencia intermedia de 60 MHz y su misión es la de procesar las señales recibidas para

- La sección de VP está compuesta por el Video Processor, Video Controller y por la Video Memory. Realiza las funciones de determinación del rumbo relativo, distancia y altitud a partir de las señales recibidas de la sección de radio frecuencia.

- La sección de DP está compuesta por los Memory Modules, ATC ARINC 429 I/O, Data Processor y Automatic Test Simulator RS232 Data Decoder I/O. Esta sección de procesamiento, está basada en un microprocesador 80386 a 32 MHz, con su coprocesador matemático (80387)

- La sección I/O está compuesta por los módulos de Analog Discrete I/O, ARINC 429 I/O, Voice Synthesizer y I/O Processor. Adquiere la mayor parte de los datos requeridos por el TCAS y transmite señales de control e información a otros equipos.

17.8.4-OPERACIÓN TCAS.

Cuando la aeronave está en vuelo, su TCAS periódicamente transmite señales de interrogación para todas las ATCRBS y aeronaves equipadas con transpondedores Modo S en las proximidades. Estas interrogaciones son recibidas por las estaciones ATCRBS de tierra y por los transpondedores de la otra aeronave. En respuesta a estas interrogaciones, los transpondedores de la aeronave vecina envía señales que contienen información del valor de su altitud. El computador TCAS calcula la distancia entre las dos aeronaves midiendo el tiempo que transcurre entre la transmisión de la interrogación y la recepción de la respuesta.

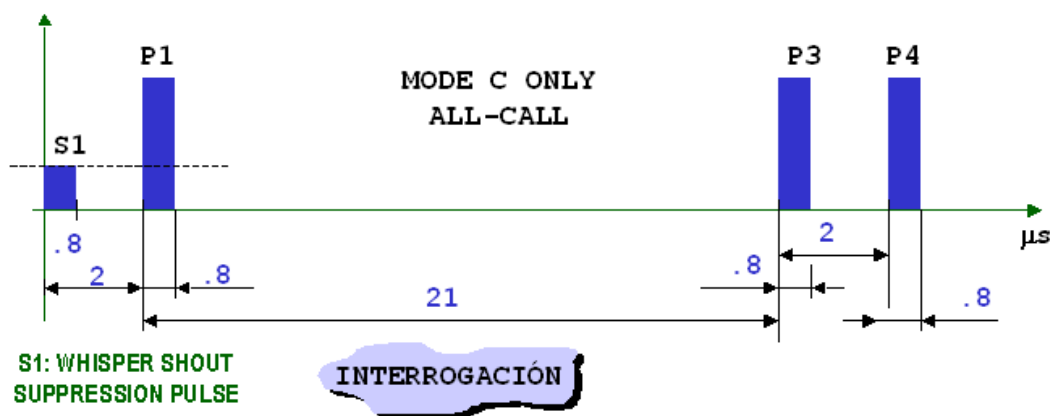
La altitud, variación de altitud (velocidad vertical), distancia y variación de distancia (velocidad) se determinan realizando una vigilancia periódica de estos cambios y los datos son utilizados para valorar la amenaza que supone la aeronave intrusa. Cada amenaza se trata de forma individual, pero el TCAS determina la mejor posibilidad de evasión con respecto a todas las aeronaves en su vecindad, a la vez que establece la coordinación de maniobras con la otra aeronave equipada con TCAS. La maniobra óptima es aquella que asegura una adecuada separación de trayectorias con la mínima tendencia de variación vertical.

La primera aeronave en detectar a la otra, inicia el procedimiento de comunicación, indica las maniobras que tiene intención de realizar y comunica órdenes a la otra aeronave solicitándole el mantenimiento de su trayectoria.

DETECCIÓN DE AERONAVE INTRUSA.

El TCAS detecta aeronaves equipadas con transpondedores Modo S, captando transmisiones periódicas que estas aeronaves realizan. Los transpondedores Modo S anuncian su presencia transmitiendo mensajes periódicos una vez cada segundo que contienen la dirección de 24 bits asignada a la aeronave. El TCAS también detecta aeronaves equipadas con transpondedores que no responden a las interrogaciones Modo S pero si a las Modo C.

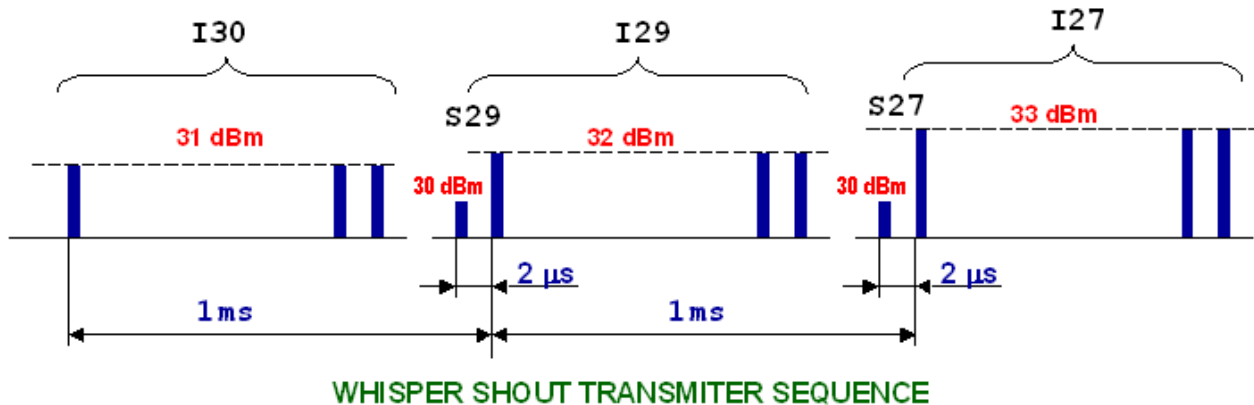
- Interrogación de aeronaves equipadas con transpondedores Modo A o Modo C:
El TCAS transmite Modo C solamente interrogación general (impulsos P1, P3 y P4). El código es similar al utilizado por las estaciones de tierra Modo A y Modo C. El impulso P4 informa a los transpondedores Modo S, que esta interrogación no está dirigida a ellos.



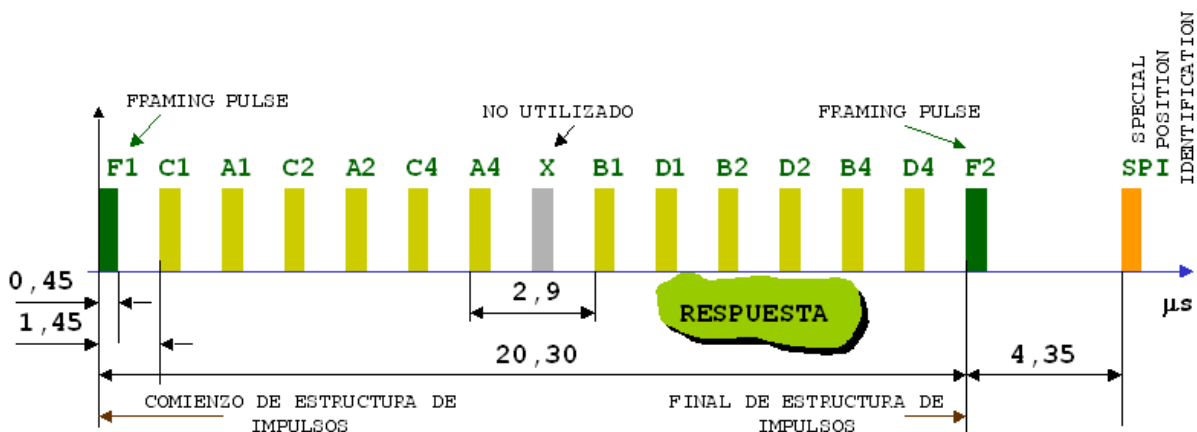
El intervalo nominal de tiempo entre dos interrogaciones es de un segundo. Para limitar interferencias radioeléctricas en áreas con alta densidad de tráfico, cada interrogación consiste en una serie de interrogaciones de potencia creciente para alcanzar aeronaves más remotas (whisper shout) con 1 ms. de periodo de tiempo dentro de las series (PRF).

La primera transmisión consiste en los impulsos P1, P3 y P4, de relativamente baja potencia, por lo tanto, solamente la aeronave más cercana va a recibir y responder a estas interrogaciones.

Después, el impulso S1 es transmitido también previo a la siguiente interrogación. Este impulso de menor amplitud, hace que la aeronave más cercana interprete la interrogación asociada como procedente de un lóbulo lateral de la estación de transmisión, no necesitando respuesta. El propósito de la secuencia Whisper-Shout es reducir el número de aeronaves que responden a una interrogación cualquiera, limitando por lo tanto la interferencia.



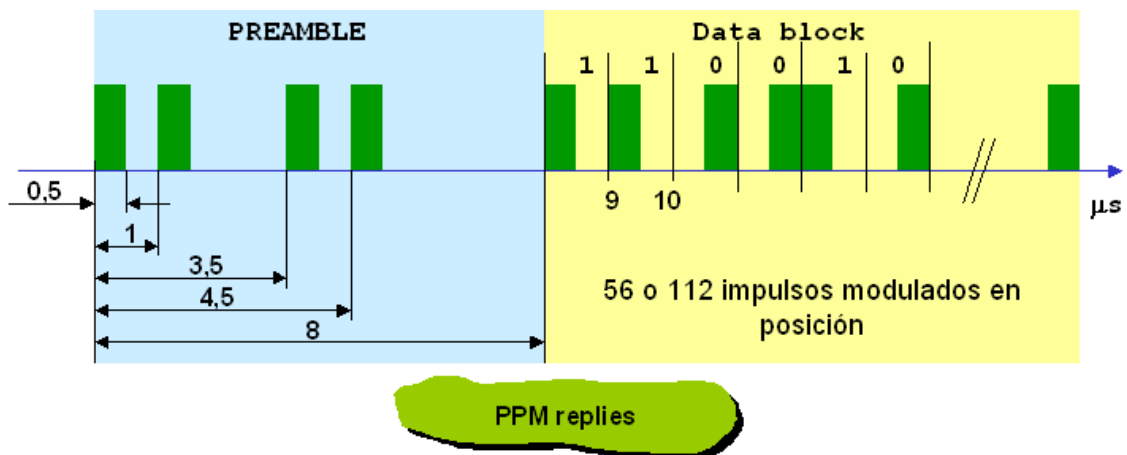
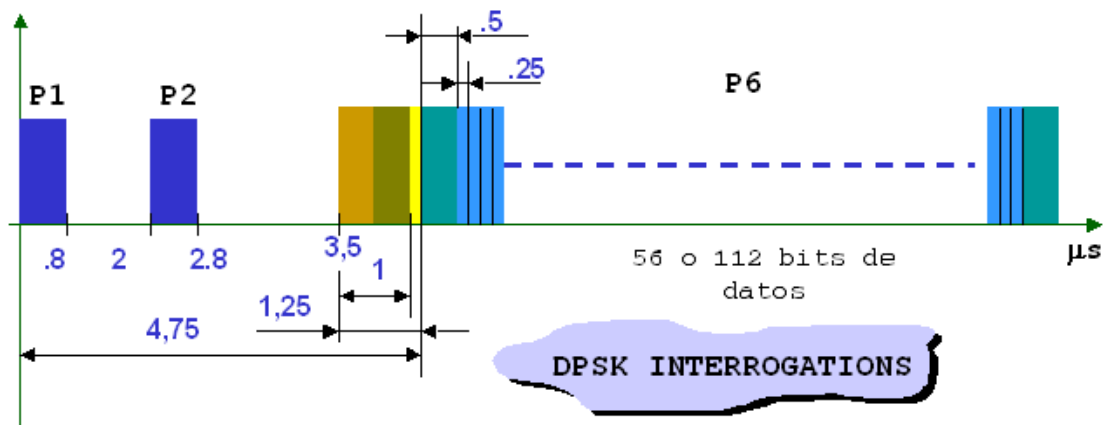
- Respuesta de aeronaves equipadas con transpondedores Modo A o Modo C: Las aeronaves equipadas con transpondedores Modo C responden transmitiendo su altitud, codificada en octal en cuatro dígitos ABCD, con un valor de 100ft para el LSB en el formato ATRBS. Las equipadas con transpondedores Modo A, responden transmitiendo su código de identificación Modo A. En este caso, la presentación de intruso en el ND se limita a mostrar su posición en distancia y rumbo.



- Interrogación de aeronave equipada con transpondedor Modo S.

a.- Codificación de transmisión. La interrogación está compuesta por tres impulsos: P1, P2 y P6. El nivel de P2 es igual o mayor que el nivel de P1, condición de no-respuesta para las aeronaves equipadas con transpondedores Modo A o Modo C. Por lo tanto, solamente las aeronaves equipadas con transpondedores Modo S responden a la interrogación. La información útil, está contenida en P6 dividida en 56 o 112 chips (un chip es un intervalo no modulado de 0,25 μ s, precedido de posibles inversiones de fase). Hay dos formatos de mensaje diferentes: (1) todas la interrogaciones Modo S (formato UPLINK) son señales binarias DPSK. (2) las respuestas Modo S (formato DOWNLINK) codifican los datos de respuesta con vía modulación PPM (Pulse Position Modulation).

La respuesta Modo S va precedida de un preámbulo basado en cuatro impulsos de duración e intervalos específicos que pretende garantizar la validez del mensaje recibido. Cualquier mensaje que no cumple con el modelo de preámbulo establecido es rechazado por el TCAS. La información está codificada en modo PPM con, para cada BIT, un nivel lógico uno si la primera mitad del intervalo esta a uno y un nivel lógico cero si está a cero (Manchester)



b.- Mensajes comunicación Modo S. El sistema de comunicación ATC Modo S que soporta intercambios personalizados entre las estaciones ATC de tierra y las aeronaves equipadas con transpondedores Modo S, se compone de un set de 25 mensajes estándar para enlace estación-aeronave (*UPLINK* format) identificado como **UFxx** y otro set de 25 mensajes estándar para enlace aeronave-estación (*DOWNLINK* format) identificado como **DFxx**.

El mensaje puede ser largo (112 bits) o corto (56 bits). Cada mensaje consiste en campos específicos con combinaciones de bits que tienen significados específicos. Sin embargo, existen dos campos que tienen una definición idéntica para todos los mensajes: (1) el campo de "mensaje tipo" consiste en los bits 1 a 5 cuya codificación convierte el valor decimal a binario y (2) el campo de dirección que contiene los últimos 24 bits del mensaje (bits 33 a 56 o bits 89 a 112) dependiendo de si el mensaje es corto o largo. Estos 24 bits contienen la dirección Modo S del transmisor.

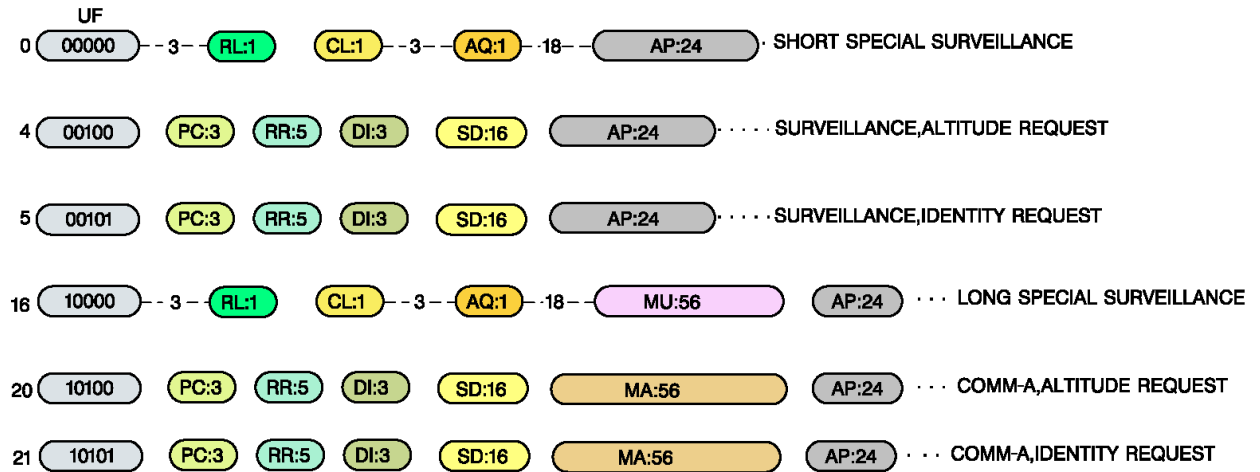
En el set de mensajes Modo S, el TCAS utiliza seis tipos de mensaje UF (UF0, UF4, UF5, UF16, UF20 y UF21) y ocho tipos de mensaje DF (DF0, DF4, DF5, DF11, DF16, DF17, DF20 y DF21)

En las siguientes tablas se representan los formatos UPLINK utilizados por el TCAS. En ellas se utiliza la siguiente nomenclatura:

XX:M indica el nombre del campo (XX) que tiene asignados M bits.

----N---- indica espacios no asignados en los que N bits deben ser puestos a 0.

Los primeros cinco bits corresponden al código binario de la interrogación.



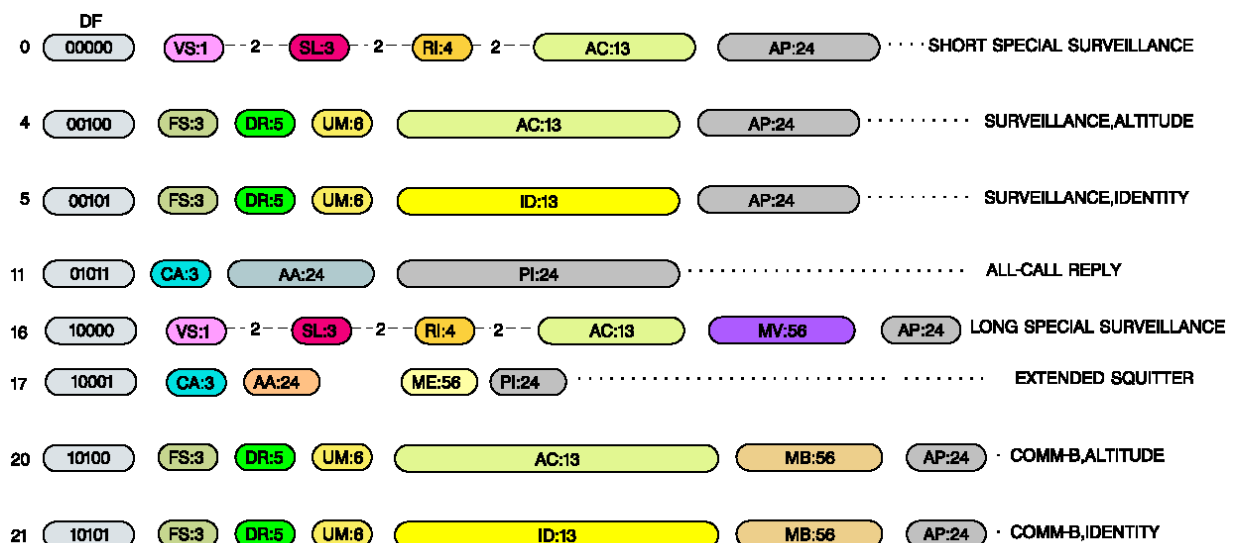
DSG	FIELD	INDICATION
AP	ADDRESS PARITY	Dirección codificada con chequeo de paridad
AQ	ACQUISITION	Indica si es un mensaje de interrogación. (1= INTERROGACIÓN)
DI	DESIGNATOR IDENT	Especifica el tipo de información contenida en el campo SD
MA	MESSAGE COMM-A	Utilizado por la estación de tierra para transmitir una orden SL TCAS a una aeronave equipada con TCAS
MU	MESSAGE COMM-U	Utilizado por el TCAS para transmitir a otra aeronave información de coordinación RA.
PC	PROTOCOL	Ordenes operacionales para el transponder
RL	REPLY LENGTH	Indica si el mensaje es corto (0) o largo (1)
RR	REPLY REQUEST	Longitud y contenido de la información de respuesta solicitada por el interrogador.
SD	SPECIAL DESIGNATOR	Contiene códigos de control que afectan al protocolo del transponder.

En las siguientes tablas se representan los formatos DOWNLINK utilizados por el TCAS. En ellas se utiliza la siguiente nomenclatura:

XX:M indica el nombre del campo (XX) que tiene asignados M bits.

----N---- indica espacios no asignados en los que N bits deben ser puestos a 0.

Los primeros cinco bits corresponden al código binario de la respuesta.



DSG	FIELD	INDICATION
AA	ADDRESS	Dirección Modo S en 24 bits
AC	ANNOUNCED ALT CODE	Información indicando la altitud de la aeronave
AP	ADDRESS PARITY	Dirección codificada con chequeo de paridad
CA	CAPABILITY	Capacidad del transponder
DR	DOWNLINK REQUEST	Extracción de peticiones del mensaje DOWNLINK por el interrogador (existiendo RA)
FS	FLIGHT STATUS	Tierra, vuelo, alerta, SPI
ID	IDENTIFICATION CODE	Contiene el código de identificación Modo A
MB	MESSAGE COMM-B	Indica contenido de recomendación para la estación de tierra
MV	MESSAGE COMM-V	Contiene subcampos ARA, RAC, VDS utilizados para coordinación
RI	REPLY INFORMATION	Tipo de respuesta y capacidad de velocidad respecto al aire
SL	SENSITIVITY LEVEL	Nivel de sensibilidad actual del TCAS
UM	UTILITY MESSAGE	Estado del transponder
VS	VERTICAL STATUS	Estado de la aeronave: 0= en vuelo, 1= en tierra

MEDICIÓN DE PARÁMETROS DE AERONAVE INTRUSA.

Una vez confirmada la recepción por parte del transponder, el TCAS comienza a interrogar a la aeronave intrusa. Su altitud se transmite directamente en la respuesta (SBA-Standard Barometric Altitude) y esta información se utiliza para determinar la altitud relativa entre las dos aeronaves. Este cálculo es solamente posible con respecto a aeronaves equipadas con transponder Modo C o Modo S.

La distancia se calcula a partir del tiempo que transcurre desde que se transmite la señal de interrogación y la vuelta de la respuesta transmitida por la aeronave intrusa.

Para calcular el acimut se utiliza la técnica de interferometría. Este principio se basa en la comparación de la fase de las señales recibidas por cuatro elementos independientes de la antena direccional tomados por pares. Con dos polos E2 y E4, la diferencia de fase de las señales recibidas en los dos elementos depende del ángulo de recepción de estas señales puesto que la diferencia en distancia a la fuente de los dos polos de recepción varía con este ángulo.

Como se conoce la distancia de separación de dos polos, y la longitud de onda de la señal fuente, se obtiene:

$$S2-S4 = (d / \lambda) 360 \sin B \quad \text{donde}$$

B: ángulo de recepción de la señal fuente

S2-S4: diferencia de fase de la señal recibida en los polos E2 y E4

D: distancia entre los dos polos.

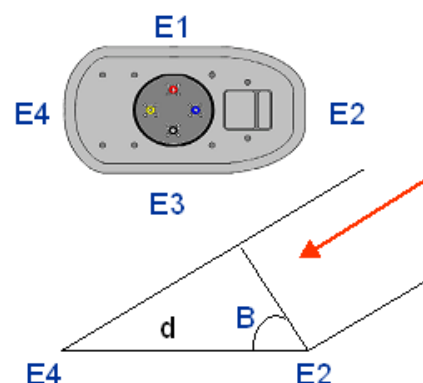
λ : longitud de onda de la señal. (para 1090 MHz $\lambda = 27,52$ cm)

$$S1-S3 = (d / \lambda) 360 \cos B$$

Por lo tanto,

$$B = \arctan [(S2-S4) / (S1-S3)]$$

Si la antena está alineada con el eje longitudinal de la aeronave, el ángulo B representa el acimut fuente y puede ser utilizado para indicar el rumbo de la aeronave intrusa en el ND.



Cuando ya están identificadas las aeronaves intrusas, estas son monitorizadas por una serie de interrogaciones y respuestas en Modo C general, para aquellas aeronaves equipadas con transpondedores Modo C, y en Modo S para aquellas equipadas con XPDR Modo S. Estos intercambios de señales, permiten al TCAS actualizar periódicamente la altitud, distancia y rumbo para cada aeronave intrusa, y calcular la variación en altitud y distancia. Con estos datos, el TCAS calcula el tiempo que separa las dos aeronaves de su punto de aproximación más cercano.

Cada 10 segundos, el TCAS transmite un mensaje para informar a las aeronaves cercanas equipadas también con TCAS de la presencia de una aeronave equipada con TCAS en su área de tráfico.

COORDINACIÓN.

En la mayor parte de los casos de encuentros de dos aeronaves equipadas con TCAS, la identificación mutua ocurre casi simultáneamente, pero con el intervalo de tiempo preciso para que se pueda establecer la prioridad necesaria para el proceso de coordinación.

La primera aeronave en detectar una configuración potencialmente peligrosa, procesa una maniobra de desviación y lo comunica a la otra aeronave. Esta aeronave tiene en cuenta la información y procesa una corrección. Si dos aeronaves se detectan una a la otra a exactamente el mismo instante, y simultáneamente transmiten mensajes de coordinación que contienen maniobras de desviación incompatibles, por convenio, la aeronave que tiene la dirección Modo S superior, la que cancela la corrección de su trayectoria.

La comunicación entre dos aeronaves comprende tres fases: (1) fase de detección, (2) fase de vigilancia (dividida a su vez en interrogación de adquisición, respuesta de adquisición, interrogación de captura y respuesta de captura) y (3) fase de coordinación (dividida a su vez en interrogación de coordinación y respuesta de coordinación).

17.8.5-TCAS BITE.

SELF TEST.

Se puede realizar presionando el botón de test que el computador TCAS tiene en su panel frontal, o a través del **CMS** (*Central Maintenance System*), aplicando el procedimiento *TCAS Functional Test en la MCDU*.

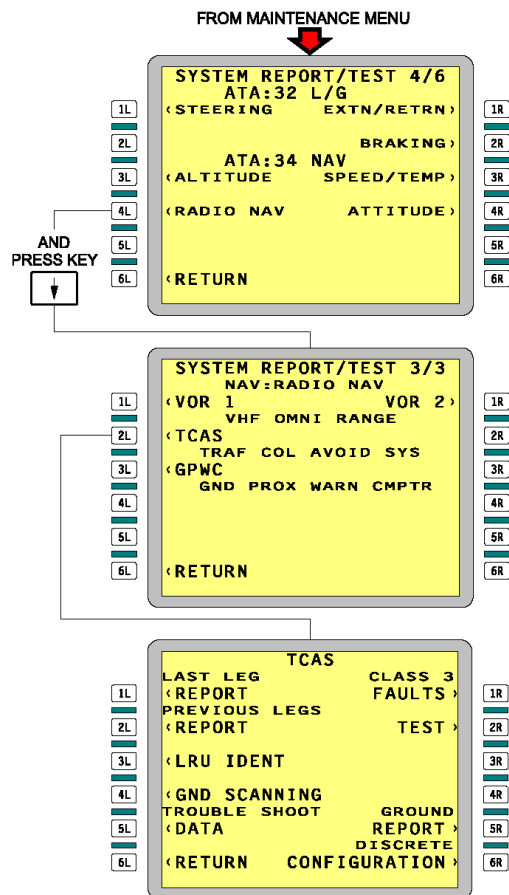
Al final de la secuencia de test el sistema genera un mensaje con voz sintética "TCAS SYSTEM TEST OK" si el sistema opera correctamente, y "TCAS SYSTEM TEST FAIL" si se ha detectado alguna anomalía. En este caso, los ND-s, los PFD-s y el EWD muestran los mensajes de fallos pertinentes. Además, en un Display LCD del frontal del computador TCAS, se muestra un código identificando el componente que ha fallado.

CMS.

El *Central Maintenance System* permite presentar los fallos del computador TCAS en una MCDU y por impresora. Este sistema, puede operar de dos modos: (1) modo MENÚ y (2) modo NORMAL.

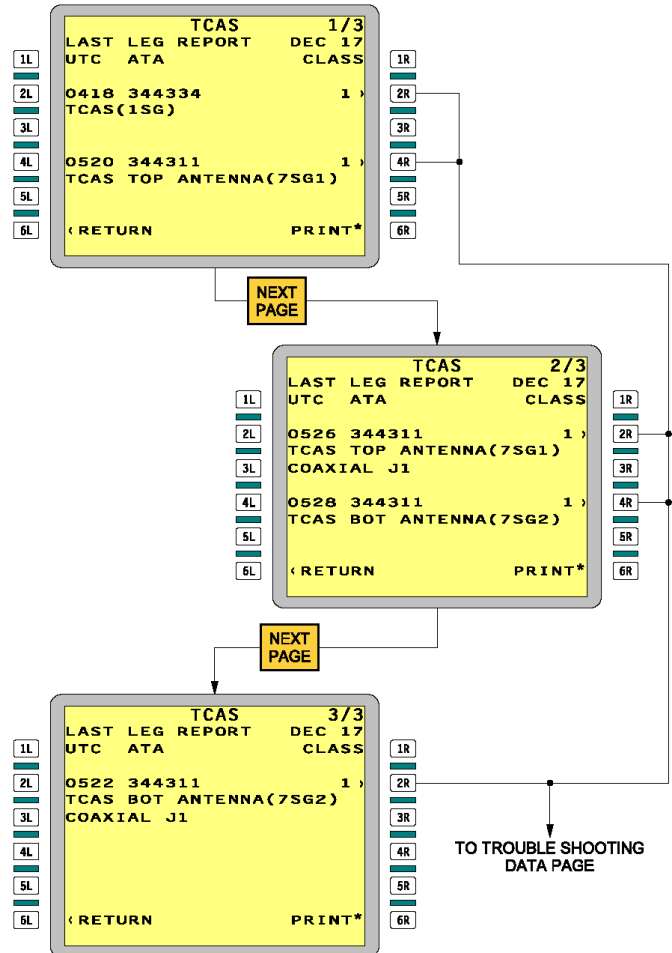
MODO MENÚ:

- Función SYSTEM REPORT/TEST. Esta función permite el acceso a todos los sistemas electrónicos y al computador TCAS.

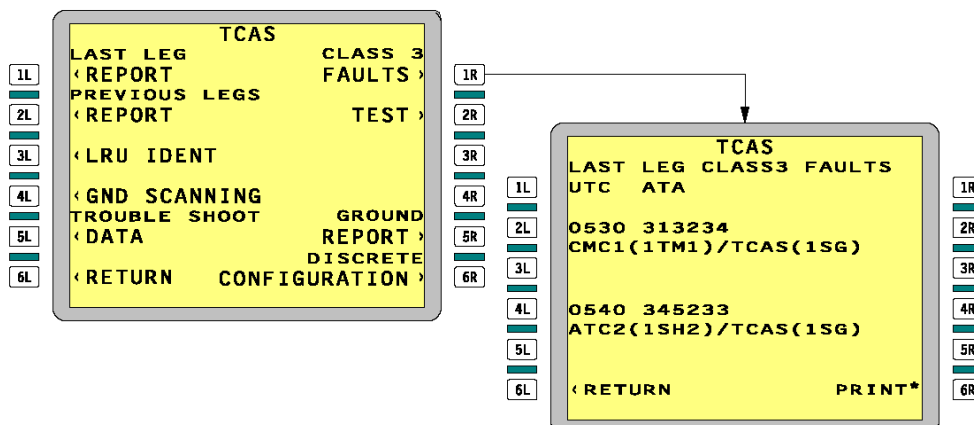


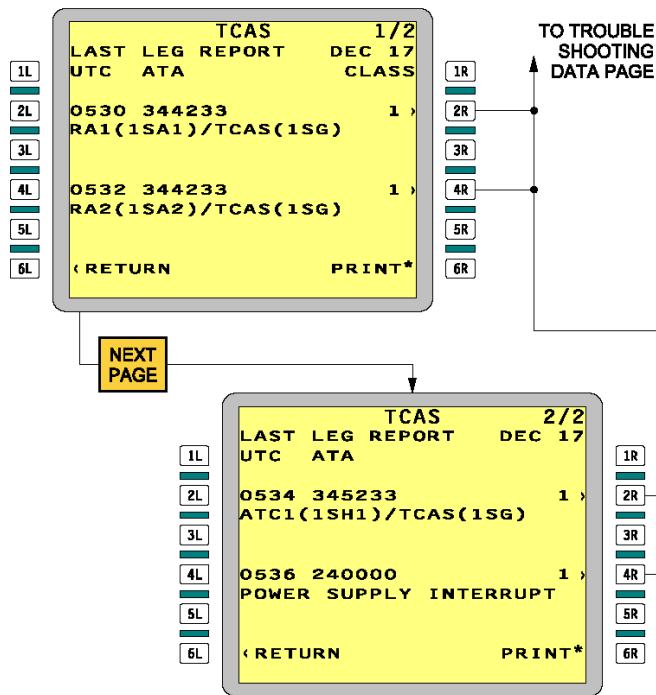
-LAST LEG REPORT. Este informe contiene los fallos internos y externos del computador memorizados durante el último vuelo.

FALLOS INTERNOS:



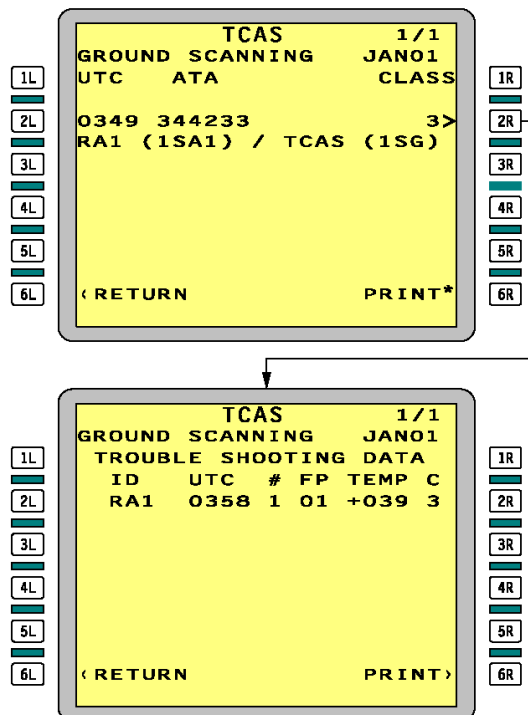
FALLOS EXTERNOS





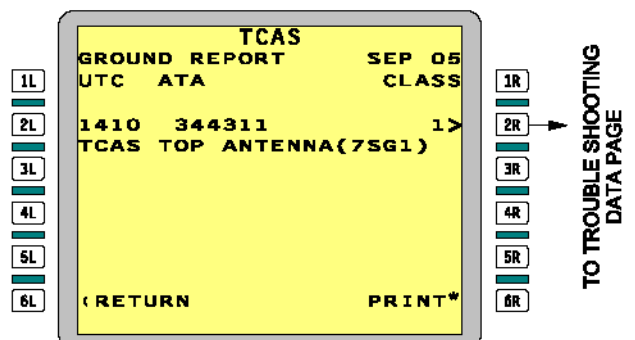
- GND SCANNING

Esta función permite la consulta de las memorizaciones de fallos del computador TCAS tal y como los define el fabricante del componente. Los tests cíclicos internos y monitorización de periféricos TCAS son usados para detectar fallos esporádicos.

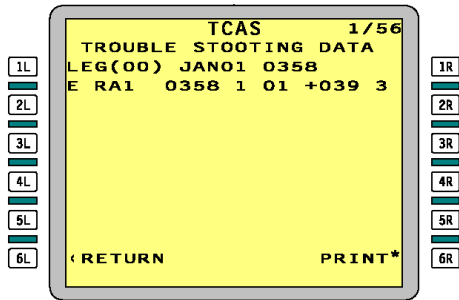


- GND REPORT.

Esta función se utiliza para presentar fallos internos clase 1, 2 o 3 cuando son detectados en tierra



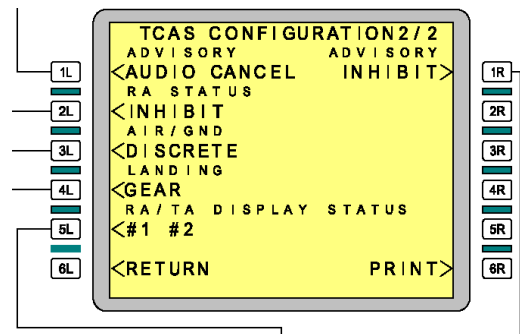
- TROUBLE SHOOT DATA.



Esta función permite analizar de forma rápida los fallos almacenados para detectar cualquier defecto de software.

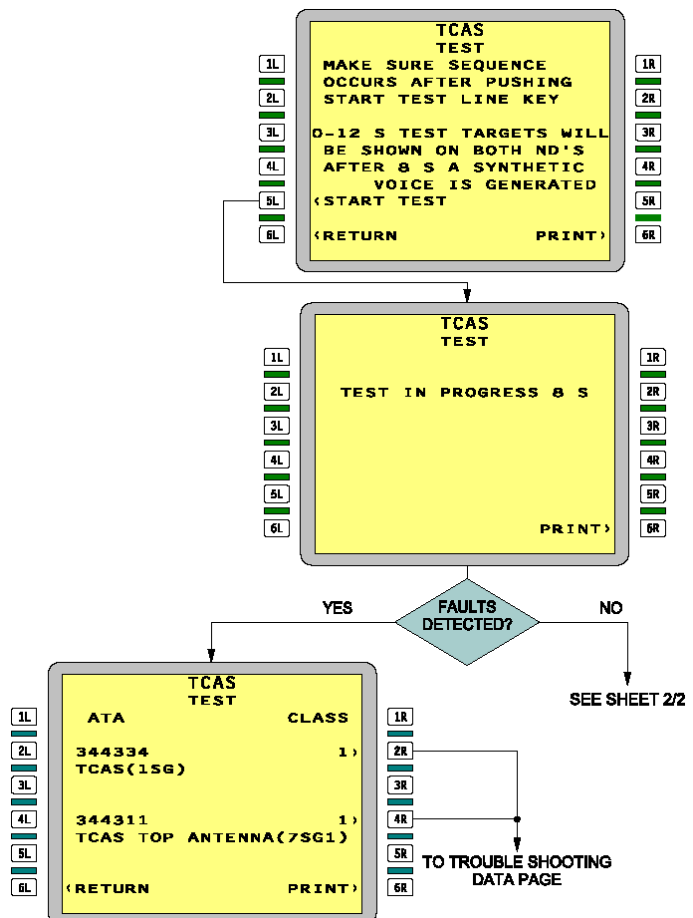
- DISCRETE CONFIGURATION.

Esta función se utiliza para presentar el estado de diversos discretos. Una pantalla informe, muestra el estado de los discretos en el momento en el que este menú es seleccionado.



- TCAS FUNCTIONAL TEST

El test finaliza con la pantalla de la MCDU mostrando bien "TCAS SYSTEM TEST OK" si no se ha detectado fallo, o "TCAS SYSTEM TEST FAIL" si se ha detectado alguna anomalía. En este caso, voces sintéticas anuncian este estado.



17.9-ATIMS (AIR TRAFFIC AND INFORMATION MANAGEMENT SYSTEM)

17.9.1-INTRODUCCIÓN A LA INFORMACIÓN POR ENLACE DE DATOS.

Esta información se divide en:

- (1) Aircraft Information Network System. (AINS).
- (2) Cockpit Information Systems.
- (3) Maintenance Information Systems.
- (4) Cabin Information Network System. (**CINS**)
- (5) Internal Wireless Datalink.
- (6) Passenger Information Network.

*(1) El AINS consiste en:

- un servidor que proporciona una plataforma para aplicaciones de mantenimiento y operaciones.
- una interfaz segura para los sistemas aviónicos.
- un enlace virtual entre la red *Ethernet Local Area Network* (**ELAN**) de la aeronave y el sistema de información basado en tierra, de la aerolínea.

El sistema adquiere, almacena, procesa y transmite datos de sistemas aviónicos a: el operador de mantenimiento para ayudar en el trouble shooting, a la tripulación de vuelo para realizar cálculos de actuación, consultar documentación, etc.; y también a la red de la aerolínea, a través de la red basada en tierra del aeropuerto.

*(2) Los sistemas de información del cockpit lo componen:

- (a) el *Air Traffic and Information Management System* (**ATIMS**).
- (b) los *Air Traffic Services* (**ATS**)
- (c) los *Airline Information Services* (**AIS**).
- (d) el *Air Traffic Service Unit* (**ATSU**) Router.

(a) ATIMS. Permite comunicaciones **CPDLC** (*Controller-Pilot Datalink Communications*) para gestión del tráfico aéreo (**ATM**).

(b) ATS. Estos servicios incluyen:

Aplicaciones *Air Traffic Control* (**ATC**) *Future Air Navigation System* (**FANS**), tales como: aplicación *ATS Facilities Notification* (**AFN**) para conexión con el centro ATC de tierra; aplicación CPDLC para comunicaciones vía enlace de datos; aplicación *Automatic Dependent Surveillance* (**ADS**) para vigilancia sobre la trayectoria de la aeronave.

Aplicaciones ATC ARINC 623 tales como: aplicación *Departure Clearance* (**DC**); aplicación *Oceanic Clearance* (**OC**) y aplicación *Automatic Terminal Information Service* (**ATIS**)

(c) AIS. Estos servicios incluyen las aplicaciones estándar *Airline Operational Control* (**AOC**). Estas aplicaciones de enlace de datos son utilizadas por la aerolínea para mantenimiento y propósitos administrativos.

(d) Router ATSU. Este router proporciona: servicios de enlace de datos para aplicaciones AOC ubicadas en periféricos de abordaje; servicios de enlace de datos para aplicaciones ATC/AOC ubicadas en la ATSU; gestión de los medios de enlaces de datos (VHF, *Satellite Communication* (**SATCOM**), HF)

(3) Los sistemas de información de mantenimiento: está compuesto por el *Maintenance Access Terminal System*. Este sistema consiste en: un dispositivo para situar un PC portátil, otro, para una impresora estándar, una zona para colocación de un CD-ROM, un inversor estático y una interfaz PC/impresora.

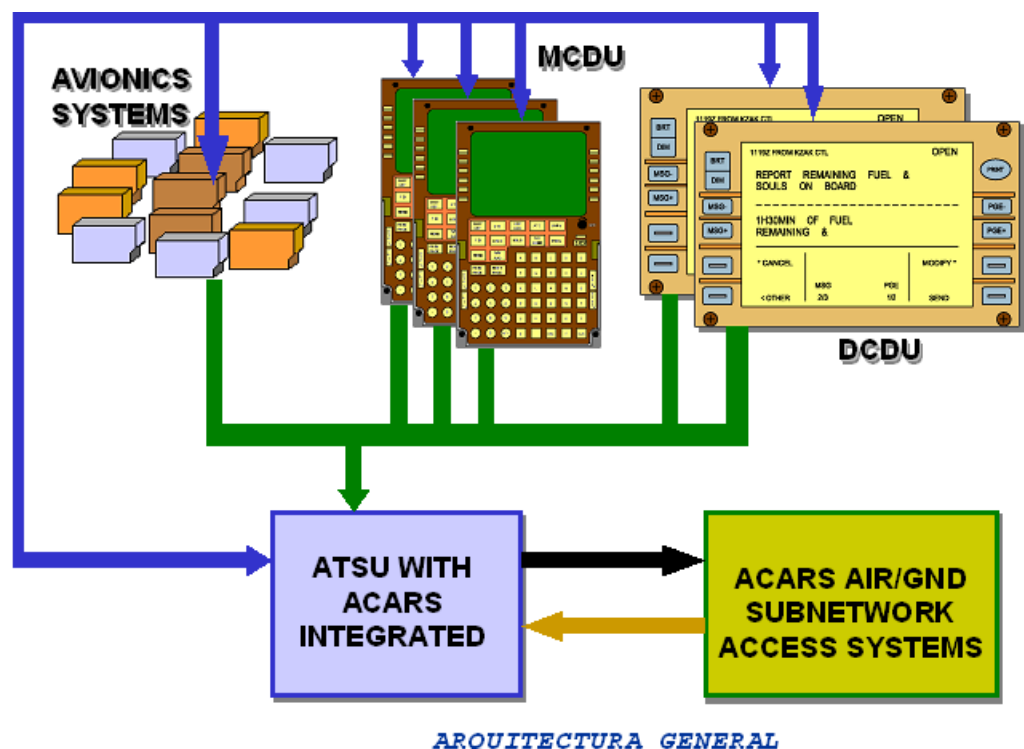
(4) El CINS proporciona a la aeronave una conexión de comunicación a alta velocidad basada en el estándar Ethernet. La tripulación de vuelo, la de cabina y el personal de mantenimiento pueden utilizar el CINS para conectar un portátil a la *Local Area Network (LAN)*

(5) Enlace de datos sin hilos interno: La wireless LAN de cabina, hace posible para los pasajeros y la tripulación de cabina conectar dispositivos electrónicos personales a la Ethernet de la aeronave.

(6) La red de información de pasajeros, permite a los pasajeros y a la tripulación de cabina tener acceso a Internet (Intranet) y servicios de e-mail durante el vuelo. Los datos digitales son enviados desde el servidor de Internet de abordó, a los asientos de los pasajeros, a través de un enlace de datos de alta velocidad.

17.9.2-ATIMS. DESCRIPCIÓN GENERAL.

El **ATIMS**, en conjunto con los **ATS** (*Air Traffic Services*), los **AIS** (*Airline Information Services*), y el **ATSU** (*Air Traffic Service Unit*) Router, forman parte del sistema de información del cockpit.



El ATIMS ha sido desarrollado para permitir comunicaciones mediante enlace de datos, e intercambio de datos complejos o informes específicos entre la aeronave y los centros de tierra. Permite funciones como:

Comunicaciones vía enlace de datos entre piloto y controlador **CPDLC** (*Controller-Pilot Datalink Communications*) (voz en HF en backup), para gestión de tráfico aéreo.

Intercambio de datos complejos o informes específicos entre la aeronave y los centros en tierra.

Informe automático (posición, intención) para vigilancia del tráfico aéreo.

Comunicaciones específicas entre la aerolínea y la aeronave (control operacional) para mejorar la flexibilidad y costes operacionales.

COMPONENTES.

El sistema está formado por una **ATSU** (*Air Traffic Service Unit*) situada en el compartimiento de aviónica, dos **DCDU-s** (*Datalink Control and Display Units*), dos **ATC MSG P/BSW** (Pushbutton Switch) con iluminación, y un **C/B RESET/ATSU1** para resetear la ATSU, situado en el overhead panel.

La ATSU es el componente principal del sistema. En esta unidad, a través de la MDDU (Multipurpose Disk Drive Unit), o del cargador portátil de datos, se carga el siguiente software para realizar funciones específicas:

-ATSU A/C INTERFACE software. Este software sirve para realizar funciones tales como la monitorización del sistema (alimentación y funciones BITE), gestión de las comunicaciones aire/tierra, gestión de la interfaz hombre/máquina (MCDU, DCDU, impresora y función de alerta), etc.

-ATSU software configuration. Este software consiste en un fichero de configuración de parámetros de comunicación ATC y **AOC** (*Airline Operational Control*)

-ATSU router parameter software. Este software contiene la base de datos del mapa del mundo DSP (Defined Service Provider)

-AOC software. Este software consiste en aplicaciones AOC que dependen de la definición de la aerolínea. Estas aplicaciones, afectan a operaciones relacionadas con el vuelo, como son los planes de vuelo, tiempo meteorológico, cantidad de combustible, gestión de personal, gestión de puertas, etc.

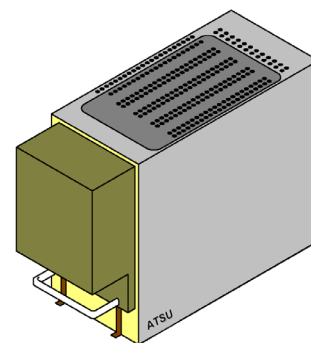
-ATC FANS A (*Air Traffic Control Future Air Navigation System A*) + application packages. Estos paquetes consisten en:

*software de utilidades ATC HMI (Human/Machine Interface), que proporciona servicios para el HMI de la aplicación ATC (DCDU, MCDU, impresora y alerta ATC)

*software de aplicación ATSU ATC FANS A, que contiene aplicación **AFN** (*ATS Facilities Notification*), aplicación CPDLC y **ADS** (Automatic Dependent Surveillance)

--El propósito de la aplicación AFN, es establecer el contacto con el centro ATC de tierra, y proporcionarle el número de registro de la aeronave y las aplicaciones de enlace de datos disponibles en la misma, con las correspondientes direcciones.

--El propósito de la aplicación CPDLC es proporcionar diálogo entre los controladores ATC y la tripulación de vuelo, utilizando para ello una comunicación vía enlace de datos, en lugar de comunicaciones por voz. Cada mensaje CPDLC está compuesto por un conjunto de elementos de mensaje, los cuales corresponden a la fraseología existente utilizada por los actuales procedimientos ATC.



Para los mensajes uplink (tierra-aeronave), la aplicación CPDLC se utiliza con el fin de mostrar los mensajes en las DCDU-s cuando son recibidos a bordo; gestionar el fichero de mensajes si se reciben varios mensajes, teniendo en cuenta la prioridad establecida; alertar a la tripulación de vuelo, a través del FWS (Flight Warning System) de la llegada de un mensaje; mostrar los mensajes en la MCDU cuando un mensaje es demasiado largo para ser mostrado en las DCDU-s, (por ejemplo el plan de vuelo); transmitir la respuesta de la tripulación de vuelo desde las DCDU-s a tierra; cargar los mensajes en el sistema de aviónicos de la aeronave (por ejemplo, el plan de vuelo en el FMS (Flight Management System)).

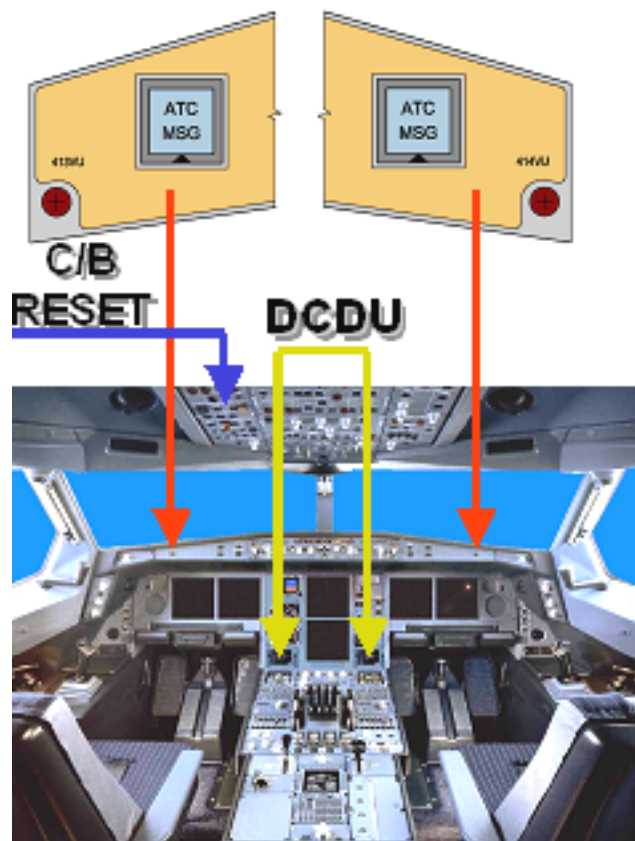
Para los mensajes downlink (aeronave-tierra), la aplicación se utiliza para gestión de la MCDU en la preparación de mensajes, mostrar los mensajes destinados a tierra en la DCDU antes de la transmisión con el fin de que puedan ser revisados por la tripulación de vuelo.

--El propósito de la función ADS es proporcionar al centro ATC de tierra datos de vigilancia de la aeronave (identificación de la aeronave, posición), mediante informes específicos determinados por un protocolo ADS.

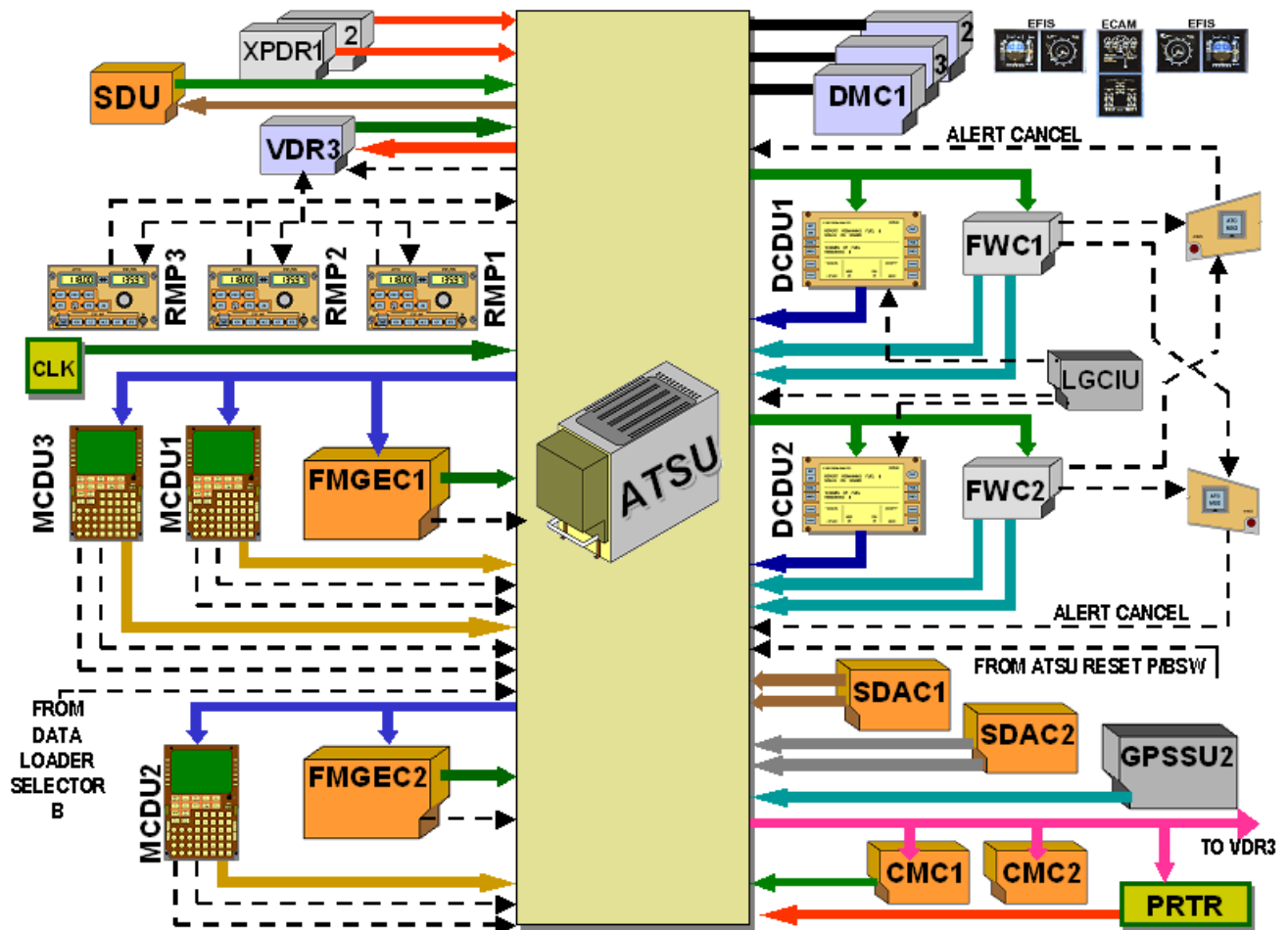
Las DCDU-s proporcionan la interfaz para las aplicaciones ATC. Proporcionan a la tripulación de vuelo capacidad de control y display, permitiendo que se envíen mensajes de respuesta al centro de tierra, y que se muestren los mensajes recibidos desde el centro ATC de tierra. Las DCDU-s son gestionadas por la ATSU, la cual procesa y organiza por un lado, los datos en páginas para ser mostrados por pantalla, y por otro lado, traduce los códigos de tecla recibidos en órdenes de la tripulación.

Los ATC MSG P/BSW-s proporcionan a la tripulación de vuelo, alerta visual en caso de recepción de un mensaje ATC. Esta alerta se cancela presionando el P/BSW. Además, se produce una alerta auditiva vía los FWC-s (simple o repetitiva, dependiendo de la prioridad del mensaje)

Las MCDU-s son utilizadas para comunicaciones AOC aire/tierra y aplicaciones ATC necesitadas además de las DCDU-s. Las funciones que realizan son: preparación y modificación de todos los mensajes preparados por la tripulación de vuelo, justificaciones a respuestas negativas o a peticiones particulares, edición de texto, configuración de aplicaciones (activación ADS, inicialización AFN) y configuración de sistemas (gestión ATSU, impresión automática o bajo pedido).



ARQUITECTURA.



El ATSU interactúa con las siguientes unidades periféricas:

- (1) La Datalink Control and Display Unit 1 and 2 (**DCDU 1 & 2**).
- (2) El ATSU RESET pushbutton switch.
- (3) Los Flight Management Guidance and Envelope Computers 1 and 2 (**FMGEC**).
- (4) Las Multipurpose Control and Display Units (**MCDU**).
- (5) El VHF Data Radio 3 (**VDR3**) transceiver.
- (6) El SATCOM Satellite Data Unit (**SDU**).
- (7) Los Radio Management Panels 1, 2 and 3 (**RMP**).
- (8) El Clock.
- (9) Los Flight Warning Computers 1 and 2 (**FWC**).
- (10) Los System Data Acquisition Concentrators 1 and 2 (**SDAC**).
- (11) Los Display Management Computers (**DMC**).
- (12) La Multipurpose Disk Drive Unit (**MDDU**).
- (13) Los Central Maintenance Computers 1 and 2 (**CMC**).
- (14) La impresora.

Opcionalmente, puede interactuar con las siguientes unidades periféricas:

- (15) Los ATC transponders.
- (16) El High Frequency Data Radio 1 (**HFDR1**) transceptor.
- (17) Los Cabin Terminals.
- (18) La Data Management Unit (**DMU**).

INTERFAZ CON OTROS SISTEMAS.

El ATSU está interrelacionado con el *Flight Management System (FMS)* para llevar a cabo las siguientes funciones:

- Adquisición por parte de la ATSU de datos y parámetros de la aeronave para aplicaciones AOC (aeropuerto de origen/destino, número de vuelo, fuel a bordo, etc.), también para funciones de comunicación aire / tierra y gestión de sistemas (número de vuelo, FMS master / ave, etc.)
- Adquisición por parte de la ATSU de ficheros de datos de navegación y de gestión de vuelo para la aplicación ADS.
- Intercambio de mensajes ARINC 619 para la aplicación AOC remota. (Para esta aplicación, el ATSU actúa como un router).
- Intercambio de mensajes ARINC 619 para la aplicación CPDLC.
- Adquisición por parte del FMS, de información de estado del enlace de datos para las aplicaciones AOC y ATC.

El ATSU está interrelacionado con los siguientes elementos del Sistema de comunicaciones:

- VDR3 para comunicaciones de datos vía VHF y control de modo voz / datos en VHF. El ATSU utiliza los servicios proporcionados por VDR3 para comunicarse con tierra en modo de voz o datos.
- RMP 1/2/3 para conmutación en VDR3 de modo voz / datos. Se utiliza un discreto de salida de la RMP, para solicitar el modo de conmutación a la ATSU. Y se utiliza un discreto de salida de la ATSU para indicar a la RMP que sistema, ATSU o RMP está controlando el VDR3.
- SDU1 para comunicación de datos vía satélite. El ATSU utiliza el servicio proporcionado por SDU1 para comunicar con tierra en modo datos. En este modo, la SDU gestiona por sí misma la sintonización de su frecuencia y conexión con la red de tierra. El ATSU no controla el modo de operación de la SDU.
- HFDR1 para comunicaciones de datos vía HF. El ATSU utiliza el servicio proporcionado por HFDR1 para comunicar con tierra en modo datos. En este modo, la HFDR1 gestiona por sí misma la sintonización de su frecuencia y conexión con la red de tierra.

ATSU / *Flight Warning System (FWS)*. Los Flight Warning Computers (FWC-s), son utilizados para generar indicaciones de avisos a la tripulación: Alertas ATC (Cuando se recibe un mensaje CPDLC, los FWC-s generan alertas ATC sonoras y visuales); Avisos de la aerolínea; alarmas del sistema de enlace de datos;

ATSU / *Central Maintenance System (CMS)*.

- La ATSU está conectada con los *Central Maintenance Computers (CMC-s)* para propósitos de mantenimiento El sistema BITE está asegurado por la ATSU.

-La ATSU, envía el número de software y P/N de cada componente de ATIMS a los CMC-s.

-Aplicación CMC / AOC. La ATSU envía a los CMC-s que tipo de informe se requiere en la transmisión automática por medio de la programación de la función activa CMC: informe BITE, informe de aviso en tiempo real, informe de fallo en tiempo real, *Post Flight Report (PFR)*.

-Parámetros de configuración de la aeronave. La ATSU toma parámetros de configuración de la aeronave procedentes de los pines programables del CMC, para saber si están instalados en la aeronave, ciertos sistemas opcionales (SDU, HFDR1, Cabin terminals, DMU).

-Identificación de la aeronave. La ATSU toma el *Aircraft Registration Number (ARN)* y el número de vuelo (obligatorio para las funciones de comunicación aire / tierra) de los CMC-s.

-Parámetros AOC. Los CMC-s proporcionan a la ATSU parámetros AOC, que son necesitados por la aplicación AOC (para los informes).

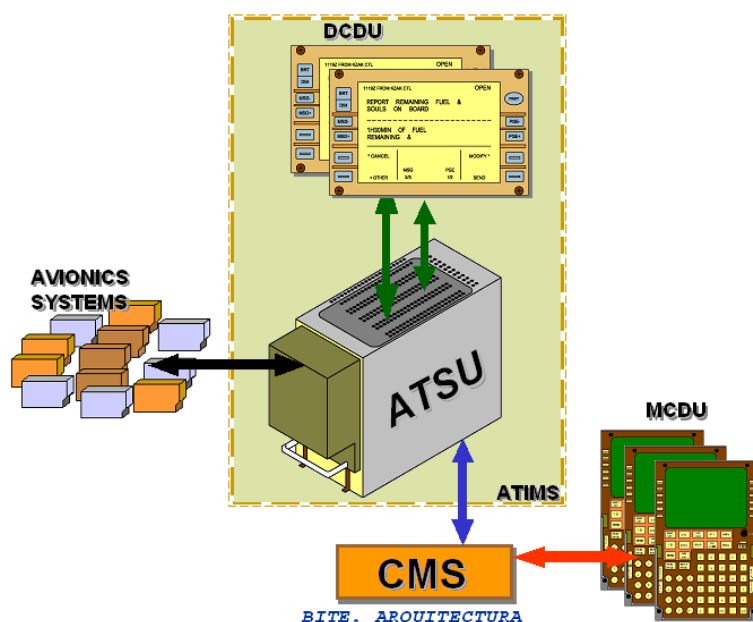
- Estatus de la comunicación por enlace de datos. La ATSU envía información de estado a los CMC-s concerniente a la disponibilidad de comunicación por enlace de datos con tierra y entre la ATSU y el CMC.

Landing Gear /ATIMS. La ATSU y ambas DCDU-s reciben información de la *Landing Gear Control Interface Unit (LGCIU)* para determinar si la aeronave está en tierra o en vuelo. Esta información corresponde a “Nose Landing Gear comprimido o no”. La información es enviada por la LGCIU por medio de tres discretos. (GND= aeronave en tierra; OPEN= aeronave en vuelo)

17.9.3-TEST BITE.

El BITE del ATIMS, está asegurado por la ATSU, la cual concentra la información de fallos proporcionada por la misma ATSU y la monitorización interna de la DCDU. El BITE, detecta, identifica y memoriza fallos internos y externos relativos al sistema, tales como: fallos internos de la ATSU, fallos internos de las DCDU-s, fallos de la interfaz entre ATSU y DCDU-s y fallos externos de la interfaz entre ATSU y sus periféricos.

Este BITE opera en los modos normal e interactivo.



MODO NORMAL. En este modo, el BITE monitoriza el estado de la ATSU y DCDU; monitoriza también las entradas de datos de varios periféricos del ATIMS (FMGEC, MCDU, CMC, etc.); transmite permanentemente el estado del ATIMS y mensaje de su identificación al CMC.

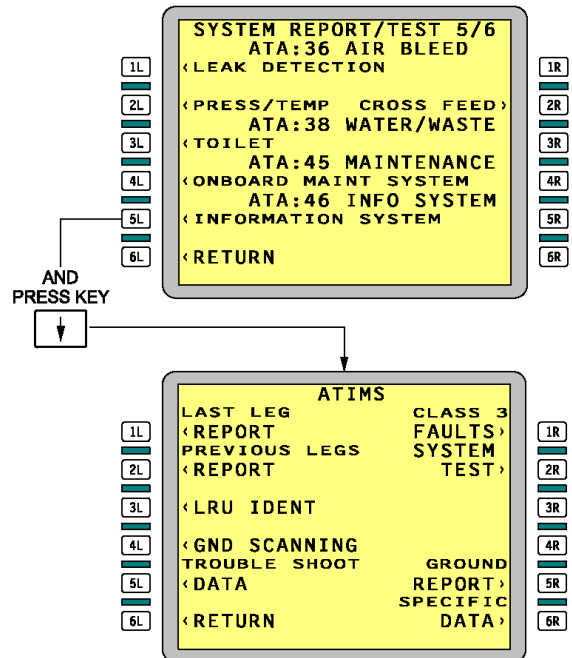
En caso de detección de fallo, el BITE guarda la información en la memoria de fallos y la transmite al CMC vía ARINC 429 (etiqueta 356).

MODO INTERACTIVO. En este modo de operación, el BITE se realiza a través de la MCDU, navegando a través de las diferentes páginas menú dedicadas al ATIMS:

-Función SYSTEM REPORT / TEST.

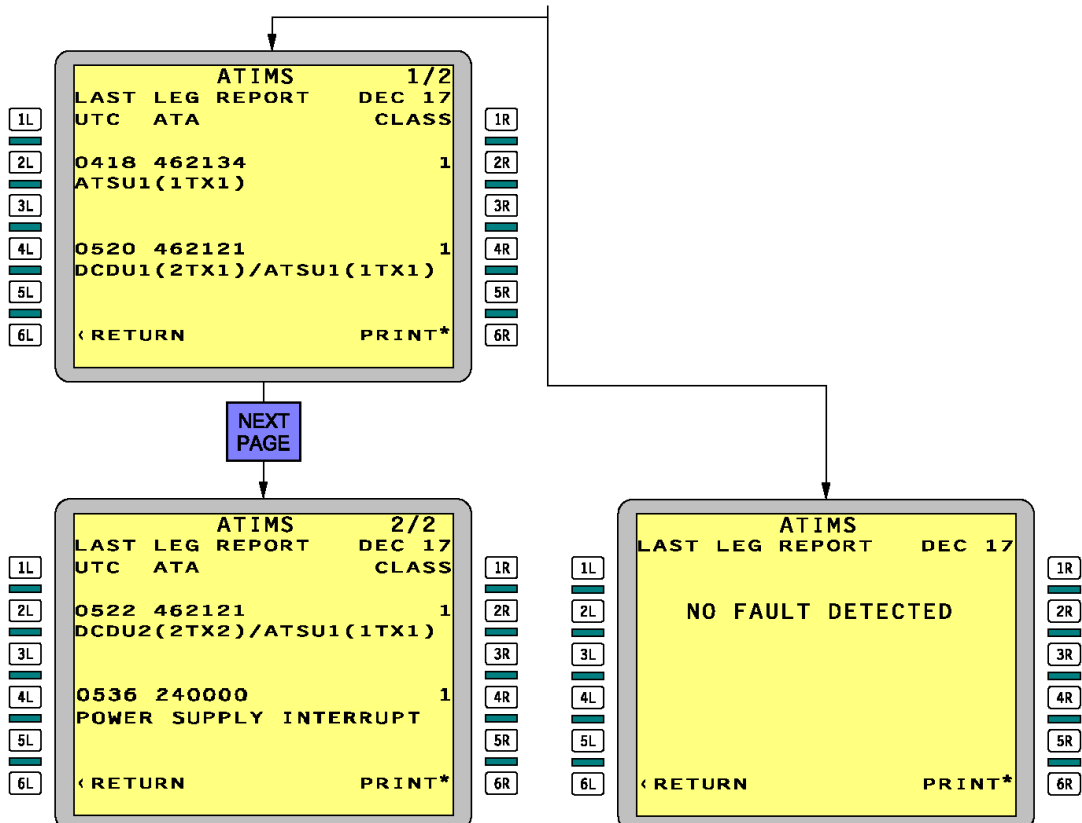
La información BITE (system report) y la petición de test (system test function) están disponibles a través de los menús de la MCDU, la cual permite comunicación con el BITE del ATIMS a través del CMC.

El menú de mantenimiento del ATIMS solamente es accesible en tierra desde el menú de mantenimiento general y la página de SYSTEM REPORT / TEST. Toda la información mostrada en la MCDU durante la configuración del test BITE, puede ser impresa por la impresora.

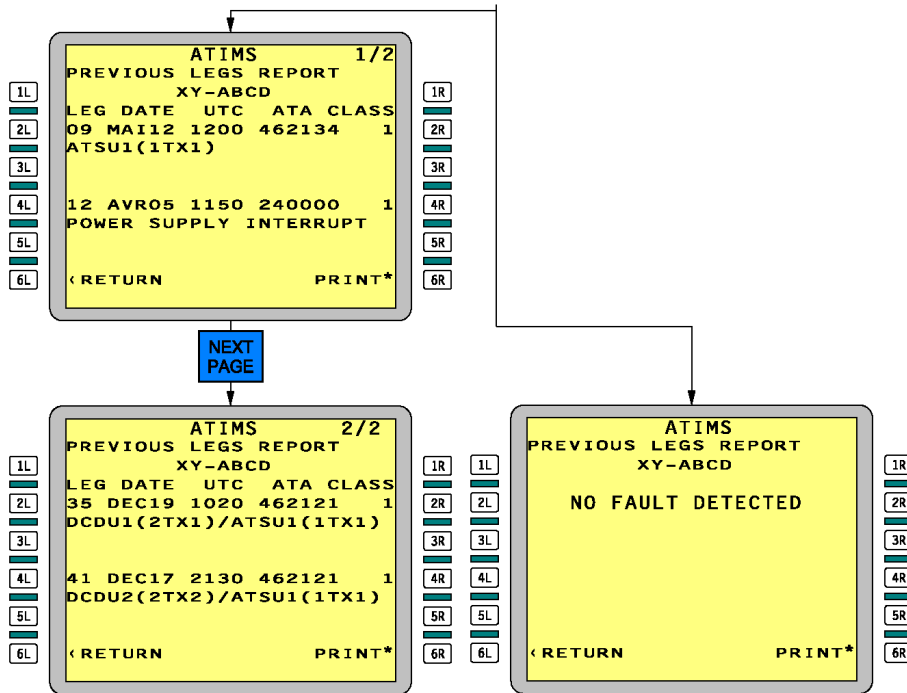


Página LAST LEG REPORT (1L)

Este informe contiene los mensajes de fallo (internos y externos, clase 1 y clase 2) grabados durante el último vuelo.



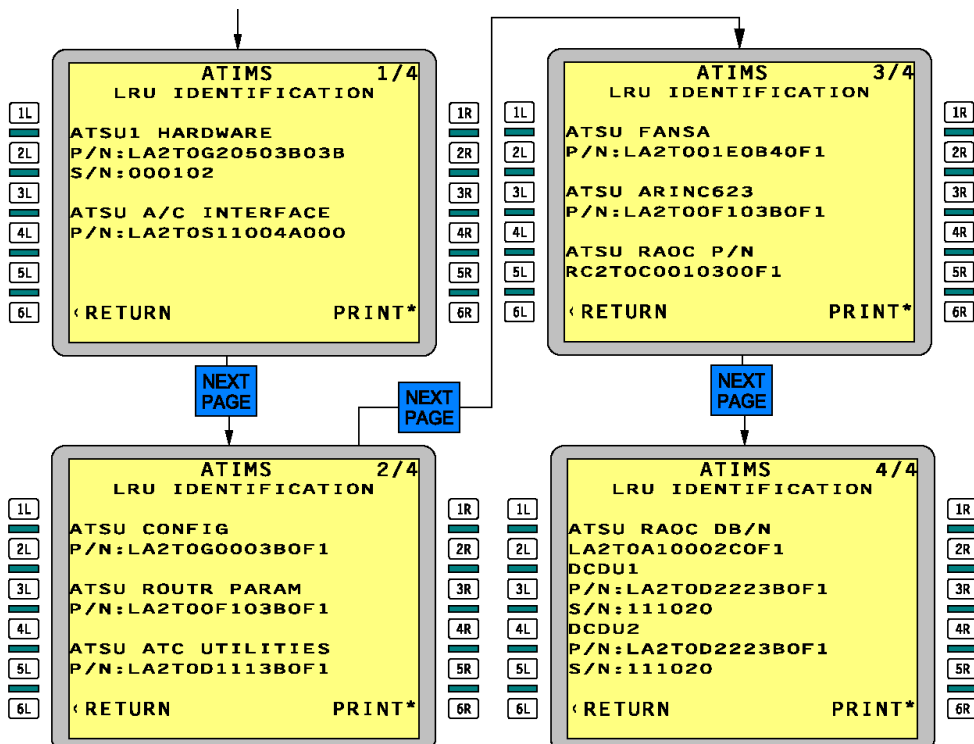
Página *PREVIOUS LEGS REPORT* (2L).

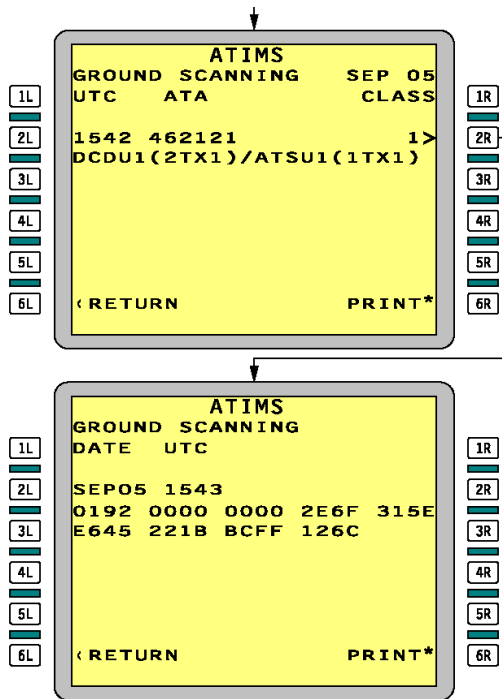


Este informe contiene los mensajes de fallo (internos y externos, clase 1 y clase 2) grabados durante los 63 vuelos anteriores.

Página *LRU IDENTIFICATION* (3L).

Este menú da la identificación de diferentes componentes del ATIMS: ATSU, DCDU1, DCDU2, software y bases de datos asociadas. La información que se presenta es la siguiente: P/N y S/N de la ATSU; P/N del software del ATSU A/C INTERFACE; P/N del software de configuración del ATSU; P/N del software de parámetros del router ATSU; P/N del software de las utilidades ATC HMI; P/N del software de las aplicaciones del ATSU ATC FANS A; P/N del software de las aplicaciones ATSU ATC ARINC 623; P/N del software de la aplicación AOC; P/N del software de la base de datos de AOC; P/N y S/N de la DCDU1 y DCDU2.



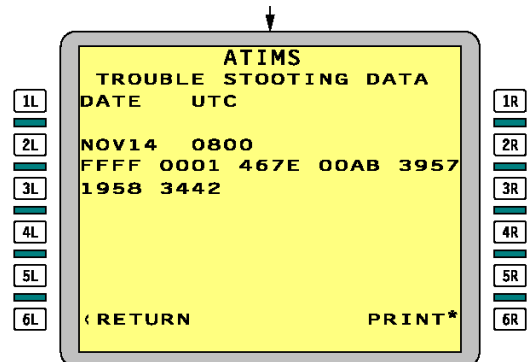


Página *GROUND SCANNING* (4L).

Esta página se basa en la monitorización y análisis de fallo durante el vuelo, y permite la consulta de las grabaciones de fallo del ATMS. La monitorización de periféricos del ATSU y los tests cíclicos internos son utilizados para detectar fallos transitorios.

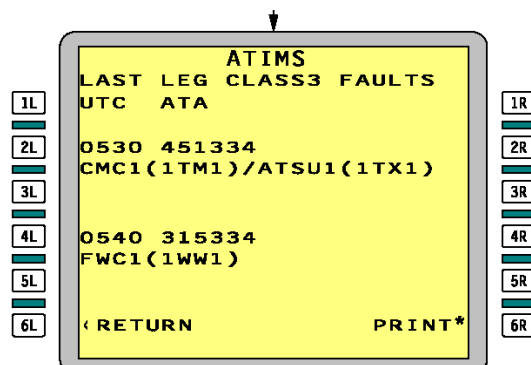
Página *TROUBLE SHOOTING DATA* (5L).

Esta función proporciona parámetros de correlación y datos rápidos relativos al fallo mostrado en la página de LAST LEG REPORT y la página PREVIOUS LEGS REPORT.



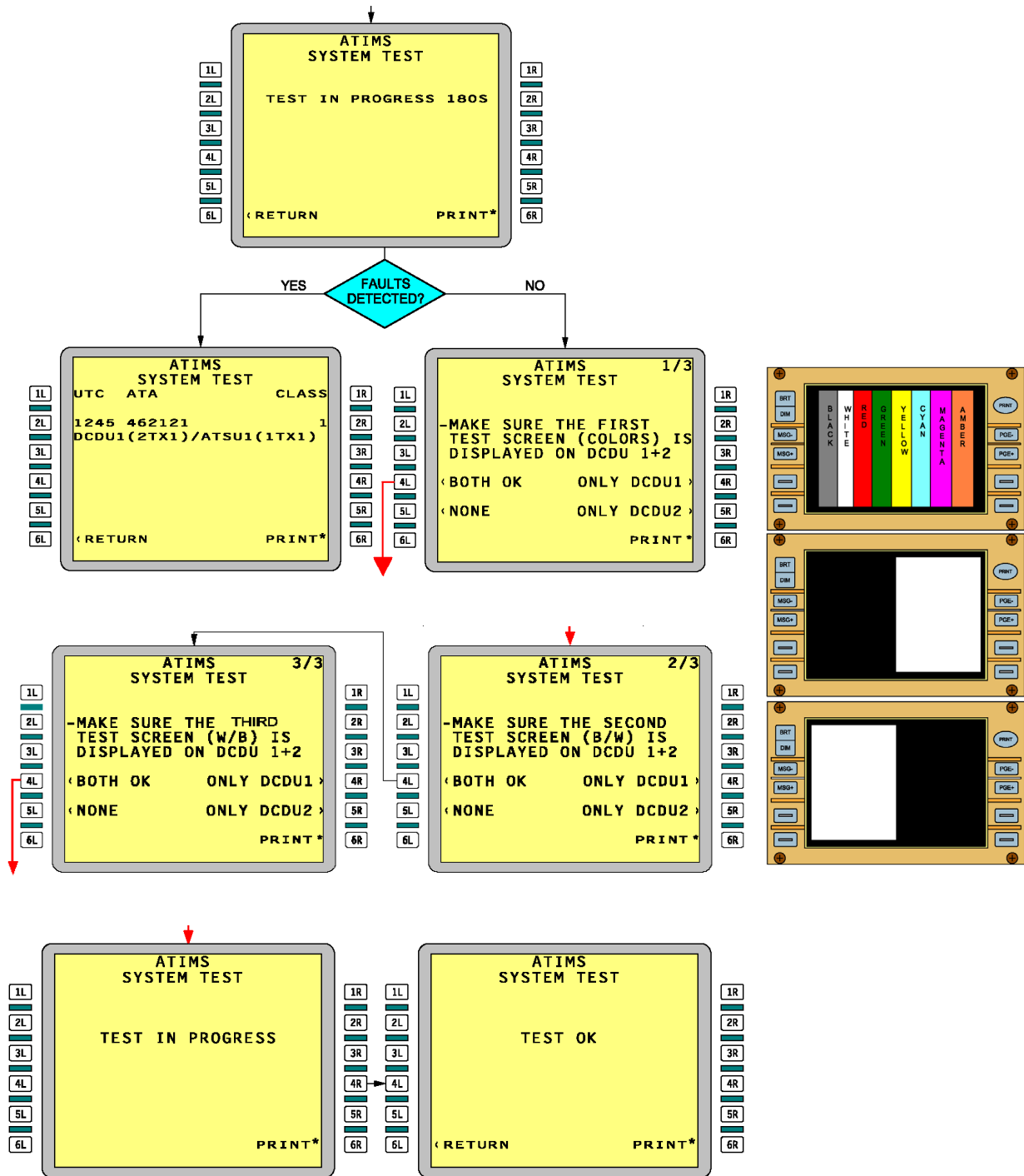
Página *CLASS 3 FAULTS* (1R).

Este menú permite mostrar los fallos clase 3 grabados durante el último vuelo.



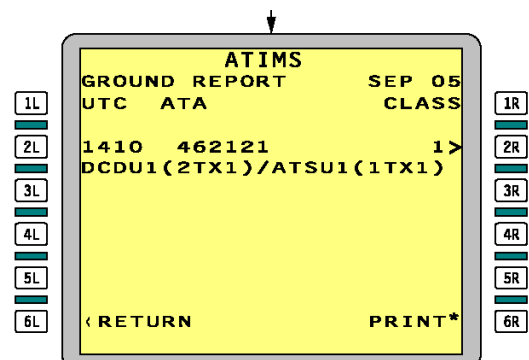
Página *SYSTEM TEST* (2R).

El test del ATIMS BITE se inicia cuando se presiona la tecla 2R, y finaliza, bien, con la presentación en la pantalla de la MCDU de la indicación de "TEST OK" cuando todos los tests han sido completados y no se ha detectado ningún fallo, o bien con los mensajes de fallo cuando se han detectado estos.



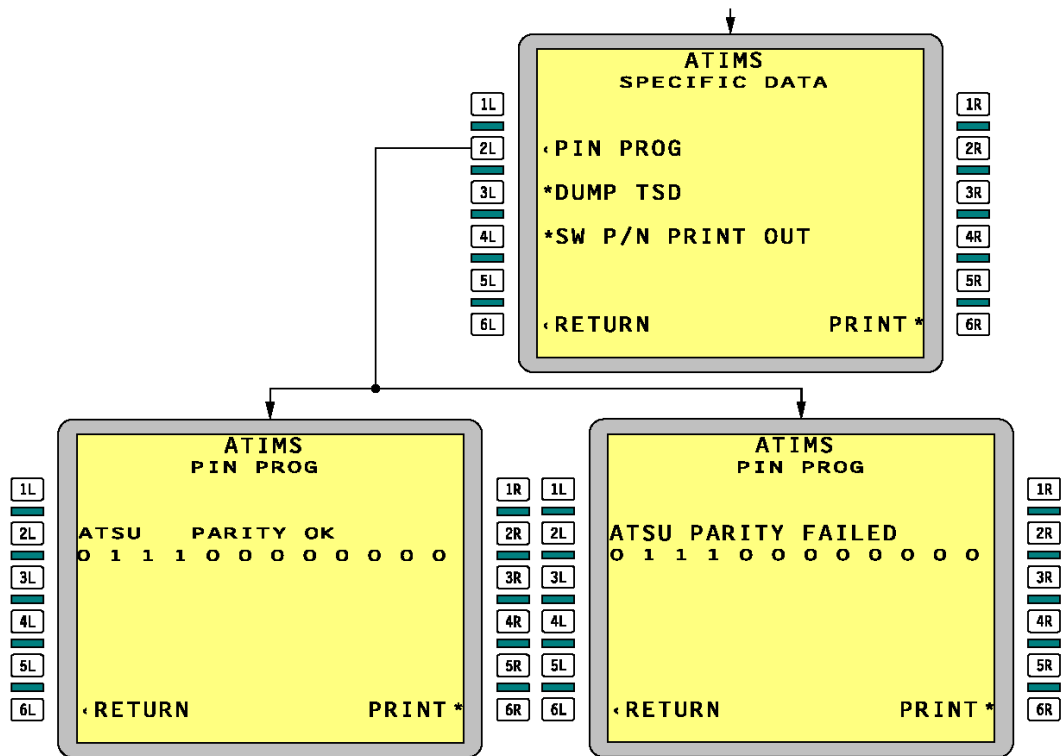
Página GROUND REPORT (5R).

Esta página se utiliza para presentar los fallos internos clase 1, 2 o 3 cuando estos son detectados en tierra. Los datos relevantes de trouble shooting son mostrados presionando la tecla adyacente a la indicación de fallo correspondiente. Estos fallos son diferentes a los mostrados en la página LAST LEG REPORT.



Página *SPECIFIC DATA* (6R)

Este menú da acceso a diferentes funciones: (2L) PIN PROGRAMMING para chequear la configuración de la ATSU con la validez de su paridad (El orden en el que se muestra el PIN PROGRAMMING, está de acuerdo con el del conector de entrada de la ATSU. Un pin programable no definido, (repuesto) es mostrado con valor 0. Un pin activo, se muestra con valor 1). (3L) DUMP TSD. (4L) SW P/N PRINT OUT.



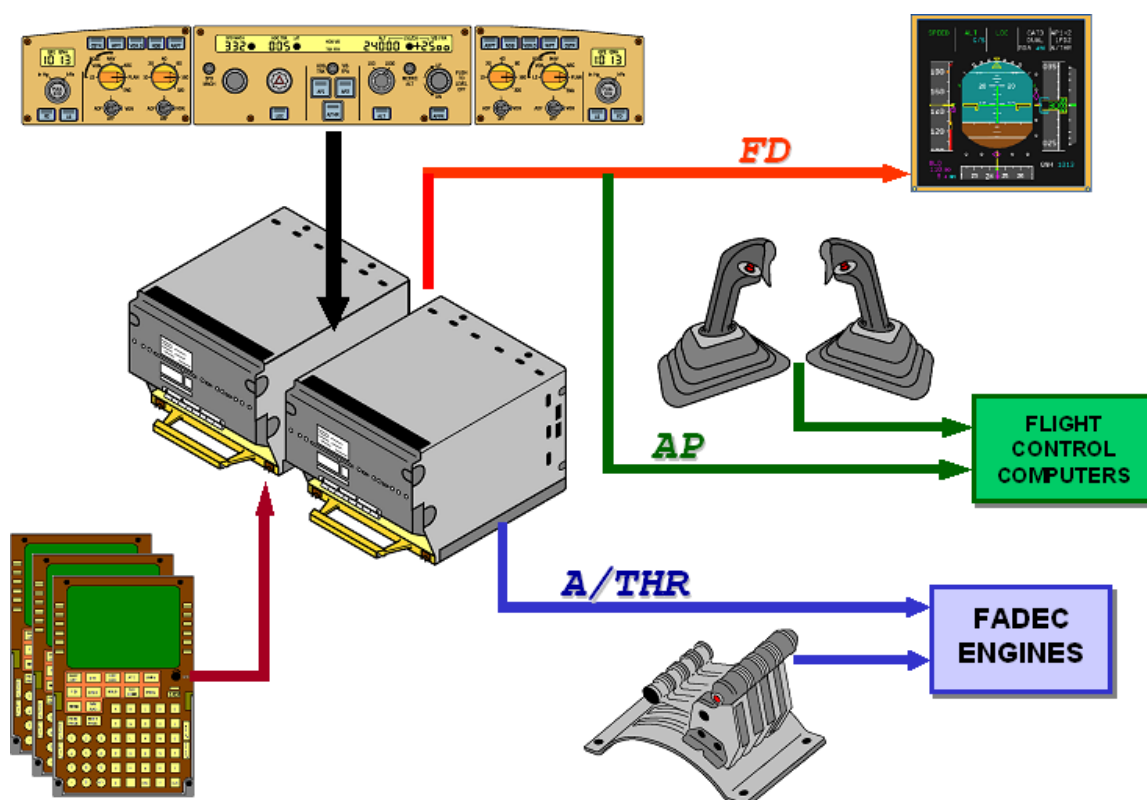
17.10-FMS (FLIGHT MANAGEMENT SYSTEM).

17.10.1-INTRODUCCIÓN. AFS

El **FMS** es una función integrada dentro de los **FMGEC-s** (*Flight Management, Guidance and Envelope Computers*), los cuales, a su vez, forman parte, junto con la **FCU** (*Flight Control Unit*) y las **MCDU-s** (*Multipurpose Control and Display Units*), del **AFS** (*Auto Flight System*).

El AFS consiste en dos FMGEC-s, tres MCDU-s y la FCU.

PRINCIPIOS BÁSICOS DE OPERACIÓN DEL AFS.



PRINCIPIOS BÁSICOS DE OPERACIÓN

El piloto introduce al sistema órdenes simples, de resolución a corto plazo, a través de la FCU y órdenes complejas, de resolución a medio/largo plazo, a través de las MCDU-s.

GENERAL. El control de la aeronave es automático (AUTOMATIC) o manual (MANUAL). Es automático cuando se realiza a través del AP o del A/THR. Es manual cuando el piloto acciona el side stick o las palancas de empuje.

El guiado de la aeronave es gestionado (MANAGED) o seleccionado (SELECTED). Es gestionado cuando la aeronave se controla utilizando parámetros de referencia manejados por el FMGEC, en concreto la función FM. Esta función tiene en cuenta los datos seleccionados por el piloto en la MCDU. Es seleccionado cuando el piloto selecciona parámetros de referencia a través de la FCU.

POWER-UP TEST / FD ENGAGEMENT. Tan pronto como esté disponible la energía eléctrica, el FD se engancha automáticamente, siempre y cuando el test de energización (power-up test) esté realizado. En el PFD no aparece ningún símbolo de guiado hasta que esté presente un modo seleccionado de fase activa.

MCDU INICIALIZATION. Primero, la página de status se muestra en la MCDU, entonces el piloto utiliza la MCDU para la preparación del vuelo lo cual incluye: la elección de la base de datos, inicialización de plan de vuelo, entradas y chequeos de las R-NAV, introducción de datos de performance V_1 V_R V_2 y FLEX temp. Antes del despegue, al menos la V_2 , debe haber sido introducida en la MCDU. La introducción del plan de vuelo (lateral y vertical) y la V_2 en la MCDU, es tomada en cuenta por la función FM y confirmada por la iluminación de las luces asociadas en la FCU.

A/THR ENGAGEMENT. El acoplamiento del A/THR, depende del movimiento de las palancas de empuje para el despegue. Cuando el piloto mueve estas palancas para el despegue:

- el FMGEC automáticamente acopla (1) los modos de despegue para el yaw y el guiado longitudinal [**SRS** (Speed Reference System) y **RWY** (RunWay)], (2) la función de autothrust (pero no la activa, solo la arma).
- los símbolos del FD aparecen en el PFD.

Para el despegue, las palancas de empuje son situadas en TO /GA o en MCT/FLX TO (Maximum Continuous Thrust /Flexible takeoff).

A la altitud de reducción de empuje, la parte de FM avisa al piloto para situar las palancas de empuje en la posición CLB, momento en que se activa el modo correspondiente del A/THR. Las palancas de empuje, normalmente se quedarán en esta posición hasta que el mensaje de audio "RETARD" pida al piloto situarlas en IDLE, antes del aterrizaje, desacoplando el sistema del A/THR.

AP ENGAGEMENT. Cualquier AP puede acoplarse 5 segundos después de la carrea de despegue una vez alcanzada la V_2 . Solamente un AP puede estar acoplado en un momento dado. El último que se acopla, es el que permanece. No obstante, después de las fases de ascenso normal, crucero y descenso, la selección del modo LAND (Autoland) permite al segundo AP acoplamiento en modo DUAL.

Después del aterrizaje, durante el modo de ROLL OUT, los AP-s permanecen enganchados para controlar la aeronave en la línea central de pista. Después, a baja velocidad, el piloto desengancha el AP y detiene la aeronave.

ARQUITECTURA GENERAL.

*Cada FMGEC se compone de cuatro secciones funcionalmente independientes. Estas secciones son:

- función de gestión de vuelo, **FM** (*Flight Management*): Navegación (lateral y vertical), procesamiento de actuación y predicciones, procesamiento de los displays y enlace de datos de comunicación / vigilancia
- función de guiado de vuelo, **FG** (*Flight Guidance*): *Autopilot (AP)*, *Flight Director (FD)* y *Autothrust (A/THR)*.
El AP procesa órdenes para la deflexión de superficies de control. El A/THR procesa la orden de empuje para el control de motores. El FD muestra al piloto las órdenes que deberían ser seguidas por el AP si este estuviera enganchado. Los parámetros de referencia, tales como la velocidad, rumbo, velocidad vertical y altitud, pueden ser seleccionados de forma manual por la tripulación, o de forma automática, por el sistema.
- función de entorno de vuelo, **FE** (*Flight Envelope*): procesamiento de la configuración de vuelo, detección de Windshear.

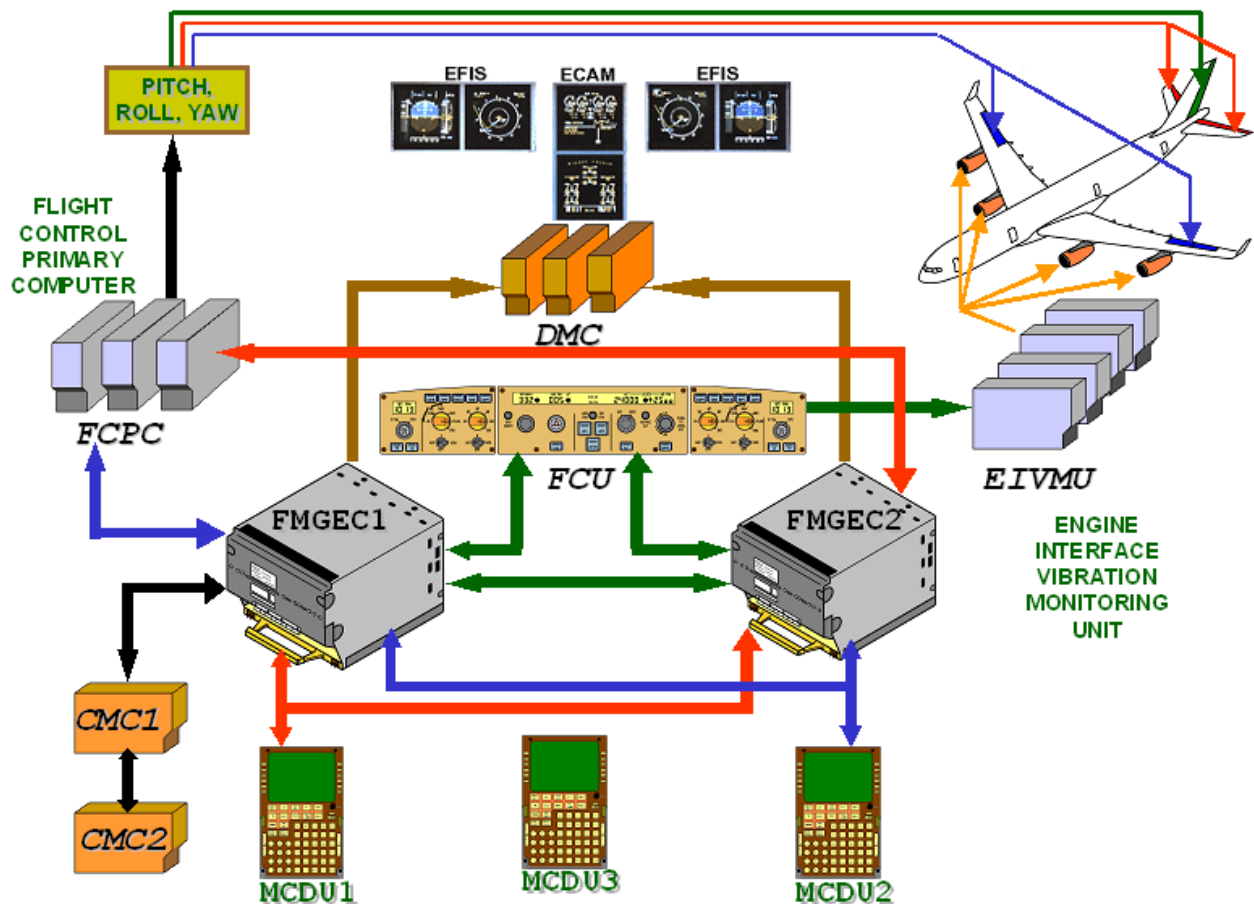
La función de FE adquiere y monitoriza parámetros, utilizados por la protección de FE, la función FM y la función FG, tales como datos del ADIRS, datos de navegación, datos de fuel, datos de la FCU, etc.

Entorno de velocidad, consolidación y monitorización de las fases de vuelo y configuración de la aeronave (posición del tren de aterrizaje, posición de flaps y slats, status de los motores, etc.), detección de windshear y detección de la posición del centro de gravedad son funciones de protección de FE.

- función de detección y aislamiento de fallos –*Fault Isolation and Detection System (FIDS)*–

La FIDS centraliza la información de fallos del AFS procedentes de los FMGEC-s, MCDU-s y FCU. El sistema envía los resultados, bajo petición, a los **CMC-s** (*Centralized Maintenance Computers*). El FIDS es totalmente independiente de otras partes del FMGEC, y está instalado solamente en el FMGEC1.

La interrogación de la función FIDS, se realiza seleccionando el control apropiado en la MCDU.



*Las tres MCDU-s conectadas a los FMGEC-s proporcionan la interfaz entre la tripulación y la función de FM (La MCDU3 es utilizada como back-up en vuelo, para cubrir el caso de fallo de cualquiera de las otras dos MCDU-s y para propósitos de mantenimiento en tierra). Esta interfaz permite:

- introducción y modificación del plan de vuelo.
- presentación, selección y modificación de los parámetros asociados con las funciones de gestión de vuelo

*La FCU se compone de tres paneles de control: un panel de control para el AFS, situado entre dos paneles de control para el EFIS. Esta unidad se utiliza para:

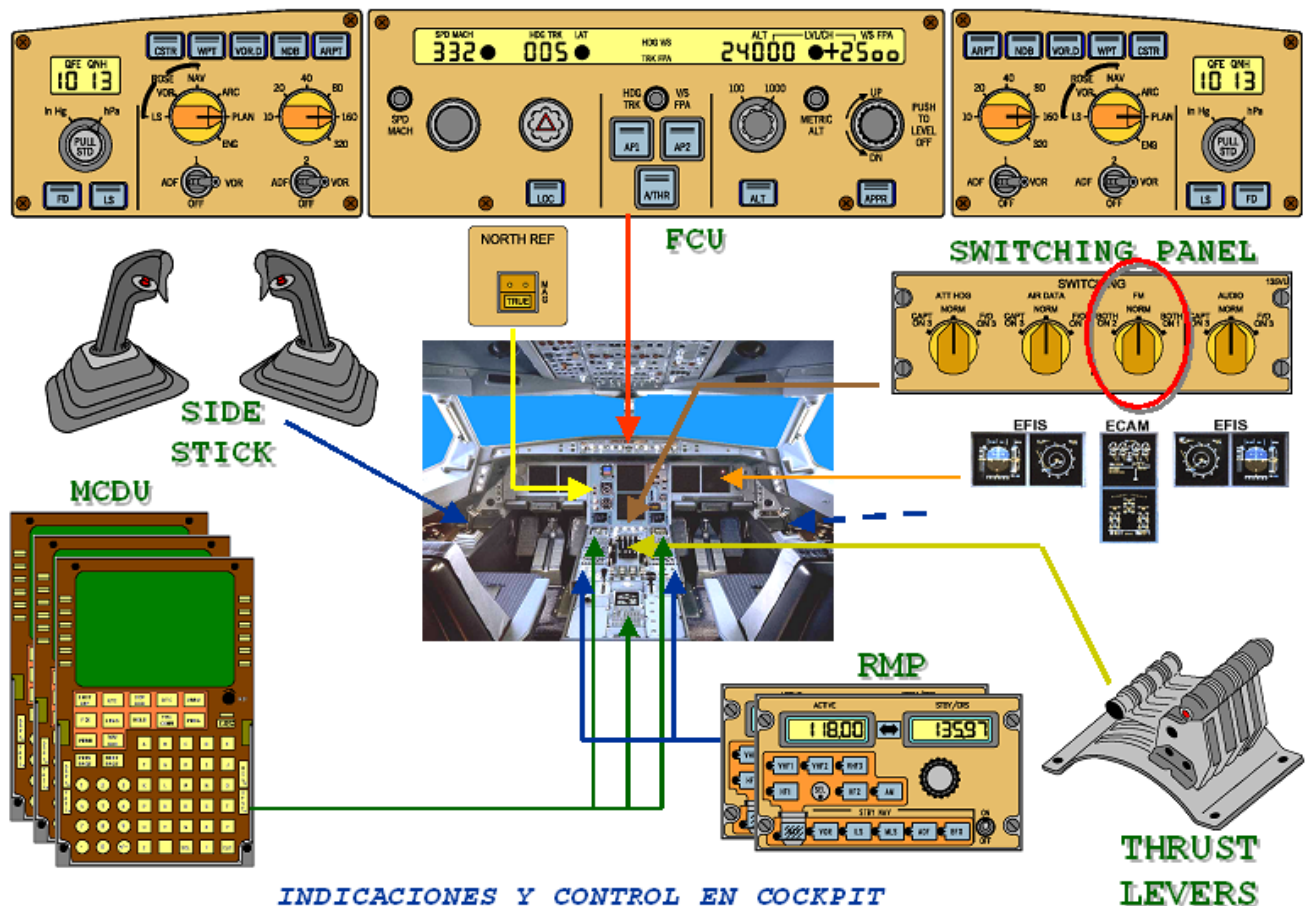
- enganchado / desenganchado de los sistemas AP/FD y del A/THR.
- selección de parámetros de vuelo (altitude, speed/Mach, vertical speed/flight path angle, heading/track).
- selección de los modos de AP/FD.
- selección de modos de presentación en el EIS.
- función de interfaz requerida para la transmisión de datos de motores desde los FMGEC-s a las EIVMU-s.

Los FMGEC-s envían órdenes de deflexión de superficies de control para la función del AP por medio de los **FCPC-s** (*Flight Control Primary Computers*) 1, 2 y 3.

También envían orden de empuje para la función A/THR a través de la FCU a las **ECU1/EEC1** (*Engine Control Unit 1/ Electronic Engine Control1*), vía **EIVMU1** (*Engine Interface Vibration Monitoring Unit 1*), **ECU2/EEC2** vía **EIVMU2**, **ECU3/EEC3** vía **EIVMU3**, **ECU4/EEC4** vía **EIVMU4**, para establecer la orden de empuje en los respectivos motores.

Los datos del AFS son mostrados en los PFD-s y ND-s a través de los DMC-s. Los ND-s son más específicos para la parte de FM y los PFD-s, son más específicos para las partes de FG y FE.

INDICACIÓN Y CONTROL DEL AFS.



switches) situados en los side stick, A/THR instinctive disconnect P/BSW-s situados en las palancas de empuje, North Ref P/BSW, SWITCHING/FM selector switch situado en el switching panel.

Las indicaciones se obtienen vía PFD-s, ND-s, EWD, y SD, y los warning luminosos MASTER WARN, MASTER CAUT, AUTO LAND Y DATA LINK ATTENTION GETTER.

FCU: El panel de control para el AFS, situado entre los dos paneles de control para el EFIS, permite y muestra el enganchado de AP y A/THR y permite la selección de modos de guiado y parámetros de vuelo. Los dos paneles de control para el EFIS controlan y muestran para cada lado EFIS (Capt y F/O) las funciones de PFD y de ND.

MCDU-s: Aunque la MCDU está dedicada a diferentes sistemas, su principal función es permitir al piloto comunicarse con el FMGEC. La MCDU es la principal interfaz de entrada /display entre el piloto y la parte de FM del FMGEC. Permite la entrada de parámetros para el control del sistema y planes de vuelo, y es también utilizada para realizar pertinentes revisiones y modificaciones. Muestra en su pantalla información concerniente a la progresión del vuelo y actuaciones de la aeronave, para revisión y monitorización por parte de la tripulación de vuelo.

EFIS: Cada ND se forma a partir de datos del plan de vuelo, datos seleccionados a través de la FCU, posición actual de la aeronave, dirección y velocidad del viento, velocidad respecto a tierra y ruta. Cada PFD muestra en el **FMA** (*Flight Mode Annunciator*) que es la zona superior del PFD que abarca tres líneas de texto, el status de enganche del AP/FD, A/THR, también se muestra en esta zona, el estado de armado/enganchado de los modos del AP/FD y A/THR, además de estado de la aproximación seleccionada. Fuera del FMA se muestran órdenes del FD y velocidades características de FE en la escala de velocidad.

THRUST LEVERS: Permiten que los modos de **TO/GA** (*Take-Off / Go Around*) y A/THR sean enganchados. Dos botones situados en los laterales de las palancas de empuje (A/THR instinctive disconnect pushbuttons) permiten que la función A/THR sea desenganchada.

SIDE STICKS: El AP se desengancha cuando se presiona el botón de prioridad de mando situado en cada side stick, o cuando se aplica una fuerza por encima de un cierto nivel en el propio side stick. (El AP se puede desconectar a través de los pedales del timón de dirección (Rudder pedals))

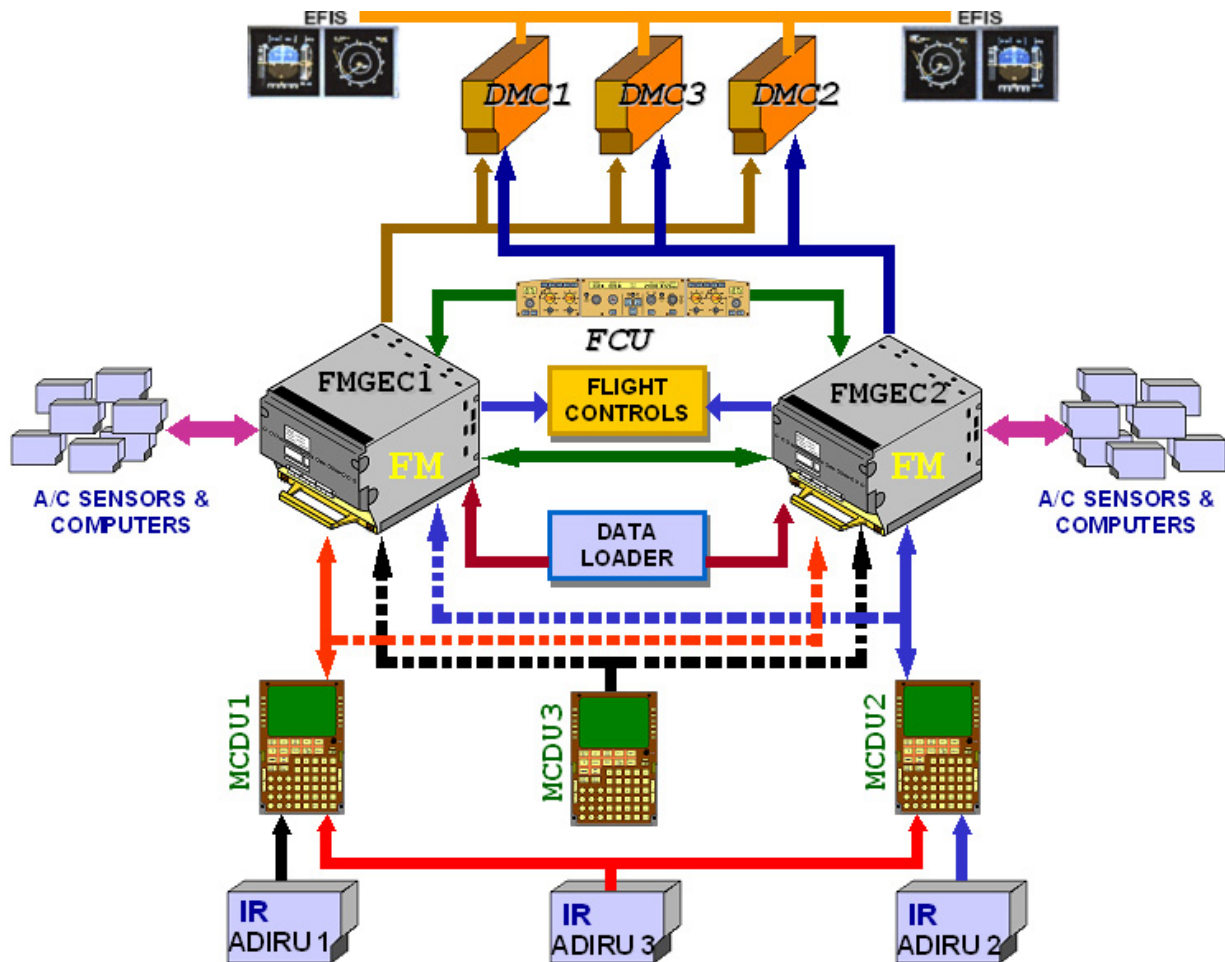
NORTH REF. SWITCH: Mediante este switch se selecciona la referencia de rumbo magnético o verdadero (magnetic/true). Cuando se alcanza 81°N o 60°S, mensajes en las MCDU-s y EFIS, avisan a la tripulación de que deben seleccionar la referencia verdadera. Si el piloto olvida conmutar la referencia, el AP se desconecta.

SWITCHING PANEL: El tercer botón, comenzando desde la izquierda, corresponde al FM Source Selector. Este botón tiene tres posiciones: NORM, BOTH ON 1, BOTH ON 2. En la posición NORM, la MCDU1 trabaja con el FM1 y la MCDU2, trabaja con el FM2. en la posición BOTH ON 1 (2), MCDU1 y MCDU2, trabajan con el mismo FM1 (2).

RMP: Los Radio Management Panels, son utilizados para la selección de nav aids.

ECAM: EWD muestra mensajes de aviso del AFS y el SD muestra en su página de status, información del AFS tal como sistemas no operativos.

17.10.2-FMS. DESCRIPCIÓN GENERAL.



La función de gestión de vuelo (Flight Management –FM–) de cada FMGEC, junto con la FCU y dos de las tres MCDU-s, lleva a cabo una serie de tareas, con el fin de ayudar a la tripulación en la gestión del vuelo. Todas las tareas se forman a partir de un plan de vuelo lateral y vertical y se pueden dividir en las siguientes cinco funciones:

- definición del plan de vuelo (lateral y vertical)

- navegación lateral: Inicialización del ADIRS, procesamiento de la posición lateral, seguimiento del plan de vuelo, selección (manual o automática) de las ayudas a la navegación (navaids).

- datos de actuación: procesamiento de la velocidad óptima, procesamiento de predicciones (durante el plan de vuelo, de acuerdo a varias limitaciones), seguimiento del perfil de vuelo, cálculo de varias actuaciones suplementarias.

- gestión de los displays: en las MCDU-s, ND-s y PFD-s.

- Enlace de datos de comunicación / vigilancia: comunicación de operación de la aerolínea y funciones **FANS** (Future Air Navigation System) [ADS (Automatic Dependent Surveillance) y CPDLC (Controller Pilot Datalink Communication)].

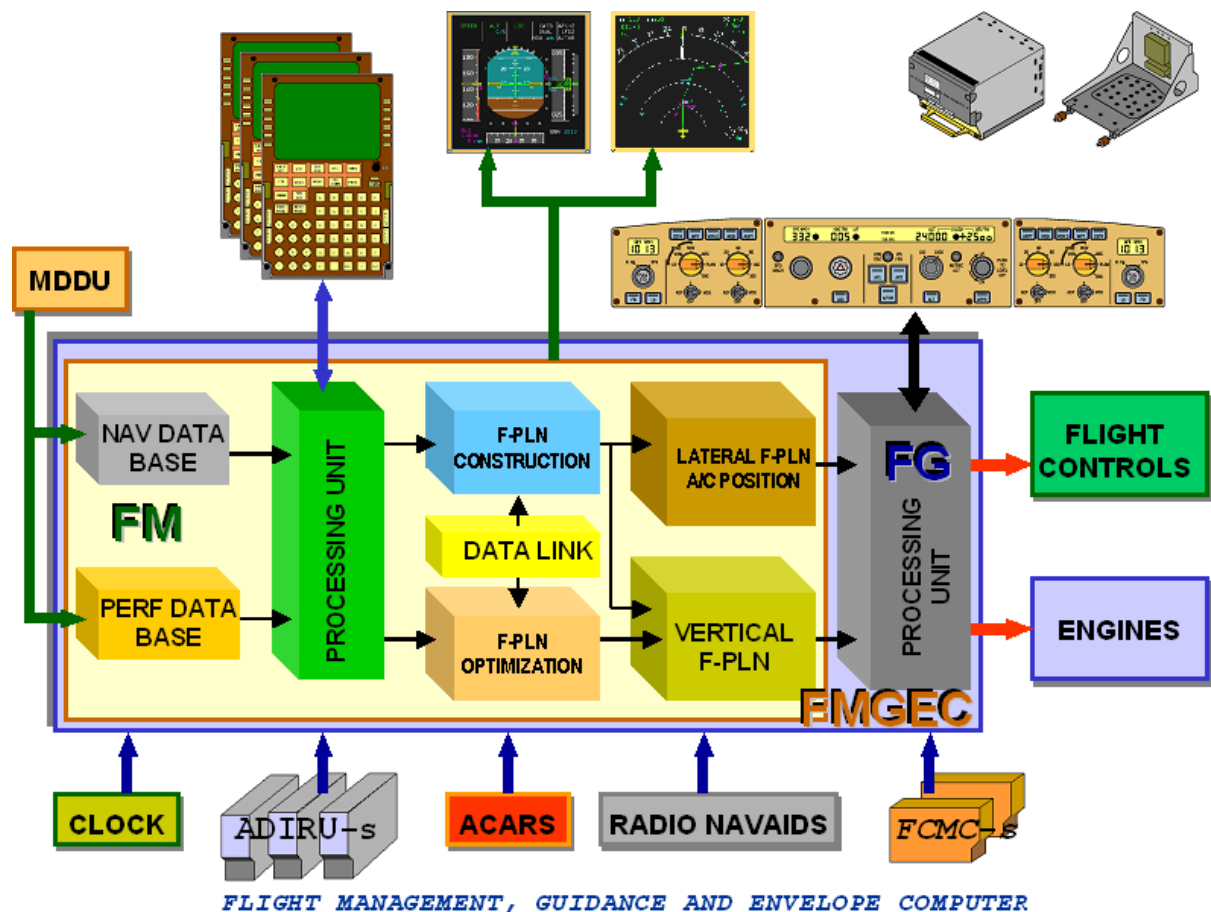
BASE DE DATOS.

La base de datos es una memoria de masa dividida en dos partes: base de datos para la navegación (NAV DATA BASE) con waypoints, nav aids, runways para la construcción del plan de vuelo lateral y la base de datos de actuación (PERF DATA BASE) con aspectos aerodinámicos de la aeronave, velocidades óptimas y modelo de motores, para la optimización del plan de vuelo. La NAV DATA BASE puede ser modificada o actualizada por la compañía, según norma ARINC 615, mientras que la PERF DATA BASE, solo puede ser modificada por el fabricante.

DEFINICIÓN DEL PLAN DE VUELO:

El plan de vuelo, está definido por varios elementos que indican las rutas que la aeronave debe seguir, con las limitaciones (principalmente de velocidad, altitud y tiempo) a lo largo de estas rutas. Está dividido en dos partes: lateral y vertical. La parte lateral da la dirección a seguir y la parte vertical, da los diferentes escalones de altitud con las restricciones de velocidades y tiempos asociadas.

Los elementos que indican las rutas que la aeronave debe seguir, se toman principalmente de una base de datos o pueden ser introducidos por el piloto. El FMGEC permite vía la MCDU, definir un plan de vuelo a partir de la base de datos de navegación almacenada internamente en una memoria específica. El piloto puede, o bien seleccionar un plan de vuelo ya construido, mediante un número de ruta de la compañía (CO ROUTE), o bien, puede construir un plan de vuelo concreto a partir de los datos contenidos en la base de datos, tales como, designación de los aeropuertos de origen y destino, waypoints, etc. En el caso de una CO ROUTE, el plan de vuelo puede contener información adicional como: la ruta de llegada, procedimientos de salida, el índice de coste (CI), aeropuerto alternativo asociado, etc.



La base de datos de navegación, debe ser actualizada por parte de la aerolínea, cada 28 días mediante un cargador de bases de datos (*Multipurpose Disk Drive Unit –MDDU-*) tal y como esta definido en ARINC 615. Esta base de datos incluye datos como: aeropuertos y procedimientos, CO ROUTE, aerovías, nav aids, waypoints, etc. Además, el piloto puede introducir de forma manual un número determinado de waypoints, nav aids, rutas y runways. Estos últimos datos pueden ser borrados de forma manual o automática bien, al final de cada vuelo, o cuando se selecciona un nuevo banco de datos; depende de la elección de la compañía (fichero AMI- Airline Modifiable Information).

La definición del plan de vuelo, determina en avance el tipo de aproximación a realizar en el aeropuerto de destino (ILS, R/NAV, VOR approach, ILS BACK BEAM).

En caso de fallo total del FM, la MCDU1 o MCDU2 puede proporcionar un modo de navegación de back-up. La activación de este modo se realiza vía teclado de la propia MCDU. En este caso, es la propia MCDU la que proporciona al sistema un plan de vuelo simplificado, basado en el que estaba disponible en el momento en el que fallo la función FM, y que comprende plan de vuelo lateral, posición de la aeronave y plan de vuelo con error de crosstrack (XTRKE). En esta situación, no se puede acoplar el AP en modo NAV.

NAVEGACIÓN LATERAL:

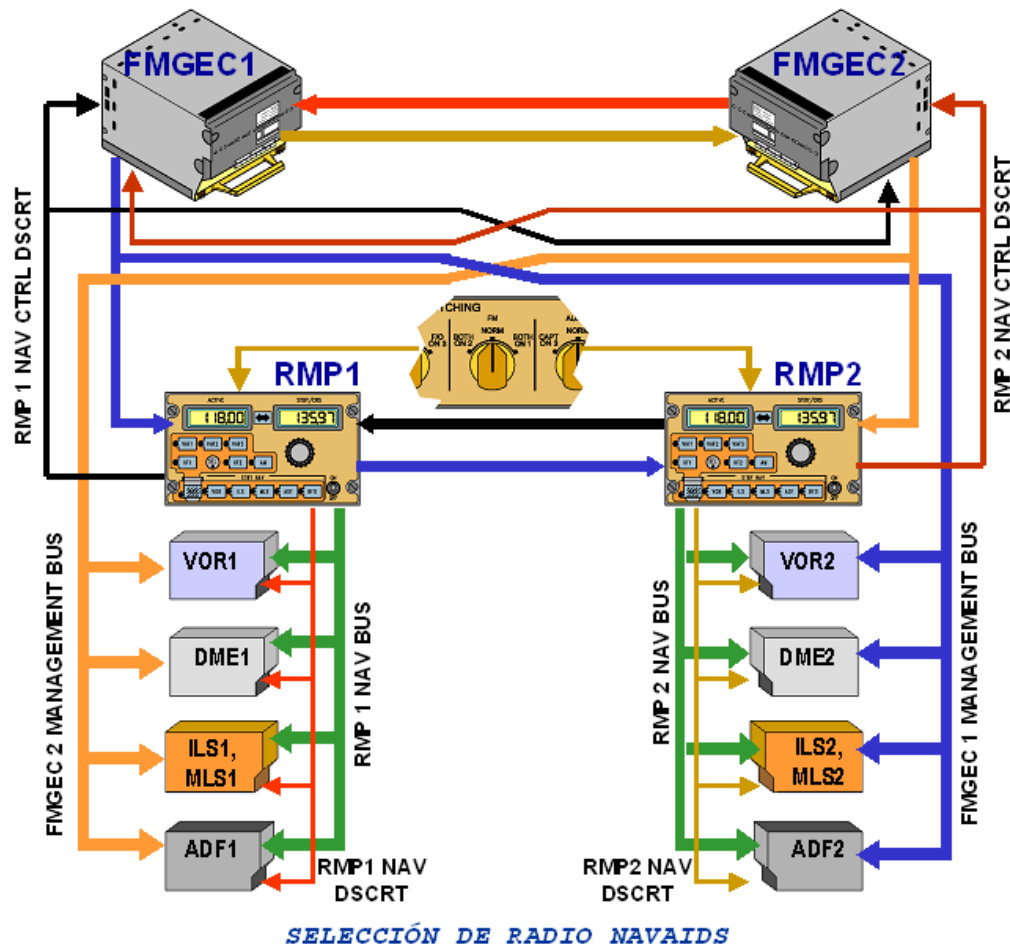
El proceso de navegación proporciona al sistema información del estado actual de la aeronave consistente en posición actual, altitud, viento, velocidad verdadera respecto al aire y velocidad respecto a tierra. Para la obtención de estos datos utiliza entradas del ADIRS, radionavegación y ACARS. La posición de la aeronave, es procesada a partir de datos suministrados por las tres ADIRU-s y radio nav aids (VOR, DME, ILS)

Las tres IR-s son inicializadas en tierra directamente desde el FM en respuesta a una petición de la MCDU a partir de una posición dada por la base de datos de navegación, o una posición definida por el piloto.

El plan de vuelo lateral proporciona los secuenciales cambios de rumbo en cada waypoint, dentro de tres secciones principales: salida, ruta y llegada.

El FMGEC proporciona navegación lateral asociada al plan de vuelo por medio del piloto automático, director de vuelo y la función de control lateral (botón de selección HDG/TRK en la FCU)

Cada parte FM, puede sintonizar todos los receptores de radio navegación (VOR, DME, ILS ADF) en ambos lados. En operación normal (con las dos FM operativas) cada FM sintoniza el sistema de radio navegación de su propio lado. En caso de fallo de un FMGEC, la función FM que queda, sintoniza las nav aids en ambos lados. También, está definido un modo back-up utilizando el **RMP** (*Radio Management Panel*) para conmutar la sintonización automática por sintonización manual en su propio lado.



DATOS DE ACTUACIONES:

El FMS permite minimizar por medio de la función de actuaciones, los costes de vuelo a través del procesamiento de una velocidad óptima para cada fase del mismo, optimización de combustible y tiempo. Este procesamiento depende del plan de vuelo lateral a seguir, índice de coste, peso de la aeronave, dirección y velocidad del viento, temperatura, modelo de los motores y parámetros aerodinámicos guardados en la base de datos de actuación.

La función de actuaciones también procesa predicciones a lo largo del plan de vuelo. Este procesamiento se realiza teniendo en cuenta las siguientes consideraciones:

- El perfil vertical de vuelo, está secuenciado en las típicas fases de vuelo: pre-vuelo (PREFLIGHT) despegue (TAKE-OFF), ascenso (CLIMB), crucero (CRUISE), descenso (DESCENT), aproximación / aproximación frustrada / go around (APPROACH / MISSED APPROACH / GO AROUND).

- La función FM por un lado, procesa actuaciones en los waypoints, tales como techo de ascenso, techo de descenso, escalón de ascenso óptimo, escalón de descenso óptimo, punto de intercepción para una altitud definida en la MCDU, límites de velocidad, consumo de fuel, tiempo para destino, etc. y por otro lado, tiene en cuenta las limitaciones de tiempo en un waypoint y limitaciones de velocidad y/o altitud en varios waypoints para procesar la velocidad, tiempo, altitud y fuel cuando sobrevuela cada waypoint.

La utilización del AP o FD permite el seguimiento del perfil vertical, cuando la velocidad vertical no ha sido impuesta por el piloto vía la FCU. El seguimiento de la velocidad vertical, se realiza por AP/FD o por el sistema de control de empuje (también cuando la velocidad no ha sido impuesta por el piloto)

El FM puede calcular actuaciones suplementarias como planificación de la altitud, altitudes óptimas y máximas, desviación vertical, predicción para los cinco aeropuertos más cercanos, predicciones para el plan de vuelo secundario, etc.

GESTIÓN DE LOS DISPLAYS:

El FM muestra información de guiado, navegación y actuaciones a través de las ND-s, MCDU-s y PFD-s, para la inicialización y seguimiento del plan de vuelo.

En las MCDU-s muestra información del plan de vuelo (waypoints, velocidad, tiempo, viento, predicciones de combustible, procedimiento de altitud), limitaciones del plan de vuelo, navais, datos de actuaciones relativos a las fases del vuelo, peso y CG, etc.

En los ND-s se muestra información de la posición de la aeronave, planes de vuelo, desviación lateral del plan de vuelo primario, mensajes, así como diversos parámetros mostrados en la FCU.

En los PFD-s se muestran mensajes, desviación de altitud respecto al descenso. Si el AP y/o FD están en modo de seguimiento del perfil de descenso, este perfil se muestra durante dicho descenso. La parte superior de los PFD-s está dedicada a los FMA-s (Flight Mode Annunciator): 5 columnas de información relativa a estatus de los AP, FD, A/Thrust, modos armados y acoplados de los canales lateral y longitudinal, datos de aproximación y aterrizaje y modos activos de A/Thrust.

FUNCIONES DE ENLACE DE DATOS DE COMUNICACIÓN / VIGILANCIA.

La interfaz **FMS / ATSU** (*Flight Management System / Air Traffic Service Unit*) soporta las siguientes funciones de enlace de datos:

Aplicaciones **FMS AOC** (*Flight Management System Airline Operational Control*) (también soportada por el ACARS MU) y aplicaciones **FMS ATC**. (*Flight Management System Air Traffic Control*).

-Funciones AOC. A través de la ACARS MU o de la ATSU, la sección de FM es capaz de enviar (downlink) información (envío de informes de vuelo) o peticiones a la estación de tierra para inicialización de datos, y también recibir (uplink) información o peticiones desde tierra (datos de inicialización del plan de vuelo, despegue y viento o peticiones de informes). Por ejemplo, la tripulación puede enviar una petición solicitando datos para el despegue relativos hasta un máximo de dos runways, indicando las condiciones de despegue en esas runways (configuración, viento, contaminación). En respuesta a esa petición, o automáticamente, tierra envía las velocidades de despegue asociadas con hasta cuatro runways y las condiciones de despegue que han sido tenidas en cuenta para elaborar esas velocidades de despegue.

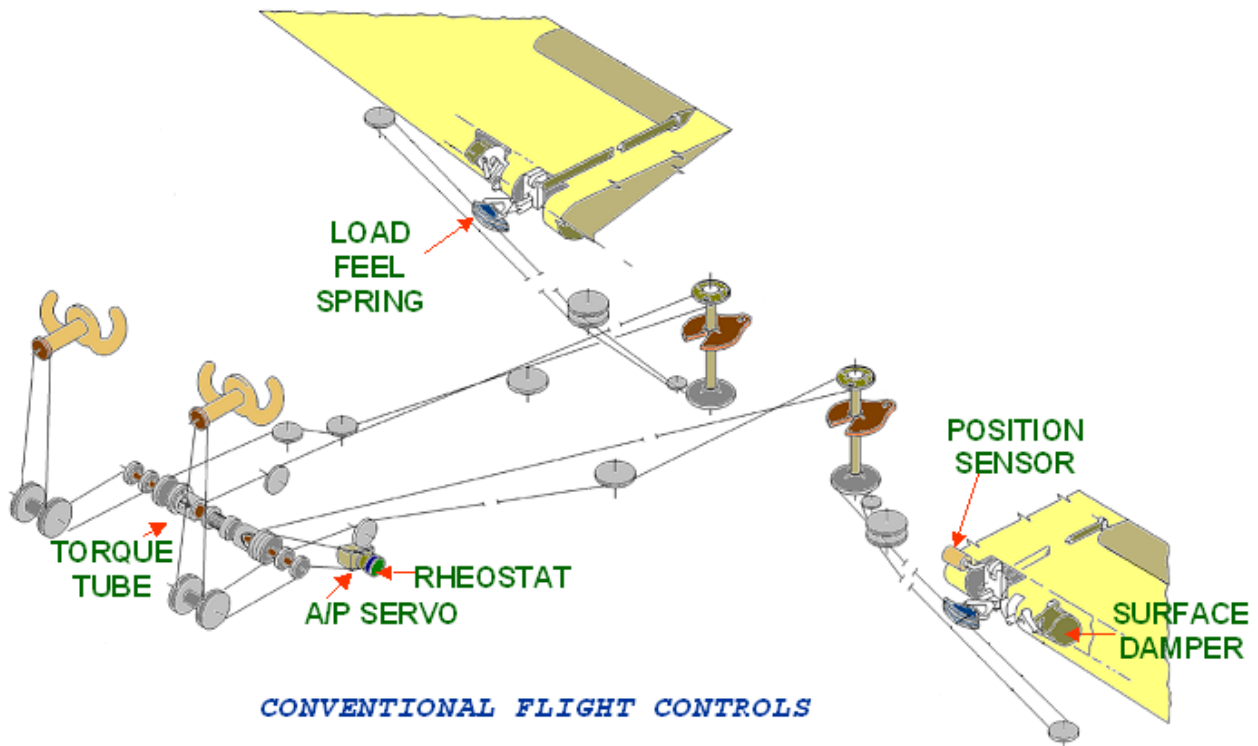
Los informes de vuelo, proporcionan en tiempo real, a tierra, información sobre la posición actual de la aeronave y su situación. Los tipos de informes de vuelo consisten en informes de posición, progresión, plan de vuelo y datos de actuaciones.

-Funciones ATC. Estas funciones reúnen las aplicaciones **CPDLC** (*Controller Pilot Datalink Communication*) y **ADS** (*Automatic Dependent Surveillance*). Las funciones CPDLC están divididas en dos categorías diferentes: (1) Recepción (uplink) de inicialización / modificación del ATC F-PLN, petición de informe ATC, informe de posición ATC, confirmación ATC, etc. (2) Transmisión (downlink) de petición de inicialización / modificación del F-PLN, respuesta al informe ATC, informe de posición, etc.

17.11-FBW (FLY BY WIRE).

Fly by wire es un método de control en lazo cerrado con el cual, primeramente, el movimiento de control o presión ejercida por el piloto, es transformado en una señal digital para ser procesada por un computador, según un programa específico. Posteriormente, dicho computador entrega una señal a los actuadores de las superficies de control para gobierno de estas, en la proporción correspondiente a la maniobra ejercida por el piloto.

17.11.1-ASPECTOS GENERALES.



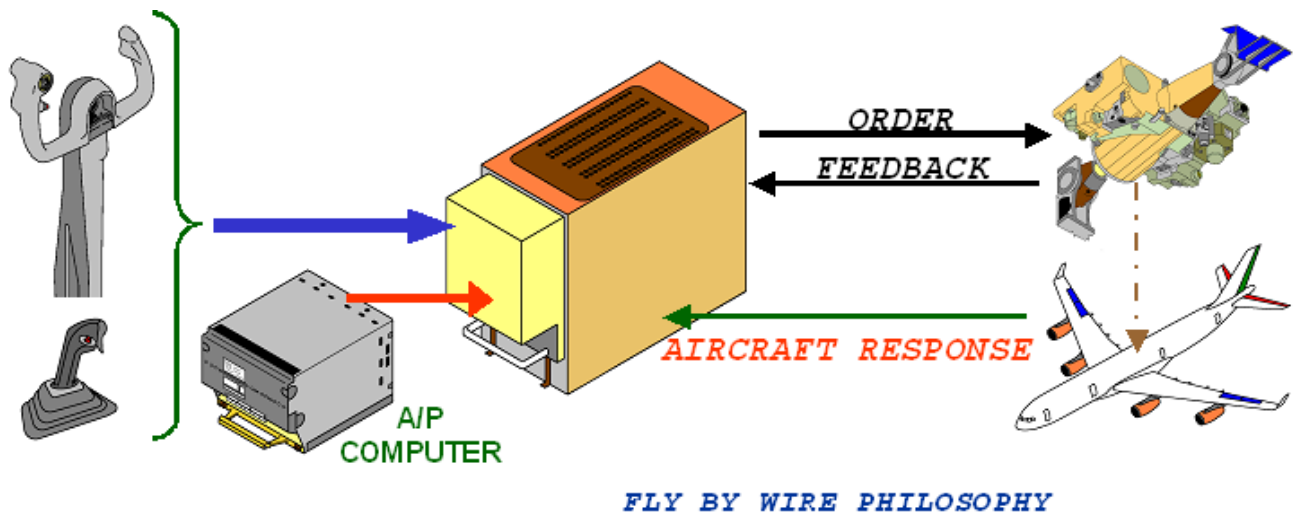
Mediante los controles convencionales de las superficies de control de la aeronave, dichas superficies son gobernadas de forma totalmente mecánica. Mediante el método FBW, todas las superficies de control de vuelo, son controladas de forma eléctrica y operadas de forma hidráulica, con posibilidad adicional de back up mecánico.

El canal mecánico se ha reemplazado por conductores eléctricos y computadores, los cuales controlan los servo-actuadores y elaboran las leyes del control del vuelo, optimizando el control de la aeronave.

Las órdenes del piloto automático, son ahora directamente transmitidas a los computadores para control de vuelo.

La realimentación mecánica de los servo-actuadores convencionales, es reemplazada por una realimentación eléctrica a los computadores.

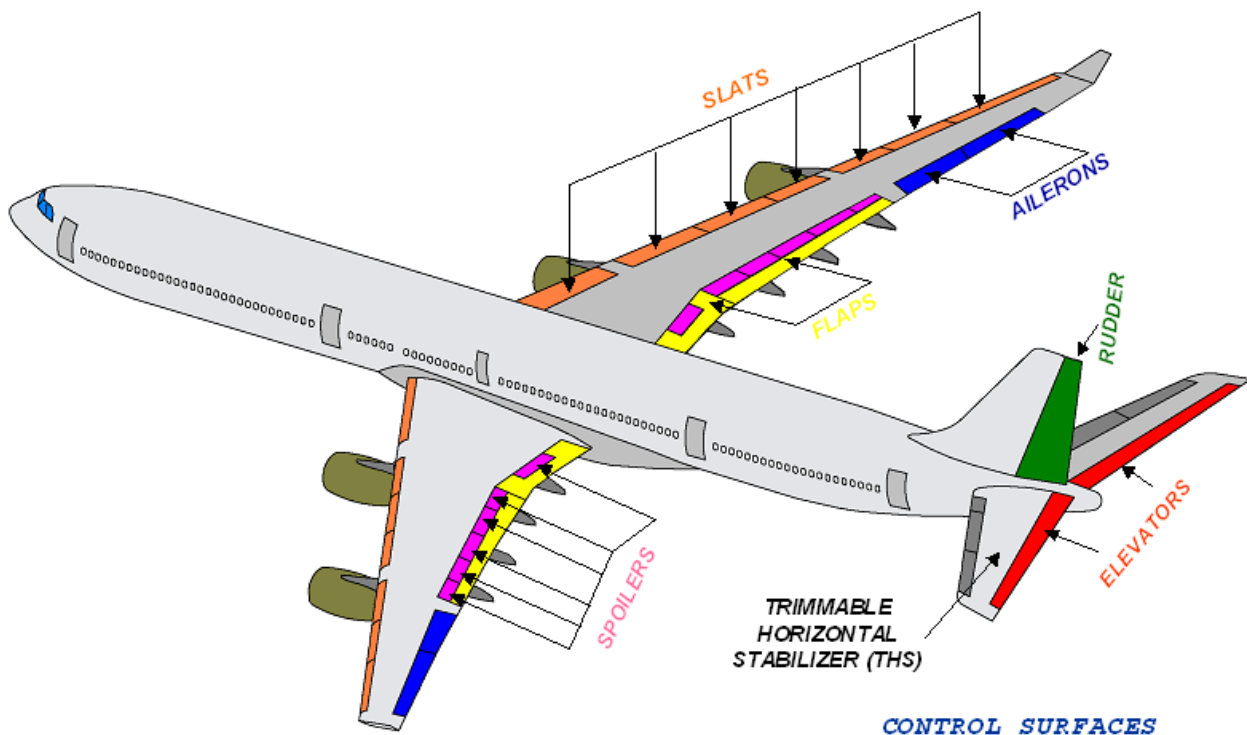
La función *Control Wheel Steering (CWS)* del AFS, está asegurada mediante los computadores de control de vuelo que seleccionan la orden adecuada, en función de la respuesta de la aeronave para mantener la actitud requerida.



El control eléctrico se realiza sobre los siguientes elementos: AILERONS, ROLL SPOILERS, RUDDER TRIM, ELEVATOR, THS, SLATS & FLAPS, SPEEDBRAKES / GROUND SPOILERS, RUDDER.

La actuación hidráulica (actuadores hidráulicos) se realiza sobre todas las superficies de control sin necesidad de utilizar aletas de compensación.

El control mecánico (back up) se realiza sobre el RUDDER y el **THS** (*Trimmable Horizontal Stabilizer*).



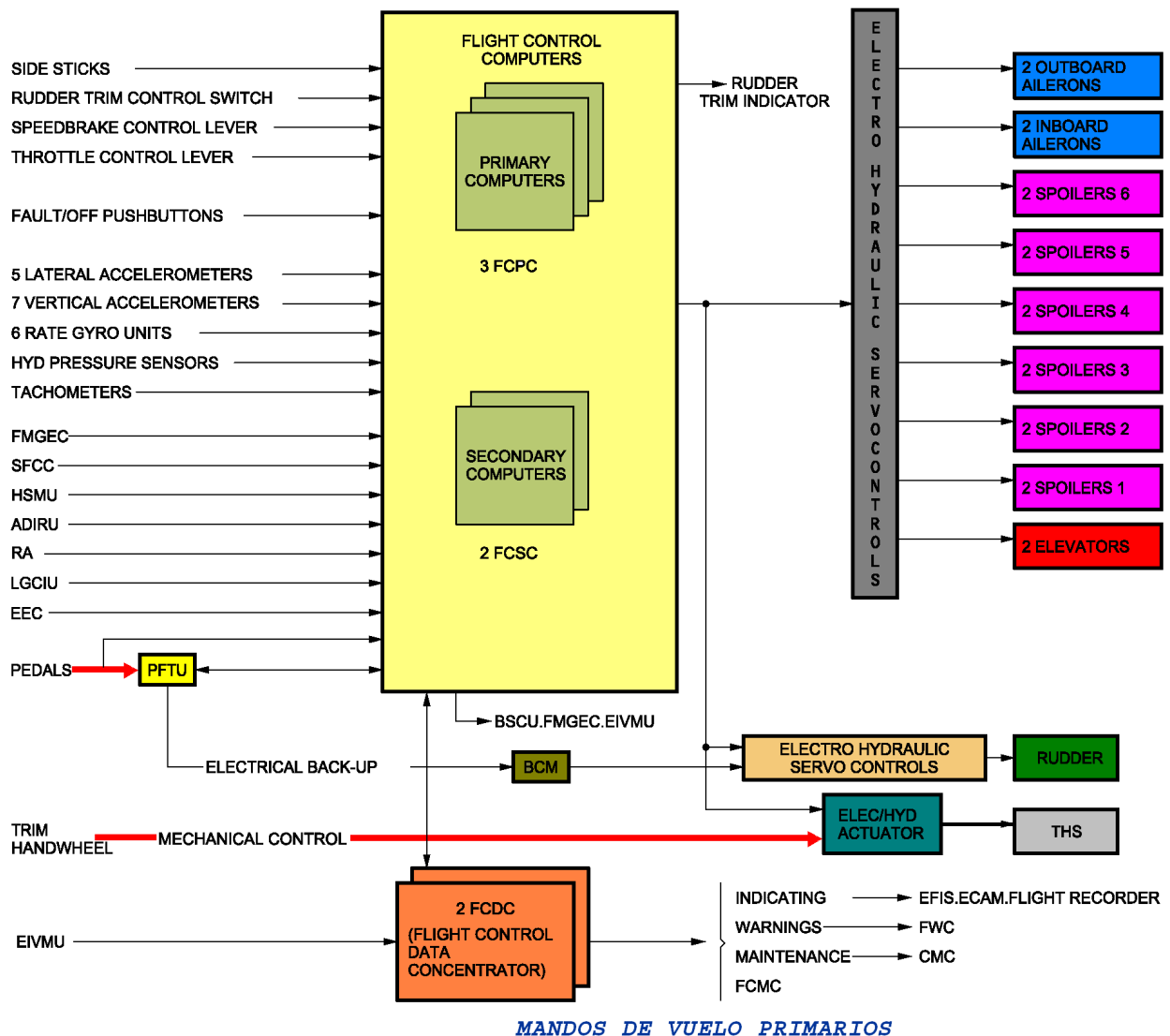
Las superficies son controladas por tres tipos de computadores:

Los *Flight Control Primary Computers (FCPC-s)*

Los *Flight Control Secondary Computers (FCSC-s)*

Los *Slat and Flap Control Computers (SFCC-s)*

Los FCPC-s y FCSC-s permiten el control de la aeronave en los ejes de alabeo (ROLL), guiñada (YAW) y cabeceo (PITCH). Los SFCC-s son utilizados para controlar los flaps y slats. Las funciones de indicación, avisos y mantenimiento son realizadas por los *Flight Control Data Concentrators (FCDC-s)*.



AILERONS: realizan las funciones de: control de alabeo en modo manual desde los side sticks o control wheel, y en modo AP desde los FMGEC-s; *Maneuver Load Alleviation (MLA)*; aumento de la sustentación, spoilers de tierra, amortiguador de turbulencia. Cada alerón es maniobrado por dos servo controles electro-hidráulicos, gobernados desde los FCPC-s o FCSC-s.

RUDDER: la función principal es la de control de la guiñada, pero también se utiliza para guiar la aeronave en tierra. El timón es maniobrado simultáneamente por tres servo-controles electro-hidráulicos. En operación normal, los servo-controles son gobernados por los FCPC-s y los FCSC-s desde los side sticks/control wheels y/o los pedales del timón (para el caso de un control manual) y desde los FMGEC-s (para el caso de control en AP)

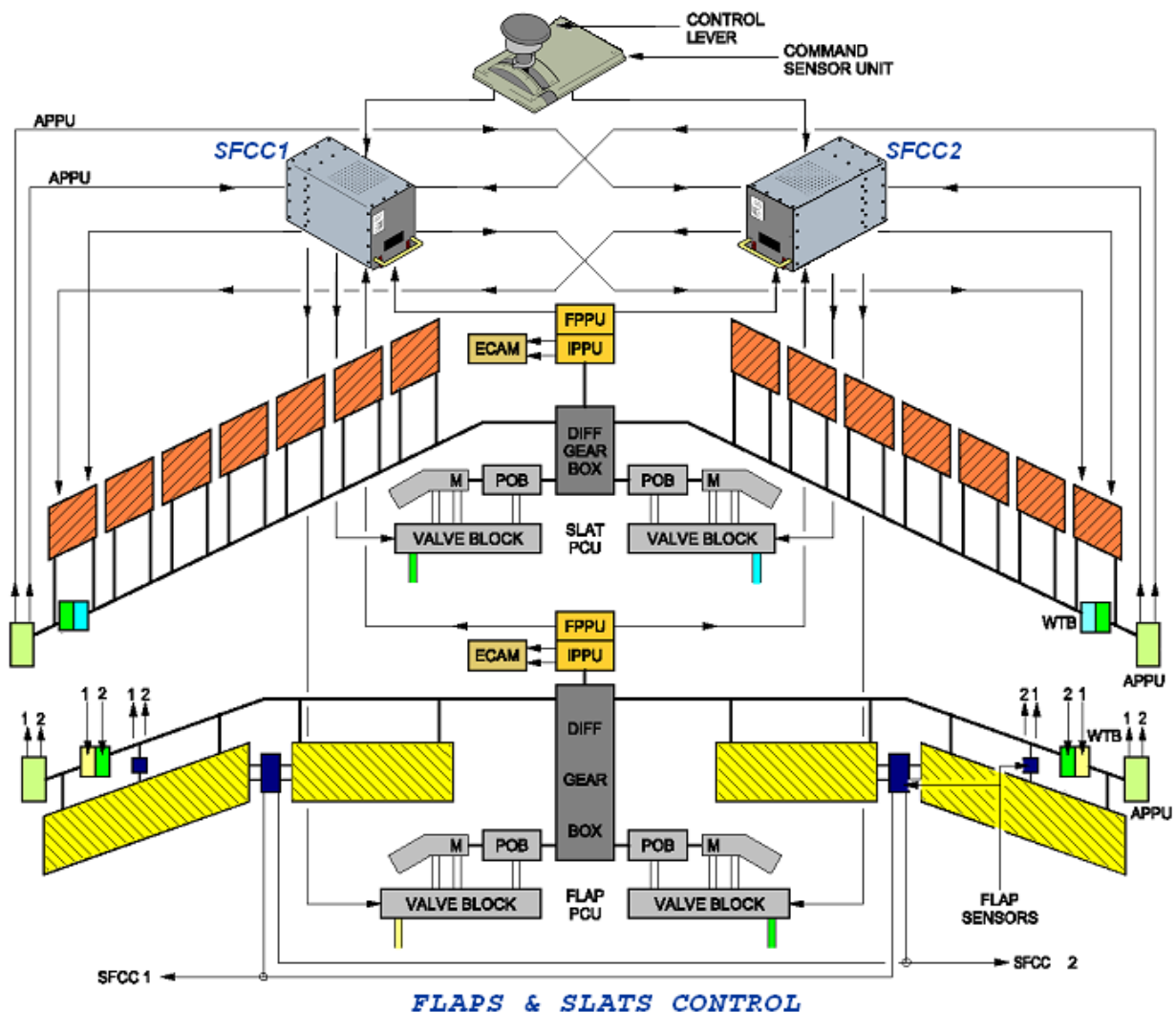
En caso de pérdida de gobierno eléctrico del timón (p. e. por fallo de los computadores), los servo-contróles son gobernados por el *Back-Up Control Module (BCM)*.

ELEVATORS: dos timones de profundidad, junto con el *Trimmable Horizontal Stabilizer (THS)* se encargan del control de pitch de la aeronave. Cada timón es maniobrado por dos servo-contróles electro-hidráulicos, controlados por los FCPC-s o los FCSC-s según órdenes de los side sticks/control wheels o los FMGEC-s.

TRIMMABLE HORIZONTAL STABILIZER (THS). El THS asegura la función de reglaje de pitch. Es maniobrado por un eje roscado gobernado por dos motores hidráulicos. Estos motores son controlados o bien de forma eléctrica por tres motores eléctricos controlados por los FCPC-s, o bien de forma mecánica desde la rueda de control de trim. En modo control por AP, son los FMGEC-s los que envían las órdenes de guiado a los FCPC-s.

SPOILERS. Son utilizados para el control de alabeo (spoilers 2 a 6), función de speed-brake y función de spoiler de tierra (spoilers 1 a 6), y también implementa la función de MLA (spoilers 4 a 6) (se numeran partiendo desde la base, hacia la punta del ala)

FLAPS & SLATS: los flaps y slats de cada ala son maniobrados mediante las *Power Control Units (PCU-s)* hidráulicas, las cuales gobiernan un enlace mecánico y actuadores rotatorios. Las PCU-s son controladas por los SFCC-s



Índice

A

ACARREO PARALELO... 152
 ACARREO SERIE 152
 ACARS 561
 acondicionador de señal.. 176
 ACTIVADOS POR FLANCO
 141
 ACTIVADOS POR NIVEL 140
 actuador eléctrico 177
 ADDRESS BUS 207
 ADM..... 556
 AEEC 393
 AIM 415
 AINS 602
 AMU 535
 analizador de espectro 485
 Ángulo crítico 378
 ánodo común 109
 AOA 557
 ARINC..... 393
 ARINC 453 539
 ATC 578
 ATIMS 602
 ATS 527

CLASIFICACIÓN DE LOS
 SISTEMAS 493
 código de paridad 30
 Código EBCDIC 29
 CÓDIGO HAMMING 32
 COMA FLOTANTE 19
 combinación prohibida 135
 COMPLEMENTO A 1 18
 COMPLEMENTO A 2 18
 conexión alambrada 168
 conexión en estrella 393
 conexión fusible 168
 configuración lineal 393
 CONTADOR ASCENDENTE
 148
 CONTADOR
 DESCENDENTE 149
 CONTADOR REVERSIBLE
 158
 CSMA/CA 424
 CUELLO 441
 CYCLIC REDUNDANCY
 CHECK 30

ECAM 532
 ECP 532
 ED-12B 469
 ED-14C 478
 EFIS 532
 EIA 66, 329
 EIS 532
 el mapa de memoria 233
 Electrostatic Discharge 84
 Enable 101
 ENSAMBLADOR 249
 entradas flotantes 73
 ESD Protected Area 88
 ESDS 86
 espectro 492
 EUROCAE 477
 EUT 479
 Even Parity 123
 EWD 533
 excepción 313
 Expansión de la capacidad
 231
 Expansión de palabra 231

B

back door coupling 482
 barrido veztorizado 443
 báscula 134
 BCD DATA 408
 biestable 134
 Binario Reflejado 27
 binarios cíclicos 26
 binarios continuos 26
 BITE 510
 BNR 408
 BNR DATA 409
 BOBINAS DE DEFLEXIÓN
 441
 Bulk Current Injection 487

C

CAÑÓN 441
 cátodo común 109
 CCITT 329
 CDS 545
 célula 498
 CELULA FLASH 211
 célula GTEM 487
 Célula TEM 486
 CHECKSUM 32
 Chip Enable 101
 Chip Select 101

D

Daisy Chain 315
 DATA BUS 207
 descarga guía escalonada
 499
 DESCOMPOSICIÓN
 POLINÓMICA 11
 DEU 545
 direccionamiento 272
 DIRECT MEMORY ACCESS
 244
 DISCRETOS 413, 474
 DISPLAYS 16-s 466
 DISPLAYS 7-s 464
 DISPLAYS DE MATRIZ DE
 PUNTOS 467
 DISTANCIA 30
 DITS 393
 DLP 463
 DMAC 356
 DMC 532
 DMD 463
 DO-160B 477
 DO-178B 469
 DO-248B 471
 dobles 420
 DSP 203

E

F

FBW 624
 FCU 535
 FETCH 357
 fibra monomodo 381
 fibra multimodo 381
 FIRMWARE 238
 Flip-Flop 134
 Floppy Disk 229
 FMS 614
 Foiled Twisted Pair 375
 FPLA 170
 framing error 324
 front door coupling 482
 Full Handshake 337
 FWC 532

G

GAL 169
 GENERADOR DE
 CARACTERES 452
 GPS 573

H

Hard Disk 229
 HARDWARE 238

I

IEC	66
IMPEDANCIA	
CARACTERÍSTICA DE LA	
LÍNEA	366
índice de refracción	377
INTERRUPT MASK	304
INTERRUPTS	244
iRAM	217

J

JEDEC	66

K

Karnaugh	53

L

LABEL	404
LATCH	142
Least Significant BIT	11
ley de Snell	378
LIFO	264
línea plana	486
liquido cristalino	457
LONG WORDS	294
LRU	527
LSK	515
luminóforos	444

M

MAPA LÓGICO	233
Master DSDL	539
MASTER/SLAVE	143
MAT	442
MAXTERMS	48
MINTERMS	48
Mode Stirred Chamber	487
Most Significant BIT	11
MSU	555
MULTIVIBRADOR ASTABLE	
.....	79
MULTIVIBRADOR	
MONOESTABLE	79

N

ND	533
nivel de los fallos	510
niveles de software	469

Non-Return-to-Zero	34
NYQUIST	188

O

OBRM	538
Odd Parity	123
Open Area Test Site	486

P

PAL	169
PANTALLA	441
Partial Handshake	337
periféricos	239
PFD	533
Phase-Encoded	34
phase-lock loop	322
PIA	339
PITOT	557
píxel	450
Priority Encoder	95
programa fuente	251
PROGRAM-CONTROLLED	
I/O	243
protocolo	243
pull-up	75
Pulse Mode Handshake ..	337

Q

Quine McCluskey	53

R

RAM de pantalla	453
Raster	442
REFLEXIONES	367
REFRESCO	217
refresco de pantalla	446
Return-to-Zero	34
RHO-THETA	443
RS 422	539
RTCA	477

S

SAE	429
SAR	196
SD	534
SDAC	535
SDI	406

SECUENCIALES

ASÍNCRONOS	134
SECUENCIALES	
SÍNCRONOS	134
SEÑAL ANALÓGICA	10
SEÑAL DIGITAL	10
SEÑAL DISCRETA	10
SEVEN SEGMENTS	109
shadow mask	444
Shielded Twisted Pair	375
SIGNO-MAGNITUD	17
SIM	421
SOFTWARE	238
SRU	528
STACK POINTER	264
STATIC PORTS	556

T

tabla de transición	135
TAMAÑO DE PASO	182
TAT	558
TCAS	588
TERMINACIONES	371
Thin Film Transistor	462
TIMER 555	80
tipo de fallos	510
TOGGLE	139
TOTEM POLE	73
transductor	176
transmisión paralelo	321
transmisión serie	321
triadas	444
triboelectricidad	84
TRIESTADO	75

U

UD	532
Unshielded Twisted Pair .	374
UUT	524

V

VIA	346

W

WAIT STATES	246
Watchdog	299
WXR	533

X

X-ON / X-OFF	332

NEMÓNICO	TAMAÑO	MODOS DE DIRECCIONAMIENTO															operación	X	N	Z	V	C		
			REG	Dn	An	(An)	(An)+	_(An)	d(An)	d(An,Xi)	Abs.W	Abs.L	d(PC)	d(PC,Xi)	#	Impl							REG	
																								MEMORIA
ABCD Add BCD	B	s= Dn d= d=	*				*											d <-- [s] + [d] + [X] (en BCD)	X	U	X	U	X	
ADD Add	B, W, L	s= Dn d= s= d=Dn	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*		d <-- [Dn] + [d] Dn <-- [s] + [Dn]	X	U	X	X	X
ADDA Add address	W L	d=An s= d=An s=	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	An <-- [s] + [An]	X					
ADDI Add immediate	B, W, L	s=Imm d=	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	d <-- s + [d]	X	X	X	X	X	
ADDQ Add quick	B, W, L	s=Imm3bits d= s=Imm3bits d=	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	d <-- QQQ + [d] (000= 8)	X	X			X	
ADDX Add extended	B, W, L	s= Dn d= s=-(An) d=	*															d <-- [s] + [d] + [X]	X	X	X	X		
AND Logical AND	B, W, L	s= Dn d= s=Dn	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	d <-- [Dn] ^ [d]			X	X	0	
ANDI AND immediate	B, W, L	s=Imm d= s=Imm d=	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	d <-- s ^ [d]			X	X	0	
ASL Arith. Shift left	B, W, L W	count=[Dn] d= count=QQQ d= count=1 d=	*	*														(a) ver dibujo al final	X	X	X	X		
ASR Arith. Shift right	B, W, L W	count=[Dn] d= count=QQQ d= count=1 d=	*	*														(b) ver dibujo al final	X	X	X	X		
BCHG Test bit & change it	B L	bit#=[Dn] d= bit#= Imm d= bit#=[Dn] d= bit#= Imm d=	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	Z <-- compl (bit# de d) después complementa el bit testado en d			X	X		
BCLR Test bit & clear it	B L	bit#=[Dn] d= bit#= Imm d= bit#=[Dn] d= bit#= Imm d=	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	Z <-- compl (bit# de d) después pone a 0 el bit testado en d			X	X		
BSET Test bit & set it	B L	bit#=[Dn] d= bit#= Imm d= bit#=[Dn] d= bit#= Imm d=	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	Z <-- compl (bit# de d) después pone a 1 el bit testado en d			X	X		
BTST Test bit	B L	bit#=[Dn] d= bit#= Imm d= bit#=[Dn] d= bit#= Imm d=	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	Z <-- compl (bit# de d)			X	X		

NEMÓNICO	TAMAÑO	MODOS DE DIRECCIONAMIENTO														operación	X	N	Z	V	C			
			Dn	An	(An)	(An)+	_(An)	d(An)	d(An,Xi)	Abs.W	Abs.L	d(PC)	d(PC,Xi)	#	Impl									
															REG									
MEMORIA																								
CHK	W		*		*	*	*	*	*	*	*	*	*	*	*		si [Dn] <0 o [Dn] > [s] se produce interrupción			X	U	U		
CLR	B, W, L		*		*	*	*	*	*	*	*	*	*	*	*		d <-- 0			0	1	0	0	
CMP	B, W, L		*	*	*	*	*	*	*	*	*	*	*	*	*		[d] - [s]			X	X	X	X	
CMPA	W		*	*	*	*	*	*	*	*	*	*	*	*	*		[An] - [s]			X	X	X	X	
Compare address	L		*	*	*	*	*	*	*	*	*	*	*	*	*					X	X	X	X	
CMPI	B, W, L		*	*	*	*	*	*	*	*	*	*	*	*	*		[d] - [s]			X	X	X	X	
Compare immediate																								
CMPM	B, W, L				*												[d] - [s]			X	X	X	X	
Compare memory																								
DIVS	W		*	*	*	*	*	*	*	*	*	*	*	*	*		d <-- [d] / [s] usando 32 bits de d y 16 de s			X	X	X	0	
DIVU	W		*	*	*	*	*	*	*	*	*	*	*	*	*		d <-- [d] / [s] usando 32 bits de d y 16 de s			X	X	X	0	
EOR	B, W, L		*	*	*	*	*	*	*	*	*	*	*	*	*		d <-- [Dn] X-OR [d]			X	X	0	0	
Exclusive OR																								
EORI	B, W, L		*	*	*	*	*	*	*	*	*	*	*	*	*		d <-- s X-OR [d]			X	X	0	0	
Exclusive OR immediate															*					X	X	0	0	
EXG	L		*	*	*												[s] <--> [d]							
Exchange																								
EXT	W		*														bits 8 a 15 de d <-- bit 7 de d bits 16 a 31 de d <-- bit 15 de d			X	X	0	0	
Sign extended	L		*														PC <-- [d]			X	X	0	0	
JMP					*										*									
Jump																								
JSR					*										*		SP <-- [SP] - 4 [SP] <-- [PC]; PC <-- [d] An <-- dirección efectiva de s							
Jump to subroutine																								
LEA	L				*												SP <-- [SP] - 4; [SP] <-- [An] An <-- [SP]; SP <-- [SP] + desplaz							
Load effective address																								
LINK	W																[c] ver dibujo al final			X	X	X	0	X
Link and allocate																				X	X	X	0	X
LSL	B, W, L		*	*	*	*	*	*	*	*	*	*	*	*	*					X	X	X	0	X
Logical shift left	W			*	*	*	*	*	*	*	*	*	*	*	*					X	X	X	0	X
LSR	B, W, L		*	*	*	*	*	*	*	*	*	*	*	*	*		(d) ver dibujo al final			X	X	X	0	X
Logical shift right																			X	X	X	0	X	
	W			*	*	*	*	*	*	*	*	*	*	*	*					X	X	X	0	X

NEMÓNICO	TAMAÑO	MODOS DE DIRECCIONAMIENTO														operación	X	N	Z	V	C
			Dn	An	(An)	(An)+	_(An)	d(An)	d(An,Xi)	Abs.W	Abs.L	d(PC)	d(PC,Xi)	#	Impl						
PEA	L	S=		*			*	*	*	*	*	*	*		SP <-- [SP] - 4						
Push effective address															[SP] <-- Dirección efectiva de s						
RESET															Activa salida línea reset						
ROL	B, W, L	count=[Dn] d =	*				*	*	*	*	*	*	*		(e) ver dibujo al final		X	X	0	X	
Rotate left without X	W	count=QQQ d=	*		*		*	*	*	*	*	*	*				X	X	0	X	
		count=1 d=															X	X	0	X	
ROR	B, W, L	count=[Dn] d =	*				*	*	*	*	*	*	*		(f) ver dibujo al final		X	X	0	X	
Rotate right without X	W	count=QQQ d=	*		*		*	*	*	*	*	*	*				X	X	0	X	
		count=1 d=															X	X	0	X	
ROXL	B, W, L	count=[Dn] d =	*				*	*	*	*	*	*	*		(g) ver dibujo al final		X	X	0	X	
Rotate left with X	W	count=QQQ d=	*		*		*	*	*	*	*	*	*				X	X	0	X	
		count=1 d=															X	X	0	X	
ROXR	B, W, L	count=[Dn] d =	*				*	*	*	*	*	*	*		(h) ver dibujo al final		X	X	0	X	
Rotate right with X	W	count=QQQ d=	*		*		*	*	*	*	*	*	*				X	X	0	X	
		count=1 d=															X	X	0	X	
RTE															SR<--[SP]; SP <-- [SP] + 2; PC <--[SP]; SP <-- [SP] + 4	X	X	X	X	X	
Return from exception																					
RTR															CCR<--[SP]; SP <-- [SP] + 2; PC <--[SP]; SP <-- [SP] + 4	X	X	X	X	X	
Return & restore CCR																					
RTS															PC <-- [SP]; SP <-- [SP] + 4						
Return from subroutine																					
SBCD	B	s= Dn d=	*				*								d <-- [d] - [s] - [X] (en BCD)	X	U	X	U	X	
Subtract BCD		s=-(An) d=														X	U	X	U	X	
Scc	B	d=	*		*		*	*	*	*	*	*	*		Pone a 1 los 8 bits de d si cc es verdad, si no a 0						
Set on condition														*	SR <-- s ; espera por interrupción	X	X	X	X	X	
STOP		s=																			
Load SR and stop																					
SUB	B, W, L	s= Dn d=	*		*		*	*	*	*	*	*	*		d <-- [d] - [s]	X	X	X	X	X	
Subtract		d= Dn s=			*		*	*	*	*	*	*	*			X	X	X	X	X	
SUBA	W	d= An s=	*		*		*	*	*	*	*	*	*		An <-- [An] - [s]						
Subtract address	L	d= An s=	*		*		*	*	*	*	*	*	*								
SUBI	B, W, L	s=Imm d=	*		*		*	*	*	*	*	*	*		d <-- [d] - s	X	X	X	X	X	
Subtract immediate																					
SUBQ	B, W, L	s=Imm3 d=	*		*		*	*	*	*	*	*	*		d <-- [d] - QQQ	X	X	X	X	X	
Subtract quick																					
SUBX	B, W, L	s= Dn d=	*		*		*	*	*	*	*	*	*		d <-- [d] - [s] - [X]	X	X	X	X	X	
Subtract extended		s=-(An) d=														X	X	X	X	X	

NEMÓNICO	TAMAÑO	MODOS DE DIRECCIONAMIENTO													operación	X	N	Z	V	C		
		Dn	An	(An)	(An)+	(An)	d(An)	d(An,Xi)	Abs.W	Abs.L	d(PC)	d(PC,Xi)	#	Impl								
		MEMORIA																				
		REG																				
SWAP	W	d=	*													[Dn]31-16 <--> [Dn]15-0		X		0	0	
Swap register halves																						
TAS	B	d=	*	*	*	*	*	*	*	*	*	*	*	*	*		Test d y actualiza los flags N y Z; Pone el bit 7 a 1		X	X	0	0
Test and set																						
TRAP			SSP <--[SSP] - 4; [SSP] <--[PC]; SSP <--[SSP] - 2; [SSP] <--[SR]; PC <-- vector																			
Trap																						
TRAPV			si V=1, entonces: SSP <--[SSP] - 4; [SSP] <--[PC]; SSP <--[SSP] - 2; [SSP] <--[SR]; PC <-- TRAP vector																			
Trap on overflow																						
TST	B, W, L	d=	*	*	*	*	*	*	*	*	*	*	*	*	*		Test d y actualiza los flags N y Z		X	X	0	0
Test																						
UNLINK			*													SP <--[An]; An <--[SP]; SP <--[SP] + 4						
Unlink																						

INSTRUCCIONES DE RAMIFICACIÓN	
NEMÓNICO	OPERACIÓN
DESPLAZ.	
BRA	PC <-- [PC] + desplaz
Branch always	
Bcc (ver tabla)	si cc es verdad, entonces: PC <-- [PC] + desplaz
Branch conditionally	
BSR	SP <-- [SP] - 4; [SP] <-- [PC]; PC <-- [PC] + desplaz
Branch to subroutine	
DBcc	si cc es falso, entonces: Dn <-- [Dn] - 1; si [Dn] no es -1, entonces PC <-- [PC] + desplaz
Decrement & branch conditionally	
DBRA	El ensamblador interpreta esta instrucción como DBF
Decrement & branch	

s	operando fuente	CCR	registro de códigos de condición
d	operando destino	Q...Q	dato inmediato rápido
An	registro de dirección	#	negación o direccionamiento inmediato (según contexto)
Dn	registro de datos		
Xn	registro usado como índice	bit#	nº de bit
PC	contador de programa	u	indefinido (flag)
SP	puntero de pila	x	afectado (flag)
SR	registro de estado		

SUFIJO	NOMBRE	TEST
cc		
T	True	Always true
F	False	Always false
HI	High	C OR Z = 0
LS	Lower or same	C OR Z = 1
CC	Carry clear	C = 0
CS	Carry set	C = 1
NE	Not equal	Z = 0
EQ	Equal	Z = 1
VC	Overflow clear	V = 0
VS	Overflow set	V = 1
PL	Plus	N = 0
MI	Minus	N = 1
GE	Greater or equal	N X-OR V = 0
LT	Less than	N X-OR V = 1
GT	Greater than	Z OR (N X-OR V) = 0
LE	Less or equal	Z OR (N X-OR V) = 1

T Y F no pueden ser usados en la instrucción Bcc

NOTA 1:

X se pone a 0 si el bit4 del operando inmediato es 0; si no, no cambia
N se pone a 0 si el bit3 del operando inmediato es 0; si no, no cambia
Z se pone a 0 si el bit2 del operando inmediato es 0; si no, no cambia
V se pone a 0 si el bit1 del operando inmediato es 0; si no, no cambia
C se pone a 0 si el bit0 del operando inmediato es 0; si no, no cambia

NOTA 2:

X cambia si el bit4 del operando inmediato es 1; si no, no cambia
N cambia si el bit3 del operando inmediato es 1; si no, no cambia
Z cambia si el bit2 del operando inmediato es 1; si no, no cambia
V cambia si el bit1 del operando inmediato es 1; si no, no cambia
C cambia si el bit0 del operando inmediato es 1; si no, no cambia

NOTA 3:

X se pone a 1 si el bit4 del operando inmediato es 1; si no, no cambia
N se pone a 1 si el bit3 del operando inmediato es 1; si no, no cambia
Z se pone a 1 si el bit2 del operando inmediato es 1; si no, no cambia
V se pone a 1 si el bit1 del operando inmediato es 1; si no, no cambia
C se pone a 1 si el bit0 del operando inmediato es 1; si no, no cambia

