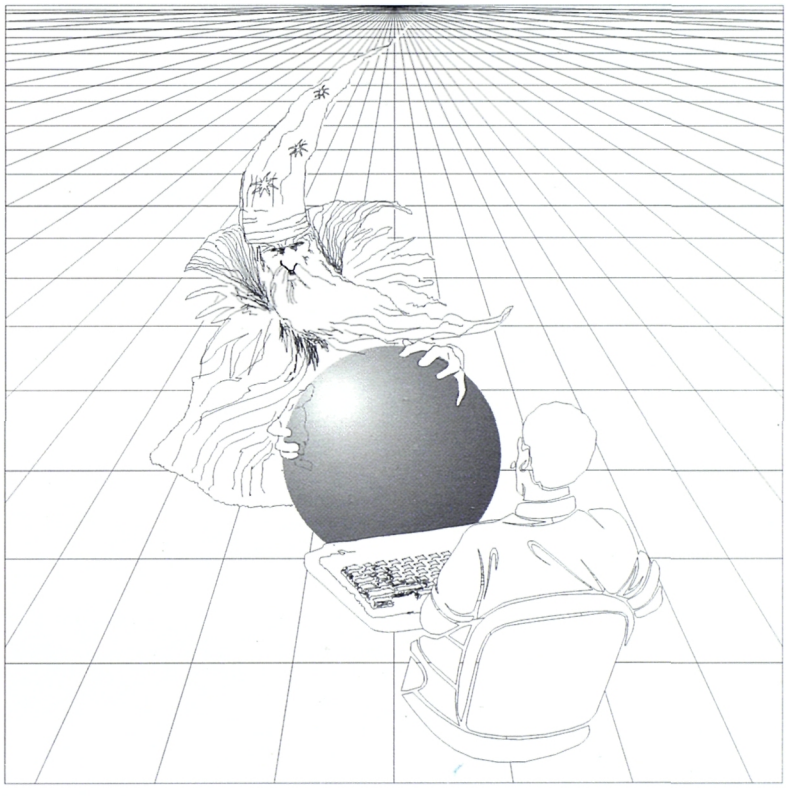
ANTONIO DÍAZ ESTRELLA

DISEÑO DE SISTEMAS DIGITALES



UNIVERSIDAD DE MÁLAGA / MANUALES

UNIVERSIDAD DE MÁLAGA
Publicaciones y Divulgación Científica

ANTONIO DÍAZ ESTRELLA

DISEÑO DE SISTEMAS DIGITALES

UNIVERSIDAD DE MÁLAGA / MANUALES

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Publicaciones y
Divulgación Científica

© Antonio Díaz Estrella
© Publicaciones y Divulgación Científica. Universidad de Málaga.

Diseño de la colección: J. M. Mercado

ISBN: 978-84-9747-176-3
Depósito Legal: MA-173/2007

Imprime: Imagraf Impresores, S.A. Tel.: 952 32 85 97



Esta obra está bajo una licencia de Creative Commons
Reconocimiento-NoComercial-SinObraDerivada 4.0 Internacional:
http://creativecommons.org/licenses/by-nc-nd/4.0/legalcode
Cualquier parte de esta obra se puede reproducir sin autorización
pero con el reconocimiento y atribución de los autores.
No se puede hacer uso comercial de la obra y no se puede alterar,
transformar o hacer obras derivadas.

Publicaciones y
Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Índice general

Prefacio xiii

1.- Introducción 1

1. Introducción . . . 1
2. Algunos conceptos básicos . . . 2
   - 2.1. Microprocesadores . . . 3
   - 2.2. Programación . . . 4
     - 2.2.1. Multitarea . . . 6
   - 2.3. Memorias . . . 7
     - 2.3.1. Jerarquía de memoria . . . 9
     - 2.3.2. Memoria cache . . . 10
     - 2.3.3. Bancos de memoria . . . 17
     - 2.3.4. Gestión de memoria . . . 18
   - 2.4. Entrada/Salida . . . 20
     - 2.4.1. Conversión analógico digital . . . 20
     - 2.4.2. Espacios de E/S . . . 21
     - 2.4.3. Modos de transferencia . . . 22
     - 2.4.4. Modos de transmisión . . . 27
   - 2.5. Microsistemas . . . 29
   - 2.6. Pipelining . . . 31
   - 2.7. Latencia y ancho de banda . . . 34
3. Evolución de los microprocesadores y microcomputadores . . . 35
   - 3.1. Los primeros tiempos . . . 35
   - 3.2. Microelectrónica . . . 36
   - 3.3. Evolución de los microprocesadores . . . 39
   - 3.4. Evolución de los microcomputadores . . . 41
   - 3.5. El interfaz hombre máquina . . . 45
   - 3.6. La multimedia . . . 47

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.7. Las comunicaciones 47
4. Últimas Tendencias 50
5. Bibliografía 51
6. Guía de estudio 52

2.- Microprocesadores: Arquitectura IA-32 de Intel 53
1. Introducción 53
2. Familia IA-32 de Intel 54
3. Microarquitecturas 58
3.1. Microarquitectura del Pentium 58
3.2. Microarquitectura P6 61
3.3. Microarquitectura Netburst 65
3.4. Microarquitectura Core 66
3.5. Ejemplos 67
3.5.1. Comparativa de prestaciones de microprocesadores IA-32 67
4. Entorno de ejecución básico 71
4.1. Modos de operación 71
4.2. Organización de memoria 72
4.2.1. Gestión de memoria 73
4.3. Registros de programación de aplicaciones 75
4.3.1. Registros de propósito general 75
4.3.2. Registros de punto flotante 77
4.3.3. Registros SIMD 78
4.4. Direccionamiento de operandos 78
4.5. Ejemplos 79
4.5.1. Uso del depurador del PC 79
5. Tipos de datos 81
6. Conjunto de instrucciones 82
6.1. Instrucciones de propósito general 82
6.2. Instrucciones de entrada/salida 83
6.3. Instrucciones x87 FPU 83
6.4. Instrucciones de gestión de estado 84
6.5. Instrucciones SIMD 84
6.5.1. Instrucciones MMX 84
6.5.2. Instrucciones SSE 85
6.6. Instrucciones del sistema 85
6.7. Ejemplos 86
6.7.1. Suma usando el depurador del PC 86

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

6.7.2. Aritmética saturante 87
6.7.3. Multiplicación SIMD 87
6.7.4. Multiplicación de matrices con MMX 89
6.7.5. Incrustación usando MMX 91
7. Interrupciones y excepciones 94
8. Entrada/Salida 95
9. Entorno de ejecución del sistema 96
9.1. Registros del sistema 96
9.2. Gestión de memoria 98
9.2.1. Segmentación 98
9.2.2. Paginación 107
9.2.3. Protección 109
9.2.4. Multitarea 112
9.3. Ejemplos 114
9.3.1. Cálculo de dirección física 114
10. Interfaz del Pentium 117
10.1. Interfaz de memoria y E/S 117
10.1.1. Operaciones de bus 120
10.2. Interfaz de interrupciones 123
10.3. Interfaz de multiproceso 124
11. Interfaz del Pentium II 125
11.1. Transacciones 125
11.2. Arbitración 129
11.3. Interfaces de control, compatibilidad y diagnóstico . . . 130
12. Interfaz de Pentium 4 131
13. Últimas tendencias 132
14. Bibliografía 134
15. Guía de estudio 137

3.- Memorias **139**
1. Introducción 139
2. Memorias semiconductoras 140
2.1. Memorias no volátiles 143
2.2. Memorias volátiles dinámicas 144
2.2.1. Memorias asíncronas 147
2.2.2. Memorias síncronas 150
2.2.3. Memorias especializadas 157
2.2.4. Empaquetado 158
2.2.5. Detección y corrección de errores de memoria . 159

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.3. Memorias volátiles estáticas 161
2.3.1. Memorias asíncronas 162
2.3.2. Memorias síncronas 163
2.3.3. Memorias especializadas 164
2.4. Ejemplos 165
2.4.1. Tiempos de acceso en modo ráfaga 165
2.4.2. Mapeo de memoria 165
2.4.3. Granularidad 166
2.4.4. Ancho de banda efectivo 167
2.4.5. Manejo de nomenclatura 167
3. Memorias magnéticas 168
3.1. Discos duros 168
3.2. Discos flexibles 171
3.3. Cintas 172
3.4. RAID 172
3.5. Ejemplos 173
3.5.1. Cálculo de parámetros de un disco duro 173
4. Memorias ópticas 174
4.1. CD-DA 174
4.2. CD para datos 176
4.3. DVD 178
4.4. Ejemplos 180
4.4.1. Cálculo de capacidad de un CD-DA 180
5. Últimas tendencias 181
6. Bibliografía 182
7. Guía de estudio 184

4.- Entrada/Salida **185**
1. Introducción 185
2. Controladores de E/S 186
2.1. Controlador de interrupciones 8259 de Intel 186
2.2. Controlador de DMA 8237 de Intel 189
2.3. Ejemplos 194
2.3.1. Acceso a puertos de E/S 194
2.3.2. Programando una RSI 194
2.3.3. Guía básica de programación de DMA 196
3. Puertos de E/S 197
3.1. Puerto serie RS-232C 197
3.1.1. Modem nulo 198

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.1.2. Control de flujo 198
3.1.3. UART 16550 de National 199
3.2. Puerto paralelo IEEE 1284 204
3.2.1. Modos 1284 206
3.3. Ejemplos 207
3.3.1. Acceso a impresora 207
4. Buses estándar de E/S 209
4.1. Evolución de buses internos de E/S 209
4.2. Bus PCI 211
4.2.1. Interfaz de memoria y E/S 212
4.2.2. Transacciones 214
4.2.3. Interfaz de interrupciones 216
4.2.4. Interfaz de arbitración 217
4.2.5. Otros interfaces 217
4.2.6. Conector PCI 219
4.2.7. Autoconfiguración 219
4.3. Bus PCI Express 221
4.4. Interfaz SCSI 223
4.5. Interfaz ATA 223
4.6. USB (Universal Serial Bus) 227
4.6.1. Topología del bus USB 228
4.6.2. Modelo de flujo de datos 228
4.6.3. Tipos de transferencias 230
4.6.4. Interfaz USB 230
4.6.5. Formato de paquetes 231
4.6.6. Enumeración USB 234
4.7. Bus I2C 235
5. Interfaz a periféricos 237
5.1. Teclado 237
5.2. Ratón 239
5.3. Joystick 242
5.4. Sistema Gráfico 244
5.4.1. Pantallas de visualización 246
5.4.2. Adaptadora gráfica 248
5.4.3. Aceleradora gráfica 252
5.4.4. AGP (Accelerated Graphics Port) 257
5.4.5. Ejemplos 258
5.5. Sistema de audio 259

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5.5.1. SoundBlaster 16 . . . 259
5.5.2. Sistema AC´97 . . . 266
5.5.3. MIDI . . . 270
5.5.4. Ejemplos . . . 273
6. Bibliografía . . . 274
7. Guía de estudio . . . 276

5.- Microcomputadores **277**
1. Introducción . . . 277
2. El IBM PC . . . 278
2.1. Diagrama de bloques . . . 280
2.1.1. Generador de reloj . . . 280
2.1.2. Buses . . . 282
2.1.3. Interrupciones . . . 282
2.1.4. Controlador de DMA . . . 283
2.1.5. Puertos paralelo . . . 283
2.1.6. Temporizadores . . . 283
2.1.7. Memoria . . . 284
2.1.8. Interfaz de teclado . . . 284
2.2. Espacio de direcciones . . . 286
2.3. Bus de expansión . . . 287
2.3.1. Ciclos de bus . . . 289
3. El PC-XT . . . 292
4. El PC-AT . . . 292
4.1. Espacio de direcciones . . . 293
4.2. Timers, interrupciones y DMA . . . 294
4.3. Bus AT . . . 296
5. PC-386 . . . 299
5.1. Chipset 82340 . . . 299
6. PC-Pentium . . . 301
6.1. ACPI . . . 302
6.2. Interfaz LPC . . . 303
6.3. Chipset 875P . . . 304
7. El PXA27x . . . 307
7.1. Controlador de memoria . . . 307
7.2. Controlador de DMA y E/S . . . 309
7.3. Controlador LCD . . . 311
7.4. Controlador de cámara . . . 311
8. Ejemplos . . . 312

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

8.1. Ejemplo de diseño de tarjeta de expansión 312
8.2. Cálculo de anchos de banda de interfaces 316
9. Bibliografía 317
10. Guía de estudio 320

6.- Microcontroladores **321**
1. Introducción 321
2. Fundamentos 322
2.1. Diagrama de bloques 322
2.1.1. UCP 322
2.1.2. UCM 322
2.1.3. UES 323
2.2. Clasificación de microcontroladores 324
2.3. Criterios de selección de microcontroladores 325
3. El 8051 de Intel 327
3.1. Espacio de direcciones 327
3.2. Registros de función especial 329
3.2.1. Registros de control de timers 332
3.2.2. Registros del puerto serie 333
3.2.3. Registros de gestión de interrupciones 334
3.2.4. Registro de control de alimentación 335
3.3. Modos de direccionamiento 335
3.4. Conjunto de instrucciones 336
3.4.1. Instrucciones de transferencia de datos 336
3.4.2. Instrucciones aritméticas 336
3.4.3. Instrucciones booleanas 337
3.4.4. Instrucciones lógicas 337
3.4.5. Instrucciones de salto 338
3.4.6. Ejemplos 339
3.5. Señales de interfaz 341
3.6. Ciclos de bus 342
3.7. Ejemplos 344
3.7.1. Programación de contador de eventos 344
3.7.2. Control de un depósito de agua 345
3.7.3. Programación del puerto serie 347
3.7.4. Entorno de desarrollo integrado de Keil 348
4. El ADuC842 de Analog Devices 350
5. Bibliografía 353
6. Guía de estudio 356

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

7.- Guía de diseño de un sistema basado en microprocesador 357
1. Introducción 357
2. Etapas de diseño de un SBM 358
2.1. Especificaciones del producto 359
2.2. Elección de la arquitectura 359
2.3. Bus del sistema 359
2.4. Elección de microprocesador 360
2.5. Partición del sistema 361
2.6. Descripción detallada del diseño 362
2.6.1. Diagrama eléctrico del circuito 363
2.7. Uso de lógica custom 363
2.8. Prototipo 364
2.9. Puesta a punto 364
2.10. Integración hardware/software 366
2.11. Análisis de rendimiento 366
2.12. Transferencia a PCB 366
3. Recomendaciones de diseño 367
3.1. Diseño de pistas de alta velocidad 367
3.2. Planos de masa y alimentación 368
3.3. Interferencias 369
3.4. Condensadores de desacoplo 370
3.5. Ejemplos 371
3.5.1. Diseño de layout 371
4. Bibliografía 372
5. Guía de estudio 373

Índice alfabético 375

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Prefacio

Any sufficiently advanced technology is indistinguishable from magic.
Arthur C. Clarke

"Manu estaba dormido cuando, de pronto, un cántico celestial inundó la alcoba y una insinuante voz repetía: Levántate y ve al templo. Al instante, Manu se despertó y alargando su mano golpeó con saña un pequeño cofre que irradiaba luces de colores. Manu no paraba de maldecir y de quejarse de algo que yo nunca llegué a entender. De pronto, la música cesó y la habitación se llenó de una gran luz blanca que me deslumbró. Cuando recuperé la vista, advertí que Manu se dirigía a la bodega. Lo seguí, y en la penumbra, observé aterrorizado cómo Manu hacía señas a un horrible dragón que al verle empezó a agitarse. Manu montó al dragón y éste, lanzando un rugido estremecedor, empezó a moverse torpemente hacia un portón echando un humo pestilente que casi me hace vomitar. El dragón parecía hablar pero Manu no le prestaba atención; sólo se concentraba en apuntar con el dedo al portón el cual como por arte de magia empezó a abrirse lentamente. Cuentan que al llegar al templo las puertas se van abriendo solas y las antorchas se encienden a su paso, mientras unos extraños seres colgados del techo clavan sus pupilas rojas en la nuca de Manu. En una oculta habitación hay un espejo mágico que Manu consulta, mediante conjuros que hace con los dedos, para saber cosas de este mundo y del otro. Dicen que, a veces, Manu cae en trance y llevándose una mano a la cabeza invoca a otros seres que parecen hablar con él. Pero yo no me lo creo".

Es posible que así experimentara un antepasado nuestro ciertas tareas de nuestra vida cotidiana. Como decía Clarke nuestra tecnología actual parecería cosa del demonio a nuestro asustado ancestro. Sin embargo, para nosotros es algo tan habitual que no le prestamos la menor atención:

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

"Manuel estaba dormido cuando, de pronto, una música atronadora procedente del reloj despertador inundó la habitación; el display de números rojos marcaba las 7 en punto y un dicharachero locutor sólo sabía repetir: Levántate que ya es tiempo. Manuel golpeó con saña al reloj hasta conseguir enmudecerlo, mientras protestaba inútilmente por haber programado la alarma una hora más tarde. A continuación, encendió la luz de la habitación y de un salto, cogió la ropa y empezó a vestirse mientras se encaminaba al garaje de la casa. Pulsó el mando a distancia de la llave de su coche y éste desbloqueó las puertas y encendió las luces de cortesía. Sin demora, arrancó, pisó fuerte el acelerador, metió la primera marcha y maniobró hasta la puerta. Allí encendió la radio para oír las noticias y apretó insistentemente el mando a distancia de la puerta que no funcionaba bien por culpa de las pilas. Manuel trabaja en un edificio inteligente donde hay muchas videocámaras de vigilancia y sensores de presencia para la iluminación del edificio y la apertura automática de casi todas las puertas. Manuel es agente comercial y se pasa toda la mañana delante del ordenador, tecleando correos electrónicos y generando informes rutinarios. Cada dos por tres, llama desde su móvil de última generación, regalo de la empresa, y no usa la agenda sino que, gracias a la tecnología de reconocimiento de voz, basta con que diga el nombre del cliente".

Realmente estamos "inundados" de cientos de sistemas digitales que juegan un papel discreto pero agradecido en muchas de nuestras actividades diarias. No hace muchos años, no sabíamos lo que era un mando a distancia y, hoy en día, no sabemos vivir sin él. La televisión, los reproductores de discos compactos y DVD, los móviles y muchos otros equipos de consumo son ya artículos de casi primera necesidad. Todos ellos son sistemas digitales que se han convertido en uno de los fundamentos de nuestra tecnología y, por qué no decirlo, de nuestra sociedad.

Los sistemas digitales, basados en el álgebra de Boole, nos permiten resolver problemas en un espacio binario donde sólo existen dos estados: El cero y el uno. No sólo nuestros números y letras se pueden codificar en binario, también nuestra voz, la música, las imágenes, incluso la materia[1]. Cualquier información se puede reducir a un formato digital a partir del cual se puede modificar, transportar, almacenar y mezclar con otras informaciones.

Sin embargo, el álgebra booleana es tan simple que si los cálculos binarios los tuviera que hacer un ser humano, éste se pasaría años para resolver

[1]Hoy en día, se pueden usar técnicas de realidad virtual para capturar las propiedades de un objeto (geometría, textura, tacto, etc.) y generar una copia virtual que se puede teletransportar a un lugar remoto donde se puede ver y tocar.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

cualquier problema medianamente complejo. Afortunadamente, disponemos de la tecnología microelectrónica, que permite crear microscópicos[2] elementos binarios tanto de proceso (microprocesadores) como de memoria que son capaces de ejecutar billones de operaciones por segundo. Esta es una de las claves[3] del éxito de los sistemas digitales programables que llamamos ordenadores.

Este manual estudia los sistemas basados en microprocesadores desde un punto de vista hardware. Consta de 7 capítulos:

1. Introducción
2. Microprocesadores: Arquitectura IA-32 de Intel
3. Memorias
4. Entrada/Salida
5. Microcomputadores
6. Microcontroladores
7. Guía de diseño de un sistema basado en microprocesador

El libro va destinado a estudiantes con conocimientos previos de sistemas digitales: Álgebra de Boole, diseño digital lógico y físico, fundamentos de microprocesadores y de su interfaz a memoria y entrada/salida, etc. No obstante, el primer capítulo repasa todos aquellos conceptos básicos[4] necesarios para la compresión de la materia.

Todos los capítulos incluyen ejemplos que permiten asimilar conceptos y/o acercan la teoría a la práctica. La bibliografía que se proporciona es muy extensa e incluye muchas páginas WWW que se deberían de consultar, ya que hablamos de una tecnología que se queda obsoleta en menos de un año.

[2]Hoy en día ya habría que decir nanoscópicos.

[3]La otra clave es la filosofía de diseño de niveles de abstracción que permite diseñar sistemas muy complejos descomponiendo su arquitectura y funcionamiento en niveles. El diseñador de un nivel hace uso del nivel inmediatamente inferior sin necesidad de saber cómo funciona internamente y, a la vez, proporciona servicio al nivel superior sin implicarlo en sus operaciones particulares. La revolución tecnológica le debe mucho a esta forma de trabajar.

[4]Este primer capítulo no describe en profundidad estos conceptos básicos, de manera que si el estudiante tiene dificultad con ellos debería también consultar la bibliografía recomendada.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Aunque el libro incluye mucha información de utilidad para el análisis y diseño de sistemas basados en microprocesador, no pretende, en ningún caso, sustituir a los manuales de diseño software y hardware que serán imprescindibles para abordar cualquier proyecto[5]. Al final de cada capítulo, se incluye una guía de estudio que ayudará al estudiante a concentrarse en los aspectos que se consideran más importantes[6].

Este manual es fruto de la experiencia docente del autor en varias asignaturas de sistemas digitales impartidas en la E.T.S.I. de Telecomunicación de la Universidad de Málaga desde hace más de 10 años. Su contenido cubre la materia necesaria para cuatro de las asignaturas impartidas en este centro: Diseño de Sistemas Digitales y Laboratorio de Diseño de Sistemas Digitales en quinto curso de Ingeniería de Telecomunicación y en tercer curso de Ingeniería Técnica de Telecomunicación especialidad Sistemas Electrónicos. A continuación, se describe el contenido de cada capítulo.

El tema 1 es una introducción a los sistemas basados en microprocesadores. Se repasan algunos conceptos básicos necesarios para la comprensión del resto de la materia. También se comentan los hitos más importantes en la reciente historia de los microprocesadores y microcomputadores, apuntando también cuáles pueden ser las tendencias en los años venideros.

El tema 2 estudia la arquitectura de microprocesadores IA- 32 de Intel, repasando las características más relevantes de todos los miembros de esta gran familia, desde el 8086 hasta el Core Duo. En particular, explica la microarquitectura de los microprocesadores Pentium, Pentium II y Pentium 4. A continuación, estudia el entorno de ejecución básico (modos de operación, organización de memoria, registros, tipos de datos, etc.), el conjunto de instrucciones y el entorno de programación del sistema, haciendo especial hincapié en las facilidades que ofrece para la gestión de memoria (multitarea, memoria virtual y protección de datos). Finalmente, muestra las señales de interfaz del Pentium y del Pentium II y sus técnicas de acceso a memoria y E/S. El tema incluye ejemplos de uso del depurador del PC, traducción de direcciones, programación con instrucciones MMX y comparativa de ejecución entre microprocesadores Pentium.

El tema 3 engloba el estudio de todos los tipos posibles de memoria: Memorias semiconductoras, memorias magnéticas y memorias ópticas. Se estudian las memorias semiconductoras no volátiles (ROM, PROM, EPROM, EEP-

[5]Ver las referencias al final de cada capítulo.

[6]Las cuestiones que se plantean en la guía de estudio tienen relación directa con los objetivos docentes específicos que este manual pretende conseguir.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

ROM y Flash) y volátiles (DRAM y SRAM). Dentro de las memorias DRAM se hace un estudio de su evolución (FPDRAM, EDO DRAM, BEDO DRAM, SDRAM, RDRAM, DDR, etc.) y se describen aspectos relacionados con su empaquetado y con la detección y corrección de errores. Un estudio similar se lleva a cabo con las memorias SRAM (ASRAM, PBSRAM, QDR, etc.). A continuación, se estudian las memorias magnéticas, prestando especial atención al disco duro (arquitectura y funcionamiento). Seguidamente se presentan las memorias ópticas (CD y DVD) y sus diferentes formatos. Se hacen ejemplos de mapeo de memoria, cálculos de capacidad, tiempos de acceso y anchos de banda.

El tema 4 está dedicado a la interfaz de E/S de los sistemas digitales y, en particular, de los ordenadores personales compatibles IBM. En primer lugar, se estudian dos controladores de Intel de gran importancia en la arquitectura de los PC: El controlador de interrupciones 8259 y el controlador de DMA 8237. A continuación, se describen los dos puertos básicos de un PC: El puerto serie RS-232C y el puerto paralelo IEEE 1284. Seguidamente, se presentan los buses de E/S, en particular los buses internos PCI, PCI express e I2C y los externos USB, SCSI y ATA. El tema continúa con el estudio de interfaces a periféricos de PC e incluye teclado, ratón, joystick, pantallas de visualización CRT y TFT, subsistemas gráficos, SoundBlaster16, AC´97 e interfaz MIDI. Se hacen ejemplos de acceso a E/S, programación de interrupciones, DMA y SoundBlaster16.

El tema 5 presenta sistemas basados en microprocesadores de propósito general que denominaremos microcomputadores. En primer lugar, se hace un estudio detallado de la arquitectura del IBM PC, base de todas las arquitecturas posteriores de IBM. Seguidamente se muestran las diferencias con el PC-XT y, con más detalle, con el PC-AT. Posteriormente, se estudian plataformas basadas en chipset para 8086 y Pentium, se hacen algunas comparativas y se describen nuevas características como ACPI y LPC. A continuación, se describe el sistema integrado PXA27x de Intel basado en la microarquitectura XScale. Para terminar, se hacen ejemplos de diseño de tarjetas de expansión y de cálculo de anchos de banda de interfaces.

El tema 6 estudia los microcontroladores. En primer lugar, enseña la arquitectura y funcionamiento de un microcontrolador típico. A continuación, hace un estudio comparativo entre familias de diferentes fabricantes y proporciona una guía de selección del más adecuado para cada aplicación. Seguidamente, se estudia el microcontrolador 8051 de Intel: Su modelo software, conjunto de instrucciones y arquitectura hardware. Posteriormente, se hacen algunos

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

ejemplos de programación y manejo de sistemas de desarrollo integrados (en particular, μvision 2 de Keil). Finalmente, se presenta el microconvertidor 842 de Analog Devices, un potente microcontrolador compatible con el conjunto de instrucciones del 8051.

El tema 7 muestra, sin entrar en profundidad, la típica metodología de diseño de los sistemas basados en microprocesadores, dando algunas recomendaciones de diseño y aportando algunos consejos basados en la experiencia. Se hacen algunos ejemplos sobre diseño de placas de circuito impreso.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

1 Introducción

1. Introducción

La convergencia de la microelectrónica, informática y telecomunicaciones ha revolucionado nuestra sociedad. Hoy en día se habla de una Sociedad de la Información y del Conocimiento donde textos, sonidos, voz, imágenes y videos se codifican digitalmente y se convierten en datos binarios que se pueden transformar y/o transportar fácilmente.

La revolución comenzó en los años setenta con la aparición del microprocesador. En los años 80, los microprocesadores facilitaron la llegada de los ordenadores personales (PC, Personal Computer) en un momento en el que nadie podía sospechar el éxito que alcanzarían. En los 90, triunfó el PC multimedia y su conexión a red junto con el auge de la telefonía móvil de segunda generación. La llegada del siglo XXI ha consolidado Internet y ha puesto las bases para el despliegue de la tercera generación de móviles. Los siguientes pasos apuntan a entornos inteligentes donde terminales multimedia inalámbricos se comunicarán de forma espontánea con dispositivos próximos y podrán acceder a cualquier red de comunicaciones.

En este capítulo, se repasan conceptos básicos de sistemas basados en microprocesadores, se establecen algunas definiciones y se comentan los hitos más importantes en la reciente historia de los microprocesadores y microcomputadores, apuntando también cuáles pueden ser las tendencias en los años venideros.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2. Algunos conceptos básicos

Un microcomputador es un sistema digital de alta integración que resuelve problemas ejecutando una secuencia de instrucciones que denominamos programa. Un microcomputador básico consta de tres elementos fundamentales (figura 1.1): Unidad Central de Proceso o UCP, Unidad Central de Memoria o UCM[7] y Unidad de Entrada/Salida (E/S) o UES.

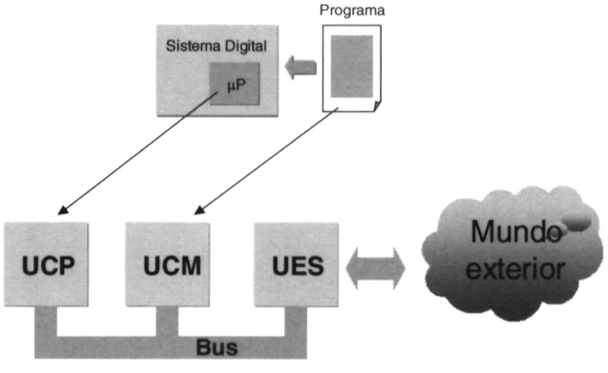


Figura 1.1: Diagrama de bloques básico de un sistema basado en microprocesador

La UCM es la encargada de almacenar el programa, la UCP de ejecutarlo y la UES de comunicar al sistema con el mundo exterior. Esta tres unidades se conectan entre sí mediante un conjunto de señales denominado bus[8].

El bus transfiere información de dirección, datos y control. Las señales de dirección permiten seleccionar localizaciones de memoria de la UCM o puertos de E/S de la UES, las señales de datos transportan instrucciones y datos entre unidades y las señales de control manejan la transferencia definiendo comienzo, duración, tipo, arbitración, etc..

Los buses pueden ser paralelo o serie. En un bus paralelo clásico cada señal de dirección, datos o control tiene asociada una línea física, mientras que en un bus serie todas las señales van por las mismas líneas físicas[9].

[7]En los sistemas actuales de propósito general, la memoria suele estar distribuida a lo largo de todo el sistema (ver apartado 2.3.1).

[8]En sistemas muy sencillos hay un único bus que comunica al procesador con la memoria y con los diferentes dispositivos de E/S. Sin embargo, en los sistemas de propósito general actuales hay múltiples buses: bus del microprocesador, bus de memoria, buses locales, buses de E/S, etc.

[9]También es típico encontrar buses paralelo que para reducir su número de señales las

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.1. Microprocesadores

Los microprocesadores son UCPs integradas en un chip[10]. La arquitectura básica de un microprocesador contiene: Una unidad de control para buscar las instrucciones y determinar su tipo, una unidad aritmético lógica para llevar a cabo las operaciones que requieren las instrucciones y un conjunto de registros para guardar información de control, operandos de entrada y resultados intermedios[11]. La figura 1.2 muestra un ejemplo de arquitectura de microprocesador.

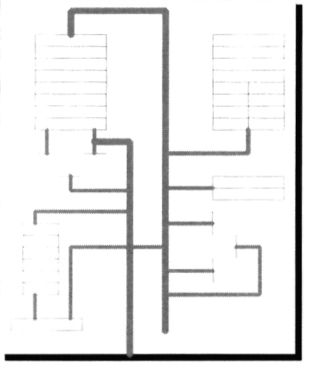


Figura 1.2: Microarquitectura del microprocesador 8086 de Intel

El espacio de direcciones de un microprocesador es el conjunto de direcciones a las que puede acceder. Este espacio es de $2^n - 1$ direcciones siendo n el número de señales del bus de direcciones del microprocesador.

Los microprocesadores son capaces de interpretar y ejecutar un conjunto determinado de instrucciones. Es el denominado lenguaje máquina.

multiplexan. Puede ver un ejemplo de esta técnica en el chip 8051 de Intel que se explica en el capítulo de Microcontroladores.

[10]Hoy en día, prácticamente todas las UCPs son microprocesadores.

[11]Sin embargo, las microarquitecturas actuales suelen ser mucho más complejas. Profundos pipelines, memorias cache, predicción de saltos, arquitecturas superescalares y ejecución especulativa son algunas de las características que consiguen mejorar drásticamente la capacidad de proceso de los nuevos microprocesadores. En apartados posteriores se ampliará esta información.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.2. Programación

El lenguaje máquina es idóneo para el hardware del microprocesador pero inadecuado[12] para el usuario que tenga que programarlo. Para resolver este problema, el programador escribe su programa en un lenguaje más apropiado a su forma de expresarse y luego hace uso de un traductor[13] o un intérprete[14] para convertirlo al lenguaje máquina del microprocesador (figura 1.3). Esta idea de abstraerse del lenguaje máquina ha sido muy provechosa y ha dado lugar a los sistemas con varios niveles de abstracción.

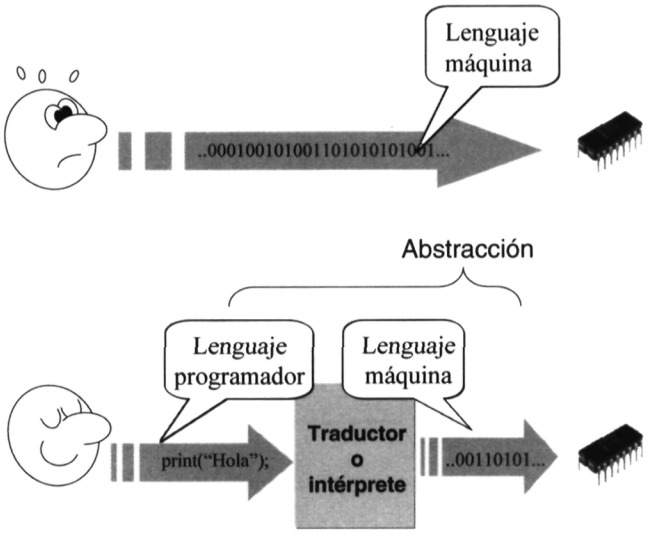


Figura 1.3: Lenguaje máquina, traductores e intérpretes

En la figura 1.4 se puede ver un ejemplo de un microcomputador con varios niveles de abstracción. El programador del sistema trabaja con los tres niveles más bajos (1,2 y 3)[15] mientras que el programador de aplicaciones sólo utiliza los niveles más altos (4 y 5).

En el nivel más bajo está el hardware del sistema que es el que realmente ejecuta las instrucciones. El segundo nivel es el lenguaje máquina del micro-

[12]Sería mejor decir incomprensible, ya que cada instrucción es una secuencia de ceros y unos.

[13]El traductor convierte el programa completo a lenguaje máquina antes de ejecutarlo.

[14]El intérprete va convirtiendo y ejecutando las instrucciones del lenguaje del programador una detrás de otra.

[15]El programador del sistema también implementa los traductores e intérpretes entre niveles.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

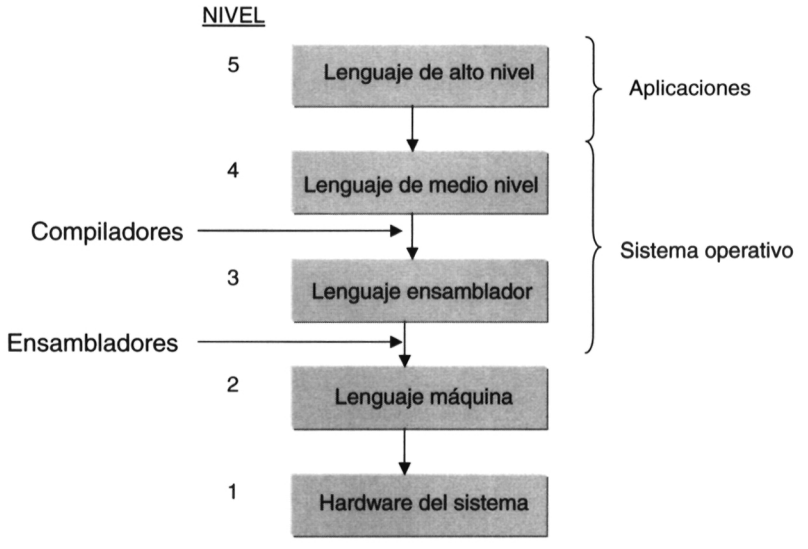


Figura 1.4: Ejemplo de niveles de abstracción de un microcomputador

procesador, una secuencia de instrucciones que el microprocesador reconoce y sabe ejecutar. El tercer nivel es el lenguaje ensamblador que proporciona un lenguaje simbólico del lenguaje máquina, aunque no deja de ser un lenguaje de bajo nivel (tedioso de programar) y específico de un microprocesador. El traductor de lenguaje ensamblador a lenguaje máquina se denomina ensamblador. El cuarto nivel es un lenguaje de medio nivel (por ejemplo, lenguaje C) que permite programar el hardware con potentes funciones pero que es independiente de la máquina[16]. El traductor de este lenguaje a lenguaje ensamblador se denomina compilador. Finalmente, el último nivel es un lenguaje de alto nivel u orientado al usuario que describe el programa en un lenguaje casi natural (C++, Pascal, etc.) y que deja la programación del hardware para niveles inferiores.

El sistema operativo (S.O.) es un conjunto de funciones escritas en lenguaje de medio y/o bajo nivel que ejecutan tareas rutinarias del sistema[17] y que liberan al programador de aplicaciones de tener que programarlas.

[16]Se puede ejecutar en máquinas con diferente hardware, es decir con diferentes microprocesadores.

[17]Por ejemplo, leer un carácter de teclado, presentar datos en pantalla, imprimir un fichero, etc.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.2.1. Multitarea

La multitarea permite compartir los recursos del microprocesador entre varios programas o tareas que parecen ejecutarse de forma simultánea. La figura 1.5 ilustra esta idea.

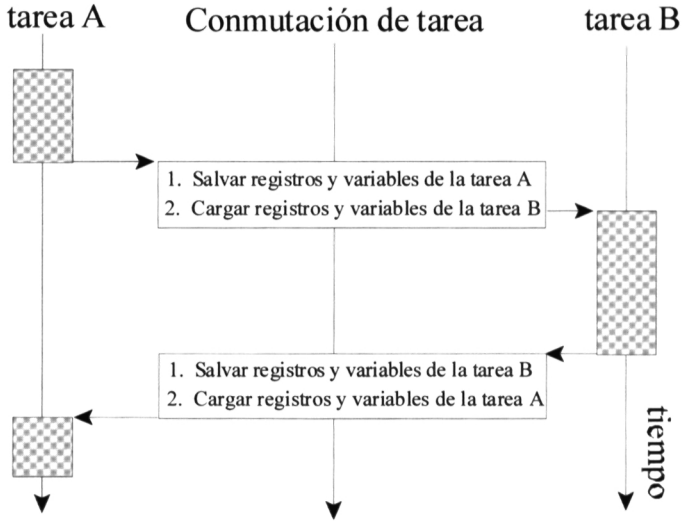


Figura 1.5: Conmutación de tareas

La tarea A se ejecuta durante cierto tiempo. En un momento determinado el S. O. decide cambiar de tarea[18]. Para ello, previamente salva los registros del microprocesador y las variables de la tarea A (que queda, de esta forma, congelada a la espera de poder reanudarse más adelante). A continuación, se recargan los registros del microprocesador con los valores que tenían cuando se abandonó la tarea B, así como las variables del programa. A partir de ese instante puede empezar a ejecutarse la tarea B que se mantendrá hasta que el S.O. decida conmutar de nuevo.

[18]Hay muchas formas de implementar la conmutación de tareas dependiendo de que haya prioridad entre tareas o no o de que el sistema trabaje en tiempo real.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.3. Memorias

La memoria se podría definir como un conjunto de registros que almacenan información. Los registros suelen tener un tamaño de 1 byte[19]. Los registros se numeran de 0 a $2^n - 1$, donde n es el número de señales del bus de direcciones que se requieren para poder acceder a cualquiera de ellos. El tamaño de la memoria se suele medir en bytes o múltiplos de bytes[20].

La figura 1.6 muestra la organización física de la memoria (izquierda) donde cada byte tiene su dirección. Sin embargo, el microprocesador también puede acceder directamente a bloques de bits más grandes: palabras (dos bytes consecutivos), dobles palabras (4 bytes consecutivos), etc.. En este caso, la dirección de estos bloques será la dirección de uno de sus bytes.

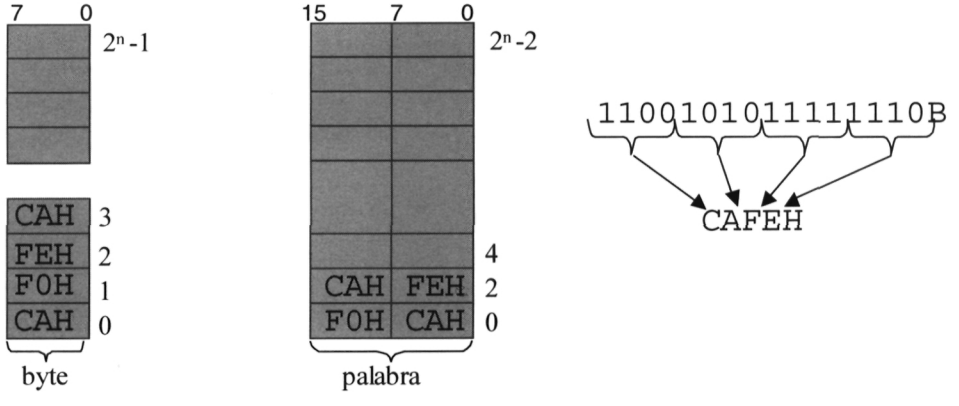


Figura 1.6: Organización física de memoria

Se dice que es una organización *little endian* cuando la dirección del bloque de bits es la dirección de su byte menos significativo. Por el contrario se dice que es *big endian*, cuando la dirección del bloque de bits es la dirección de su byte más significativo[21]. En la figura 1.6 (derecha) la palabra de dirección `2H` contiene el dato `CAFEH` si el convenio es *little endian* y `FECAH` si es *big endian*.

[19] 1 byte=8 bits.

[20] 1Kbyte (Kilo) = 2^{10} bytes, 1Mbyte (Mega) = 2^{20} bytes, 1Gbyte (Giga) = 2^{30} bytes, 1Tbyte (Tera) = 2^{40} bytes, 1Pbyte (Peta)= 2^{50} bytes, 1Ebyte (Exa) = 2^{60} bytes, 1Zbyte (Zetta) = 2^{70} bytes y 1Ybyte (Yotta)= 2^{80} bytes.

[21] Esta nomenclatura procede de un artículo de Cohen del año 1981 (ver bibliografía) y se refiere a un cuento que se relata en "Los Viajes de Gulliver"de Jonathan Swift donde se provocó una guerra debido a una discusión sobre por dónde se debían de romper los huevos, si por la parte grande o por la pequeña.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

La memoria puede ser semiconductora, magnética u óptica. La primera se utiliza en todas las unidades centrales del sistema: UCP, UCM y UES, mientras que las dos últimas son exclusivas de la UES (para periféricos de almacenamiento masivo).

La memoria semiconductora puede ser volátil o no volátil.

La memoria no volátil conserva la información almacenada aunque se retire la alimentación del sistema; es básicamente una memoria de lectura (ROM, Read Only memory) aunque algunas implementaciones admiten su borrado y re-escritura[22].

Las memorias volátiles pierden la información almacenada cuando se retira la alimentación. Son memorias muy rápidas de lectura y escritura. Se denominan memorias RAM (Random Access Memory) ya que son de acceso directo (no secuencial). Se clasifican en memorias DRAM (Dynamic RAM) y memorias SRAM (Static RAM).

La figura 1.7 compara las memorias SRAM y DRAM en función de su coste, densidad y velocidad. Se puede observar que las memorias DRAM son más baratas y ocupan menos espacio (mayor densidad) que las SRAM; sin embargo, son más lentas.

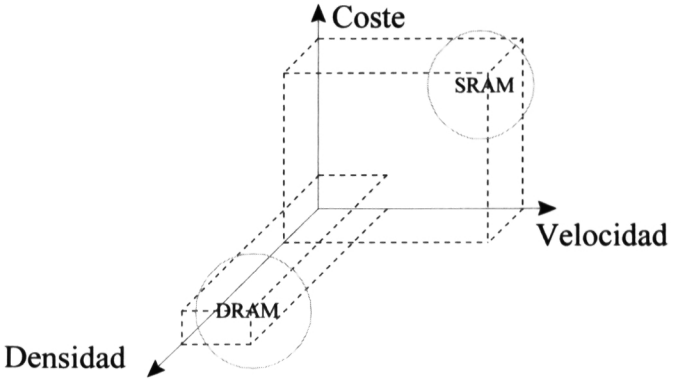


Figura 1.7: Comparativa entre memorias SRAM y DRAM

[22]PROM, EPROM, EEPROM, Flash, etc.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.3.1. Jerarquía de memoria

En un microcomputador es muy probable que la información solicitada por el microprocesador:

1. Vuelva a ser requerida por el microprocesador (localidad temporal). Por ejemplo, las instrucciones de un bucle, los datos de una pila o las variables del programa
2. Esté en direcciones de memoria consecutivas(localidad espacial). Es la estructura típica de un programa o la de una cadena de datos.

Para aprovechar este comportamiento, la jerarquía de memoria distribuye de forma inteligente la memoria a lo largo de todo el sistema (ver figura 1.8). Las memorias más próximas al microprocesador (niveles bajos) son muy rápidas aunque muy costosas, mientras que las más alejadas son cada vez más lentas aunque mucho más baratas y densas. El objetivo es trasvasar la información de unas memorias a otras de manera que el microprocesador siempre tenga a su disposición las instrucciones que necesita.

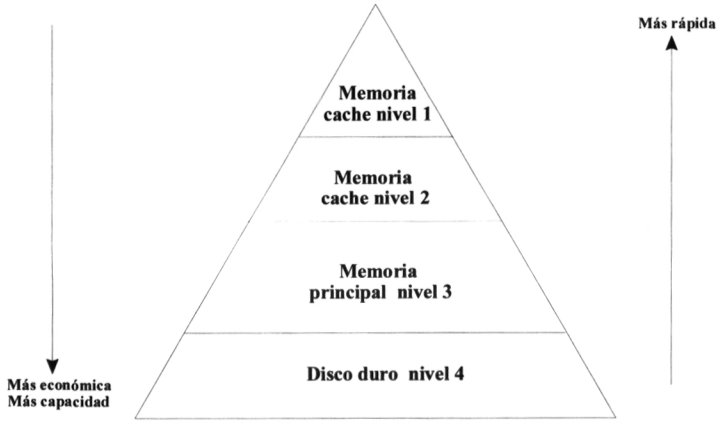


Figura 1.8: Jerarquía de memoria

Un típico ordenador personal usa un complejo sistema de jerarquía de memoria que requiere el uso de muchas tecnologías:

- SRAM para las memorias cache que guardan una copia de las instrucciones y datos que han sido (o podrían ser) recientemente usados por el microprocesador.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- DRAM para la memoria principal o UCM, también llamada primaria, que almacena los programas y datos que está usando (o va a usar a corto plazo) el microprocesador.
- Memorias magnéticas u ópticas para la memoria que guarda todos los programas y datos del sistema que no están siendo ejecutados (o no van a serlo a corto plazo) por el microprocesador. Su forma de acceso difiere de la memoria primaria.

El elemento fundamental de la jerarquía de memoria es la memoria cache.

2.3.2. Memoria cache

En la figura 1.9 se puede ver un sistema clásico de memoria cache. El microprocesador busca las instrucciones y datos en la cache. Cuando no las encuentra (fallo de cache), va a buscarlas a memoria principal y, de paso, las copia en la cache.

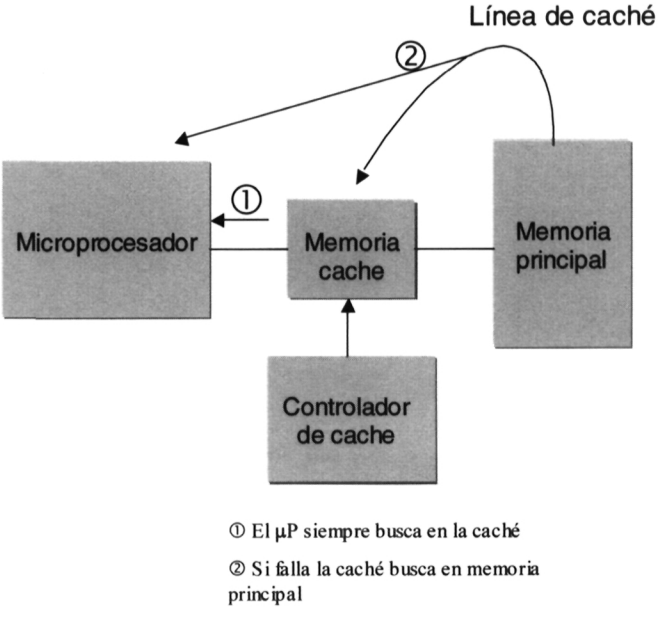


Figura 1.9: Sistema de memoria cache

Las instrucciones y datos llegan a la cache en bloques, también llamados líneas de cache. Estos bloques son los datos de varias direcciones consecutivas

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

de la memoria principal. El tamaño del bloque es una solución de compromiso entre rendimiento y coste[23].

Un controlador de cache es el encargado de gestionar todo el proceso. El controlador de cache decide qué información se mantiene, cuál se sobre-escribe y cuándo se actualizan los datos en memoria principal.

2.3.2.1. Actualización de caches Hay varias estrategias de escritura: Reescribir los bloques en modo secuencial, en modo aleatorio o en modo LRU (Least Recently Used)[24]. El rendimiento de cada estrategia depende del comportamiento del programa.

Para actualizar los datos de la cache en el sistema hay varias técnicas: Escritura continua (write through), escritura continua con buffer (write buffering) y postescritura (write back).

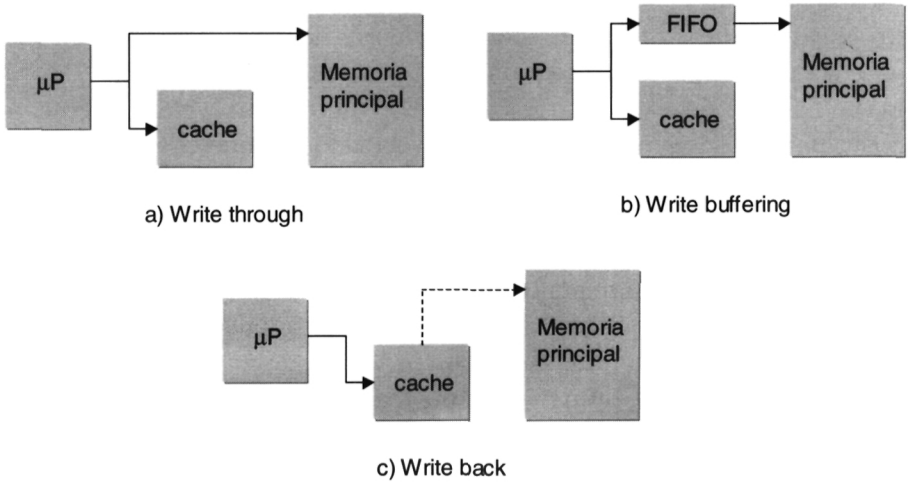


Figura 1.10: Estrategias de actualización de cache

En la escritura continua el controlador copia los datos de escritura en la memoria principal inmediatamente después de ser escritos en la cache. Es el método más sencillo pero también el de menos rendimiento.

[23]Si el tamaño del bloque es demasiado pequeño la tasa de aciertos se puede reducir. Sin embargo, si es muy grande habrá menos bloques en la cache, más lejos estará el dato adicional del solicitado y por tanto menos probable será su uso; esto obliga a ampliar la memoria y su velocidad. Las memorias cache pueden conseguir tasas de acierto superiores al 90 %.

[24]En modo LRU, el controlador necesita un flag LRU por bloque que indique el bloque a reescribir. Los flags se actualizan en cada transacción de cache.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

En la escritura continua con buffer se emplea un buffer para los accesos a memoria principal; de esta forma el procesador puede arrancar un nuevo ciclo de bus antes de que el ciclo a memoria principal haya terminado[25].

En postescritura, los bloques que se reescriben en la cache no se actualizan en memoria principal hasta que no sea necesario. Es el método de mayor rendimiento aunque el más costoso[26].

2.3.2.2. Arquitecturas de caches Hay varios tipos de memorias cache: Mapeo directo, completamente asociativa, asociativa por conjuntos, etc.

En la figura 1.11 se ve un ejemplo de memoria cache completamente asociativa. Las direcciones de memoria `A[31:0]` se descomponen en tres campos: campo de línea de cache de 5 bits (`A[4:0]`, campo de etiqueta de 15 bits (`A[19:5]`) y campo de direcciones altas de 12 bits (`A[31:20]`). La memoria cache almacena líneas de cache a las que asocia una etiqueta. Cuando el microprocesador genera una dirección de memoria, la memoria cache compara el campo de etiqueta de esta dirección (`A[19:5]`) con todas sus etiquetas. Si coincide con alguna de ellas se habrá producido un acierto de cache. La memoria completamente asociativa puede almacenar cualquier línea de cache lo que le da máxima flexibilidad. Su inconveniente es que requiere hacer muchas comparaciones, lo que redunda en retardo y coste.

Las caches de mapeo directo sólo requieren una comparación. Su inconveniente es que son mucho menos flexibles que las completamente asociativas aunque mucho más económicas. En la figura 1.12 se puede ver un ejemplo de memoria cache de mapeo directo. En este caso, las direcciones de memoria se descomponen en cuatro campos: campo de línea de cache de 5 bits, campo de índice de 8 bits, campo de etiqueta de 6 bits y campo de direcciones altas de 12 bits. La memoria cache almacena líneas de cache a las que asocia una

[25] Si a un acceso de escritura sigue otro de lectura que sea un acierto en la cache, el acceso de lectura se podrá realizar mientras la memoria principal se actualiza. Si embargo si hay dos accesos de escritura seguidos a memoria principal, el procesador tiene que esperar. Un fallo en la lectura también provoca que el procesador tenga que esperar.

[26] En postescritura el campo de etiqueta de cada bloque de la cache incluye un bit de alteración. Este bit se activa cada vez que el bloque es reescrito con nuevos datos y, por tanto, contiene datos más recientes que la memoria principal. Cuando se va a escribir un bloque de la cache si el bit de alteración está activado 1) escribe el bloque en la memoria y 2) escribe los nuevos datos en la cache; en caso contrario escribe los nuevos datos en la cache. La lógica controladora es más compleja. Todos los bloques alterados se deben escribir en la memoria principal antes de que otro procesador acceda a memoria principal. Si el sistema cae, los datos de la cache se pierden y no sabremos qué datos de la memoria principal están viciados.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

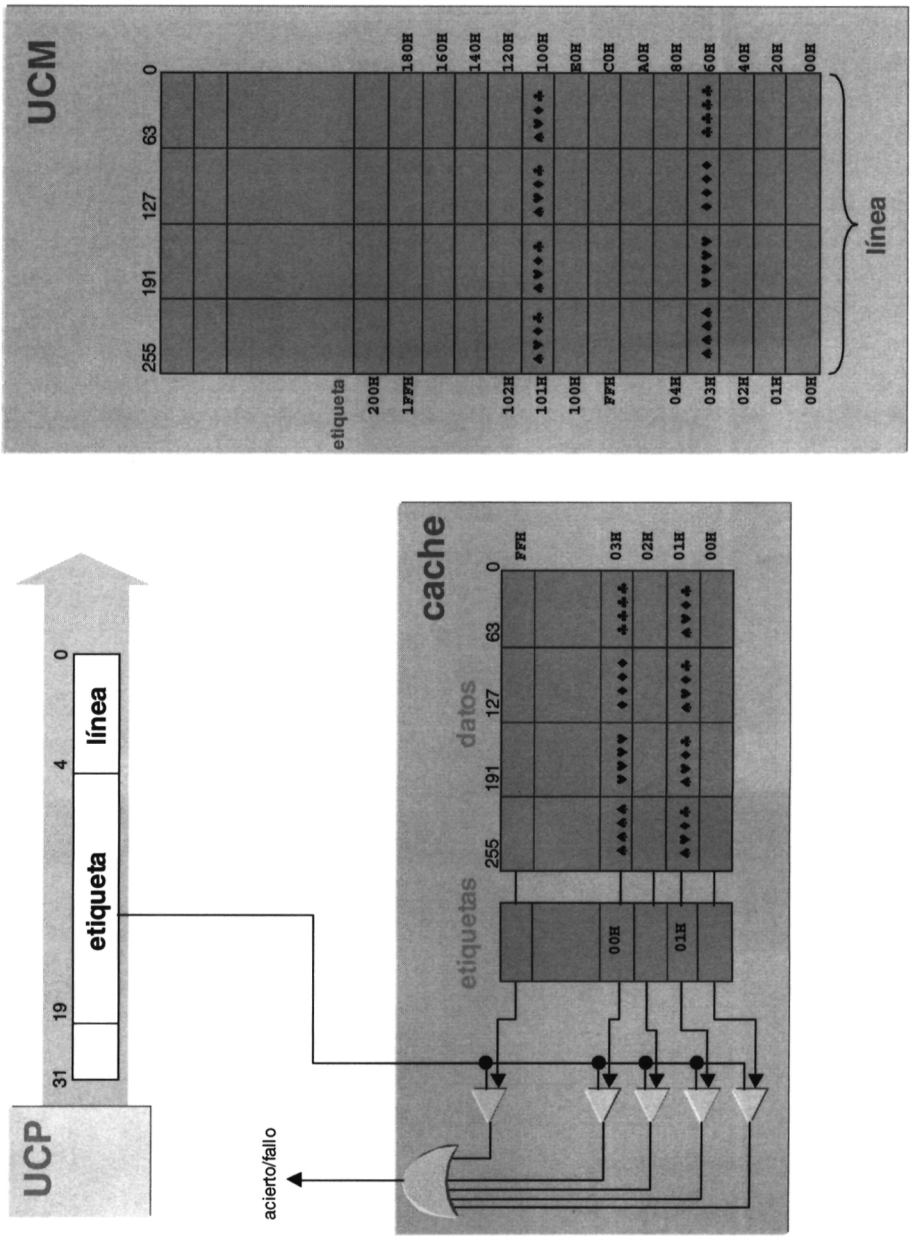


Figura 1.11: Ejemplo de memoria completamente asociativa

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

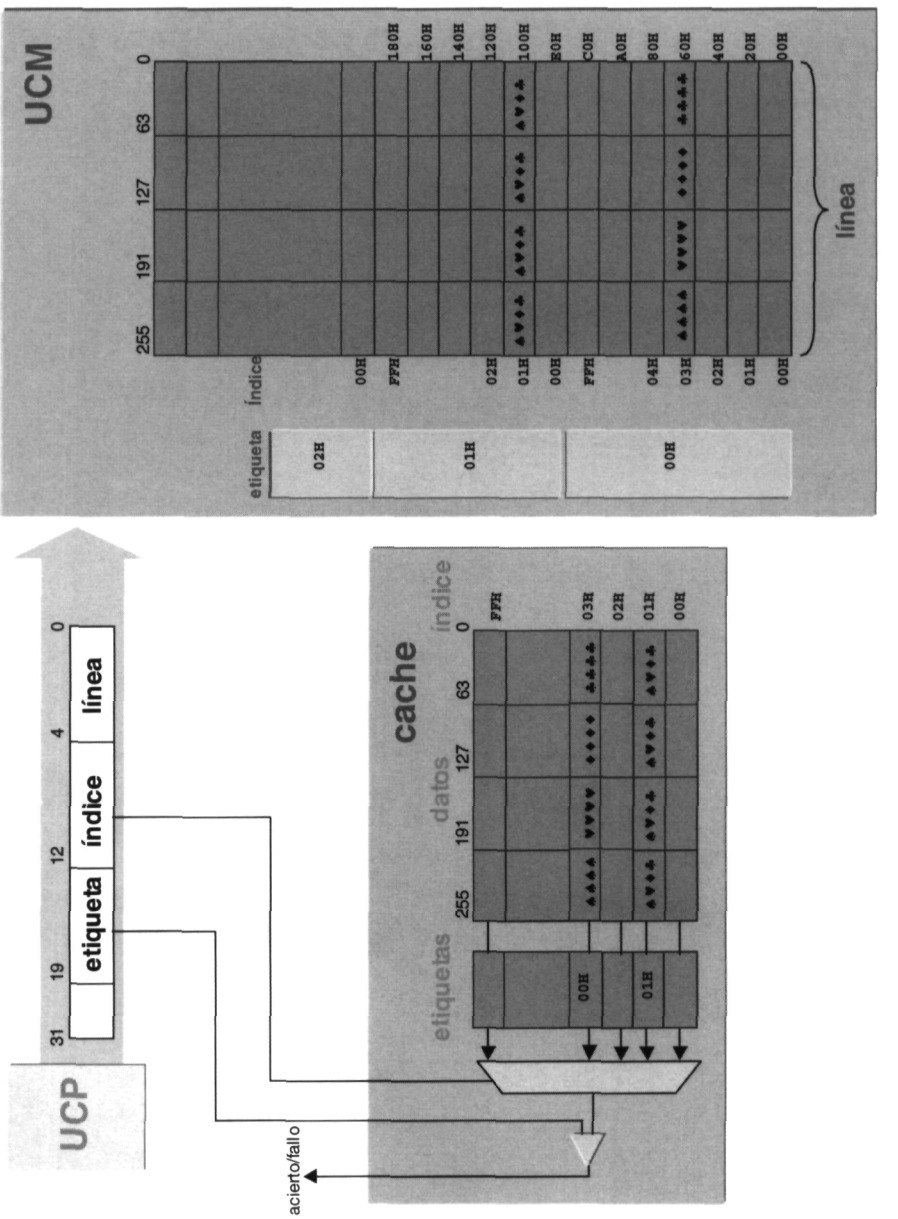


Figura 1.12: Ejemplo de memoria cache de mapeo directo

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

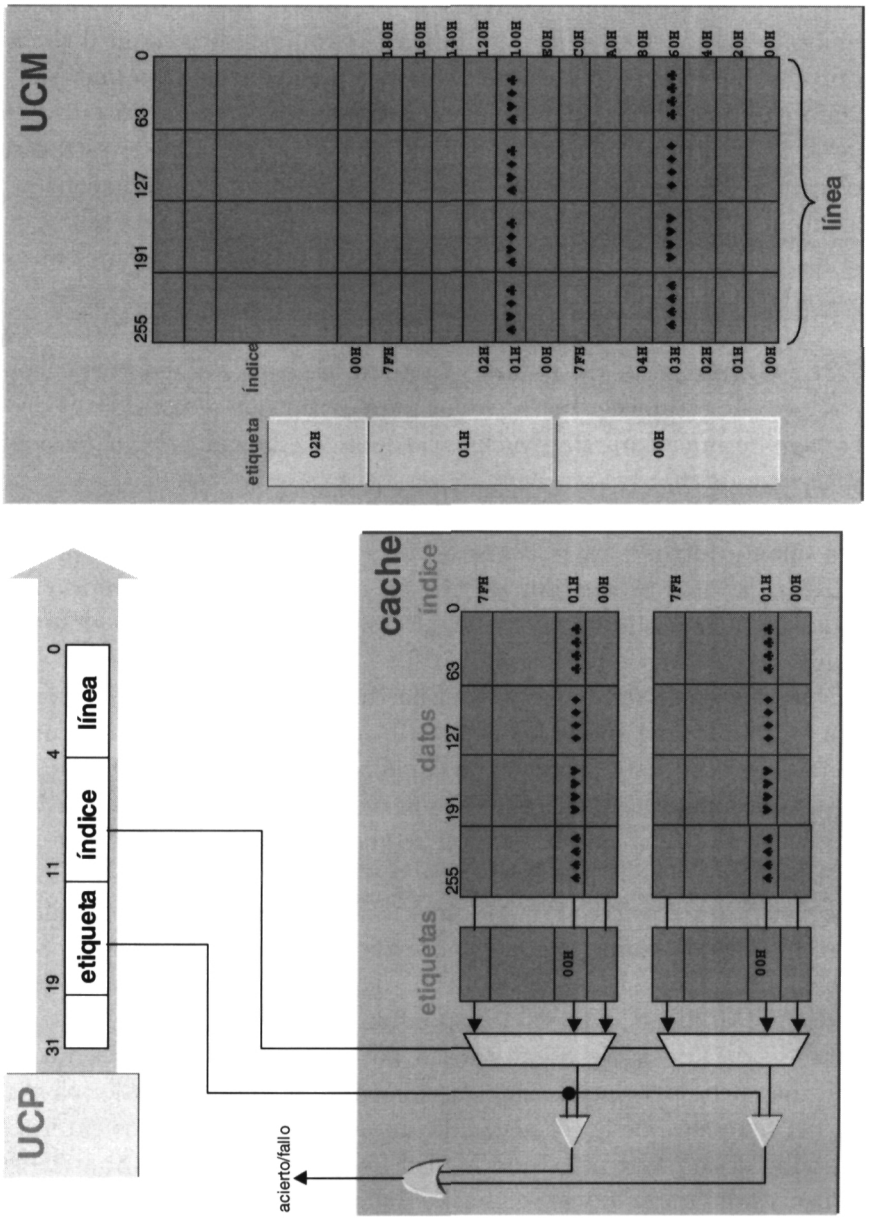


Figura 1.13: Ejemplo de memoria cache asociativa de dos conjuntos

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

etiqueta pero de forma muy restrictiva: La dirección de almacenamiento en la memoria cache debe ser la que indica el campo índice de la dirección del microprocesador. Esto significa que no podrá almacenarse otra línea que tenga el mismo campo índice. Cuando el microprocesador genera una dirección de memoria, la memoria cache compara el campo de etiqueta de esta dirección con el campo de etiqueta de la linea de cache que tiene almacenada en la dirección índice. Si coincide se habrá producido un acierto de cache.

La memoria asociativa por conjuntos es una solución de compromiso entre las dos anteriores (figura 1.13).

2.3.2.3. Coherencia de cache Cuando hay varias memorias cache en el sistema hay que establecer mecanismos para evitar que se usen datos obsoletos (coherencia de la cache). Hay varios métodos: vigilancia del bus (snooping) y transparencia del hardware.

En el snooping el controlador de cache monitoriza el bus de direcciones del sistema cuando otros procesadores están accediendo a la memoria principal compartida. Si otro procesador escribe en una posición de memoria compartida que también reside en su cache, el controlador invalida el bloque que la contiene.

En transparencia del hardware, el hardware garantiza la coherencia de la cache asegurando que todos los accesos a memoria mapeados por una cache son vistos por ella. Esto se consigue canalizando los accesos de todos los dispositivos a la memoria principal a través de la misma cache.

El protocolo MESI usa la técnica de snooping para controlar la coherencia de las caches. Cada línea de cache puede estar en alguno de los cuatro estados del protocolo MESI[27]: Modified, Exclusive, Shared o Invalid. Tanto el microprocesador como el controlador de cache u otros microprocesadores pueden modificar el estado de las líneas de cache. Cuando una linea está en el estado M (Modified) sólo está disponible en una cache y puede ser accedida sin necesidad de accesos externos. Una linea en estado E(Exclusive) sólo está en una cache pero no ha sido modificada por lo que su valor es el mismo en todo el sistema; puede ser accedida sin necesidad de accesos externos pero si se escribe debe pasar al estado M. Un línea en el estado S(Shared) está en varias caches del sistema y si se escribe debe hacerse externamente de manera que las otras caches puedan invalidar (estado I (Invalidated)) sus valores que quedan obsoletos.

[27] Observe que las iniciales de los 4 estados configuran el acrónimo MESI.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.3.3. Bancos de memoria

Las memoria DRAM son más lentas que las SRAM, entre otras razones, porque requieren un tiempo de recuperación entre accesos aleatorios[28]. Una forma de aliviar este problema es mediante la técnica de *interleaving* que divide el sistema de memoria en bancos con la idea de reducir la probabilidad de que un acceso a memoria requiera esperar este tiempo de recuperación. En la figura 1.14 se ilustra un ejemplo teórico de la mejora en ancho de banda que se obtiene con un sistema de 2 bancos de memoria.

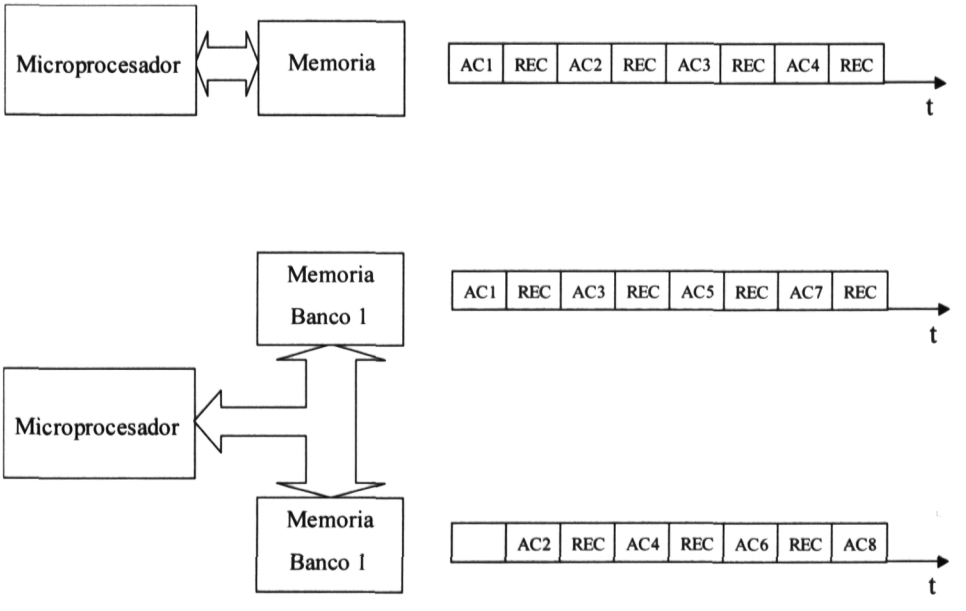


Figura 1.14: Comparativa entre un sistema de memoria convencional y uno con acceso interpaginado (interleaving) de dos bancos

Suponga que se accede a direcciones consecutivas de memoria y que el sistema de 2 bancos distribuye direcciones pares en un banco e impares en otro. En el caso de que no haya bancos, cada vez que se hace un acceso **ACx** hay que esperar un tiempo de recuperación **REC**. Sin embargo, si hay 2 bancos mientras un banco accede el otro se recupera.

[28] Cuando una memoria DRAM quiere hacer un acceso a una página distinta de la seleccionada debe esperar un tiempo denominado tiempo de precarga; esto hace que el sistema se ralentice bastante.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.3.4. Gestión de memoria

La gestión de memoria permite implementar sistemas multitarea, memoria virtual y protección de datos. La gestión de memoria distingue entre espacio de direcciones lógico y espacio de direcciones físico (figura 1.15). El espacio lógico de direcciones es el que ve el programador, mientras que el espacio físico de direcciones es el implementado por hardware. La gestión de memoria traduce las direcciones lógicas presentadas por los programas a las direcciones físicas que se envían al bus de memoria.

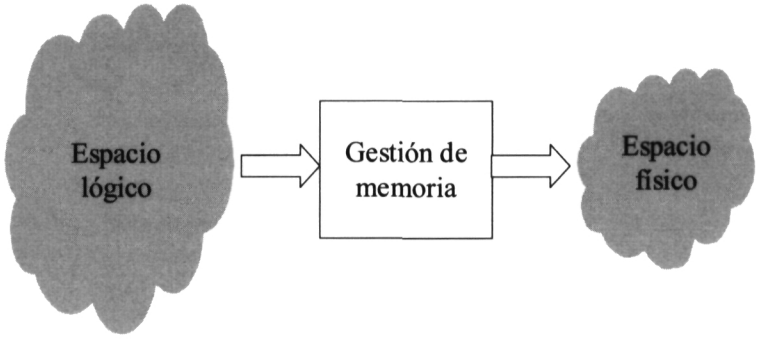


Figura 1.15: Gestión de memoria

Hay dos técnicas para llevar a cabo esta traducción: segmentación y paginación (figura 1.16).

Los segmentos son trozos de un programa lógicamente independientes y de tamaño variable que sólo se relacionan con otros segmentos mediante llamadas y transferencias de argumentos. La gestión de memoria tiene mecanismos de protección para proteger a los segmentos de accesos no autorizados. En segmentación, una dirección lógica está formada por dos campos: *segmento* y *desplazamiento*. El campo de segmento proporciona información sobre el segmento que contiene a la dirección lógica (dirección física de comienzo, tamaño y atributos). La dirección física se obtiene sumándole a la dirección física de comienzo del segmento el *desplazamiento*. El espacio lógico puede ser mucho más grande que el espacio físico.

Una página es un trozo de programas y/o datos de tamaño fijo sin ninguna estructura ni entidad propia. Se usa para implementar el mecanismo de memoria virtual. En paginación, una dirección lógica está formada por dos campos: *página* y *desplazamiento*. El campo de página proporciona información sobre la página que contiene a la dirección lógica (dirección física de comienzo y

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

atributos). La dirección física se obtiene sumándole a la dirección física de comienzo de la página el *desplazamiento*. El espacio lógico y físico coinciden en tamaño.

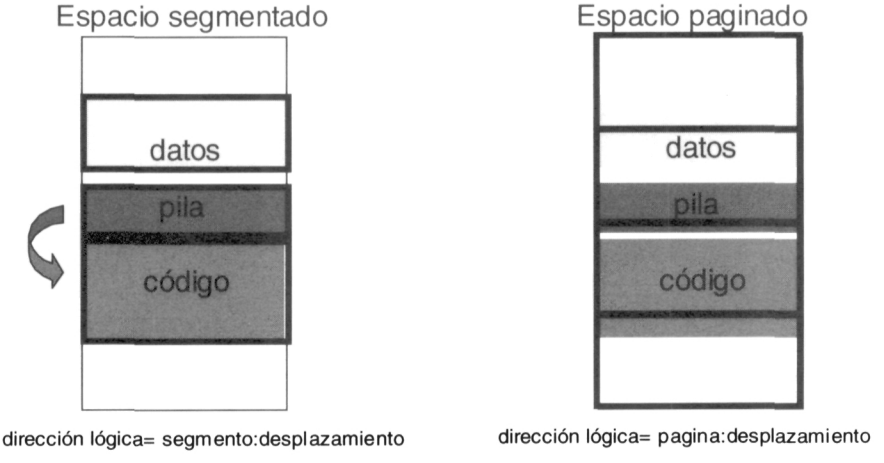


Figura 1.16: Segmentación y paginación

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.4. Entrada/Salida

La Unidad de Entrada/Salida (E/S) permite conectar el sistema con el mundo exterior. Hay muchos y variados dispositivos de E/S que requieren interfaces más o menos complejos y con muy distintas características. La mayoría de las señales reales serán analógicas, lo que obligará a usar circuitos acondicionadores de señal y conversores analógico digitales[29] para poder convertir la señal a un formato que entienda el sistema (el formato digital). Además las características eléctricas de las señales digitales pueden ser muy variadas y, en ocasiones, habrá que transformarlas en señales compatibles con el sistema. Otros aspectos importantes que afectan a la complejidad del diseño son la velocidad de operación (latencia y ancho de banda), la disponibilidad de la información o la necesidad de sincronismo.

2.4.1. Conversión analógico digital

La conversión analógico digital toma periódicamente muestras de la señal analógica y las aproxima a un conjunto de valores discretos denominados niveles de cuantificación(ver figura 1.17). Con n bits se pueden codificar $N = 2^n$ niveles de cuantificación.

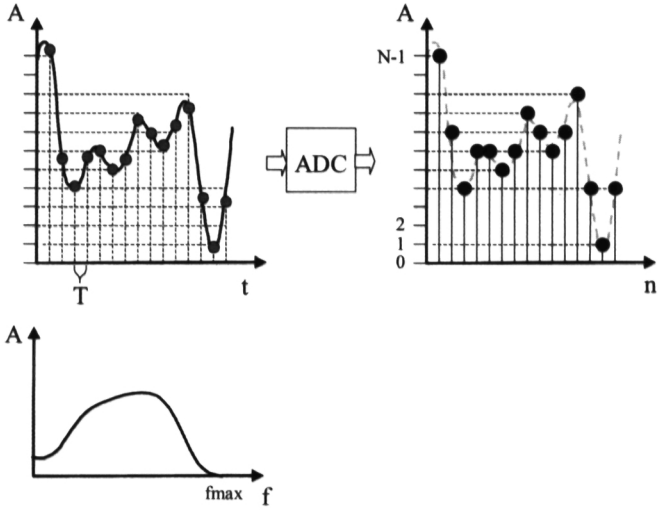


Figura 1.17: Parámetros básicos de muestreo de señales analógicas

[29] Analog Digital Conversor(ADC) y Digital Analog Conversor(DAC).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

La frecuencia de muestreo, $f_m = 1/T$, debe ser el doble de la máxima frecuencia a muestrear ($f_m = 2f_{max}$); con esto se asegura que se podrá recuperar la señal y evitar la distorsión antialiasing (teorema de Nyquist). En cuanto al número de niveles de cuantificación N, cuanto mayor sea, menor será el error cometido entre el valor analógico y el digital (error de cuantificación). Otros aspectos a tener en cuenta como la imperfección en filtros y falta de linealidad del sistema obliga a veces a incrementar tanto el valor de f_m como el de N.

2.4.2. Espacios de E/S

El microprocesador también influye en el diseño del interfaz de E/S ya que puede haber varias forma de acceso: Espacio de direcciones único (memoria y E/S) o espacio de direcciones separado (ver figura 1.18).

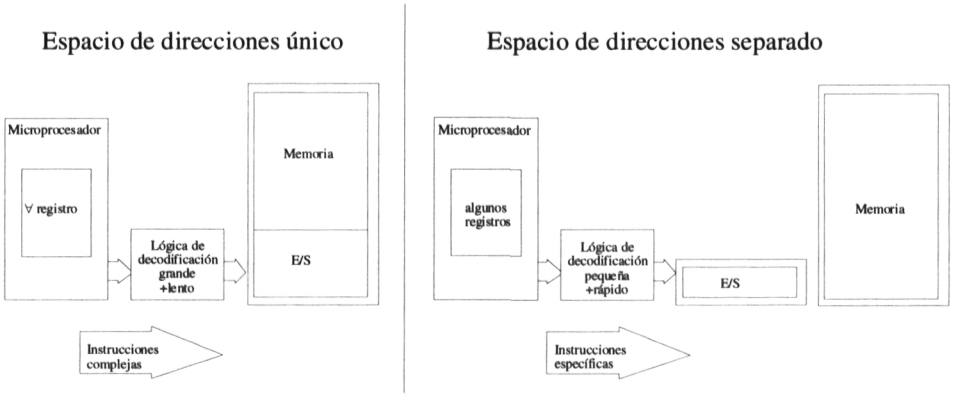


Figura 1.18: Tipos de espacios de direcciones

En el espacio de direcciones único[30], la transferencia se efectúa entre el puerto y cualquier registro, el espacio de direcciones es mayor y permite usar modos de direccionamiento. Sin embargo, se pierde parte del espacio de direcciones, la transferencia es más lenta (ya que la instrucción puede tardar más en ejecutarse) y, lo más importante, se requiere mayor lógica de decodificación de direcciones.

En el espacio de direcciones separado, se necesita menos lógica de decodificación y se utilizan instrucciones específicas de E/S (a veces requiere el uso de registros específicos). El espacio de direcciones es mucho más pequeño que el de memoria.

[30] También denominado E/S mapeado en memoria.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.4.3. Modos de transferencia

El interfaz más básico de E/S es el denominado de transferencia directa (figuras 1.19 y 1.20). Se utiliza cuando no hay problemas de sincronización con la operación de E/S y los datos están siempre disponibles. El hardware necesario es muy simple.

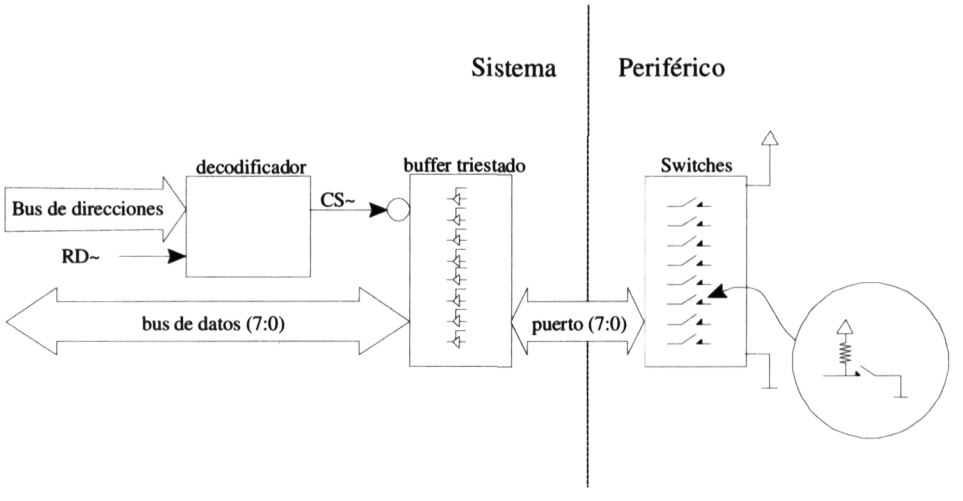


Figura 1.19: Transferencia directa: Lectura

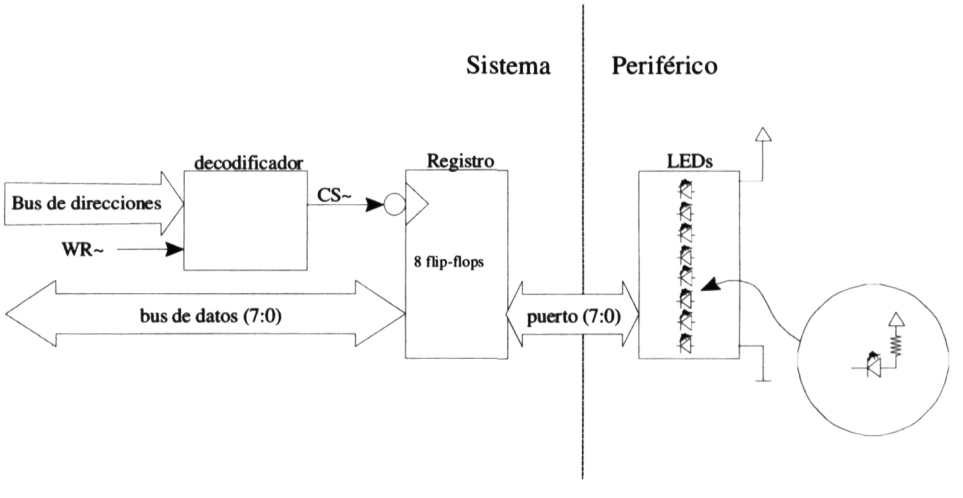


Figura 1.20: Transferencia directa: Escritura

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Para implementar un puerto de entrada sólo se necesita un buffer triestado que se habilita cuando se selecciona la dirección del puerto y se activa la señal de lectura (figura 1.19). Para el caso de un puerto de salida, es un registro (flip-flops) que almacena el dato de salida cuando se selecciona la dirección del puerto y se activa la señal de escritura (figura 1.20).

Cuando los datos de entrada no están siempre disponibles en el puerto de entrada o los datos de salida no son siempre aceptados por el dispositivo de salida, la transferencia directa no sirve. La solución inmediata es usar polling.

2.4.3.1. Polling La transferencia por polling utiliza una señal de estado (flag o registro) que está siempre disponible (puede leerse por transferencia directa) y que indica si se puede realizar la transferencia o no (figura 1.21).

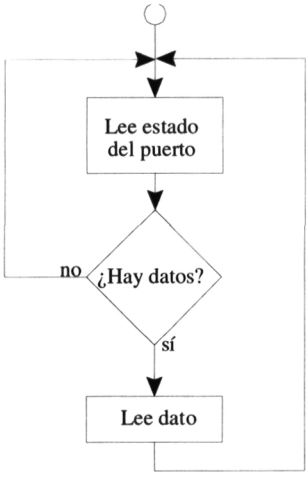


Figura 1.21: Lectura de un puerto de E/S mediante polling

El hardware es similar al de transferencia directa con la incorporación de una lógica para la lectura del flag de estado. Sus ventajas esenciales son que es fácil de programar, requiere poco hardware adicional y permite manejar dispositivos de E/S de varias velocidades. Sin embargo, puede llegar a ser muy ineficiente para dispositivos muy lentos, ya que el microprocesador tendría estaría constantemente preguntando por el flag de estado. La solución para estos casos es usar interrupciones.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.4.3.2. Interrupciones La transferencia por interrupciones es un mecanismo hardware del microprocesador que detecta solicitudes de transferencia de dispositivos de E/S y las satisface de manera (casi) transparente para el programa en curso. El hardware es relativamente complejo ya que suele manejar varias solicitudes simultáneas de interrupción (lo que obliga a establecer prioridades), requiere almacenar las direcciones de las diversas subrutinas de servicio de interrupción y debe salvar y restaurar el contenido de los registros del microprocesador.

En la figura 1.22 se ilustra el proceso básico:

1. Un periférico solicita una interrupción (`INT`). Si en el microprocesador están habilitadas las interrupciones hardware (máscara), el microprocesador termina la instrucción en curso y arranca un ciclo de reconocimiento de interrupción, activándose la señal `INTA` hacia el periférico. Si varios periféricos solicitan una interrupción al mismo tiempo habrá que establecer prioridades. Las prioridades las puede establecer el microprocesador, el software (mediante sondeo de interrupciones), los dispositivos de E/S (encadenamiento de interrupciones) o un controlador específico de interrupciones.

2. La dirección de la rutina de servicio de interrupción (RSI) está almacenada en memoria en la tabla de vectores de interrupción. El número de vector de interrupción correspondiente al periférico puede estar proporcionado por el microprocesador o por el dispositivo de E/S (que lo envía a la recepción de una señal `INTA`).

3. Antes de saltar a la RSI, el microprocesador salva sus registros en la pila (ubicada en memoria).

4. A continuación el microprocesador carga en su contador de programa el contenido del vector de interrupción: la dirección de comienzo de la RSI.

5. Empieza a ejecutarse la RSI. Si hay más de un periférico o causa que pueda haber provocado la interrupción, la RSI tendrá que averiguar quién fue el que solicitó la interrupción.

6. Antes de retornar al programa principal, la RSI restaura los valores de los registros del microprocesador.

7. Se sigue ejecutando el programa principal por el punto donde se había dejado.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

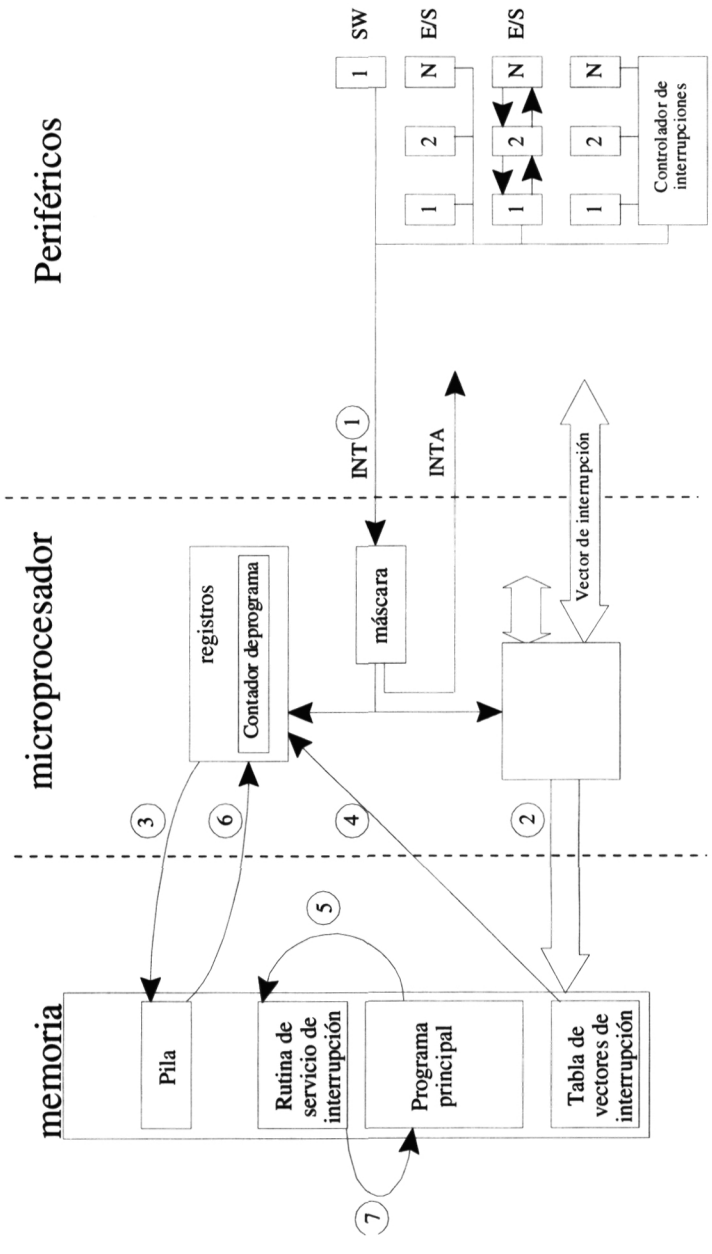


Figura 1.22: Procedimiento de interrupción

La ventaja esencial de este tipo de transferencia es que el programa en curso se libera de atender a los dispositivos. Su inconveniente es que el tiempo de respuesta puede no ser adecuado para velocidades de transferencia de datos elevadas. En este caso, se suele utilizar la transferencia por acceso directo a memoria (DMA, Direct Memory Access).

2.4.3.3. Acceso directo a memoria Se realiza una transferencia directa entre el dispositivo de E/S y la memoria, donde el control de la transferencia no lo soporta el microprocesador sino un dispositivo denominado controlador de DMA. El hardware es muy complejo, pero a cambio evita la búsqueda y ejecución de un ciclo de instrucciones para la transferencia de cada palabra de datos. El coste sólo lo hace viable cuando se necesita un velocidad de transferencia grande (>1Mbyte/s). En la figura 1.23 se muestra una configuración típica.

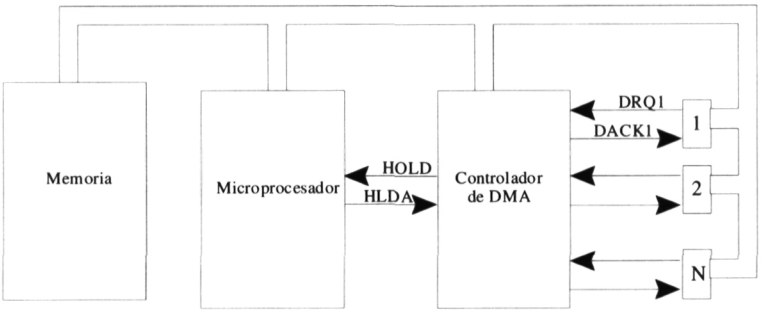


Figura 1.23: Sistema con acceso directo a memoria

Los periféricos solicitan una canal DMA activando una señal `DRQx`, el controlador de DMA gestiona esta solicitud (establece prioridades entre canales, enmascara solicitudes, etc.) y, si procede, solicita el bus al microprocesador a través de la señal `HOLD`. El microprocesador acepta la solicitud poniendo sus señales de bus en alta impedancia y activando la señal `HLDA`. A continuación, el controlador de DMA comunica al periférico que se va a llevar a cabo la transferencia activando la señal `DACKx`. Es importante destacar, que el controlador de DMA no actúa como buffer intermedio de los datos transferidos, sólo se limita a activar las señales de control de E/S y memoria simultáneamente para que los datos pasen de memoria al puerto de E/S o viceversa.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.4.4. Modos de transmisión

Hay dos formas básicas de transmisión: Serie y paralelo (figura 1.24).

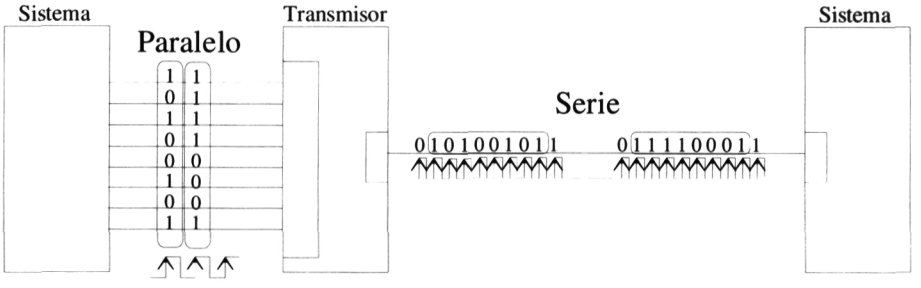


Figura 1.24: Modos de transmisión serie y paralelo

La transmisión paralelo es muy rápida pero no es adecuada cuando el cable es largo o la frecuencia es demasiado elevada. Esto se debe a que pueden aparecer diafonías entre señales adyacentes y/o distorsiones de señal. Un ejemplo de este tipo de transmisión es el puerto paralelo del PC.

La transmisión serie es más inmune al ruido, la longitud del cable puede llegar a ser muy larga y es más barato (menos cables). Tradicionalmente, su velocidad ha sido inferior a la de trasmisión paralelo. Un ejemplo de este tipo de transmisión es el puerto serie del PC.

2.4.4.1. Transmisión síncrona y asíncrona En una transmisión asíncrona (figura 1.25) los datos se envían y se reciben carácter a carácter. En reposo, el transmisor envía unos binarios continuamente. Cuando quiere transmitir un carácter envía un bit de arranque (cero binario) y, a continuación, los bits de datos. El número de bits de datos depende de la codificación utilizada (7 u 8 son los valores más habituales). También se suele incluir un bit adicional de paridad. Cuando el receptor recibe el bit de arranque activa un reloj[31] que muestrea los bits de datos. Cuando termina la transmisión de los datos, el transmisor envía 1 ó 2 bits de parada (unos binarios) y vuelve a la condición de reposo. Este procedimiento se repite carácter a carácter hasta que se envía el mensaje entero.

La transmisión asíncrona no es adecuada para altas velocidades ya que

[31] Los relojes de transmisión y recepción tienen la misma frecuencia nominal pero son completamente independientes.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

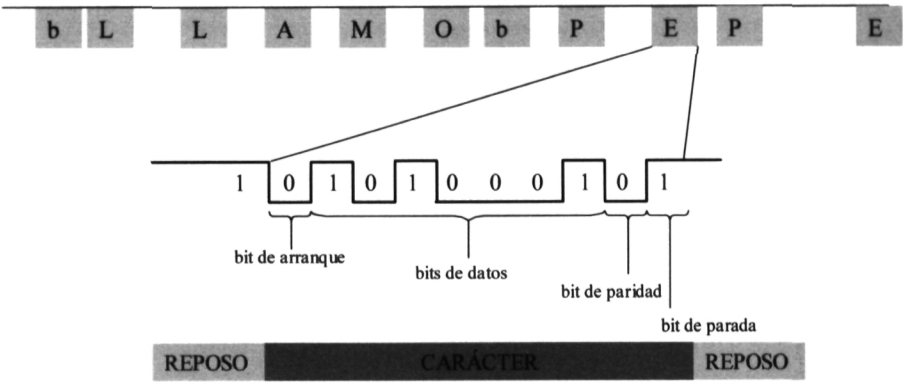


Figura 1.25: Ejemplo de transmisión asíncrona

es poco eficiente[32] y tiene un escaso control de errores[33]. Además, al ser los relojes de transmisión y recepción independientes, el tamaño de los carácteres (número de bits) no puede ser muy largo ya que conforme aumenta el tamaño del carácter aumenta el desfase entre relojes y, en consecuencia, la dificultad de recibir datos válidos.

En una transmisión síncrona los datos transmitidos siempre van acompañados de un reloj que garantiza la recuperación de los datos en recepción. En muchas ocasiones, la información se codifica para integrar en una única señal física el reloj y los datos.

El mensaje se envía por paquetes de información. El paquete tiene varios campos: sincronismo, número de paquete, datos, chequeo de errores, etc.. El campo de sincronismo alerta al receptor de que llegan datos, el campo de número de paquete facilita la recuperación de paquetes perdidos, el campo de chequeo de errores no sólo detecta errores sino que también es capaz de corregirlos. Dentro de un paquete, el porcentaje de bits que no son de datos (overhead) es muy inferior al caso de la transmisión asíncrona.

[32] Observe en la figura que de cada 10 bits enviados sólo 8 son de datos.
[33] El bit de paridad sólo sirve para detectar un error en un bit.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.5. Microsistemas

Los sistemas basados en microprocesador se podrían clasificar en sistemas de propósito general y sistemas de propósito específico.

En sistemas de propósito general (figura 1.26), el objetivo básico es conseguir alta velocidad, gran capacidad de proceso y almacenamiento y posibilidad de expansión. Son sistemas que usan periféricos estándar (el teclado, el ratón, el monitor, etc.) y un S.O. con gestión de memoria (memoria virtual, multitarea y protección). La memoria está distribuida a lo largo de todo el sistema (ver apartado 2.3.1 de jerarquía de memoria). Usan memoria no volátil (tipo ROM) para el arranque e inicialización del sistema y volátil (tipo DRAM) para almacenar el S.O. y los programas de aplicaciones. También disponen de controladores integrados específicos[34] que liberan al microprocesador de tareas rutinarias que consumen demasiados recursos y tiempo del microprocesador. En ocasiones la UCP contiene varios microprocesadores (multiproceso) que se reparten la ejecución de las diferentes tareas. El ejemplo, más típico es el ordenador personal.

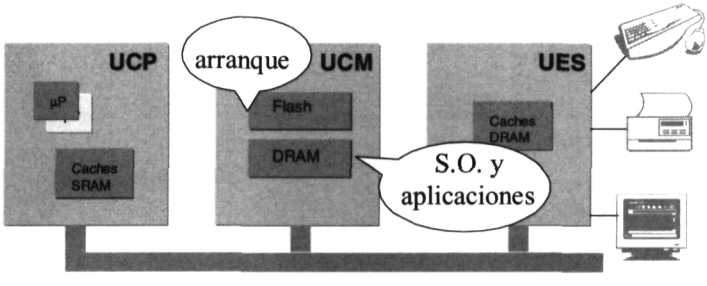


Figura 1.26: Microcomputador de propósito general

En sistemas de propósito específico (figura 1.27) se imponen otro tipo de requerimientos como son el consumo, el espacio disponible o el coste. A veces se suelen denominar sistemas empotrados (embedded) ya que están insertos dentro de otro sistema[35]. En muchos casos, no requieren un S.O. ya que siempre ejecutan el mismo programa; en este caso suelen usar memoria no volátil (tipo ROM) para almacenar el código y volátil (tipo SRAM) para guardar las vari-

[34] Puentes, controladores de interrupciones, de acceso directo a memoria, gráficos, de comunicaciones, etc.

[35] Control de alarmas, cámaras digitales, teléfonos móviles, máquinas fax, etc.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

ables del sistema. Los microcontroladores son sistemas de propósito específico integrados en un chip y, normalmente, enfocados hacia el manejo de eventos. En la figura 1.28 se puede ver la microarquitectura del microcontrolador 8051 de Intel.

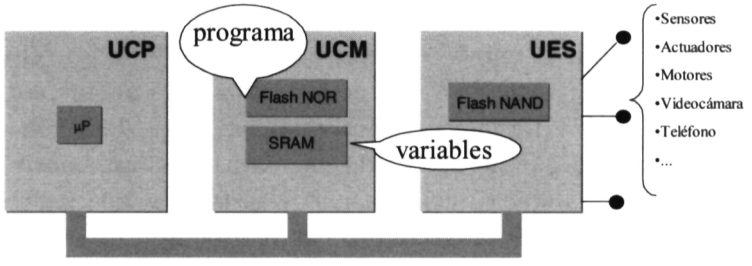


Figura 1.27: Microcomputador de propósito específico

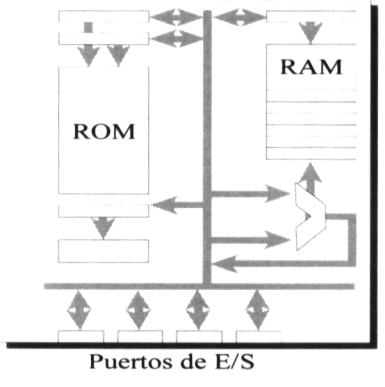


Figura 1.28: Microarquitectura del microcontrolador 8051 de Intel

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.6. Pipelining

El concepto de pipelining o segmentación es una de las técnicas más utilizadas en la arquitectura de microprocesadores y memorias para mejorar su rendimiento. Consiste en solapar procesos mediante su descomposición en etapas que se pueden ejecutar de forma independiente.

Este concepto se puede explicar con la siguiente analogía: Una fábrica tiene una cadena de montaje que ensambla productos(ver figura 1.29).

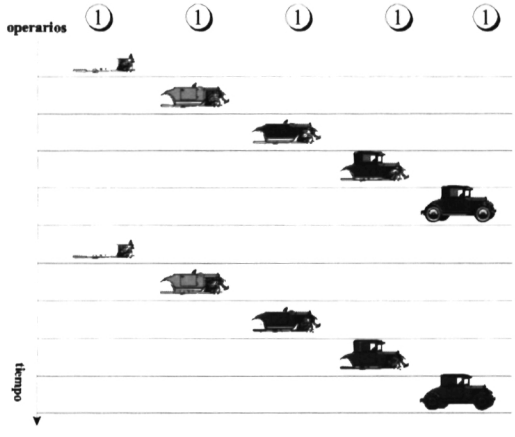


Figura 1.29: Analogía entre una cadena de montaje y un pipeline: sin segmentación

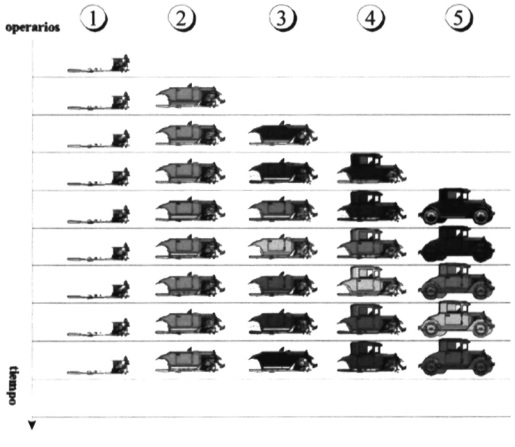


Figura 1.30: Analogía entre una cadena de montaje y un pipeline: con segmentación

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

En un sistema sin segmentación, solo hay un operario que lleva a cabo todas las etapas del montaje, por lo que tarda 5 unidades de tiempo[36] en terminar cada producto.

En un sistema con segmentación (ver figura 1.30) hay 5 operarios y cada uno de ellos se encarga de una etapa distinta. En T=1, el operario 1 lleva a cabo su tarea sobre el producto y pasa el resultado al operario 2. En T=2, el operario 1 repite la operación sobre un segundo producto, mientras que el operario 2 ejecuta la etapa 2 y pasa el resultado al operario 3. Siguiendo este proceso, en el instante T=5 todos los operarios estarán trabajando y se habrá ensamblado el primer producto. A partir de este instante, en cada nueva unidad de tiempo se ensamblará un producto. Es como si el ensamblado durara sólo una unidad de tiempo cuando en realidad dura 5.

La fábrica es un microprocesador, memoria, sistema gráfico o bus, la cadena de montaje es el pipeline y el ensamblado es el procesado o la transferencia de los productos (instrucciones y datos). En la figura 1.31 se puede ver el pipeline de un microprocesador de 5 etapas: 1) Búsqueda de instrucciones, 2)Decodificación, 3)Búsqueda de operandos, 4)Ejecución y 5) Escritura de resultados.

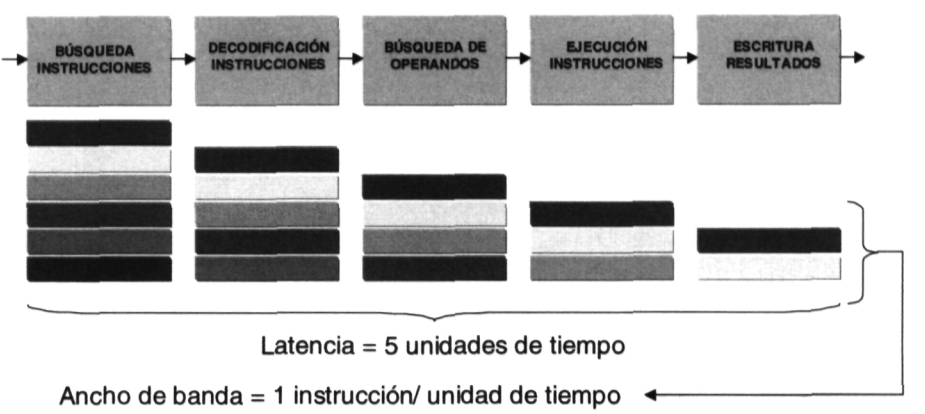


Figura 1.31: Ejemplo de pipeline de 5 etapas de un microprocesador

En la etapa 1 se busca la instrucción de memoria y se guarda en un buffer hasta que se solicite. En la etapa 2 se decodifica la instrucción determinando su tipo y los operandos que necesita. En la etapa 3 se localizan y traen los operandos tanto de registros como de memoria. En la etapa 4 se ejecuta la

[36]Supondremos que todas las etapas duran lo mismo: Una unidad de tiempo cada una.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

instrucción. Finalmente, en la etapa 5 se escribe el resultado en el registro que corresponda.

Supongamos que cada etapa se ejecuta en un ciclo de reloj y que el pipeline ya está lleno de instrucciones; aunque cada instrucción tarda en ejecutarse 5 ciclos de reloj, el microprocesador consigue ejecutar 1 instrucción por ciclo de reloj.

Sin embargo, el microprocesador no se comporta exactamente como la cadena de montaje. En primer lugar, los fallos de cache hacen que el pipeline esté desocupado hasta que lleguen las instrucciones de memoria. Además, las instrucciones de salto, que siempre habrá en el programa, no se pueden resolver hasta que no llegan a la etapa de ejecución.

La solución está en el uso de memorias cache no bloqueantes y técnicas de predicción de saltos. No obstante, hay que decir que aunque las soluciones apuntadas mejoran considerablemente la eficiencia del pipeline no resuelven el problema de forma definitiva. Téngase en cuenta que las técnicas de memoria cache y predicción de saltos son predicciones estadísticas.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.7. Latencia y ancho de banda

La latencia y el ancho de banda son dos medidas de rendimiento muy utilizadas en el diseño de microcomputadores.

La latencia mide cuánto tiempo tarda en llevarse a cabo un proceso determinado. Se usa para calcular lo que tarda en ejecutarse una instrucción o el tiempo que se requiere para llevar a cabo un acceso a memoria o E/S.

El ancho de banda mide la velocidad de dicho proceso. Cuando se utiliza para establecer la capacidad de transferencia de datos de un bus se puede estimar como:

$$B_{bus} = \frac{f_{CK} \times S}{C} \tag{1.1}$$

donde f_{CK} es la frecuencia de reloj de bus, S el tamaño del bus de datos y C el número de ciclos de reloj que requiere la transferencia. Se mide en MBytes o Gbytes por segundo.

Cuando se usa para determinar la capacidad de proceso del microprocesador se puede estimar como:

$$B_{micro} = \frac{f_{CK} \times V}{C} \tag{1.2}$$

donde f_{CK} es la frecuencia de reloj del microprocesador, V es el número de vias de ejecución y C el número de ciclos de reloj que requiere la instrucción. Se mide en millones de instrucciones por segundo o MIPS.

En cualquier caso, es común medir las prestaciones de un microprocesador comparándolo con otro y determinando cuántas veces es más rápido que el de referencia en la ejecución de un conjunto escogido de aplicaciones. Un buen ejemplo son las medidas SPECint y SPECfp.

La ecuaciones 1.1 y 1.2 consideran que todas las transferencias son iguales y se suelen usar para el cálculo del ancho de banda de pico B_p (el mejor caso). El cálculo del ancho de banda medio o efectivo B_e no es fácil ya que depende de tipo y número de transferencias y aplicaciones que corren en el sistema. A veces, los fabricantes definen un factor de utilización F para calcular B_e como $F \times B_p$.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3. Evolución de los microprocesadores y microcomputadores

3.1. Los primeros tiempos

En el año 1971, Intel lanzó el chip 4004 considerado como el primer microprocesador de propósito general del mundo. Este pequeño microprocesador de 4 bits con sólo 2300 transistores era mucho más potente que el ordenador ENIAC[37]. El 4004 requería mucha lógica adicional para implementar un sistema digital completo y sus aplicaciones eran muy limitadas: calculadoras, juguetes electrónicos o aplicaciones de control sencillas.

Muy poco tiempo después ya aparecieron microprocesadores de 8 bits[38]. Estos chips incluían más hardware de interfaz, eran más rápidos y tenían un conjunto de instrucciones mucho más potente. La tecnología de fabricación evolucionó a NMOS (antes era PMOS) y posteriormente a CMOS. Estos microprocesadores abrieron el campo de aplicaciones a controladores industriales más complejos como: Sistemas de adquisición de datos y control de procesos, instrumentación, terminales inteligentes y equipos periféricos, aplicaciones militares, comunicaciones, cajas registradoras, impresoras, etc.

En 1973, nació el primer microcontrolador. Texas Instrument desarrolló para sus calculadoras el TMS-1000. Era un chip que trabajaba a 300KHz, e integraba 1KB de ROM, 32 bytes de RAM, una UCP de 4 bits y puertos de E/S.

A mediados de los setenta empezaron a aparecer lo que se podría denominar computadores personales para aficionados. Entre ellos estaban el kit de montaje Altair 8800 de MITS basado en el 8080-8 de Intel[39], el Apple I diseñado en un garaje por Stephen Wozniac y Steven Jobs[40], y el barato PET de Commodore, fabricante de calculadoras que convirtió los ordenadores personales en objetos de consumo. Sin embargo, el auténtico boom de los computadores personales fue el Apple II (1977) basado en el 6502 de MOS Technology con su pantalla gráfica en color (era una enorme innovación) y la unidad de disco (la DiskII de 5,25") que ofreció en 1978.

[37]El ENIAC(Electronic Numerical Integrator And Calculator) vio la luz en 1946 en la Universidad de Pensilvania. Tenía 30m de largo, 1m de ancho y 3m de alto. Consumía 140KW y costaba un poco menos de medio millón de dólares.

[38]Los 8008, 8080 y 8085 de Intel, el 6800 de Motorola, el 6502 de MOS Technology, el F8 de Fairchild, el Z80 de Zilog, etc.

[39]El Altair 8800 incorporaba un intérprete BASIC desarrollado por una nueva compañía llamada Micro-soft de Bill Gates y Paul Allen.

[40]Stephen Wozniac tenía 26 años y Steven Jobs 21.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

En sus primeros tiempos, los ordenadores personales se vieron como juguetes y no se les advertía ninguna aplicación importante para el mundo de los negocios, industria o investigación. Esto cambió a partir de 1979 con la hoja de cálculo VisiCalc de Dan Bricklin y Bob Frankston y el procesador de textos WordStart de Seymour Rubinstein. Poco después, llegaría el programa de base de datos dBase de Wayne Ratliff y el procesador de textos WordPerfect de Satelite Software Internacional. El programa HyperCard (1987) de Bill Atkinson, para la creación de interfaces gráficos de usuario usando técnicas visuales, fue pionero en una tendencia creciente hoy en día (Visual C++ , Visual Basic, etc.).

3.2. Microelectrónica

La tecnología de los semiconductores ha evolucionado mucho desde que se inventó el transistor en 1947. La figura 1.32 muestra el número de transistores por chip que se han podido integrar en las últimas décadas, para el caso de los microprocesadores de Intel.

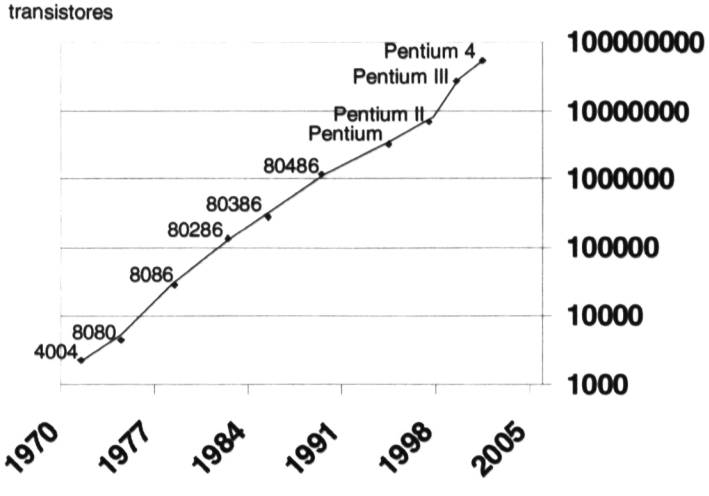


Figura 1.32: Crecimiento de la densidad de transistores en los microprocesadores IA-32 de Intel

En los años 60, Gordon Moore observó que el número de transistores de los circuitos integrados se doblaba cada 18 meses. Esta tendencia se ha mantenido a lo largo de los años, por lo que se ha convertido en la famosa "ley de Moore".

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

La causa del espectacular aumento en el número de transistores por chip es la fotolitografía. Los circuitos integrados se fabrican depositando patrones metálicos o de silicio químicamente tratado (dopado), unos encima de otros, sobre un substrato de silicio. El diseño de estos patrones conforma los transistores que constituyen las puertas lógicas del circuito integrado. Para reproducir con precisión los trazos de estos patrones sobre la pastilla, la longitud de onda de la luz debe ser al menos tan pequeña como los propios trazos. Los procesos litográficos están en constante evolución. Recientemente, los láser excimer han reemplazado a la lámparas de mercurio como fuente de luz litográfica. Este hecho ha acelerado la reducción de los trazos. Se ha barrido así el espectro electromagnético desde la luz visible, pasando por la luz ultravioleta y aproximándonos a los rayos X, donde se plantean nuevos retos tecnológicos.

El factor principal en el incremento del número de transistores por chip es la denominada norma de diseño (viene a ser el tamaño del trazo).

La figura 1.33 muestra la evolución para los microprocesadores de Intel. Esta tendencia se mantendrá gracias al desarrollo de nuevas técnicas litográficas como, por ejemplo, la ultravioleta extrema (EUV) que promete normas de diseño de 30 nm. Sin embargo, hay que decir que mientras la norma de diseño se reduce, el coste del equipo necesario para fabricar estos dispositivos se dispara.

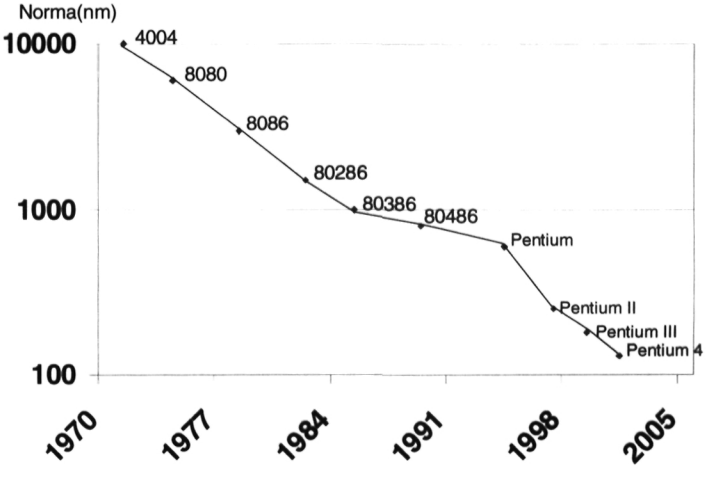


Figura 1.33: Disminución de la norma de diseño en la fabricación de microprocesadores IA-32 de Intel

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

La velocidad de reloj también se incrementa cuando se reduce la norma de diseño, ya que la velocidad de conmutación dentro del chip está limitada por las capacidades entre interconexiones de una etapa a la siguiente (buses y conexiones largas) o la capacidad de las puertas de la siguiente etapa (conexiones cortas).

Puede pensarse a primera vista que mientras se pueda reducir el tamaño de los transistores se podrá aumentar la integración (más funciones) y la velocidad (mejores prestaciones). Sin embargo, hay un factor que limita este hecho: el consumo de potencia. Actualmente los microprocesadores parecen "estufas" que hay que refrigerar si no queremos que el equipo salga ardiendo. La figura 1.34 muestra la evolución del consumo de potencia en los diferentes microprocesadores de la familia IA-32 de Intel.

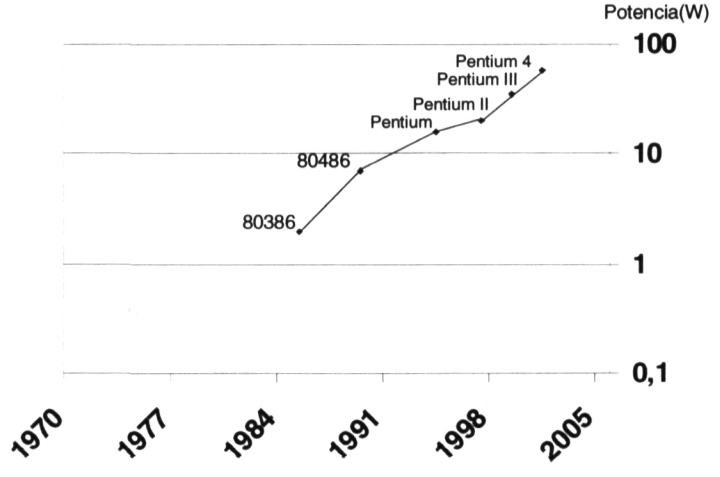


Figura 1.34: Evolución del consumo de potencia en los microprocesadores de la familia IA-32 de Intel

En los microprocesadores CMOS, la potencia disipada es proporcional al producto del efecto capacitivo del chip, su frecuencia de reloj y el cuadrado de su tensión de alimentación[41]. Las técnicas que se han venido utilizando para reducir la disipación de potencia del chip han sido disminuir la tensión de alimentación y sustituir las líneas de interconexión por materiales de menor resistividad.

[41] Otro efecto a tener en cuenta es que cuando un transistor se hace demasiado pequeño tiene una conmutación deficiente, de manera que consume corriente cuando no debiera hacerlo.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.3. Evolución de los microprocesadores

A finales de los 70, llegaron los microprocesadores de 16 bits[42]. Eran microprocesadores mucho más complejos que los de la generación anterior. Soportaban el conjunto completo de instrucciones de un miniordenador y eran fáciles de programar. La integración y velocidad fueron en aumento. Se incluyó arquitectura segmentada (pipeline) y, en algunos, gestión de memoria. La tecnología solía ser HCMOS. En cuanto a las aplicaciones, se ampliaron a proceso de datos, control en tiempo real, comunicaciones avanzadas, redes de proceso distribuido y, sobre todo, potentes ordenadores personales.

A primeros de los ochenta, empezaron a aparecer los microprocesadores[43] de 32 bits. Su arquitectura mejoró sustancialmente el rendimiento de la transferencia de datos, la capacidad de direccionamiento y de proceso, el conjunto de instrucciones, la gestión de memoria y la capacidad del coprocesador. También se empezó a integrar memoria cache y pipelines más profundos para incrementar la velocidad de procesamiento y reducir sobrecargas en el bus. El campo de aplicaciones siguió en aumento: Automatización de oficinas, aplicaciones típicas de miniordenadores, estaciones de trabajo CAD (Computer Aided Design) / CAM (Computer Aided Manufacturing), ordenadores personales de altas prestaciones, robótica, inteligencia artificial, tratamiento de señales, etc.

Los microprocesadores RISC (Reduced Instruction Set Computer) han sido otro hito muy importante. Sus primeras implementaciones se llevaron a cabo en universidades[44] apareciendo los primeros chips comerciales[45] en el año 1986. El objetivo básico de estos microprocesadores no era exactamente reducir el número de instrucciones, sino conseguir chips sencillos, baratos, rápidos, con un ancho de banda de 1 instrucción por ciclo de reloj y donde el compilador y el microprocesador formaran un conjunto bien adaptado; para ello hacían un uso extensivo de la técnica de pipeline. Los defensores de la tradicional[46] arquitectura CISC (Complex Instruction Set Computer) argumentaban que sus diseños mantenían compatibilidad software, su código era más compacto, y podían conseguir las mismas prestaciones que los RISCs. Estas afirmaciones

[42] El TI9900 de TI, el 8086 de Intel, el 6809 de Motorola, el Z8000 de Zilog, el 68000 de Motorola, etc.

[43] El NS32032 de National, el 68020 de Motorola, el 80386 de Intel, el Z80000 de Zilog, etc.

[44] El RISC I (Reduced Instruction Set Computer) de la Universidad de Berkeley y el MIPS (Microprocessor without Interlocked Pipe Stages) de la Universidad de Stanford.

[45] El R2000 de MIPS, la arquitectura SPARC (Scalable Processor ARChitecture) de Sun, la arquitectura PA HP (Precision Architecture Hewlett Packard) de HP, etc.

[46] El acrónimo CISC se acuñó para distinguir las microarquitecturas convencionales de las microarquitecturas RISC.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

se materializaron en el Pentium de Intel. Sin embargo, posteriores implementaciones de esta firma han tendido a RISC (familia Pentium II e Itanium) aunque manteniendo la compatibilidad hacia atrás.

En la década de los 90 llegó el paralelismo a nivel de instrucción: Los microprocesadores superescalares que permitían ejecutar más de una instrucción por ciclo de reloj. Las primeras implementaciones ejecutaban las instrucciones en orden[47], sin embargo la tendencia ha sido a ejecutarlas fuera de orden[48] con el objeto de reducir las dependencias de programa. La complejidad de estas arquitecturas es muy grande y gran parte de su circuitería se usa para resolver dependencias entre instrucciones. Estos problemas se aliviaron dejando que fuera el compilador el que resolviera las dependencias[49]. A finales de los noventa aparecieron nuevos microprocesadores[50] y una fuerte tendencia hacia los denominados mediaprocessors y SOC(System-On-a-Chip).

La figura 1.35 ilustra una comparativa entre microarquitecturas CISC, RISC y superescalares.

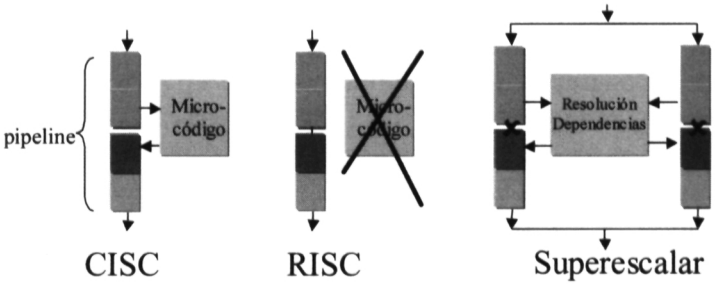


Figura 1.35: Microarquitecturas de microprocesadores

El mercado de los microprocesadores empotrados[51] (embedded) de 32 bits ha crecido en los últimos años debido a la demanda de videojuegos[52], portátiles,

[47]El Alpha 21164 de DEC, el UltraSparc de Sun, el Pentium de Intel, etc.

[48]El R10000 de MIPS, el PA8000 de HP, el PowerPC 620 de IBM, el Pentium II de Intel, etc.

[49]El Itanium de Intel hace uso de esta técnica.

[50]El 21264 de Digital, el Power3 de IBM, la UltraJava y UltraSparcIII de Sun, el PentiumII de Intel, el Cayenne de Cyrix, el K6 de AMD, el R12000 de Silicon Graphics, etc.

[51]Hay cientos de microprocesadores empotrados de 32 y 64 bits. Se pueden destacar las arquitecturas StrongARM de VLSI, SuperH de Hitachi, PowerPC de IBM, i960 de Intel, ColdFire de Motorola, Mips de NEC, XScale de Intel, etc.

[52]La Play Station de Sony y Nintendo 64 usan microprocesadores MIPS de 32 bits y 64 bits, respectivamente.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

cámaras digitales y teléfonos celulares. La potencia de cálculo de las modernas consolas de videojuegos es equivalente a la de las estaciones de trabajo de Silicon Graphics de no hace muchos años.

Actualmente, hay varias tendencias en el diseño de los futuros microprocesadores: arquitecturas superescalares avanzadas que son capaces de ejecutar decenas de instrucciones por ciclo de reloj, ejecución superespeculativa mucho más agresiva que la actual, microprocesadores multithread que ejecutan tareas en paralelo (como el Pentium 4), microprocesadores con varios procesadores (cores) que trabajan en paralelo, etc..

3.4. Evolución de los microcomputadores

En 1981, el anuncio del IBM PC (figura 1.36) aturdió al mundo de la computación.



Figura 1.36: IBM PC. Cortesía de IBM

A pesar de la ventaja que le llevaban otros fabricantes, IBM introdujo, no sabemos si intencionadamente, un concepto que revolucionaría el mercado de los ordenadores personales: una arquitectura abierta que podría crecer en el futuro. Esta estrategia, combinada con la enorme influencia de IBM y el lanzamiento de la hoja de cálculo Lotus 1-2-3, un año más tarde, hizo que el mundo de los negocios le prestara atención.

Para hacerse con el mercado en el menor tiempo posible, IBM decidió encargar el S.O. a otra compañía. Digital Research estaba desarrollando el CP/M

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

para 8086 pero no acababa de terminarlo, así que IBM buscó otra empresa: Microsoft. Durante muchos años, IBM lamentaría la decisión de no escribir su propio S.O.. Finalmente lo hizo con el OS/2.

El IBM PC se basó en el procesador 8088 de Intel. La decisión de IBM de usar la arquitectura 8086 fue muy criticada, y llevó al PC y a sus descendientes a encarar muchos problemas que otras máquinas no tuvieron que resolver, principalmente a causa del modelo segmentado de memoria[53]. La arquitectura abierta del PC, basada en el 8088 de Intel llegó a dominar completamente el mercado de microcomputadores[54]; prueba ésta de que la superioridad técnica a veces no es un requisito imprescindible para triunfar[55].

Aprovechando el éxito del PET , Commodore lanzó nuevos equipos (el VIC-20 y el Commodore64) de color muy baratos basados en el 6502. Mientras que los críticos cortos de vista seguían preguntando para qué servían estas máquinas, Commodore fue la primera compañía en introducir a millones de personas en la computación personal[56].

El portátil de Compaq (1983) fue el primer PC clónico de éxito. Compaq fue ganando reputación rápidamente gracias a su calidad y a su compatibilidad 100 % IBM (mediante ingeniería inversa). En 1986, los PCs clónicos de ALR y Compaq se hicieron con el control del mercado al introducir los primeros PCs basados en el 80386: El Acces386 y el Deskpro386. Esto marcó el fin[57] del predominio de IBM en el mercado de los PCs. En 1991, AMD lanzó el primer microprocesador 80386 no fabricado por Intel, lo que trajo consigo una importante reducción de precios no sólo en los microprocesadores sino también en los ordenadores personales.

A principios de los 80, fabricantes de miniordenadores como Hewlett-Packard, Digital Equipment Corp. y Tektronix, empezaron a desarrollar estaciones de trabajo basadas en microprocesadores. Pero fue la SPARCstation de Sun (1989) la que consiguió que las palabras SPARCstation y estación de trabajo fueran sinónimos para mucha gente. En años posteriores, aparecieron

[53]Según parece, los propios ingenieros de IBM querían usar el 68000, pero IBM ya había obtenido los derechos para fabricar el 8086 (para el uso en su máquina de escribir inteligente Displaywriter), a cambio de dar a Intel los derechos de su tecnología de memoria de burbuja. Otro factor era que el 8088 podía usar componentes de 8 bits existentes de bajo coste, mientras que los componentes del 68000 eran más caros y poco disponibles todavía.

[54]Los descendientes del IBM-PC fueron el PC-XT y el PC-AT .

[55]Muy pronto, Intel lanzó un nuevo microprocesador con un modo protegido que resolvía el problema del modelo segmentado de memoria del 8086.

[56]Se estima que se vendieron unos 22 millones de equipos.

[57]IBM estaba ocupada desarrollando sus sistemas PS/2 basados en un nuevo bus de expansión denominado MicroChannel.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

sistemas SPARCstation y clónicos más grandes y mejores, haciendo de la SPARCstation la segunda plataforma del mercado (la IBM PC era la primera).

En 1993, Apple volvió a innovar con su MessagePad[58] (figura 1.37). El MessagePad era lo que luego vendría a llamarse un PDA (Personal Digital Assistant). Tenía un microprocesador RISC, pantalla LCD, infrarrojos y reconocimiento de escritura manual.



Figura 1.37: MessagePad de Apple. Cortesía de Apple

Los PCs actuales tienen una capacidad de cálculo y de almacenamiento impensable hasta hace pocos años. El bus del memoria es de varios Gigabytes/s. Las memorias dinámicas han cambiado su arquitectura para conseguir tiempos de acceso en modo ráfaga similares a los de las memorias estáticas. Los tradicionales buses paralelo del sistema, como ISA o PCI , han dejado paso a nuevos buses serie de mayor velocidad y menor complejidad de diseño hardware[59]. La capacidad gráfica se ha disparado gracias a las nuevas instrucciones multimedia de los microprocesadores, los nuevos buses locales (PCI Express), y las baratas y potentísimas tarjetas 3D que han hecho entrada en el mercado en estos últimos años. La capacidad de los discos duros ha crecido exponencialmente pasando de los 10MB de 1983 a decenas de Gigabytes. El almacenamiento óptico y semiconductor está sustituyendo a los disquetes y

[58]El MessagePad acabó llamándose Newton, el nombre de su S.O. Tenía un microprocesador RISC ARM610 de 20 MHz, 4 MB de ROM y 640KB de RAM (disponibles para el usuario 150KB). Su coste era de unos $700.

[59]Hoy en día, un PC empieza a parecerse a una pequeña red de área local con calidad de servicio donde los nodos son la memoria, el microprocesador y los diferentes dispositivos de E/S.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MURCIA

nuevos buses como USB (Universal Serial Bus) están acabando con la "Torre de Babel"de interfaces a periféricos.

Tampoco hay que olvidar el esfuerzo de la industria en el desarrollo de los pequeños PCs (portátiles, PDAs(ver figura 1.38), tablet PC, etc.). Es más, los portátiles están empezando a relevar a los ordenadores de sobremesa gracias a sus increíbles prestaciones multimedia. También se están desarrollando completos sistemas integrados en un chip (SOC, System On Chip) que pueden llegar a ser una realidad en breve tiempo y a cubrir necesidades muy diversas: Electrodomésticos inteligentes, monitorización de personas y hogares, radioPCs, ordenadores de a bordo en automóviles, WebTV, etc.



Figura 1.38: Workpad de IBM. Cortesía de IBM

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.5. El interfaz hombre máquina

A finales de los 70 y principios de los 80, Xerox Alto comenzó la revolución de los interfaces gráficos de usuario (GUI, Graphics User Interface) los cuales barrerían la industria de los microcomputadores de la década siguiente (figura 1.39). Xerox PARC (Palo Alto Research Center) desarrolló las ideas básicas del GUI[60]: El ratón, los iconos, las ventanas, los menús, etc.. Sin embargo, aunque las ideas de Xerox Alto eran revolucionarias, fueron un enorme fracaso comercial[61].

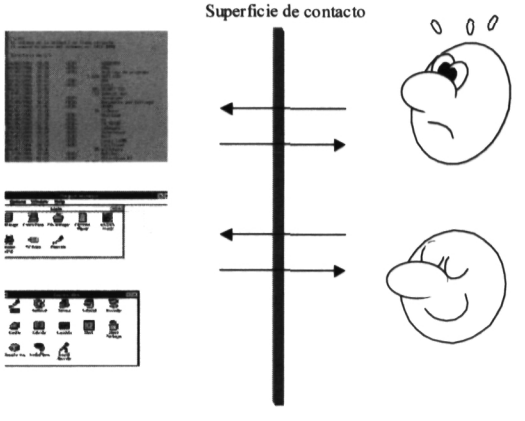


Figura 1.39: Interfaz de usuario de comandos y GUI

El Macintosh de Apple (1984) fue el que tomó el relevo (ver figura 1.40). Apple incluía varias aplicaciones innovadoras como MacPaint que demostró las posibilidades del ratón y MacWrite que descubrió el procesamiento de textos WYSIWYG (What You See Is What You Get). El "Mac" llegó a ser la herramienta elegida para cualquier empresa editora[62].

Algunos años después apareció la versión 3.0 de Windows de Microsoft que disponía de una impresionante línea ascendente de software de aplicaciones[63].

[60] Inicialmente se usó el acrónimo WIMP (Window, Icon, Menu, Pointing device) para definir esta nueva forma de interacción. Sin embargo ha ido quedando obsoleto debido a que este término tiene en inglés un significado ofensivo. Es más, ha sido utilizado, en ocasiones, por los defensores del rudo interfaz de línea de comandos para ridiculizar a los entusiastas del GUI.

[61] Costaba 50.000 dólares.

[62] La combinación de Macintosh, PageMaker e impresora LaserWriter de Apple (1985) basada en postcript dominó la industria de las publicaciones de aquellos años.

[63] Apple se querelló contra Microsoft porque consideró que le había copiado su GUI.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA



Figura 1.40: Macintosh de Apple. Cortesía de Apple

Entre éstos estaban el procesador de textos Word y de hoja de cálculo Excel.

La evolución de los GUIs ha pasado por los browsers de Internet con su hipertexto y conexión a cualquier PC del mundo. Ya están empezando también a aparecer GUIs en 3 dimensiones y es probable que, en pocos años, asistamos a la implantación de nuevos periféricos que faciliten la inmersión en mundos virtuales interactivos.

Es posible que en esta década se imponga definitivamente el interfaz de voz. Visiones más alejadas predicen interfaces gestuales y emocionales (figura 1.41) donde baste un gesto para que nuestro PC interprete qué queremos hacer, interfaces inteligentes que conozcan nuestras costumbres y sean capaces de anticiparse a nuestras exigencias e incluso implantes en el cuerpo humano.

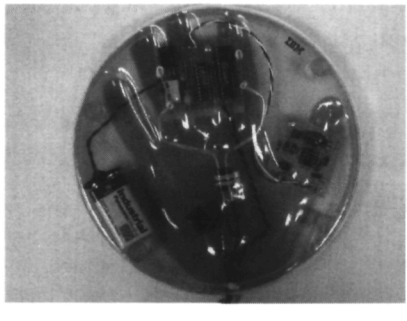


Figura 1.41: Ratón emocional de IBM. Cortesía de IBM

No obstante, hay que recordar que las predicciones de los expertos no suelen

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

cumplirse y que ciertas innovaciones[64], no consideradas por los grandes de la industria, son las que al final mueven el mundo.

3.6. La multimedia

El Amiga de Commodore (1985) fue el primer ordenador personal multimedia. Sin embargo, en sus tiempos, sólo se le consideró una máquina de juegos y poca gente captó la importancia de esta plataforma que combinaba sonido, gráficos avanzados, sistema operativo multitarea e interfaz gráfico de usuario. El Amiga usaba tres chips diseñados a medida (Agnes, Denise y Paula) para hacer gráficos avanzados y sonido.

El interés por la multimedia explotó con la aparición de las tarjetas de sonido y las unidades de CD-ROM. A partir de los 90, los microprocesadores empezaron a ampliar su conjunto de instrucciones para mejorar las prestaciones de las aplicaciones multimedia (MPEG, síntesis de audio, proceso de imagen, 3D, realidad virtual, etc.). La base de estas nuevas instrucciones está en la implementación de operaciones SIMD (Single Instruction Multiple Data). HP fue la primera en añadir estas instrucciones[65] a su arquitectura RISC. Finalmente, equipos como la WebTV forzarán el desarrollo de nuevos tipos de microprocesadores: Los mediaprocessors que estarán especializados en algoritmos de comunicaciones y procesado de imagen, video y sonido.

3.7. Las comunicaciones

Procedente de proyectos militares, Internet empezó siendo una pequeña red de computadores (Arpanet) usada en universidades americanas a principios de los 70, siendo una de sus primeras aplicaciones, el correo electrónico. Arpanet, que adoptó el estándar TCP/IP de transmisión de datos en modo paquete, creció libre y a una velocidad de vértigo[66]. Otro hito a destacar fue la red ethernet (1980) de Robert Metcalfe. En 1982 ya existía una tarjeta adaptadora para PC: la Etherlink. La LAN (Local Area Network), basada en ethernet, era un sistema de oficina que enlazaba estaciones de trabajo, servidores, e impresoras permitiendo de esta manera compartir recursos.

Aunque Internet ya llevaba varios años funcionando, fue en los 90, con la introducción de la WWW (World Wide Web) de Tim Betners-Lee, cuando

[64] Por ejemplo, el PC o la misma red Internet.

[65] Después llegaron VIS (Visual Instruction Set) de UltraSparc, MMX de Intel, 3DNow de AMD, etc.

[66] TCP/IP era de dominio público.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

se hizo tremendamente popular. La Web ofrecía, mediante el hipertexto, una sencilla, amigable y gráfica manera de buscar información o entretenimiento. Se puede decir sin lugar a dudas que la Web ha iniciado una nueva era en la industria de los computadores.

Además, Internet se ha convertido, en muy pocos años, en una herramienta imprescindible. La página web es, hoy en día, la sede virtual de una empresa o de un particular desde donde se ofertan y demandan todo tipo de servicios y productos: compra/venta directa, cartera de productos, soporte técnico y comercial, gestión administrativa, entretenimiento, intercambio de información, etc..

Los 90 también trajeron el anhelo de integrar servicios multimedia (voz, video y datos) en las redes de comunicaciones. La tecnología necesaria se desarrolló (ATM, Asynchronous Transfer Mode) pero la demanda de aplicaciones que exigieran grandes anchos de banda y una buena calidad de servicio no llegó[67]. Internet, una red específica de datos, sorprendió con algunas aplicaciones de telefonía y videoconferencia que, aún siendo de baja calidad, acabaron gozando de gran aceptación. Es más, el entusiasmo de Internet está llevando a desarrollar protocolos que proporcionan cierta calidad de servicio (RSVP, IPv6, etc.) y redes IP de muy altas velocidades.

La telefonía móvil digital[68] también ha sido otro hito muy importante de los 90 que actualmente ha incorporado servicios de datos; por ejemplo, los mensajes cortos SMS (Short Message Service) y el acceso a Internet en modo paquete GPRS (General Packet Radio Service). GPRS ha sido sólo un paso intermedio para alcanzar UMTS (Universal Mobile Telecommunication System), la tercera generación de móviles, que pretende ofrecer servicios multimedia de alta velocidad con calidad de servicio y acceso a cualquier red de comunicaciones.

La movilidad ha llegado a las redes de datos (las redes de área local inalámbricas WiFi). También han surgido miniredes denominadas redes de area personal que usan tecnologías avanzadas como bluetooth o UWB (Ultra Wide Band) para interconectar periféricos con diferentes requerimientos de calidad de servicio. Otra tecnología prometedora es RFID (Radio Frequency Identification) que proporciona dispositivos de muy bajo coste y consumo nulo. La conjunción de todas estas nuevas tecnologías está proporcionando el punto de arranque para el desarrollo de sistemas de Inteligencia Ambiental (AmI), un

[67] Además, una tarjeta ethernet era 10 veces más barata que una ATM.

[68] GSM (Global System for Mobile communications) en Europa, PDC (Personal Digital Cellular) en Japón e IS-95 en Estados Unidos.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

nuevo paradigma que creará un entorno inteligente y transparente al usuario y que establecerá enlaces de forma espontánea (figura 1.42).

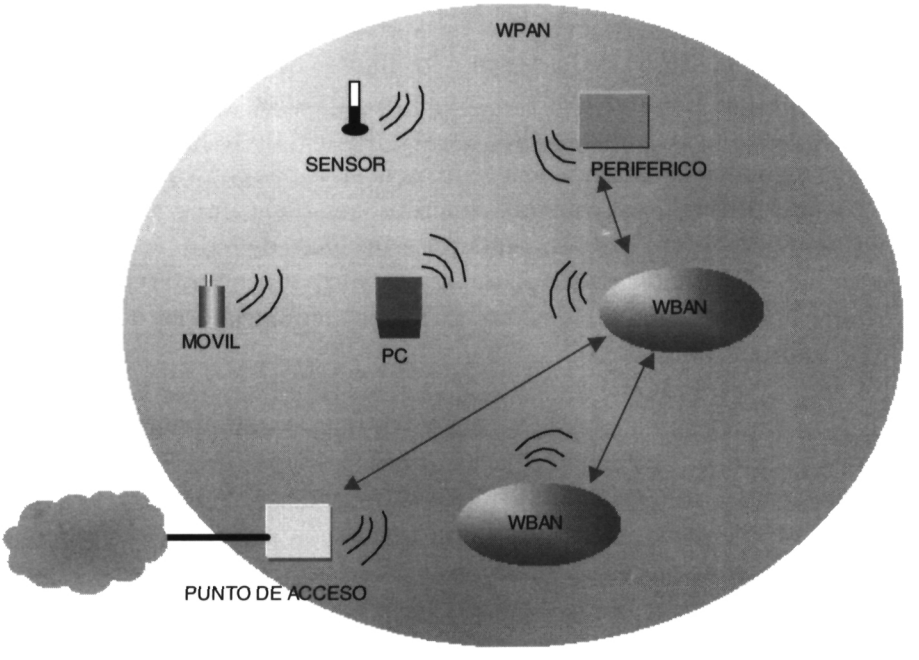


Figura 1.42: Inteligencia Ambiental. Redes de Área Corporal(BAN, Body Area Networks) se comunican con el entorno: Redes de Área Personal(PAN, Personal Area Networks), sensores inalámbricos, redes móviles, Internet, etc.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4. Últimas Tendencias

Los microprocesadores siguen reduciendo la norma de diseño, lo que repercute en una mayor integración y en un aumento de la frecuencia de trabajo. Sin embargo, el consumo de los chips y los límites de la fotolitografia plantean nuevos retos que tendrán que ser resueltos atacando en varios flancos: microarquitectura, nuevos materiales, nuevos diseños de bajo consumo, etc. En particular, parece que la tendencia será hacia sistemas con varios procesadores (cores) y multithreading. Los nuevos microprocesadores integrarán unidades de propósito general junto con procesadores específicos de bajo consumo que aceleren tareas importantes de los sistemas actuales: descarga de datos de Internet, gráficos, reconocimiento de voz o de imagen, codificadores/decodificadores MPEG, interfaz radio, etc.

En cuanto a los microcomputadores, se observa una clara tendencia hacia la normalización (S.O., interfaces, etc.), ya que la competencia entre fabricantes repercute en una mejora de los productos y en una sustancial bajada de los precios. La conexión de los microcomputadores a Internet es un hecho inevitable y está provocando una nueva revolución de los ordenadores personales. Es más, en esta década aparecerán nuevos PCs de bolsillo con acceso a internet móvil.

Con respecto al software, es como un gas que se expande hasta ocupar todo el espacio disponible[69]. Da lo mismo que los ingenieros hardware redoblen sus esfuerzos en conseguir más velocidad, más prestaciones o más capacidad de memoria, los ingenieros software siempre sabrán cómo aprovecharlo[70].

Hoy en día, el desafío es continuar fabricando sistemas digitales cada vez más pequeños, rápidos, fiables, seguros, ubicuos y, sobre todo, potentes para un mercado de consumo donde el usuario es incapaz de manejar las complejidades de esta impresionante tecnología. Esto obligará a desarrollar plataformas virtuales que oculten al usuario cualquier aspecto técnico de manejo o mantenimiento y le ofrezcan un interfaz obvio.

La unión de las telecomunicaciones y los microcomputadores presentan un nuevo marco sobre el que trabajar que afectará de lleno a muchos aspectos de la sociedad: trabajo cooperativo, teletrabajo, comercio mundial virtual con eliminación de intermediarios, reconocimiento de voz, traducción multilingüe, nuevos interfaces de realidad virtual, operaciones médicas a distancia y muchas otras aplicaciones que hoy se consideran todavía ciencia ficción y fantasía.

[69] "Ley" de Nathan Mybrvold, ejecutivo de Microsoft.

[70] De manera, que el equipo más "rumboso" del mercado siempre acabará yendo lento.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5. Bibliografía

En las referencias que se adjuntan hay algunos libros recomendados para repasar conceptos básicos de la materia. Los artículos son interesantes por su valor histórico y porque también aclaran algunos conceptos fundamentales de los sistemas basados en microprocesadores. Finalmente se proponen algunas direcciones que pueden ser de interés y de consulta.

Libros

González-Parada, E., Herrero-Reder,I. y Urdiales-García, C.: Manual de Sistemas Digitales. Servicio de Publicaciones e Intercambio Científico de la Universidad de Málaga, Málaga, 2002. ISBN 84-7496-963-8.

Tanenbaum, A.S., Structured Computer Organization, 5ª Edición, Prentice-Hall International, New Jersey, USA, 2005.

Artículos

Moore, G.E.: "Cramming more components onto integrated circuits", Electronics, Vol. 38, No. 8 April 19, 1965.

Faggin, F., Hoff, M.E., Mazor, S., and Shima, M.: "The History of the 4004", IEEE Micro, December 1996, pp. 10-20.

Gelsinger, P.P. et al.: "Microprocessors circa 2000", IEEE Spectrum October 1989, pp. 43-47.

Bourianoff,G.: "The Future of Nanocomputing", IEEE Computer, August 2003, pp. 44-53.

Flynn, M.J.and Hung, P.: "Microprocessor Design Issues: Thoughts on the Road Ahead", IEEE Micro, May-June 2005, pp. 16-31.

Algunas direcciones de Internet

Bayko, J.: "Great Microprocessors of the Past and Present", 1996, bwrc.eecs.berkeley.CIC/archive/cpu_history.html

-: "Most Important Software Products", 20th Anniversary, BYTE, September 1995. www.byte.com/art/9509/sec7/art5.htm.

-: "Top 20 Technologies", 20th Anniversary, BYTE, September 1995. www.byte.com/art/9509/sec7/art16.htm.

IEEE Micro. www.computer.org

EDN. www.edn.com

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

6. Guía de estudio

1. Definir los conceptos de microprocesador, sistema basado en microprocesador, sistema empotrado y microcontrolador. Distinguir entre ellos.
2. Describir los bloques de un sistema basado en microprocesador. Explicar la función de cada bloque y del bus.
3. Definir los diferentes niveles de abstracción de un microcomputador y establecer la relación entre ellos.
4. Definir el concepto de multitarea.
5. Distinguir entre organizaciones de memoria little endian y big endian.
6. Describir la técnica interleaving de memoria.
7. Definir el concepto de jerarquía de memoria.
8. Explicar el funcionamiento básico de una memoria cache y distinguir entre arquitecturas (mapeo directo, asociativa por conjuntos,etc.).
9. Definir el concepto de coherencia de cache y describir la técnica de snooping
10. Distinguir entre segmentación y paginación.
11. Definir la frecuencia de muestreo y los niveles de cuantificación en una conversión analógico digital.
12. Distinguir entre los diferentes modos de transferencia de E/S: Transferencia directa, polling, interrupciones y DMA.
13. Comparar las características de la transmisión serie y paralelo.
14. Distinguir entre transmisión asíncrona y síncrona.
15. Explicar las diferencias entre un sistema de propósito general y uno específico, indicando los requerimientos de cada uno.
16. Definir el concepto de pipelining. Enumerar qué comportamientos del sistema pueden complicar el funcionamiento ideal del pipeline.
17. Definir latencia y ancho de banda.
18. Definir la ley de Moore.
19. Explicar la relación entre la norma de diseño y la velocidad de un microprocesador.
20. Enumerar los factores que influyen en la potencia disipada de un microprocesador y las técnicas que permiten reducirla.
21. Enumerar las características básicas de una microarquitectura RISC.
22. Distinguir entre microprocesadores CISC, RISC y superescalares.
23. Enumerar las posibles causas del éxito del IBM PC.
24. Definir el concepto de inteligencia ambiental.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2 Microprocesadores: Arquitectura IA-32 de Intel

1. Introducción

La arquitectura IA-32 de Intel ha estado siempre a la vanguardia del diseño de los ordenadores personales. Se podría decir, si tenemos en cuenta el número de ordenadores que la usan, que es la arquitectura más popular y, quizás también, la más importante. El éxito de la arquitectura IA-32 se debe sobre todo a su compatibilidad software; uno de sus más importantes logros es que el código objeto creado para los primeros microprocesadores IA-32 de finales de los años 70 todavía se ejecuta en los microprocesadores IA-32 de ultima generación[71]. Otra causa importante es el hecho de que cada nueva generación de microprocesadores IA-32 se entrega con un aumento considerable de prestaciones.

Este capítulo estudia la arquitectura IA- 32 de Intel, repasando las características más relevantes de todos los miembros de esta gran familia, desde el 8086 hasta el Pentium 4. Se explica su microarquitectura, su entorno de ejecución, su conjunto de instrucciones y su interfaz con el sistema.

[71] La arquitectura IA-32 se refiere a microprocesadores con registros generales de 32 bits. Sin embargo, Intel considera incluidos en esta familia a sus microprocesadores de 16 bits ya que IA-32 mantiene compatibilidad software con ellos.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2. Familia IA-32 de Intel

La tabla 2.1 resume algunas de las características de los microprocesadores más relevantes de la familia IA-32.

Procesador	Fecha	Frecuencia reloj	Densidad transistores	Tamaño registros	Tamaño bus datos	Ancho banda bus datos	Espacio Direcciones	cache
8086	1978	8 MHz	29K	16 PG	16		1MB	
286	1982	12,5 MHz	134K	16 PG	16		16MB	
386DX	1985	20 MHz	275K	32 PG	32		4GB	
486DX	1989	25 MHz	1,2M	PG:32 PF:80	32		4GB	L1:8KB
Pentium	1993	60 MHz	3,1M	PG:32 PF:80	64		4GB	L1:16KB
PentiumPro	1995	200 MHz	5,5M	PG:32 PF:80	64		64GB	L1:16KB L2:256KB o 512KB
PentiumII	1997	266 MHz	7M	PG:32 PF:80 MMX:64	64		64GB	L1:32KB L2:256KB o 512KB
PentiumIII	1999	500 MHz	8,2M	PG:32 PF:80 MMX:64 XMM:128	64		64GB	L1:32KB L2:512KB
Pentium4	2000	1,5 GHz	42 M	PG:32 PF:80 MMX:64 XMM:128	64	3,2 GBps	64GB	ETC: 12K L1:8KB L2:256KB
Pentium4 HT	2002	3 GHz	55 M	PG:32 PF:80 MMX:64 XMM:128	64	4,2 GBps	64GB	ETC:12K L1:64KB L2: 1MB
PentiumM	2003	1,6 GHz	77 M	PG:32 PF:80 MMX:64 XMM:128	64	3,2 GBps	4GB	ETC: 12K L1:64KB L2:1MB
PentiumM 775	2004	2 GHz	140 M	PG:32 PF:80 MMX:64 XMM:128	64	3,2 GBps	4GB	ETC: 12K L1:64KB L2:2MB
Pentium Ext. Ed.	2005	3,2 GHz	230 M	PG:32,64 PF:80 MMX:64 XMM:128	64	6,4 GBps	64GB	ETC: 12K L1:16KB L2:1MBx2
Pentium Xeon	2005	3,33 GHz	675 M	PG:32,64 PF:80 MMX:64 XMM:128	64	5,3x2 GBps	64GB	ETC: 12K L1:16KB L2:1MB L3:8MB
Core Duo T2600	2006	2,16GHz	152 M	PG:32 PF:80 MMX:64 XMM:128	64	5,3 GBps	4GB	L1:64KB L2:2MB

Tabla 2.1: Características de los microprocesadores IA-32 de Intel. PG: Propósito General, PF: Punto Flotante, MMX: MultiMedia eXtensions, XMM: MultiMedia eXtensions para punto flotante, ETC: Execution Trace Cache.

El 8086 (1978) es el primer microprocesador de 16 bits[72] de Intel, con un bus de datos de 16 bits y un bus de direcciones de 20 bits que le permite un direccionamiento de 1MB (2^{20} bytes = 1MB). Una característica muy peculiar

[72]Significa que sus registros son de 16 bits.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

de este microprocesador es que su espacio de direcciones se divide en segmentos de 64KB y sólo puede haber 4 segmentos activos a la vez: uno para código, otro para la pila y dos para datos. La dirección física de memoria se obtiene combinando una dirección base con un desplazamiento:

$$direccion\ fisica = direccion\ base \cdot \texttt{10H} + \ desplazamiento \tag{2.1}$$

La dirección base está en un registro de segmento y el desplazamiento en un registro general. En 1979, Intel lanzó el 8088, un 8086 con el bus de datos de 8 bits[73].

El 80286 (1982) es también un microprocesador de 16 bits pero de prestaciones muy superiores al 8086. Tiene un bus de datos de 16 bits y un bus de direcciones de 24 bits. Es capaz de direccionar, por tanto, 2^{24} bytes = 16 MB. Introduce un nuevo modo de funcionamiento: El modo protegido que soporta gestión de memoria[74] y libera a la IA-32 del insólito espacio de direcciones segmentado del 8086.

El 80386 (1985) es el primer microprocesador de 32 bits de Intel[75]. Tiene un bus de datos y de direcciones de 32 bits. Es capaz de direccionar, por tanto, 2^{32} bytes = 4 GB. Es compatible con los microprocesadores anteriores ya que los registros de 16 bits del 8086/80286 se "mapean" en los 16 bits menos significativos de sus registros de 32 bits. Su modo protegido está muy mejorado. La gestión de memoria añade paginación y un modo virtual 8086 que permite ejecutar programas 8086 en un entorno multitarea. Es el primero en incluir pipelining.

El 80486 (1989) es un diseño mejorado del 80386 que integra un coprocesador matemático 387, un controlador de memoria caché 82385 y 8KB de memoria caché (para instrucciones y datos). Incluye nuevas instrucciones para caché y para S.O. multiproceso. Posteriores versiones del 486 incorporan mecanismos de gestión de consumo de potencia[76].

El Pentium[77] (1993) es un microprocesador de 32 bits con una microarquitectura superescalar de dos vías. Integra 16 KB de caché L1 (8KB para datos y 8KB para código). Usa técnicas de predicción de saltos para mejorar

[73] El IBM PC se basó en el microprocesador 8088 de Intel. El 8088 era un 20 % más lento que el 8086, pero los buses de 8 bits eran críticos para mantener un bajo coste del sistema.

[74] Nunca llegó a funcionar bien.

[75] En 1981 y antes del 80386 hubo un intento fallido: el chipset iAPX432.

[76] Estas características fueron inicialmente desarrolladas para una gama de microprocesadores que se destinaron al mercado de portátiles (386SL y 486SL).

[77] Debería haberse llamado 80586, pero por cuestiones de derechos de autor se decidió cambiar el nombre.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

la eficiencia de sus dos pipelines. Mejora la gestión de memoria (Modo virtual 8086 más eficiente y páginas de 4KB y 4MB) y facilita el diseño de sistemas multiproceso. Tiene un bus de datos de 64 bits y un bus de direcciones de 32 bits. También incluye un controlador de interrupciones programable avanzado (APIC, Advanced Programmable Interrupt Controller).

El Pentium Pro(1995) tiene una microarquitectura superescalar de 3 vías con superpipeline y ejecución dinámica[78]. Integra una memoria caché L2 de 256 KB que trabaja a la frecuencia de reloj del microprocesador y dispone de un bus específico. Está optimizado para código de 32 bits[79]. Tiene un bus de direcciones de 36 bits (direcciona 2^{36} bytes = 64 GB) y un bus de datos de 64 bits orientado a transacción.

El Pentium MMX (1997) añade al Pentium la tecnología MMX (MultiMedia Extensions): Nuevas instrucciones, nuevos tipos de datos y un modelo de ejecución SIMD (Single Instruction Multiple Data) que aceleran drásticamente las aplicaciones multimedia. También duplica el tamaño de la caché L1 (de 16 a 32 KB) y mejora la arquitectura.

El Pentium II (1997) mejora la microarquitectura del Pentium Pro y añade la tecnología MMX. Integra una memoria caché L1 de 32KB y una memoria caché L2 de 256 (ó 512 ó 1MB) que trabaja a la mitad de la frecuencia de reloj del microprocesador. También mejora su gestión de consumo de potencia. También hay versiones de muy altas prestaciones para servidores multiproceso (Xeon[80]) y para PCs de consumo (Celeron[81]).

El Pentium III (1999) añade al Pentium II las Streaming SIMD Extensions (SSE): Una ampliación de la tecnología MMX que permite trabajar con registros de 128 bits y operandos de punto flotante. El Pentium III Xeon incluye una nueva caché L2 de muy altas prestaciones: ATC (Advanced Transfer Cache)[82].

El Pentium 4 (2000) introduce una nueva microarquitectura que le permite alcanzar frecuencias de reloj más altas y mejores prestaciones que sus antecesores. Esta microarquitectura, denominada NetBurst, es superescalar de 3 vías con hyperpipeline, ejecución dinámica avanzada y un novedoso subsis-

[78] La ejecución dinámica incluye análisis de flujo de microdatos, ejecución fuera de orden, predicción de saltos mejorada y ejecución especulativa.

[79] Adecuado para S.O. de 32 bits como Windows NT o OS/2.

[80] El Pentium II Xeon dispone de una caché L2 de 2MB que corre a la velocidad del microprocesador.

[81] El Pentium II Celeron es un diseño de bajo coste con una caché L2 de 128KB.

[82] ATC está integrada en el microprocesador, tiene un bus de datos de 256 bits y una latencia muy reducida que consigue anchos de banda de 88GBps.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

tema de caché de trazas de ejecución que reduce la latencia del decodificador. Además, el Pentium 4 vuelve a ampliar la tecnología MMX con las SSE2 (más instrucciones y operandos de punto flotante de doble precisión) y soporta un bus de sistema de 400MHz.

Las últimas versiones del Pentium 4 (2004) incorporan tecnología hyperthreading, para ejecutar concurrentemente varios hilos de ejecución de un programa, más instrucciones multimedia (SSE3) y el modo EM64T (Extended Memory 64 Technology) que implementa un espacio de direcciones plano de 64 bits.

Intel también apostó desde hace tiempo por microprocesadores específicos para portátiles. El Pentium M (2003) tiene una microarquitectura especial con unas prestaciones similares a los Pentium 4 pero con un consumo muy reducido.

El Pentium Extreme Edition(2005) es un Pentium 4 de última generación que introduce tecnología dual-core (dos procesadores[83] dentro del chip) para potenciar el procesamiento multithreading. El Pentium D es una versión recortada del anterior que no incluye tecnología hyperthreading.

Finalmente, el Core Duo (2006) combina las microarquitecturas Netburst y M incluyendo varias innovaciones relacionadas con la ejecución de instrucciones, el acceso a memoria y la gestión de potencia.

[83]Intel habla de núcleo cuando se refiere a cada procesador. Por tanto este chip sería de doble núcleo.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3. Microarquitecturas

En los siguientes apartados se describen las últimas microarquitecturas de la familia IA-32: Pentium, P6 (Pentium Pro, Pentium II y III) y NetBurst (Pentium 4).

3.1. Microarquitectura del Pentium

En la figura 2.1 se puede ver un diagrama de bloques del microprocesador Pentium.

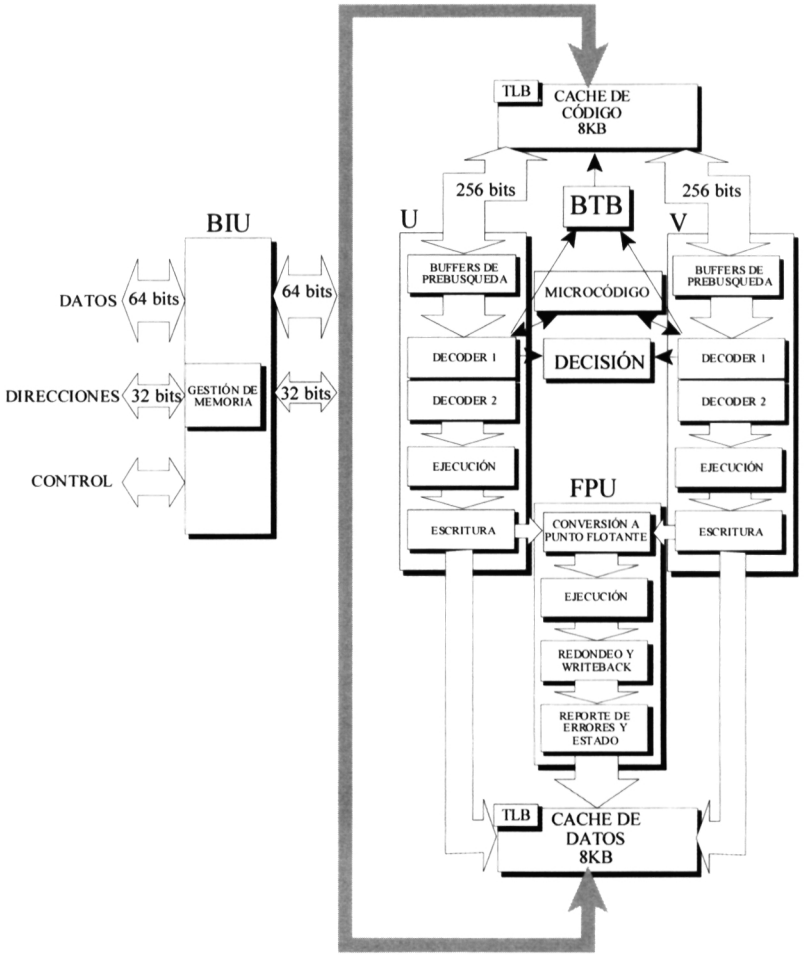


Figura 2.1: Microarquitectura del Pentium

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

La arquitectura es superescalar de 2 vías (U y V) lo que permite ejecutar 2 instrucciones de enteros en un ciclo de reloj. Las memorias caché L1 presentan una arquitectura de memoria Harvard que facilita el acceso simultáneo de instrucciones y datos.

Las vías U y V son pipelines independientes de 5 etapas:

1. PF: Prebúsqueda de instrucciones
2. D1: Decodificación
3. D2: Decodificación
4. EX: Ejecución
5. WR: Escritura

La vía U puede ejecutar cualquier instrucción mientras que la vía V sólo puede ejecutar instrucciones simples[84].

La unidad de punto flotante (FPU) tiene 8 etapas, algunas de ellas compartidas con la unidad U (PF, D1, D2, EX, WB/Conversión a punto flotante X1, ejecución X2, redondeo y write-back WF y reporte de errores y de estatus ER). A continuación, se describen las etapas de prebúsqueda, decodificación y ejecución.

La etapa de prebúsqueda está formada por la memoria caché de código[85] y los buffers de prebúsqueda. Los bytes de instrucción que se extraen de la caché se almacenan en uno de los 4 buffers de prebúsqueda. Cada buffer tiene un tamaño de una línea de caché. El acoplo entre la caché de instrucciones y los buffers de prebúsqueda garantiza, virtualmente, que el Pentium nunca tendrá que esperar a que le lleguen instrucciones, excepto en el caso de fallos de caché o saltos mal predichos.

El Pentium tiene un BTB (Branch Target Buffer) que predice si un salto se va a producir o no. Se basa en una caché asociativa de 4 vías de 1KB y 256 líneas. Cada línea contiene una dirección de una instrucción de salto condicional y 2 bits de historia.

[84] El pipeline U es más potente que el V ya que es el único que añade el hardware necesario para llevar a cabo las instrucciones de desplazamiento y rotación. Las instrucciones simples se ejecutan por hardware (no por microprograma) en sólo un ciclo de reloj. Ejemplos de este tipo de instrucciones son: MOV, INC, DEC, PUSH y POP.

[85] 8KB, asociativa de 2 vías con líneas de 32 bytes, LRU (Least Recently Used), snooping en hardware para resolver la coherencia entre cachés on-chip y memoria externa. TLB (Translation Lookaside Buffer) asociativa de 4 vías con 32 entradas y pseudo LRU (ver apartado 9.2.2 de paginación).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Los bits de historia (ver figura 2.2) se actualizan cada vez que se ejecuta la instrucción. En el estado inicial 11, el microprocesador siempre salta; si acierta se mantiene en este estado pero si se equivoca pasa al estado 10. En el estado 10, el microprocesador "insiste" en saltar; si acierta retorna al estado 11 pero si se equivoca pasa al estado 01. En el estado 01, el microprocesador no salta; si acierta pasa al estado 00 y si falla vuelve al estado 10. Finalmente, en el estado 00 el microprocesador tampoco salta; si acierta se mantiene en ese estado y si se equivoca retorna al estado 01.

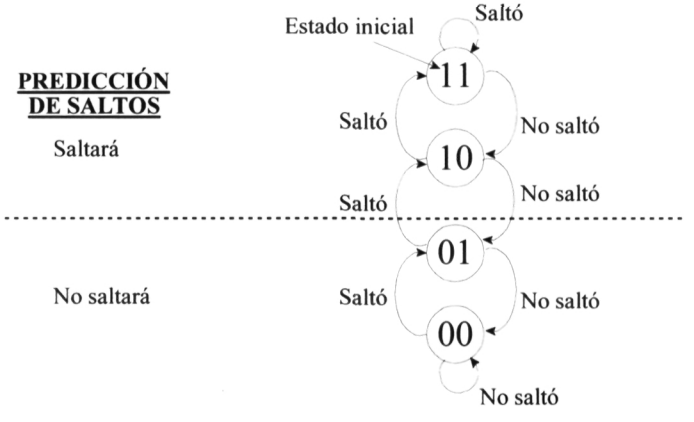


Figura 2.2: Predicción de saltos en el Pentium

Por tanto, salvo que los estados sean 00 o 01, el Pentium considera que se va a producir el salto y la etapa de prebúsqueda se adelanta tomando de la caché la instrucción correspondiente. Cuando aparece una nueva instrucción de salto que no está en el BTB, se reemplaza aleatoriamente una de sus posiciones.

La decodificación se descompone en dos subetapas: D1 y D2. Las funciones básicas de D1 son decodificar las instrucciones y resolver las dependencias entre instrucciones. Los pipelines U y V pueden ejecutar simultáneamente dos instrucciones `I1` e `I2` simples si se cumplen ciertas condiciones:

```
IF (I1 e I2 son simples AND I1 no es un salto AND
    el destino de I1 no es ni fuente ni destino de I2)
        Se envía I1 e I2 a los pipelines U y V respectivamente
ELSE
        Se envía I1 al pipeline U
```

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

La decisión se toma en la etapa D1 de ambos pipelines. Si se ejecuta `ELSE`, `I2` queda en la etapa D1 para ser comparada con `I3`.

La mayoría de las instrucciones se decodifican por hardware pero algunas muy complejas, rémora de su compatibilidad con los microprocesadores anteriores, necesitan microcódigo. El microcódigo del Pentium tiene un tamaño de 4Kpalabras de 92 bits.

La etapa D2 es necesaria para resolver los complejos modos de direccionamiento del microprocesador. Contiene un sumador de 4 entradas de direcciones correspondientes a 1) segmento, 2) base, 3) índice y 4) desplazamiento de la dirección efectiva (ver apartado 4.4). Esto permite reducir a uno el número de ciclos de reloj necesarios para el cálculo de los modos de direccionamiento. Otra función que tiene es la de leer operandos de los registros para su uso en la etapa de ejecución.

La etapa de ejecución incluye la ALU y la caché de datos[86].

3.2. Microarquitectura P6

La microarquitectura P6 es superescalar de tres vías con superpipeline de 10 etapas, ejecución dinámica y dos niveles de caché no bloqueantes[87]. La caché L2 dispone de un bus independiente por lo que P6 admite el acceso simultáneo a la caché L2 y a la memoria principal.

La ejecución dinámica ejecuta las instrucciones en un orden distinto al de la secuencia del programa (ejecución fuera de orden). Esta técnica permite mantener lleno el superpipeline y reducir la latencia de memoria. P6 hace uso de dos conceptos de proceso de datos: análisis de flujo de datos y ejecución especulativa.

- El análisis de flujo de datos dinámico analiza en tiempo real el flujo de datos para determinar dependencias y para detectar oportunidades de ejecución de instrucciones fuera de orden. El núcleo de ejecución puede monitorizar muchas instrucciones y ejecutarlas en el orden que mejor optimice el uso de las unidades de ejecución multiples del microprocesador, mientras mantiene la integridad de los datos.

[86] 8KB, doble puerto, interpaginado de 8 bancos, asociativa de 2 vías con líneas de 32 bytes, protocolo writeback MESI, TLB dual asociativa de 4 vías y 64 entradas y doble puerto.

[87] Caché L1 de código: 8KB en Pentium Pro y 16KB en Pentium II y III, asociativa de dos vías, línea de 32 bytes, 2 puertos, MESI. Caché L1 de datos: 8KB en Pentium Pro y 16KB en Pentium II y III, asociativa de dos vías, 2 puertos, MESI. Caché L2: 256KB ó 512KB en Pentium Pro y Pentium II y 512KB ATC (Advanced Transfer Cache) en Pentium II.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- La ejecución especulativa ejecuta instrucciones que están más allá de un salto condicional que no ha sido resuelto todavía. Para hacer posible la ejecución especulativa, es necesario desacoplar el despacho y ejecución de instrucciones de la actualización de los registros.

La microarquitectura P6 (ver figura 2.3) tiene 3 unidades básicas: Búsqueda/decodificación, Despacho/Ejecución y Retirada. Las tres unidades se comunican con un almacén de microinstrucciones.

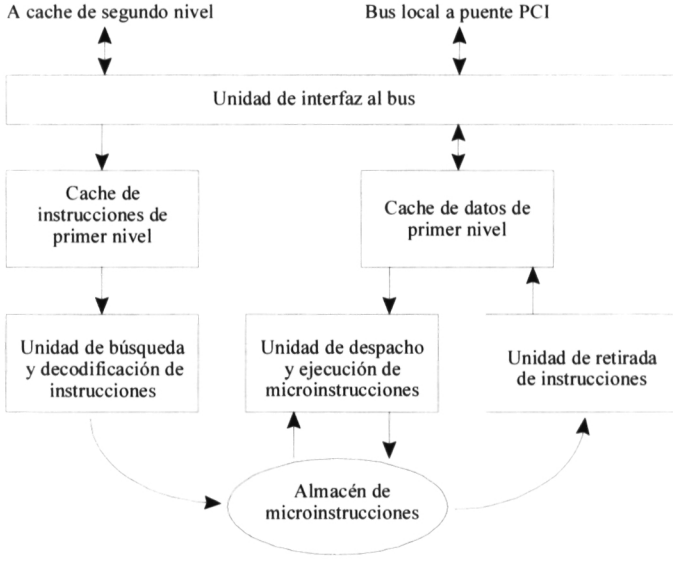


Figura 2.3: Ejecución dinámica de la microarquitectura P6 de Intel

La unidad de búsqueda y decodificación busca instrucciones y las descompone en microinstrucciones que deposita en el almacén. La unidad de despacho y ejecución toma las microinstrucciones del almacén y, si procede, las ejecuta. La unidad de retirada completa la ejecución de cada instrucción (una o varias microinstrucciones) y actualiza los registros. Las microinstruciones entran el almacén en orden, pueden ser ejecutadas en desorden y son retiradas, de nuevo, en orden.

La unidad de interfaz de bus es la responsable de comunicarse con la memoria del sistema (caché L2 y memoria principal). La caché L2 tiene un bus específico.

En la figura 2.4 se puede ver un diagrama más detallado de la microarquitectura.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

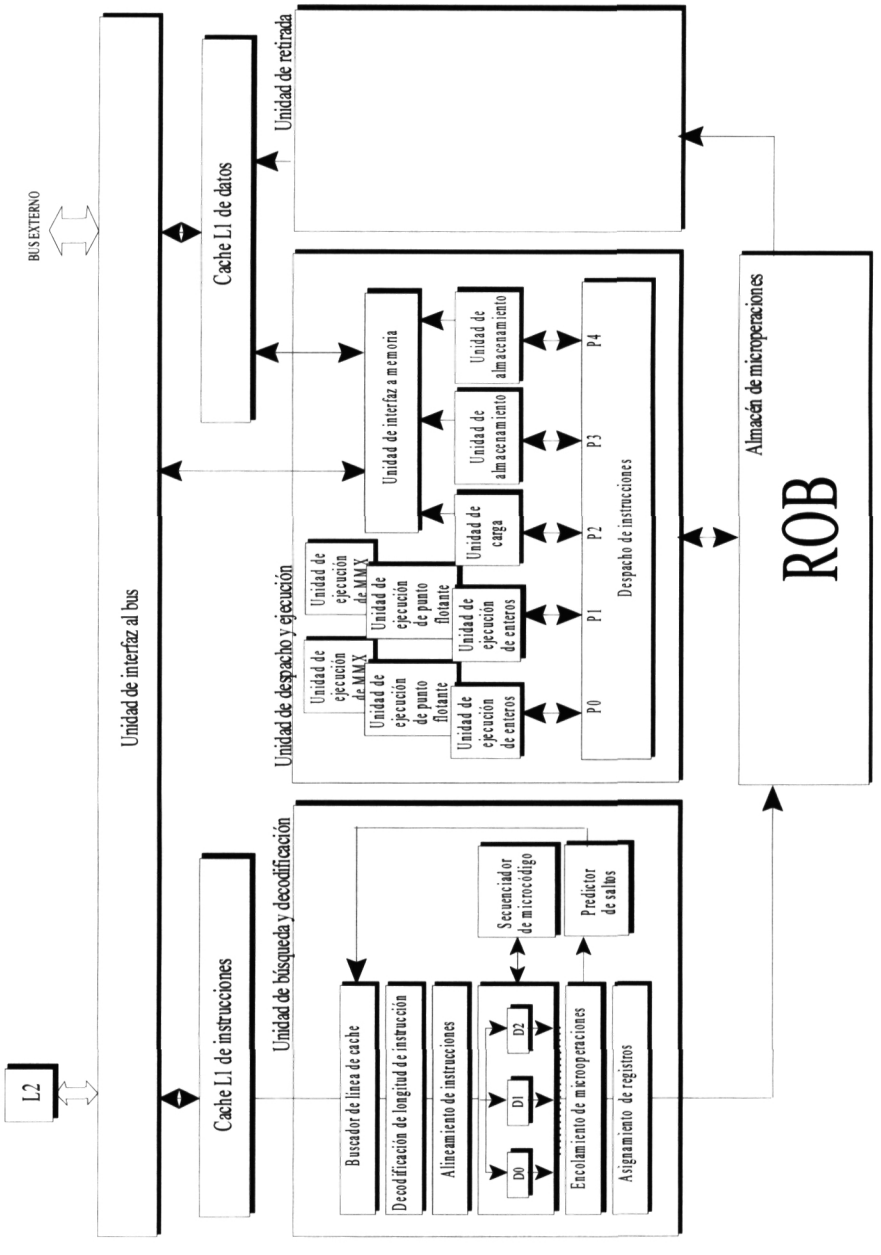


Figura 2.4: Diagrama de bloques de la microarquitectura P6 de Intel

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

La etapa de búsqueda de instrucciones lee una línea de caché de código (32 bytes) en cada ciclo de reloj, determina dónde empieza y termina cada instrucción IA-32 (alineamiento) y predice saltos si es necesario. El predictor de saltos BTB le dice a la caché de instrucciones donde buscar la nueva instrucción a ejecutar[88]. A continuación, transfiere las instrucciones (16 bytes) a la etapa de decodificación.

La etapa de decodificación consta de tres decodificadores que descomponen las instrucciones IA-32 en microinstrucciones llamadas μops. Hay dos decodificadores de instrucciones sencillas y uno para instrucciones complejas[89]. Las μops son de longitud fija y están formadas por un código de operación, 2 operandos fuente y uno destino. Los decodificadores generan hasta 6 μops en un ciclo de reloj.

Las referencias a los registros IA-32 se mapean en 40 registros auxiliares transparentes para el programador[90]. Las μops se envían al almacén de instrucciones que puede contener de 20 a 30 μops a la espera de su ejecución.

La unidad de despacho y ejecución controla la ejecución dinámica. Resuelve dependencias y conflictos de recursos y despacha las μops en cuanto tiene sus operandos de entrada disponibles y está libre alguna etapa de ejecución. Los resultados se devuelven al almacén de instrucciones.

Hay 5 puertos que admiten la ejecución de hasta 5 μops en un ciclo de reloj[91]. Las unidades de ejecución son de enteros, punto flotante, saltos, y generación de direcciones. En realidad, todos los puertos están conectados entre sí, de manera que se pueden intercambiar resultados en el siguiente ciclo de reloj.

La unidad de retirada verifica si las instrucciones han sido completamente ejecutadas y es su turno en la secuencia de ejecución. Solo se pueden retirar hasta 3 instrucciones por ciclo de reloj.

La unidad de interfaz de memoria se encarga de planificar los accesos a memoria, reordenando algunas solicitudes para prevenir bloqueos; en este sentido, realiza cargas especulativas. Los almacenamientos siempre se despachan en orden y no se permiten acciones especulativas.

Las cachés L1 y L2 son no bloqueantes. Si hay un fallo de caché, el acceso es "aparcado", de manera que se puedan hacer subsiguientes accesos a la

[88] El BTB usa un algoritmo adaptativo de 2 niveles con 512 entradas.

[89] Este tipo de instrucciones hacen uso de un microcódigo que las convierte en una cadena de hasta 4 μops.

[90] Es el denominado renombramiento de registros. En una máquina fuera de orden es fundamental para evitar dependencias de almacenamiento.

[91] En el mejor caso. En media se ejecutan 3.

Publicaciones y Divulgación Científica
UMA UNIVERSIDAD DE MÁLAGA

caché (hasta 4). Esto es importante en una máquina fuera de orden ya que de otra manera un fallo de caché, el cual podría haberse provocado especulativamente, atascaría todos los accesos posteriores.

3.3. Microarquitectura Netburst

La microarquitectura NetBurst (figura 2.5) es superescalar de 3 vías con hyperpipeline de 20 etapas. Su ejecución dinámica es muy avanzada y admite hasta 126 μops en ejecución. Usa una caché L1 de código especial denominada caché de trazas de ejecución (ETC, Execution Trace Cache). La caché de datos L1[92] es de baja latencia (dos ciclos de reloj) y la caché L2 de tipo ATC (Advanced Transfer Cache)[93]. El bus del sistema es de 400MHz[94].

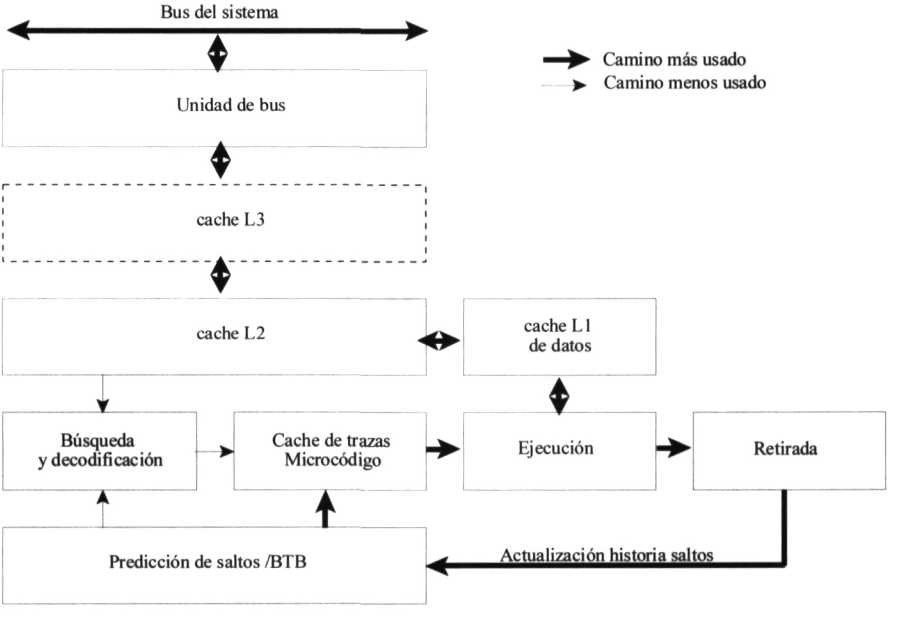


Figura 2.5: Microarquitectura NetBurst

La caché de trazas[95] de ejecución (ETC) almacena instrucciones IA-32 ya decodificadas. Esto permite en muchos casos eliminar la etapa de decodifi-

[92] 8KB asociativa de 4 vías y líneas de 64 bytes. writethrough.

[93] 256KB asociativa de 8 vías y líneas de 128 bytes, writeback.

[94] En realidad, la frecuencia del reloj es de 100MHz que se cuadruplica usando una técnica de quad pump.

[95] Se denomina traza a una secuencia ordenada de μops.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

cación del bucle de ejecución. El decodificador[96] sólo se usa cuando hay un fallo ETC y hay que ir a buscar las instrucciones a la caché L2. La ETC entrega hasta 3 μops por ciclo de reloj al núcleo de ejecución. Su capacidad es de $12K\mu$ops.

Netburst incorpora tecnología HT (HyperThreading) para ejecutar concurrentemente varios hilos de ejecución de un programa. La tecnología HT fue desarrollada para mejorar las prestaciones de la IA-32 cuando ejecuta código *multithread* y *single-threaded* bajo entornos multitarea. Un microprocesador IA-32 que soporta HT consta de 2 o más microprocesadores lógicos, cada uno de los cuales tiene su propia arquitectura software.

3.4. Microarquitectura Core

La microarquitectura Core combina características de las microarquitecturas Netburst y M incluyendo mejoras en la ejecución de instrucciones, el acceso a memoria y la gestión de energía.

Core usa *Wide Dynamic Execution*, una mejora que permite a cada núcleo (core) procesar hasta 4 instrucciones de forma simultánea (Netburst y M sólo llegan hasta 3). Además se reduce el tiempo de ejecución de las instrucciones mediante técnicas de microfusión y macrofusión. La microfusión, utilizada antes en la microarquitectura M, consiste en la agrupación (fusión) de microinstrucciones. La macrofusión agrupa pares de instrucciones en una simple microinstrucción. Para instrucciones multimedia, la técnica *Advanced Digital Media Boost* permite ejecutar instrucciones SIMD en 1 ciclo de reloj.

El acceso a memoria cache L2 está compartido por todos los núcleos del chip, lo que redunda en un aumento de eficiencia y en una disminución del consumo global; Intel llama esta técnica *Advanced Smart Cache*. El acceso a memoria principal usa *Smart Memory Access*, que optimiza el ancho de banda disponible y oculta la latencia de memoria mediante cargas especulativas.

Finalmente, la gestión de energía de Core denominada *Intelligent Power Capability* controla el consumo de energía de los núcleos, subsistemas lógicos y buses, facilitando para cada uno de estos elementos estados de bajo consumo pero sin descuidar el tiempo de respuesta del sistema.

[96] El decodificador decodifica 1 instrucción IA-32 por ciclo de reloj.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.5. Ejemplos

3.5.1. Comparativa de prestaciones de microprocesadores IA-32

Se explica un ejemplo propuesto por Intel que aclara como se ejecutan las instrucciones en el Pentium y en la microarquitectura P6. Se verá que la arquitectura P6 tiene una eficiencia muy superior a la del Pentium.

El ejemplo consiste en ejecutar un conjunto de instrucciones de un hipotético programa (ver parte izquierda de la figura 2.6).

Las instrucciones se van introduciendo en el microprocesador por orden (1,2,3,4,5,6,...). Observe que la instrucción 3 depende de la 1(por ejemplo, puede que necesite el resultado de la 1 para ejecutarse). Otro tanto ocurre entre las instrucciones 5 y 4. La instrucción 6 es una instrucción de salto; consideraremos que el microprocesador acierta en su predicción y que el programa salta a una subrutina cuya primera instrucción es la 42. A partir de aquí, se introducirán en el microprocesador las instrucciones 43, 44, 45,.. hasta la vuelta (el microprocesador vuelve a acertar en su predicción) al programa principal que se reanuda con la instrucción 7.

A la derecha de la figura se representa lo que tarda cada instrucción en "ejecutarse" y se mide en unidades de ciclo de reloj[97]. En primer lugar, veremos como se comporta el Pentium y a continuación repetiremos el proceso para la microarquitectura P6.

Todas las instrucciones son sencillas excepto la 4 y 7 que requieren dos ciclos. Las instrucciones 3,45 y 9 provocan un fallo de caché que se traduce en 4 ciclos de reloj.

[97] Observe que en este ejemplo el tiempo de ejecución no se refiere a la latencia de la instrucción. Una instrucción sencilla se "ejecuta" en un ciclo de reloj si el pipeline está lleno y todas las etapas se han ejecutado en un ciclo de reloj.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

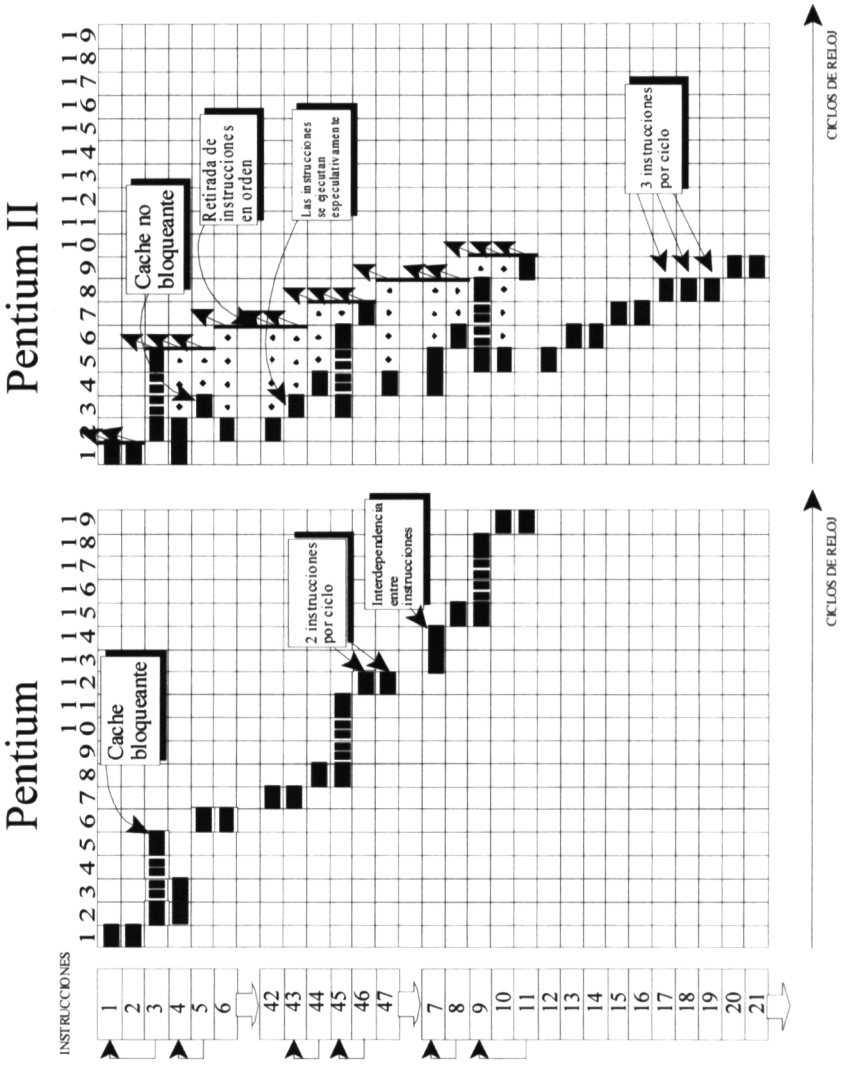


Figura 2.6: Comparativa de ejecución de instrucciones en dos microarquitecturas: Pentium y P6.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Ejecución de un Pentium

- Ciclo 1: Se introducen las instrucciones 1 y 2 y se completan en un ciclo de reloj. El IPC(Instrucciones por ciclo de reloj) es de 2/1=2.
- Ciclo 2: Se introducen las instrucciones 3 y 4. La 4 se completa en 2 ciclos de reloj. La 3 es un fallo de caché y no se completa hasta el quinto ciclo de reloj. La búsqueda de nuevas instrucciones se bloquea. El IPC baja a 4/5= 0.8.
- Ciclo 6: Se introducen las instrucciones 5 y 6. La 6 es un salto a subrutina que el Pentium predice correctamente. La dirección de salto está en la caché de primer nivel. La dos instrucciones se completan en un ciclo de reloj. El IPC sube a 6/6=1.
- Ciclo 7: Se introducen las instrucciones 42 y 43 y se completan en un ciclo de reloj. El IPC sube a 8/7=1.14.
- Ciclo 8: Se introducen las instrucciones 44 y 45. La 44 se completa en un ciclo de reloj. La 45 es un fallo de caché y no se completa hasta el ciclo 11. El IPC baja a 10/11=0.9.
- Ciclo 12: Se introducen las instrucciones 46 y 47 y se completan en un ciclo de reloj. Se retorna al programa principal. El IPC aumenta a 12/12=1.
- Ciclo 13: Se introduce la instrucción 7 y se completa en dos ciclos de reloj. No se puede ejecutar en paralelo con la 8 ya que son dependientes. El IPC baja a 13/14=0.92.
- Ciclo 15: Se introducen las instrucciones 8 y 9. La 8 se completa en un ciclo de reloj. La 9 es un fallo de caché y no se completa hasta el ciclo 18. La búsqueda de nuevas instrucciones se bloquea. El IPC baja a 15/18=0.83.
- Ciclo 19: Se introducen las instrucciones 10 y 11 y se completan en un ciclo de reloj. El IPC sube a 17/19=0.89

Ejecución en una microarquitectura P6

- Ciclo 1: Se introducen las instrucciones 1, 2 y 4. La 1 y 2 se completan en un ciclo de reloj. La 4 se completa en 2 ciclos de reloj. La instrucción 3 no puede empezar ya que depende del resultado de la instrucción 1. Se retiran las instrucciones 1 y 2. El IPC es de 2/1=2.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Ciclo 2: Se introducen las instrucciones 3, 6 y 42. La 3 es un fallo de caché y no se completa hasta el ciclo 5. La instrucción 5 no puede empezar ya que depende del resultado de la instrucción 4. La instrucción 4 se completa pero tiene que guardar su resultado en un registro auxiliar (ya que hay que guardar la secuencia original, la instrucción 3 no se ha ejecutado). Las instrucciones 6 y 42 se completan en un ciclo de reloj. No se retira ninguna instrucción. El IPC se reduce a 2/2=1.

- Ciclo 3: Se introducen las instrucciones 5, 43 y 45. La instrucción 44 no puede empezar ya que depende del resultado de la instrucción 43. Todavía se está esperando la instrucción 3. La 45 es un fallo de caché y no se completa hasta el ciclo 6. Las instrucciones 5 y 43 se completan en un ciclo de reloj. El IPC baja a 2/3=0.67.

- Ciclo 4: Se introducen las instrucciones 44, 47 y 7. La instrucción 46 no puede empezar ya que depende del resultado de la instrucción 45. El IPC sigue bajando: 2/4=0.5.

- Ciclo 5: Se introducen las instrucciones 9,10 y 12. La 9 es un fallo de caché y no se completa hasta el ciclo 8. Se completa la instrucción 3. La 8 no se empieza ya que depende del resultado de la 7. La 11 no se empieza ya que depende del resultado de la 9. Se retiran las instrucciones 3,4 y 5. El IPC sube a 5/5=1.

- Ciclo 6: Se introducen las instrucciones 8,13 y 14. Se completan en un ciclo de reloj. La 11 no empieza. Se acaba la 45, luego se puede introducir la 46. Se retiran la 6,42 y 43. El IPC sube a 8/6=1.33.

- Ciclo 7: Se introducen las instrucciones 46, 15 y 16 y se completan en un ciclo de reloj. Se retiran la 44, 45 y 46. El IPC sube a 11/7=1.57.

- Ciclo 8: Se introducen las instrucciones 17, 18 y 19 y se completan en un ciclo de reloj. Se retiran 47,7,8. El IPC sube a 17/9=1.9.

- Ciclo 9: Se introducen las instrucciones 11, 20 y 21 y se completan en un ciclo de reloj. Se retiran 9,10,11. El IPC llega a 17/9=1.9 y además hay 10 instrucciones ejecutadas especulativamente esperando a ser retiradas.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4. Entorno de ejecución básico

El entorno de ejecución básico es el conjunto de recursos que tiene el microprocesador IA-32 para ejecutar instrucciones y almacenar información (código, datos y estado). Se excluyen recursos adicionales de la arquitectura a nivel de sistema que proveen soporte al sistema operativo y al software de desarrollo del sistema.

4.1. Modos de operación

El modo de operación determina qué instrucciones y características arquitecturales están accesibles. Se distinguen tres modos[98]: Modo protegido, modo real y modo de gestión del sistema.

- El modo protegido es el estado natural del microprocesador. En este modo están habilitadas todas las instrucciones y características arquitecturales, estando disponible, por tanto, toda la capacidad y prestaciones del microprocesador. Éste es el modo recomendado que todas las nuevas aplicaciones y S.O. deben seleccionar. Entre las capacidades del modo protegido está su habilidad para ejecutar directamente programas 8086 en modo de direccionamiento real en un entorno multitarea protegido. Esta característica se denomina modo virtual-8086 o modo V86[99].

- El modo real o modo de direccionamiento real provee el entorno de programación del microprocesador 8086, con algunas extensiones. La inicialización RESET coloca al microprocesador en modo real, mientras que una sencilla instrucción, puede conmutar a modo protegido.

- Modo de gestión de sistema. Este modo (SMM, System Management Mode), introducido en el 386SL y 486, provee un S.O. y un mecanismo transparente e independiente de la aplicación para implementar gestión de alimentación del sistema y características de diferenciación OEM (Original Equipment Manufacturer). SMM se activa mediante una interrupción externa o un comando del APIC (Advanced Programmable

[98] El Pentium 4 EM64T (Extended Memory 64 Technology) introduce un modo adicional denominado IA-32e que le permite trabajar con una arquitectura de registros de 64 bits (IA-64). No se estudia en este apartado.

[99] Este modo no es realmente un modo del microprocesador, es simplemente un atributo que puede ser habilitado para cualquier tarea (con el software apropiado) mientras se está en el modo protegido.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Interrupt Controller), que conmuta a un espacio de direcciones separado mientras que salva el contexto entero del microprocesador. El código específico SMM puede entonces ser ejecutado de forma transparente.

4.2. Organización de memoria

La memoria que direcciona el bus del microprocesador se denomina memoria física (figura 2.7). Esta memoria se organiza como una secuencia de bytes, donde cada byte tiene asignada una única dirección, denominada dirección física. La dirección física puede variar[100] de 0 hasta $2^{36} - 1$. Por tanto, el espacio de direcciones de memoria puede[101] llegar a ser de 64GB.

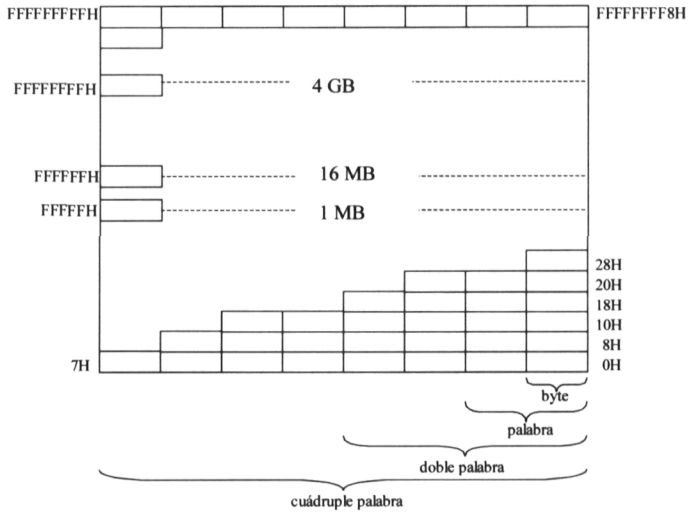


Figura 2.7: Organización de memoria de la arquitectura IA-32 de Intel

Se pueden direccionar bytes, palabras, dobles palabras y cuádruples palabras. Es una organización little endian. En la figura 2.8 se pueden ver algunos ejemplos del orden de los bytes cuando se referencian como operandos de memoria. Las palabras, dobles palabras y cuádruples palabras no necesitan estar alineadas en memoria en sus límites naturales[102].

[100] Si el microprocesador soporta EM64T es espacio es mayor.

[101] Depende del microprocesador, consulte la tabla 2.1.

[102] Los límites naturales de palabras, dobles palabras y cuádruples palabras son direcciones pares, múltiplos de 4 y múltiplos de 8 respectivamente.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

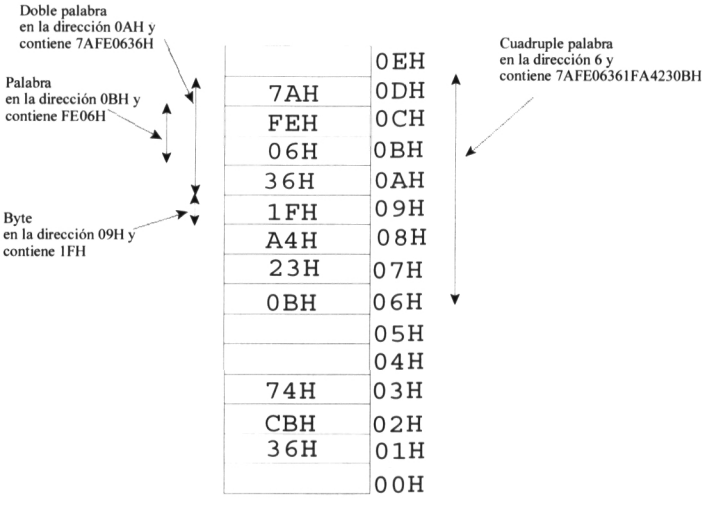


Figura 2.8: Ejemplos de direccionamiento de datos

Todas las transferencias de datos ocurren como resultado de uno o dos ciclos de bus. Si los datos no están alineados (una palabra o doble palabra cruza una dirección múltiplo de 4, o una cuádruple palabra que cruza una dirección múltiplo de 8) son necesarios dos ciclos de bus.

4.2.1. Gestión de memoria

En modo protegido, el microprocesador usa gestión de memoria. Esto se traduce en que los programas no direccionan directamente memoria física sino un modelo de memoria denominado espacio lógico de direcciones[103].

Se pueden establecer espacios lógicos segmentados y/o paginados:

- Espacio segmentado: La dirección lógica tiene dos componentes: un selector de segmento y un desplazamiento dentro del segmento seleccionado. Una tabla de descriptores de segmento traduce la dirección lógica a dirección lineal. Esta dirección lineal es la dirección física si no se implementa paginación. Puede haber hasta 16383 segmentos de diferentes tamaños, siendo el máximo tamaño de cada uno de ellos de 4GB.

- Espacio paginado: El espacio físico de direcciones de 4GB se divide en

[103] Ver apartado 9.2 para obtener información más detallada.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

páginas de 4KB. Una tabla de páginas traduce la dirección lineal a dirección física. Hay, por tanto, 2^{20} páginas de 4KB cada una.

A partir de la familia P6, la arquitectura IA-32 soporta 64GB de memoria física aunque requiere la ayuda de la gestión de memoria para poder acceder a todo el espacio.

En la figura 2.9 se puede ver el mecanismo de gestión de memoria que implementan los microprocesadores IA-32.

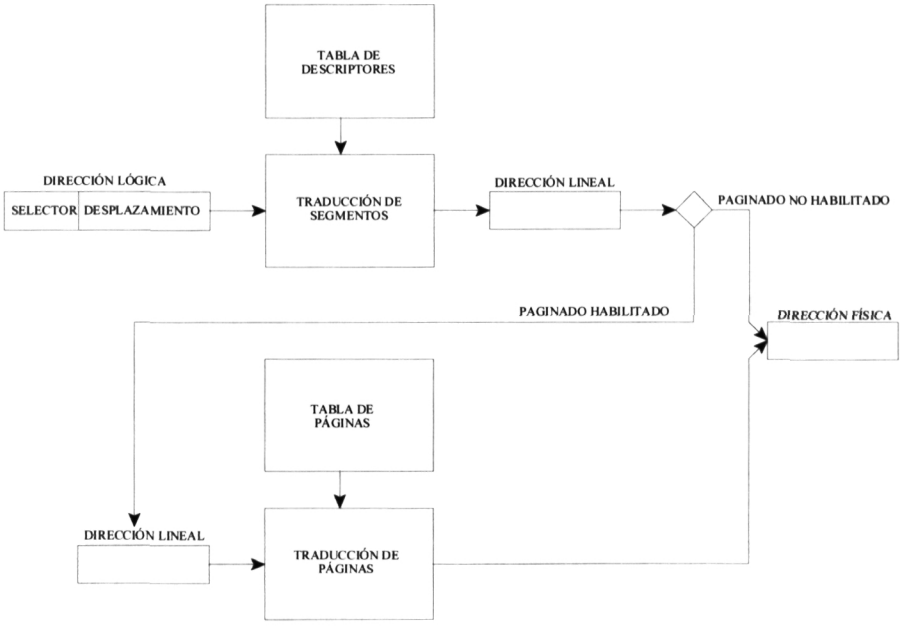


Figura 2.9: Mecanismos de gestión de memoria de IA-32

El espacio segmentado se puede convertir en un espacio lineal o plano de 4GB sin ninguna otra estructura, donde la dirección de cada byte se denomina dirección lineal y va de 0 a $2^{32} - 1$. En este caso, la dirección lógica coincide con la dirección lineal.

En modo real, existe el espacio de direccionamiento real que es el espacio pseudosegmentado (ya que no hace uso de la gestión de memoria) que tenía el primer microprocesador IA-32: El 8086. En la figura 2.10 se ilustra su funcionamiento. Los segmentos son de 64KB y el espacio está limitado a 1MB.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

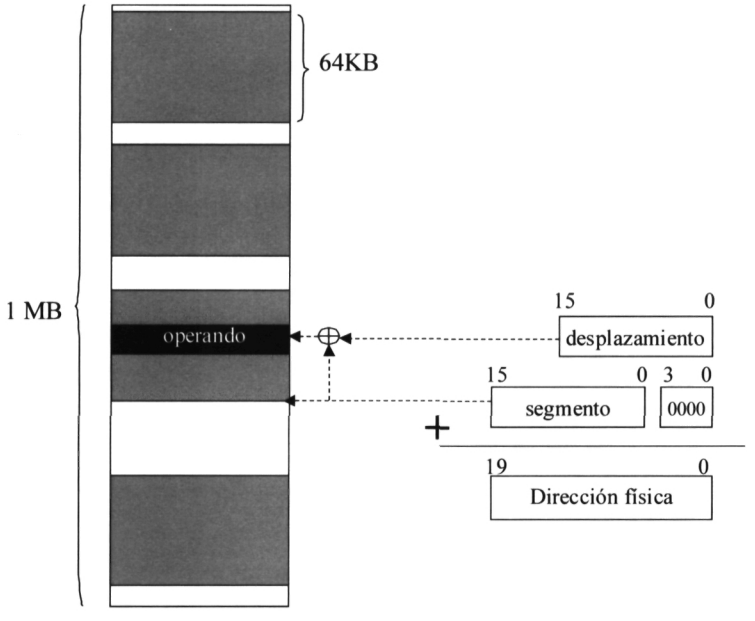


Figura 2.10: Segmentación en el 8086

4.3. Registros de programación de aplicaciones

La figura 2.11 muestra el conjunto de registros del modo de ejecución básico. Se distinguen tres grupos: Registros de propósito general, registros de punto flotante y registros SIMD.

4.3.1. Registros de propósito general

La arquitectura IA-32 dispone de 16 registros de propósito general. Estos registros se pueden agrupar en 8 registros generales, 6 registros de segmento, un registro de estado y control y un puntero de instrucciones.

El puntero de instrucciones (EIP, Extended Instruction Pointer) es un registro de 32 bits que apunta a la dirección de la siguiente instrucción a ejecutar. Se usa en combinación con los registros de segmento, ya que lo que contiene, en realidad, es el desplazamiento dentro del segmento de código.

Los 8 registros generales son de 32 bits y guardan operandos para operaciones aritméticas y lógicas y para cálculo de direcciones[104]. Los nombres de

[104]Excepto ESP que no puede ser usado como operando índice.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

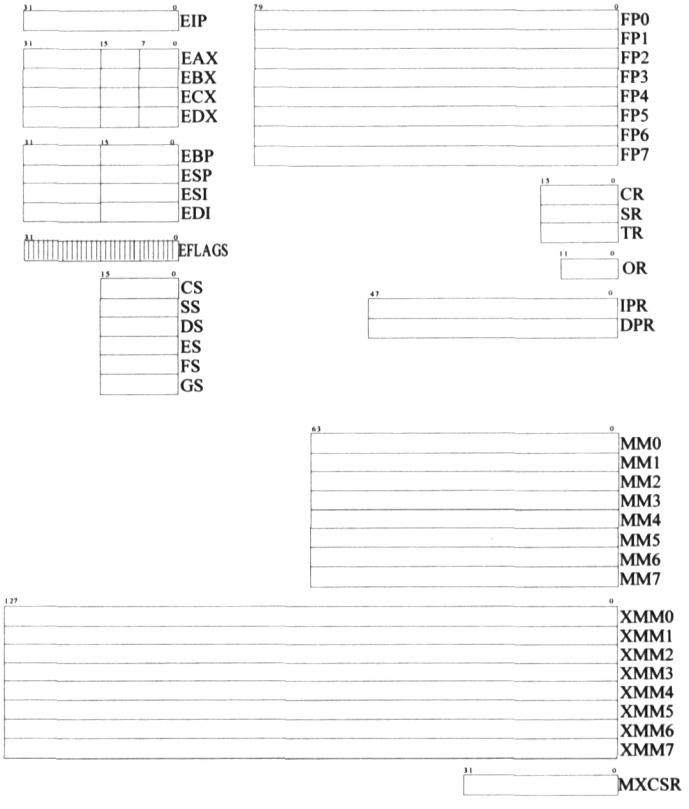


Figura 2.11: Registros básicos

estos registros se derivan de los nombres de los registros del 8086 (AX, BX, CX, DX, SP, BP, SI y DI). Podemos distinguir entre:

- 4 registros de datos: EAX,EBX,ECX y EDX
- 2 registros de punteros: ESP y EBP
- 2 registros índice: ESI y EDI

Por compatibilidad con el 8086, en los registros de datos se puede direccionar la palabra menos significativa, denominándose en este caso registros AX, BX, CX y DX. También se pueden direccionar los bytes alto y bajo de este palabra; por ejemplo, en el registro AX el byte alto y bajo se denominan AH y AL, respectivamente. Sin embargo, los registros de punteros e indices sólo admiten direccionamientos de 32 bits (ESP, EBP, ESI y EDI) y de su palabra menos significativa (SP, BP, SI y DI).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Estos registros son de propósito general, aunque con algunas excepciones, ya que hay instrucciones que asignan registros específicos para su procesamiento[105].

Los 6 registros de segmento (CS,DS,SS,ES, FS,GS) son de 16 bits y sirven para determinar en cada momento qué segmentos de memoria están disponibles. Cada registro se asocia con un tipo particular de segmento:

- El registro CS, hace referencia al segmento de código. El microprocesador busca las instrucciones en el segmento de código, utilizando como desplazamiento el contenido del registro EIP. El registro CS se carga como resultado de interrupciones, excepciones, o instrucciones que transfieren el control entre segmentos (por ejemplo, CALL, RET y JMP).

- El registro SS, hace referencia a la pila. Antes de que un procedimiento sea llamado, hay que asignar una zona de memoria a la pila. La pila guarda la dirección de retorno, parámetros pasados a la rutina llamada y variables temporales asignadas al procedimiento. El registro SS puede ser modificado, lo que permite establecer varias pilas. El contenido de ESP marca la cima de la pila.

- Los registros DS, ES, FS y GS, hacen referencia a distintos segmentos de datos. Por tanto, pueden estar disponibles 4 segmentos al mismo tiempo.

El registro de estado y control EFLAGS es de 32 bits. Es un registro que reporta y modifica el estado del microprocesador.

4.3.2. Registros de punto flotante

Los registros x87 FPU proveen un entorno de ejecución para operaciones con datos en punto flotante. Hay 14 registros:

- 8 registros de datos de punto flotante de 80 bits: FP[7:0]
- 1 registro de control de 16 bits: CR
- 1 registro de estado de 16 bits: SR
- 1 registro de etiqueta de 16 bits: TR
- 1 registro de código de operación de 11 bits: OR

[105] Las instrucciones que usan registros específicos incluyen multiplicación y división de doble precisión, E/S, cadenas, traslación, bucles, rotación y desplazamiento variable e instrucciones de pila.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- 1 registro puntero de instrucciones de 48 bits: IPR
- 1 registro puntero de datos de 48 bits: DPR

4.3.3. Registros SIMD

Los registros SIMD proveen un entorno de ejecución para operaciones SIMD (Single Instruction Multiple Data) sobre datos empaquetados. Hay 17 registros:

- 8 registros de 64 bits:MM[7:0]
- 8 registros de 128 bits: XMM[7:0]
- 1 registro de estado y control de 32 bits: MXCSR

Los registros MM[7:0] se "mapean" sobre los registros de datos de punto flotante FP[7:0]. Este mapeo evita añadir nuevos estados facilitando la compatibilidad, pero tiene como inconveniente el que no se puedan usar simultáneamente instrucciones de punto flotante y MMX.

4.4. Direccionamiento de operandos

Los operandos de una instrucción pueden estar en la misma instrucción (operando inmediato), en un registro, en memoria o en un puerto de E/S. La dirección de un operando en memoria se determina especificando el segmento que contiene al operando y el desplazamiento del operando dentro del segmento, también llamado dirección efectiva. La dirección efectiva es la suma de varios componentes:

$$direccion\ efectiva = base + (indice \times escala) + desplazamiento \qquad (2.2)$$

En la tabla 2.2 se muestran los registros que se pueden utilizar para cada campo.

Campo	Registros
SEGMENTO	CS,SS,DS,ES,FS,GS
BASE	EAX,EBX,ECX,EDX, ESP,EBP,ESI,EDI
ÍNDICE	EAX,EBX,ECX,EDX,EBP,ESI,EDI
ESCALA	1,2,4,8
DESPLAZAMIENTO	ninguno, 8 bits, 16 bits,32 bits

Tabla 2.2: Registros utilizados en los diferentes campos de la dirección efectiva

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4.5. Ejemplos

4.5.1. Uso del depurador del PC

El depurador del PC (debug) es un sencillo programa MS-DOS que se encuentra en todos los PCs compatibles. En este ejemplo, sólo se presenta la manera de ejecutarlo, su menú de ayuda y un comando de lectura de registros.

Para entrar en el depurador del PC, abra una ventana del sistema y ejecute el comando `debug`:

```
c:\>debug
-
```

Verá que cambia el cursor ">", típico de MSDOS, por otro símbolo: "-". A continuación escriba y ejecute el comando de ayuda:

```
-?
```

En su pantalla aparecerán todos los comandos del depurador (ver figura 2.12).

```
C:\>debug
-?
assemble     A [dirección]
compare      C intervalo de direcciones
dump         D [intervalo]
enter        E dirección [lista]
fill         F lista de rango
go           G [=dirección] [direcciones]
hex          H valor1 valor2
input        I puerto
load         L [dirección] [unidad] [primer sector] [número]
move         M intervalo de direcciones
name         N [ruta] [lista de argumentos]
output       O byte de puerto
proceed      P [=dirección] [número]
quit         Q
register     R [registro]
search       S lista de rango
trace        T [=dirección] [valor]
unassemble   U [intervalo]
write        W [dirección] [unidad] [primer sector] [número]
allocate expanded memory        XA [N.páginas]
deallocate expanded memory      XD [identificador]
map expanded memory pages       XM [páginaL] [páginaP] [identificador]
display expanded memory status  XS
-
```

Figura 2.12: Comandos del depurador del PC

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Por ahora sólo ejecutaremos el comando `r` que nos permite visualizar el valor de todos los registros del microprocesador[106]:

```
-r AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0B9A ES=0B9A SS=0B9A CS=0B9A IP=0100 NV UP EI PL NZ NA PO NC
0B9A:0100 0F              DB    0F
```

Además de los registros, el depurador muestra en su tercera línea la dirección de la próxima instrucción que ejecutará el microprocesador: `0B9A:0100`, donde `0B9A` es el contenido del registro de segmento de código CS y `0100` el desplazamiento ubicado en el registro IP[107].

Si queremos modificar el valor de un registro escribiremos el comando r seguido del registro a modificar. El depurador nos mostrará su valor actual y nos permitirá introducir un valor nuevo, por ejemplo 3a7:

```
-r ax AX 0000: 3a7
```

Si ejecutamos de nuevo el comando veremos que el registro ha cambiado de valor.

[106] Los comandos y datos introducidos se muestran en minúsculas y las presentaciones que va mostrando la ventana del sistema en mayúsculas.

[107] Observe que en su ordenador el valor de CS puede que no coincida con el aquí mostrado.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5. Tipos de datos

Los tipos de datos se pueden clasificar en: Fundamentales, numéricos, punteros, campo de bits, cadenas, SIMD y BCD:

- Fundamentales: Son bytes, palabras, dobles palabras (a partir del 386), cuádruples palabras (a partir del 486) y dobles cuádruples palabras (a partir del Pentium III).
- Numéricos: Son interpretaciones adicionales de los datos fundamentales para permitir operaciones sobre tipos de datos numéricos: Enteros con y sin signo y números de punto flotante.
- Punteros: Son direcciones de localizaciones de memoria. Se definen dos tipos: Puntero cercano y puntero lejano. Un puntero cercano es un desplazamiento de 32 bits (también llamado dirección efectiva) dentro de un segmento. Un puntero lejano es una dirección lógica de 48 bits que incluye un selector de segmento de 16 bits y un desplazamiento de 32 bits.
- Campos de bits: Son secuencias contiguas de bits. Pueden empezar en cualquier posición de bit de cualquier byte en memoria y pueden contener hasta 32 bits.
- Cadenas: Son secuencias continuas de bits, bytes, palabras o dobles palabras. Las cadenas de bits pueden empezar en cualquier posición de cualquier byte y pueden contener hasta $2^{32} - 1$ bits. Las cadenas de bytes pueden contener bytes, palabras y dobles palabras pudiendo ir desde 0 hasta $2^{32} - 1$ bytes.
- SIMD: Son datos empaquetados de 64 bits y 128 bits. Consisten en tipos de datos fundamentales (bytes empaquetados, palabras empaquetadas, dobles palabras empaquetadas y cuádruples palabras empaquetadas) e interpretaciones numéricas de los tipos fundamentales para su uso en operaciones de datos empaquetados enteros y de punto flotante.
- BCD. Son datos BCD que se ubican en bytes o en bloques de 80 bits. En el caso de bytes, pueden ser empaquetados[108] o no[109]. El bloque de 80 bits permite almacenar 18 números BCD y un campo de signo.

[108] Cada nibble del byte guarda un número BCD.

[109] El número BCD se almacena en el nibble bajo del byte.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

6. Conjunto de instrucciones

El conjunto de instrucciones se puede dividir en los siguientes grupos: Propósito general, entrada/salida, x87 FPU, gestión de estado, SIMD y sistema.

6.1. Instrucciones de propósito general

Las instrucciones de propósito general son las que los programadores usan habitualmente para escribir las aplicaciones y programas del sistema. Operan sobre datos y direcciones contenidos en memoria y en los registros de propósito general. Se clasifican en:

- Instrucciones de transferencia de datos. Mueven operandos entre memoria y registros. Hay instrucciones de propósito general (del tipo `MOV`), manipulación de pila (cargan y descargan datos de la pila) y de conversión de tipo[110].

- Instrucciones aritméticas binarias. Operan sobre datos codificados en binario. Incluyen suma, resta, multiplicación, división, incremento, decremento, comparación y cambio de signo.

- Instrucciones aritméticas decimales. Combinan instrucciones binarias y aritméticas. Se utilizan en ajuste BCD empaquetado y ajuste BCD no empaquetado.

- Instrucciones lógicas. Incluyen operaciones booleanas (`AND`,`OR`,`XOR` y `NOT`), bit test and modify, bit scan, desplazamiento, rotación, condición byte-set-on, y de test[111].

- Instrucciones de transferencia de control. Incluye saltos incondicionales y condicionales e interrupciones software.

- Instrucciones de cadenas. Operan sobre cadenas de bytes que se mueven a/desde la memoria.

[110] Convierten bytes en palabras, palabras en dobles palabras y dobles palabras en cuádruples palabras.

[111] *bit test and modify* salva el valor de un bit y lo modifica, *bit scan* busca el valor de un bit dentro de una palabra o doble palabra y *condición byte-set-on*pone un byte a 0 ó 1 dependiendo de cierta condición.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Instrucciones de llamadas a procedimientos: `CALL`, `RET`, `ENTER` y `LEAVE`. Estas instrucciones usan la pila para salvar el estado del procedimiento llamante, pasar parámetros a los procedimientos llamados y almacenar variables locales para el procedimiento que se está ejecutando.
- Instrucciones de control de flag. Cambian el estado de los bits del registro EFLAGS.
- Instrucciones de registro de segmento. Permiten a los punteros lejanos ser cargados en los registros de segmento.
- Otras instrucciones que no encajan en los grupos anteriores son: carga de dirección efectiva, no operación, identificación del microprocesador, etc.

6.2. Instrucciones de entrada/salida

Las instrucciones de entrada/salida (E/S) mueven datos entre los puertos de E/S y los registros del microprocesador o memoria. Se clasifican en:

- Instrucciones de transferencia sencillas (byte, palabra o doble palabra) entre un puerto de E/S y un registro de propósito general.
- Instrucciones de transferencia de cadenas (cadenas de bytes, palabras o dobles palabras) entre un puerto de E/S y memoria.

6.3. Instrucciones x87 FPU

Las instrucciones x87 FPU trabajan con datos en punto flotante, enteros y BCD. Se clasifican en:

- Instrucciones de transferencia de datos. Transfieren operandos en punto flotante, enteros y BCD entre memoria y los registros x87FPU.
- Instrucciones aritméticas. Llevan a cabo operaciones básicas aritméticas con operandos enteros y de punto flotante.
- Instrucciones de comparación. Examinan o comparan operandos enteros y de punto flotante.
- Instrucciones trascendentales. Ejecutan operaciones básicas logarítmicas y trigonométricas sobre operandos en punto flotante.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Instrucciones de carga de constantes. Cargan constantes comunes[112] en los registros de punto flotante.

- Instrucciones de control FPU x87. Opera sobre la pila de registros x87 FPU y salva y restablece el estado x87 FPU

6.4. Instrucciones de gestión de estado

Son dos instrucciones que salvan y restablecen el estado de los registros x87, MMX, XMM y MXCSR.

6.5. Instrucciones SIMD

Una instrucción SIMD (Single Instruction Multiple Data) se caracteriza porque se ejecuta sobre varios datos a la vez. Los datos son operandos en punto flotante y enteros empaquetados, contenidos en los registros MMX y XMM. Se distinguen cuatro grupos: Instrucciones MMX, instrucciones SSE, instrucciones SSE2 e instrucciones SSE3.

6.5.1. Instrucciones MMX

Las instrucciones MMX (tabla 2.3) usan registros MMX de 64 bits para almacenar bytes (8 bytes empaquetados), palabras (4 palabras empaquetadas), dobles palabras (2 dobles palabras empaquetadas) o una cuádruple palabra.

TIPO	Instrucciones
Aritméticas	PADD[B,W,D], PADDS[B,W], PADDUS[B,W] PSUB[B,W,D], PSUBS[B,W], PSUBUS[B,W], PMULHW, PMULLW, PMADDWD
Lógicas	PAND,PANDN,POR,PXOR
Comparación	PCMPEQ[B,W,D], PCMPGT[B,W,D]
Transferencia de datos	MOV[D,Q]
Conversión	PACKUSWB, PACKSS[WB,DW], PUNPCKH[BW,WD,DQ], PUNPCKL[BW,WD,DQ]
Desplazamiento	PSLL[W,D,Q], PSRL[W,D,Q], PSRA[W,D]
Gestión de estado	EMMS

Tabla 2.3: Conjunto de instrucciones MMX. Todas las instrucciones empiezan por P. Los operandos de las instrucciones pueden ser bytes (B), palabras (W), dobles palabras (D) o cuádruples palabras (Q).

Una instrucción de 8 bytes empaquetados calcula 8 resultados en paralelo. De forma similar, una instrucción de 4 palabras empaquetadas genera 4 resultados y una instrucción de 2 dobles palabras empaquetadas 2 resultados.

[112] Como por ejemplo π.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Las instrucciones MMX se pueden clasificar en aritméticas (suma, suma saturante, resta, resta saturante, multiplicación y suma ponderada), lógicas, comparación, transferencia de datos, conversión (empaquetado y desempaquetado de datos), desplazamiento y gestión de estado (para conmutar entre MMX y FP).

6.5.2. Instrucciones SSE

Las instrucciones SSE, SSE2 y SSE3 son extensiones de las instrucciones MMX que hacen uso de los registros XMM de 128 bits para empaquetar los datos.

Las instrucciones SSE trabajan sobre operandos empaquetados en punto flotante de precisión sencilla y se dividen en los siguientes grupos: Transferencia de datos, aritmética empaquetada, comparación, lógicas, shuffle y desempaquetado y conversión.

Las instrucciones SSE2 permiten trabajar con operandos empaquetados en punto flotante de doble precisión. Se clasifican en: Instrucciones SIMD de punto flotante, instrucciones SIMD de enteros, instrucciones de conversión entre punto flotante y enteros e instrucciones de conversión entre XMM Y MMX. También hay instrucciones que permiten a los programadores de aplicaciones controlar la "cacheabilidad" de los datos.

Las instrucciones SSE3 son 13 nuevas instrucciones que mejoran las prestaciones de las SSE2. Entre ellas, hay instrucciones específicas para acelerar la sincronización de los threads en el Pentium HT.

6.6. Instrucciones del sistema

En modo protegido, existen instrucciones del programador de sistemas que no son reconocidas fuera del modo protegido. Se incluyen instrucciones de verificación de parámetros de punteros, direccionamiento de tablas del descriptores, multitarea, punto flotante, E/S, interrupciones, depurado, gestión de caché y control del sistema.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

6.7. Ejemplos

6.7.1. Suma usando el depurador del PC

Este ejemplo consiste en ejecutar en modo real una instrucción de suma de registros haciendo uso del depurador del PC. La operación que se quiere llevar a cabo es la siguiente:

```
AX = 3A7H    ;Carga el valor 3A7H en el registro AX

BX = 92AH    ;Carga el valor 92AH en el registro BX

ADD AX,BX    ;AX = (AX)+(BX)
```

A continuación, se muestran los comandos y datos introducidos (minúsculas) y las presentaciones (mayúsculas) que va mostrando la ventana del sistema. En primer lugar, cargamos los registros con los nuevos valores:

```
c:\>debug

-r AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0B9A ES=0B9A SS=0B9A CS=0B9A IP=0100 NV UP EI PL NZ NA PO NC
0B9A:0100 0F              DB    0F

-r ax AX 0000: 3a7

-r bx BX 0000: 92a

-r AX=03A7 BX=092A CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0B9A ES=0B9A SS=0B9A CS=0B9A IP=0100 NV UP EI PL NZ NA PO NC
0B9A:0100 0F             DB    0F
```

A continuación, escribimos en memoria de código, justo en la dirección a la que apunta el puntero de instrucciones, el código de la instrucción `ADD AX,BX` que es `d801`:

```
-e 100 0B9A:0100 0F.01

-e 101 0B9A:0101 0F.d8
```

Si ejecutamos el comando `r`, confirmaremos en la tercera línea que efectivamente este código es el correspondiente a la instrucción que queremos ejecutar:

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

```
-r AX=03A7 BX=092A CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0B9A ES=0B9A SS=0B9A CS=0B9A IP=0100 NV UP EI PL NZ NA PO NC
0B9A:0100 01D8 ADD AX,BX
```

A continuación hacemos uso del comando `t` para ejecutar una instrucción:

```
-t AX=0CD1 BX=092A CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0B9A ES=0B9A SS=0B9A CS=0B9A IP=0102 NV UP EI PL NZ AC PE NC
0B9A:0102 B93028 MOV CX,2830
```

Podemos comprobar que la instrucción se ha ejecutado, AX ahora contiene el resultado de la suma e IP apunta a la siguiente instrucción a ejecutar.

Se podría haber hecho mucho más sencillo, usando el comando `a`, que nos permite introducir código ensamblador directamente:

```
-a
0B9A:0100 ADD AX,BX
0B9A:0102
-t
```

6.7.2. Aritmética saturante

Una imagen en color se representa mediante píxeles, donde cada pixel tiene tres componentes: Rojo, verde y azul. Cada componente se codifica con un número determinado de bits. Suponga que usa un byte para codificar cada componente y que suma dos intensidades del componente rojo de un pixel. El resultado no puede ser nunca mayor que `FFH` (rojo) ni menor que `00H` (negro) (ver figura 2.13).

MMX incorpora aritmética saturante para facilitar este tipo de operaciones (ver figura 2.14). En aritmética saturante un overflow provoca que el resultado sea fijado a su valor máximo y un underflow a su valor mínimo. Esta técnica evita las largas series de chequeos de overflows y underflows necesarios con instrucciones tradicionales.

6.7.3. Multiplicación SIMD

Las multiplicaciones sencillas son manejadas por `PMULHW` y `PMULLW`. Estas instrucciones operan sólo sobre valores de 16 bits generando 4 resultados en paralelo. La instrucción `PMULHW` calcula sólo la palabra más significativa del resultado en el registro destino. En algunas situaciones, este resultado de 16

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

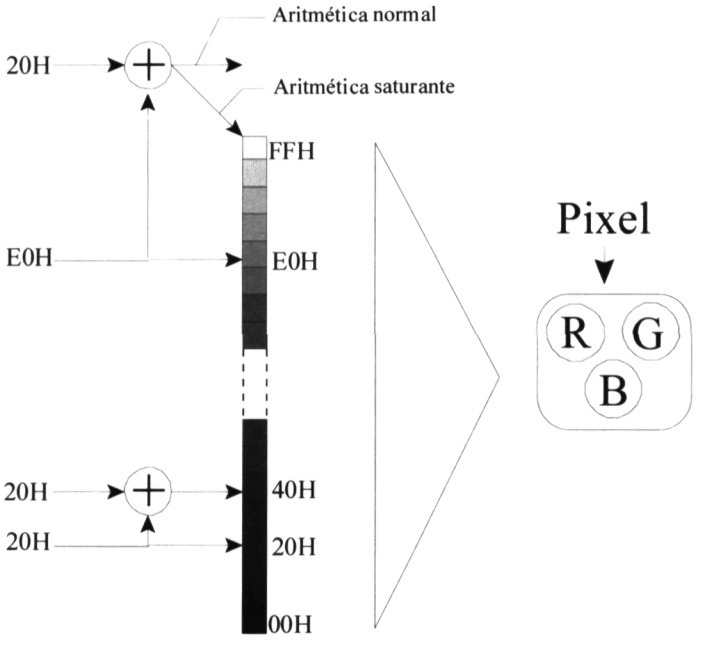


Figura 2.13: Ejemplo de aritmética saturante: Operaciones con pixeles

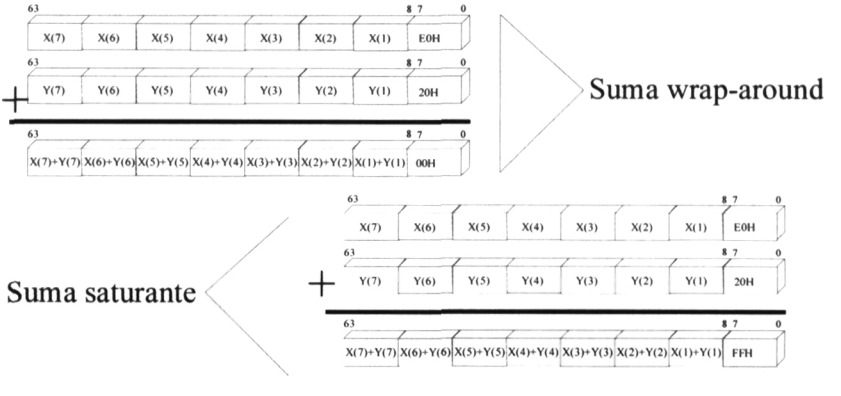


Figura 2.14: Comparativa entre suma wrap-around y saturante

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

bits proveerá precisión adecuada. Para precisión de 32 bits, la segunda mitad del resultado es generada por `PUNPCKHWD` y `PUNPCKLWD`, las cuales intercambian los dos registros de destino.

El código de la figura 2.15 multiplica las 4 palabras, X[3:0], contenidas en MM1 por las 4 palabras, Y[3:0], contenidas en MM2 y almacena los 4 productos resultantes como 2 dobles palabras en MM1 y 2 dobles palabras en MM2. Este código calcula 4 productos en 6 ciclos de reloj sobre un PentiumMMX, mientras que un Pentium no MMX requiere 10 ciclos para completar una sencilla operación de enteros de 16x16→32.

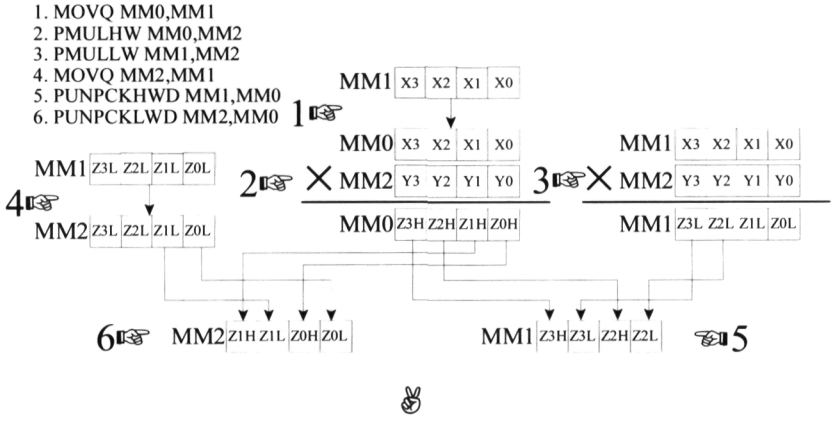


Figura 2.15: Multiplicación SIMD

Las instrucciones `MOVD` y `MOVQ` pueden mover datos a/desde memoria usando los mismos modos de direccionamiento que `MOV`. También mueven datos de un registro `MM` a otro. La instrucción `MOVD` puede incluso intercambiar datos con los registros enteros.

6.7.4. Multiplicación de matrices con MMX

En aplicaciones 3D, se requieren múltiples operaciones geométricas para ubicar y mover objetos dentro del espacio tridimensional, así como para modificar los puntos de vista del espectador[113]. Este tipo de operaciones, denomi-

[113]Los objetos 3D se componen de polígonos, normalmente triángulos, que se definen por sus vértices. Cada vértice tiene una coordenada 3D.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

nadas transformaciones (traslaciones, rotaciones, cambios de escala y perspectiva), se reducen al cálculo de las nuevas coordenadas de cada punto 3D (ver figura 2.16).

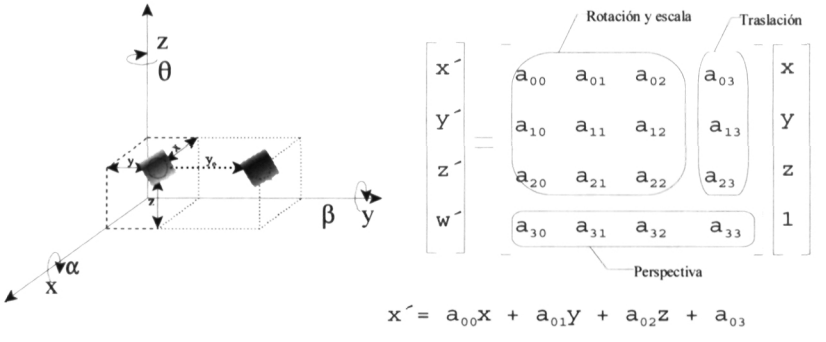


Figura 2.16: Matriz de transformación 3D

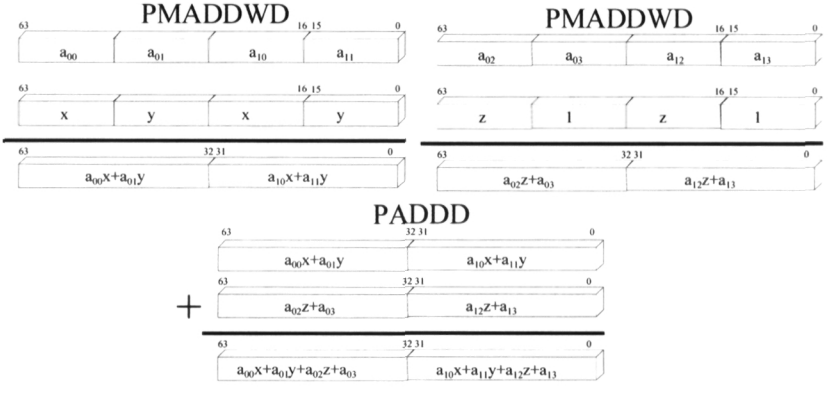


Figura 2.17: Multiplicación de matrices

En este cálculo se usa una matriz de 4x4, de manera que la nueva coordenada requiere la resolución de 4 sumas ponderadas. MMX ofrece una instrucción específica para este propósito: `PMADDWD`. Esta instrucción multiplica dos pares de palabras de 16 bits y, a continuación, suma cada par produciendo dos re-

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

sultados de 32 bits. En la figura 2.17 se puede ver cómo se resuelve el cálculo de las nuevas coordenadas de la matriz anterior. La operación de multiplicación-suma es el corazón de muchos algoritmos de audio y video, tal como la transformada rápida de Fourier.

6.7.5. Incrustación usando MMX

Una técnica común de video conocida como incrustación (chroma keying) permite que un objeto (por ejemplo, el hombre del tiempo) situado enfrente de una pantalla azul[114] sea superpuesto sobre otra imagen (el mapa del tiempo). MMX puede acelerar esta técnica utilizando instrucciones SIMD lógicas y de comparación. En la figura 2.18 se puede ver un ejemplo ilustrativo.

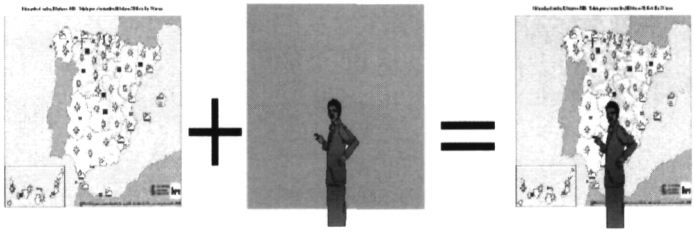


Figura 2.18: Ejemplo de técnica de incrustación

Para conseguir este efecto lo primero que hay que obtener es la máscara del objeto a incrustar. Esto se consigue comparando una imagen del objeto sobre fondo azul con una imagen completamente azul (ver figura 2.19). Se puede ver como la instrucción `PCMPEQW` compara pixels de 16 bits de cuatro en cuatro. El resultado es `0000H` (color negro) si la comparación es falsa y `FFFFH` (color blanco) si la comparación es verdadera.

Una vez disponemos de la máscara se realizan las siguientes operaciones (ver figura 2.20):

1. Se multiplica cada pixel de la máscara por la imagen de fondo. Para ello se usa la instrucción `AND` que realiza la operación sobre 4 píxeles a la vez. Se obtiene así una imagen del fondo con la silueta del objeto superpuesta.

[114] Puede ser otro color. La clave es que el color no se encuentre en el objeto que se quiere incrustar.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

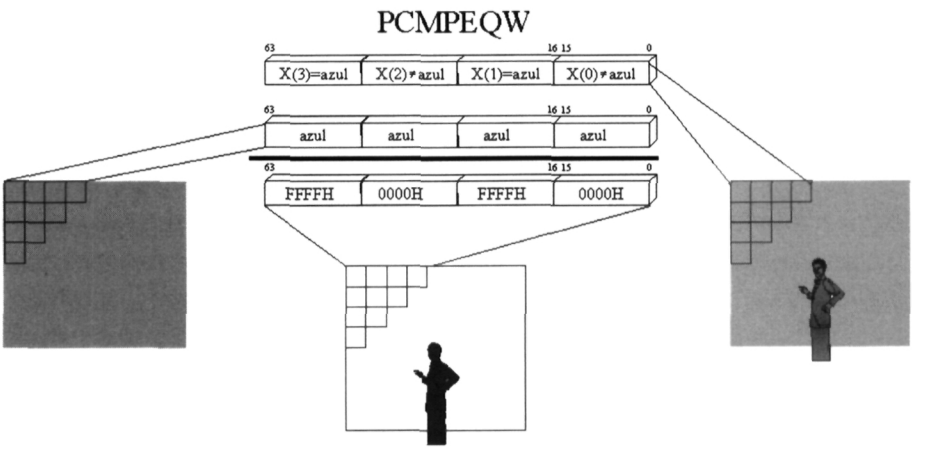


Figura 2.19: Ejemplo de incrustación: Cálculo de la máscara

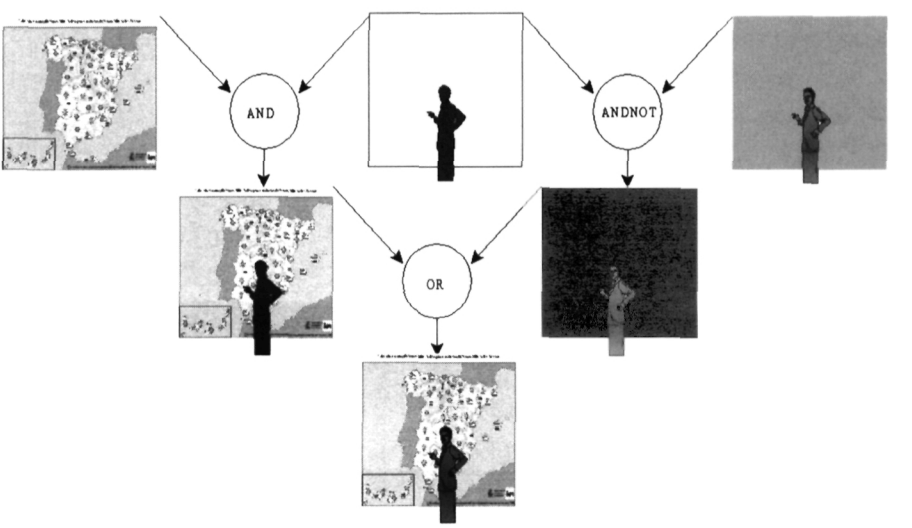


Figura 2.20: Ejemplo de incrustación: Composición

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2. Se multiplica cada pixel de la máscara invertida por la imagen del objeto sobre fondo azul. Se usa la instrucción `ANDNOT` que primero invierte los valores de la máscara y luego lleva a cabo la multiplicación (de nuevo sobre 4 píxeles a la vez). Se obtiene así una imagen del objeto sobre fondo negro.

3. A continuación, se suman las dos imágenes obtenidas anteriormente y se obtiene el efecto deseado (recuerde que el color negro es `0000H`).

En la figura 2.21, se ilustran las operaciones lógicas llevadas a cabo. El conjunto de operaciones lógicas (`AND, AND NOT` y `OR`) operan bit a bit, de modo que no requieren sufijo del tipo de datos.

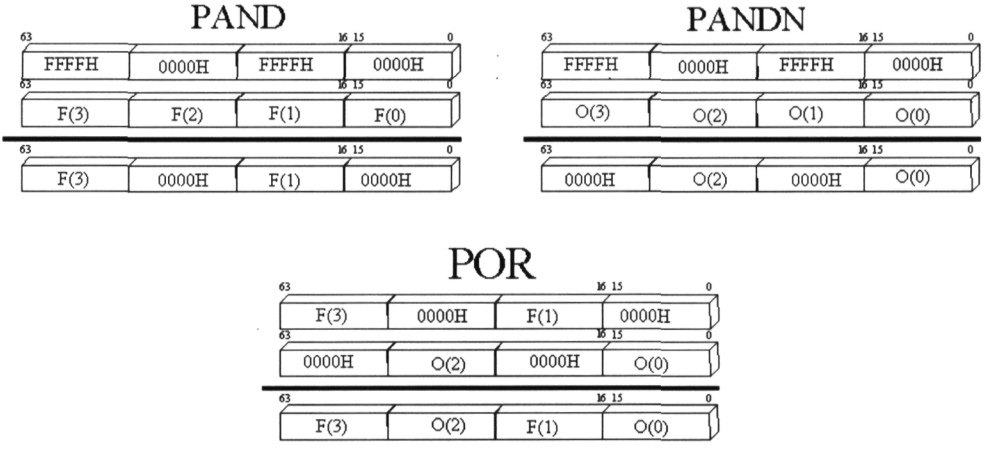


Figura 2.21: Ejemplo de incrustación: Instrucciones MMX

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

7. Interrupciones y excepciones

El microprocesador tiene dos mecanismos para interrumpir la ejecución del programa:

- Excepciones: Son eventos síncronos que son respuestas del procesador a ciertas condiciones detectadas durante la ejecución de una instrucción. Pueden ser excepciones detectadas por el microprocesador (fallos, trampas y abortos) o programadas (interrupciones software).
- Interrupciones: Son eventos asíncronos disparados por dispositivos y periféricos que necesitan atención. Pueden ser enmascarables y no enmascarables.

Normalmente, los programadores de aplicaciones no manejan estas interrupciones y excepciones[115]. El microprocesador asocia un número de identificación (número de vector) a cada tipo diferente de interrupción o excepción (ver tabla 2.4).

Numero de vector	Descripción
0	División por cero
1	Llamada al depurador
2	Interrupción NMI
3	Punto de ruptura (INT 3)
4	Detección de desbordamiento (INTO)
5	Rango BOUND excedido
6	Código de operación inválido
7	Dispositivo no disponible
8	Doble fallo
9	Reservado
10	Segmento de estado de tarea inválido
11	Segmento no presente
12	Excepción de la pila
13	Protección general
14	Fallo de página
15	Reservado
16	Error de punto flotante
17	Chequeo de alineamiento
18	Excepción de chequeo de máquina
19	Excepción punto flotante SIMD
20-31	Reservado
32-255	Interrupciones enmascarables

Tabla 2.4: Excepciones e interrupciones de IA-32

Estos números se utilizan como índices para buscar en la tabla de vectores (modo real) o descriptores (modo protegido) de interrupción la dirección de comienzo de la subrutina de servicio de interrupción (ver figura 2.22).

[115] Es una labor del S.O. y de los drivers de dispositivo.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

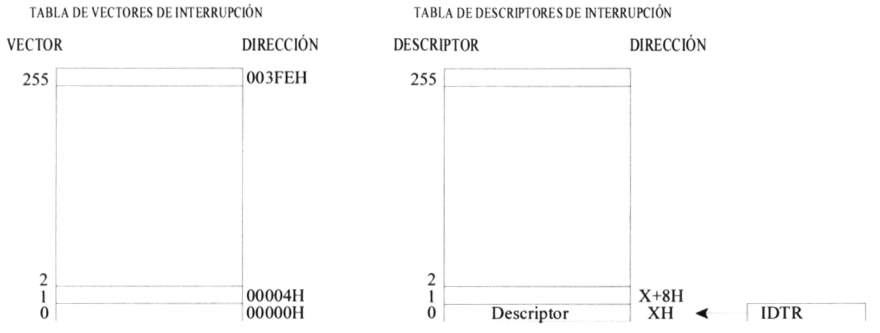


Figura 2.22: Tabla de vectores de interrupción (modo real) y tabla de descriptores de interrupción(modo protegido)

8. Entrada/Salida

La arquitectura IA-32 permite crear para E/S espacios de direcciones único o separado.

En el caso de un espacio de direcciones separado, el espacio de E/S es de 64KB y se puede direccionar[116] como puertos de 8 bits numerados desde el 0 hasta el FFFFH. También se puede acceder a puertos de 16 y 32 bits; en este caso las direcciones deben ser múltiplo de 2 (para 16 bits) o de 4 (para 32 bits) si se quiere que el acceso se haga en un único ciclo de bus. A diferencia de los accesos a memoria, el microprocesador no maneja errores de paridad de datos. Una ventaja de usar el espacio de direcciones separado es que el microprocesador garantiza que las escrituras a puertos de E/S se completan antes de que se ejecute la siguiente instrucción.

En el caso de un espacio de direcciones único, hay que prevenir que estas direcciones se guarden en las memorias cache. A partir del Pentium Pro, esto puede ser controlado programando el rango de direcciones no cacheable. Para microprocesadores anteriores se hace mediante la señal hardware `KEN#` (ver apartado 10.1). Todos los microprocesadores IA-32 con caches integradas también proveen un flag en las entradas de páginas para deshabilitar el cacheado de páginas.

[116]Las direcciones F8H y FFH están reservadas.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

9. Entorno de ejecución del sistema

El microprocesador tiene un entorno de ejecución del sistema exclusivo del modo protegido desde el cual tiene acceso ilimitado a todas sus funciones y características arquitecturales: Gestión de memoria, protección, multitarea, excepciones e interrupciones, E/S, inicialización y modo de conmutación, gestión de la unidad de punto flotante, depurado, gestión de caché y multiproceso. En este capítulo, sólo se abordará la gestión de memoria y la multitarea.

9.1. Registros del sistema

Los registros del modo protegido o del sistema (ver figura 2.23) se pueden clasificar en:

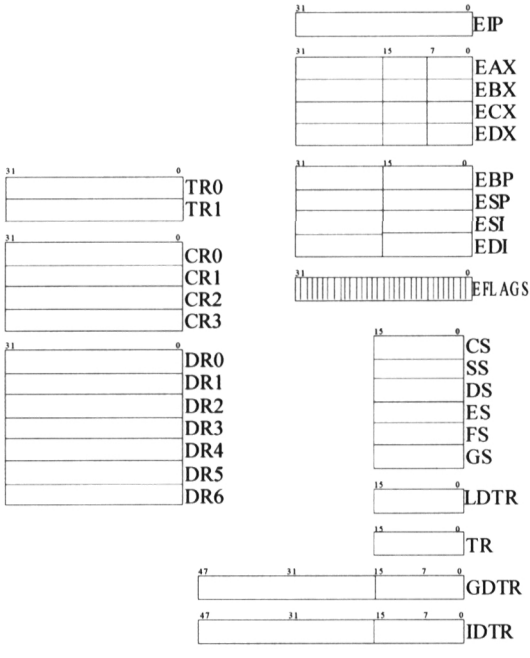


Figura 2.23: Registros del sistema

- Puntero de instrucciones EIP y registros generales. Tienen la misma funcionalidad que en el modo de ejecución básico.
- Registros de segmento. En modo protegido contienen un selector de segmento.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Registro de estado. Son algunos flags de EFLAGs, denominados flags del sistema. Controlan el espacio de E/S, las interrupciones enmascarables, el depurado, la conmutación de tareas y el modo virtual 8086.

- Registros de gestión de memoria. Estos registros especifican las localizaciones de varias estructuras de datos necesarias para implementar la gestión de memoria. El entorno de ejecución básico no tiene acceso a estos registros[117]:

  - GDTR(Global Descriptor Table Register): Registro de la tabla de descriptores globales. Define la dirección de comienzo de la tabla de descriptores globales. Es un registro de 48 bits con dos campos que indican el tamaño y la ubicación en memoria de la tabla[118].
  - IDTR(Interrupt Descriptor Table Register): Registro de la tabla de descriptores de interrupción. Similar al anterior[119].
  - LDTR(Local Descriptor Table Register): Registro de la tabla de descriptores locales. Es un registro de 16 bits que contiene un selector que identifica la ubicación de la tabla en memoria[120].
  - TR(Task Register): Registro de tarea. Es un registro de 16 bits que contiene un selector que identifica la tarea en curso[121].

- CR[0:3](Control Register): Registros de control. En CR2 se guarda la dirección lineal si hay un fallo de página. CR3 es el registro de base del directorio de páginas: PDBR (Page Directory Base Register). Usa sólo 20 bits.

- DR[0:6](Debugger Register): Registros de depurado. Proporcionan al microprocesador capacidades de depurado avanzadas, incluyendo puntos de ruptura.

- TR[0:1](Test Register): Registros de test. Permiten testear el microprocesador.

[117] En apartados posteriores se describe con más detalle el funcionamiento de estos registros.

[118] El campo de tamaño de segmento son los dos bytes menos significativos; por tanto puede llegar a tener un tamaño de 2^{16}. Los 4 bytes más significativos guardan la dirección física de comienzo de la tabla.

[119] En el IDTR el campo LIMITE está sobredimensionado ya que sólo hay 256 interrupciones.

[120] Este selector apunta a la GDT.

[121] Este selector apunta a la GDT.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

9.2. Gestión de memoria

9.2.1. Segmentación

Los elementos básicos para implementar segmentación son los selectores de segmento, las tablas de descriptores de segmento y los descriptores de segmento.

Los selectores de segmento (figura 2.24) se almacenan en los registros de segmento y permiten seleccionar descriptores de segmento.

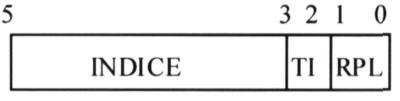


Figura 2.24: Selector de segmento

El selector tiene tres campos :

- INDICE (bits[15:3]). INDICE*8 es la dirección relativa del descriptor de segmento dentro de la tabla de descriptores correspondiente.
- TI (bit 2). Determina si el descriptor está en la GDT (T=0) o en la LDT(T=1).
- RPL (bits [1:0]. Indican el nivel de privilegio del selector.

Hay tres tipos tablas de descriptores: la tabla de descriptores globales o GDT (Global Descriptor Table), las tablas de descriptores locales o LDTs (Local Descriptor Tables) y la tabla de descriptores de interrupciones o IDT (Interrupt Descriptor Table).

La GDT es una tabla de propósito general donde se ubican, entre otros, los programas de S.O. Las LDTs son específicas de multitarea existiendo una LDT por cada tarea del sistema. La IDT contiene descriptores de puertas de trap, de interrupción y de tarea. Los descriptores de puertas de trap e interrupciones permiten pasar el control a rutinas de servicio que están localizadas dentro de la tarea en curso. Los descriptores de puertas de tarea permiten pasar el control del programa a otra tarea.

En la figura 2.25 se describe someramente el proceso que se lleva a cabo desde que se carga el selector en el registro de segmento hasta que que el descriptor correspondiente es volcado en el microprocesador[122].

[122] El microprocesador guarda la información de los descriptores de segmento que está usando en unos registros invisibles para el usuario; la finalidad de esto es evitar la consulta a las tablas de descriptores cada vez que se hace un acceso a memoria.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

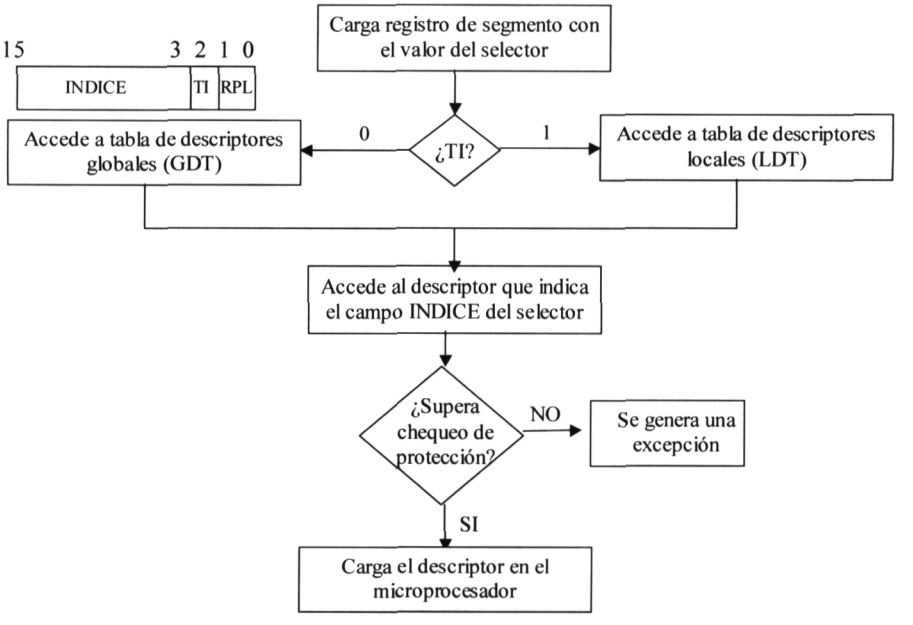


Figura 2.25: Proceso de carga de descriptor de segmento

En la figura 2.26 se ilustra el procedimiento básico del microprocesador para traducir una dirección lógica (definida por un selector de segmento más un desplazamiento) a una dirección física. Con el selector se selecciona un descriptor de segmento que contiene la dirección de comienzo del segmento al que queremos acceder, su tamaño y una serie de atributos que el sistema de gestión de memoria chequea para verificar si el acceso está permitido. En caso afirmativo, la dirección física se forma sumándole a la dirección de comienzo del segmento[123] el desplazamiento.

[123] Observe que un programador de aplicaciones no conoce el contenido del descriptor y por tanto no sabe cuál es la dirección física a la que accede.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

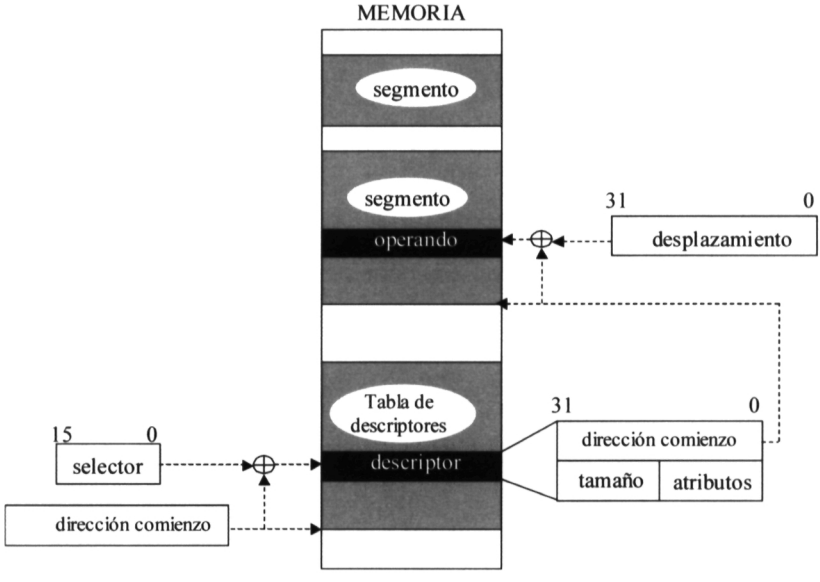


Figura 2.26: Traducción de dirección lógica a dirección física en modo protegido

Un segmento se describe mediante una estructura especial denominada descriptor de segmento (figura 2.27).

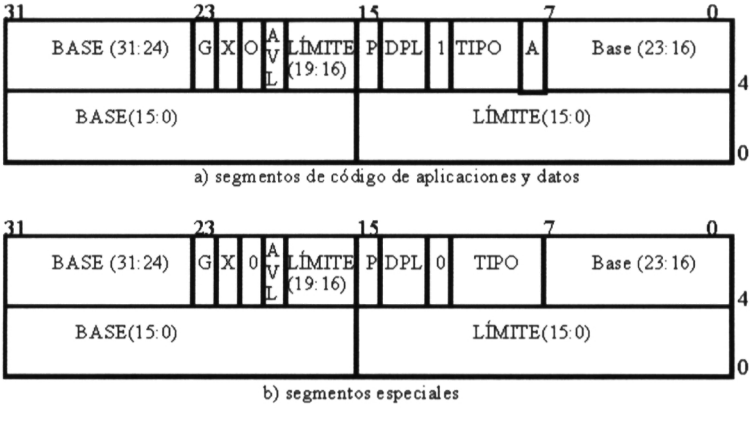


Figura 2.27: Descriptor de segmento

Los descriptores aportan los datos necesarios para convertir la dirección lógica en dirección lineal y son creados por el S.O.. El descriptor ocupa 8

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

bytes siendo sus campos básicos los siguientes:

- BASE: Define la dirección de comienzo del segmento.
- LIMITE: Define el tamaño del segmento. Se mide en unidades de bytes (bit G=0) o en unidades de 4Kbytes (bit G=1).
- TIPO: Distingue entre varios tipos de descriptores: descriptores de segmentos de datos, descriptores de segmentos de código de aplicaciones y descriptores de segmentos especiales del sistema.
- DPL: Define el nivel de privilegio del descriptor. Este valor se compara con el campo RPL del selector para determinar los derechos de acceso al segmento.
- P: Si P está desactivado (0) no se puede traducir el segmento y el microprocesador genera una excepción[124].
- A: Se activa cuando se accede al segmento que describe el descriptor. Es útil para implementar mecanismos de memoria virtual.

En la figura 2.28, se puede ver un ejemplo de traducción de dirección lógica usando la tabla de descriptores globales. El GDTR guarda la dirección de comienzo de la GDT y su tamaño. Por otro lado, la dirección lógica a la que se quiere acceder está formada por un selector (almacenado en alguno de los registros de segmento) y un desplazamiento (almacenado en alguno de los registros generales). En este caso, el selector tiene su campo TI a 0 indicando que el descriptor que tenemos que buscar está en la GDT. El campo INDICE del selector apuntará al descriptor en cuestión. Este descriptor tiene cierto nivel de privilegio en su campo de atributos, nivel que comparará con el nivel de privilegio del selector (sito en el campo RPL). Si los criterios de acceso se cumplen, la dirección de comienzo de segmento que contiene el descriptor sumada al desplazamiento será la dirección física que andamos buscando.

[124] Señala que el segmento no está presente en memoria y que probablemente habrá que ir a buscarlo al disco duro.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

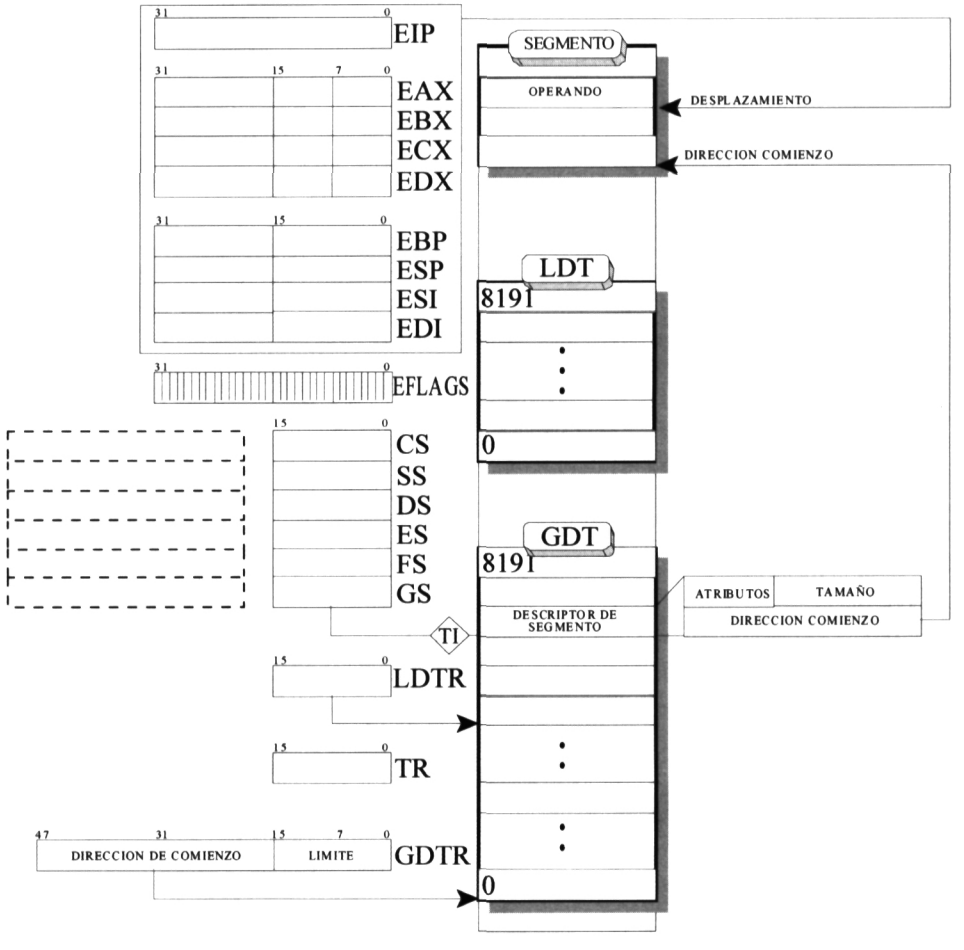


Figura 2.28: Ejemplo de traducción de segmento de la GDT

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Si el descriptor hubiera estado en una tabla de descriptores locales (campo TI del selector a 1), el procedimiento habría sido similar una vez se hubiera obtenido la dirección de comienzo de la LDT. El LDTR es un registro de 16 bits que contiene un selector que apunta a un descriptor de la GDT. Este descriptor contiene la dirección de comienzo de la LDT (ver figura 2.29).

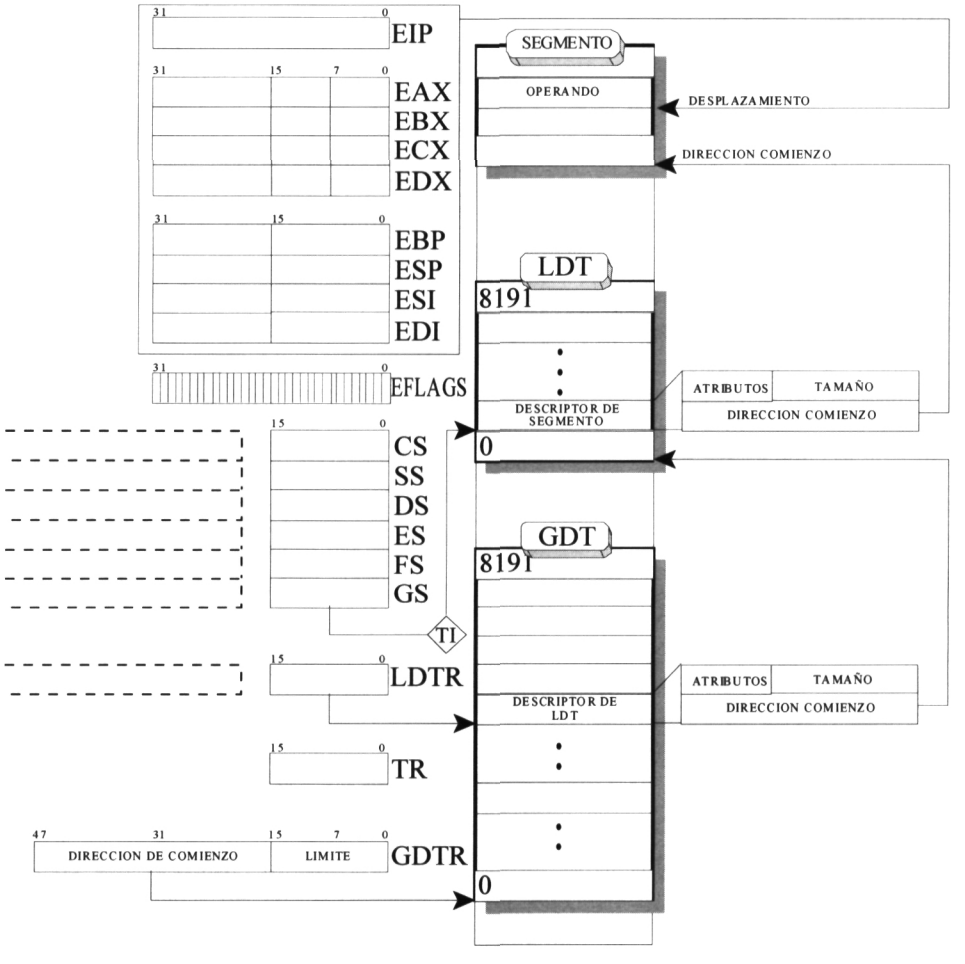


Figura 2.29: Ejemplo de traducción de segmento de la LDT

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Para permitir que un usuario acceda a los servicios del S.O. se definen unos puntos de entrada llamados puertas (figura 2.30). Una puerta contiene la dirección lógica de un punto de entrada y un nivel de privilegio. Un procedimiento que llama a una puerta debe tener el mismo privilegio que dicha puerta.

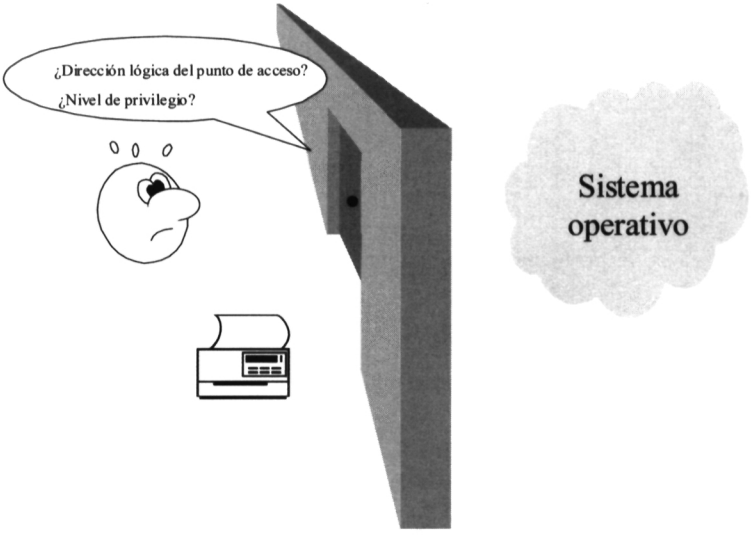


Figura 2.30: Puertas

En la figura 2.31 se muestra un ejemplo de acceso a través de una puerta. Cuando se hace una llamada a una subrutina, la dirección lógica de la instrucción a la que se quiere acceder está formada por un selector (almacenado en el registro de segmento de código) y un desplazamiento (desconocido por el usuario). Dependiendo del valor del campo TI del selector, el campo INDICE del selector apuntará a un descriptor de la GDT (TI=0) o de la LDT (TI=1). Este descriptor es especial y se denomina puerta de llamada.

La puerta de llamada contiene tres campos básicos: a) un selector que apunta al descriptor del segmento donde está ubicada la subrutina a la que queremos acceder, b) un desplazamiento que se aplicará sobre la dirección de comienzo del segmento de la subrutina y c) unos atributos que indicarán su nivel de privilegio, nivel que se comparará con el nivel de privilegio del selector.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

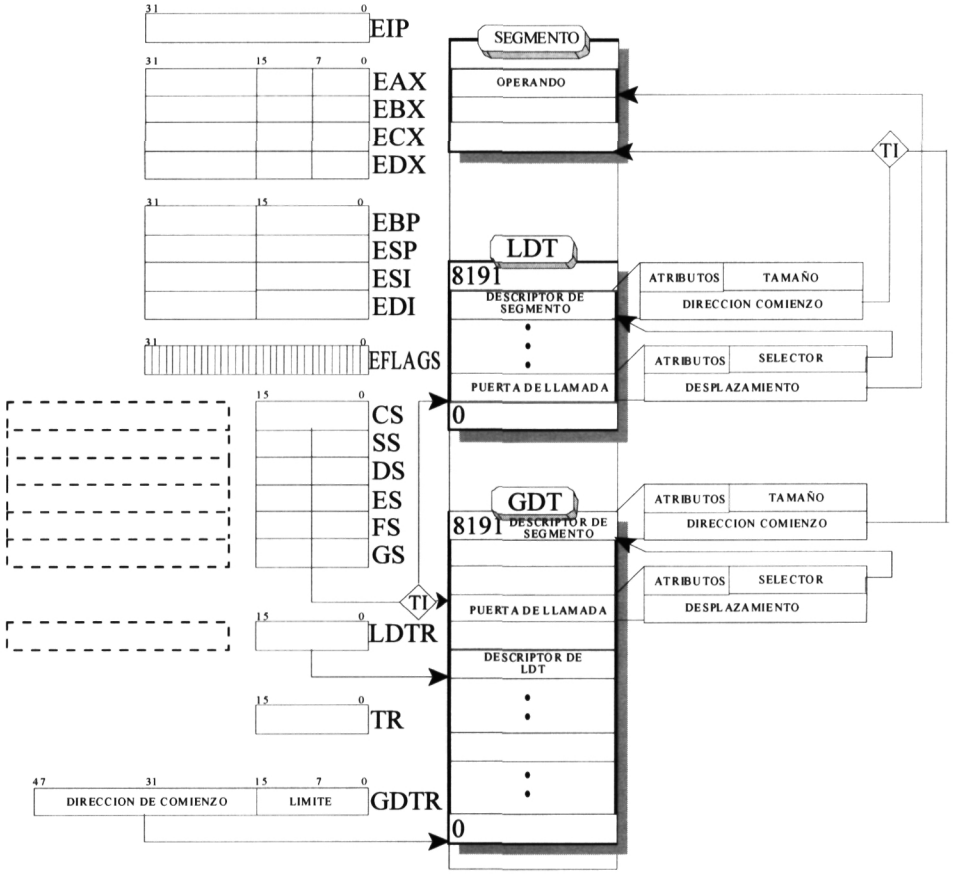


Figura 2.31: Ejemplo de procedimiento de puerta

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Cuando se genera una excepción/interrupción, el microprocesador accede a un descriptor de puerta de trap/interrupción en la IDT. Este descriptor y el procedimiento a seguir es similar al de la puerta de llamada (ver figura 2.32).

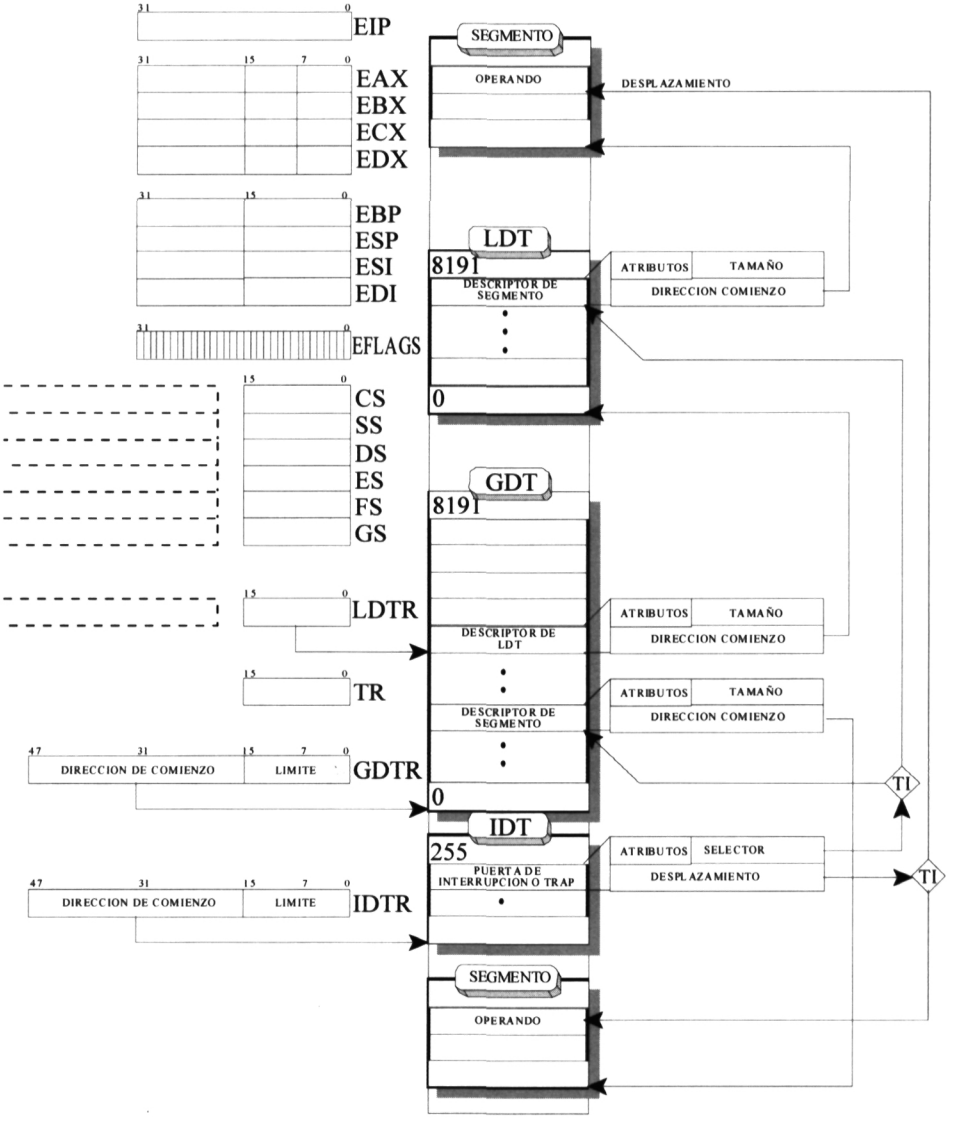


Figura 2.32: Ejemplo de procedimiento de interrupción

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

9.2.2. Paginación

La paginación permite la implementación de memoria virtual. La traducción de dirección lineal a dirección física requiere la consulta de las tablas de entradas de página (análogas a las tablas de descriptores en segmentación).

En la figura 2.33 se puede ver un ejemplo de traducción de dirección lineal a dirección física.

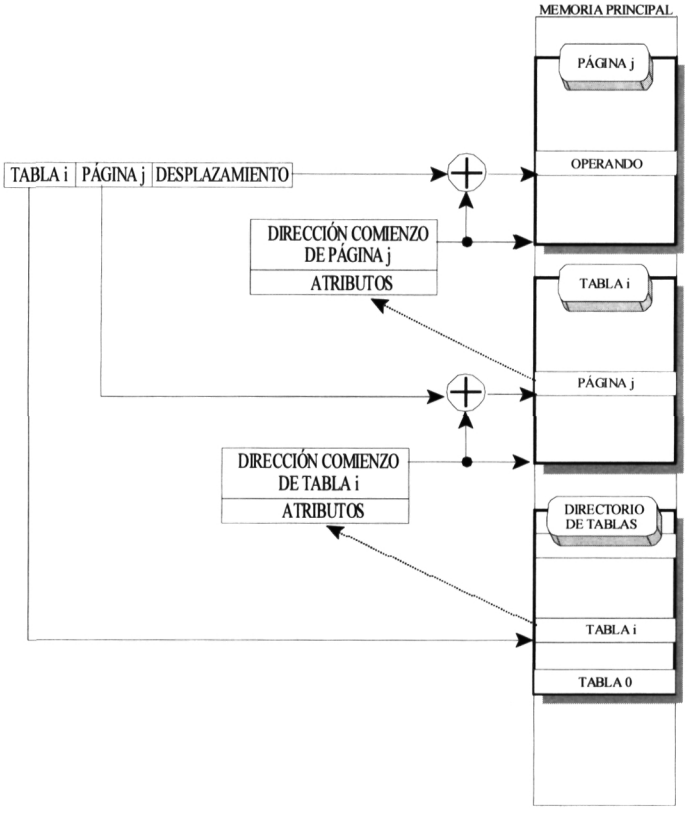


Figura 2.33: Traducción de página

La dirección lineal se descompone en tres campos: Tabla de página, página y desplazamiento. En memoria se ubica el directorio de tablas de páginas, las tablas de páginas y las propias páginas. El tamaño del directorio, de las tablas y de las páginas es el mismo (4KB). El directorio contiene 1024 entradas de página. Cada entrada de página del directorio apunta al comienzo de alguna tabla de páginas. A su vez, cada tabla de páginas tiene 1024 entradas de

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

páginas. Finalmente, cada entrada de página de cada tabla de página apunta a una página de 4KB. La dirección física se obtiene sumando la dirección de comienzo de la página y el desplazamiento. Las últimas direcciones usadas se guardan en una caché denominada caché de traducción de páginas (TLB, Translation Lookaside Buffer). Se puede observar que 1024 entradas/directorio $\times$ 1024 entradas/tabla $\times$ 4KB/página = 4GB.

La estructura de las entradas de las tablas de página se muestra en la figura 2.34.

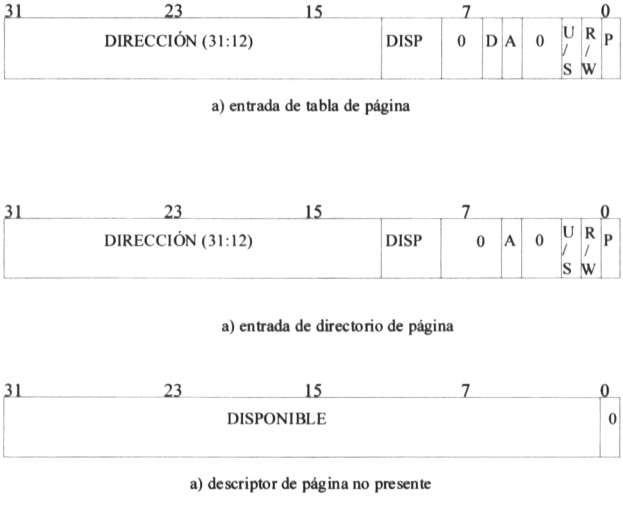


Figura 2.34: Entrada de página

Se distinguen los siguientes campos:

- DIRECCIÓN: DIRECCIÓN$\times 2^{12}$ es la dirección de comienzo de una tabla de páginas (entrada de directorio) o de una página (entrada de tabla).
- P: Si P es 0 no se efectúa la traducción y el microprocesador genera una excepción. La instrucción que provoca la excepción puede reejecutarse[125].
- R/W: bit de permiso de escritura.

[125]Indica que la página no está presente en memoria y que tendrá que activarse el mecanismo de memoria virtual.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- U/S: Selecciona dos niveles de privilegio: Usuario o Supervisor.
- A: Se activa en las dos tablas antes de producirse un acceso a una página[126].
- D: Se activa, en la tabla de páginas, antes de producirse una escritura en una página[127].

9.2.3. Protección

La protección (figura 2.35) es necesaria para evitar que un usuario corrompa información del S.O. o de otro usuario, lea información confidencial del S.O. o de otro usuario, modifique registros del microprocesador o un mal diseño perjudique al sistema[128].

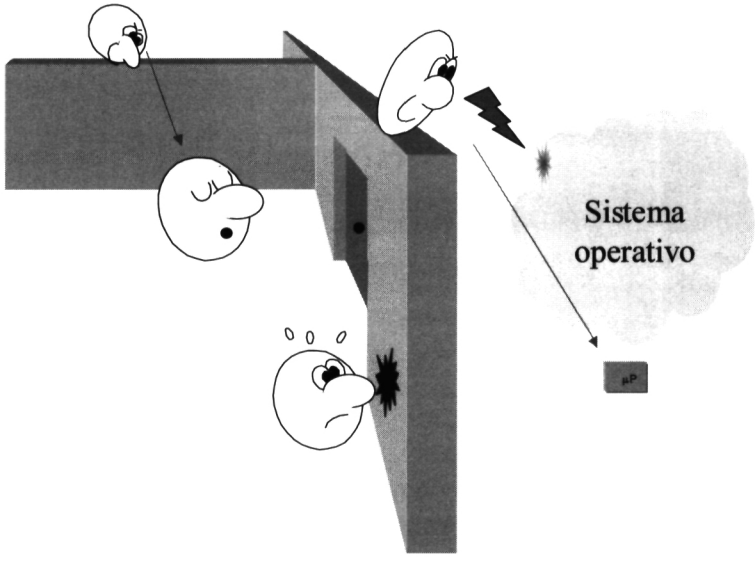


Figura 2.35: Protección

Para ello el microprocesador analiza cada instrucción de acuerdo a ciertos criterios de protección. Los mecanismos de protección se reparten entre los niveles de privilegio y la gestión de memoria.

[126] Se usa para implementar mecanismos de memoria virtual de sustitución de páginas.

[127] Se usa para implementar mecanismos de memoria virtual de mantenimiento de coherencia entre memorias.

[128] Un caso típico es cuando una pila de un programa de usuario crece indefinidamente y acaba borrando programas de otros usuarios o del S.O.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Los niveles de privilegio determinan qué instrucciones se pueden ejecutar y a qué subconjunto de segmentos y/o páginas de su espacio de direcciones se puede referir. El microprocesador comprueba la consistencia entre el nivel de privilegio del selector (RPL) y el nivel de privilegio del segmento (DPL) o página (U/S) objetivo de la instrucción. Cualquier intento por parte de la tarea de usar un segmento o página con un nivel de privilegio mayor hará que el procesador genere una excepción de protección general.

Hay cuatro niveles de privilegio (o prioridad) que varían de 0 a 3 donde el nivel 0 es el de mayor prioridad y el nivel 3 el de menor prioridad (figura 2.36).

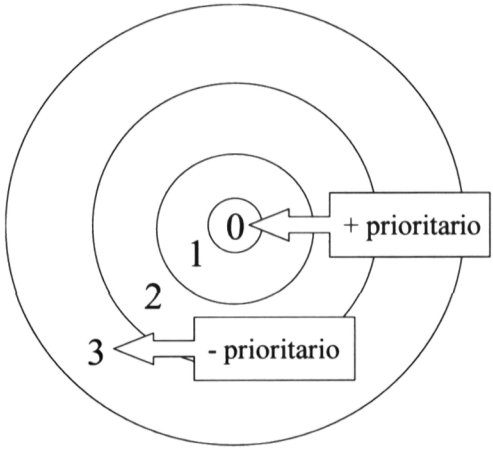


Figura 2.36: Niveles de privilegio

La gestión de memoria atrapa la mayoría de los errores de programación ya que separa los espacios de direcciones de cada tarea[129], tipifica cada segmento (código, datos, etc..), define los derechos de acceso de cada segmento o página (solo lectura, escritura, etc..) y chequea el tamaño de cada segmento.

Las comprobaciones de protección son realizadas automáticamente por el microprocesador cuando el selector de un descriptor se carga dentro de un registro de segmento y con cada acceso al segmento.

En la figura 2.37 se pueden ver varios ejemplos de sistemas de protección.

El caso *a)* es un sistema sin protección (por ejemplo, un sistema de propósito específico que siempre ejecuta el mismo código) donde todos los programas de aplicaciones y S.O tienen el máximo nivel de privilegio.

[129]Gracias al mecanismo de tablas de descriptores locales.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

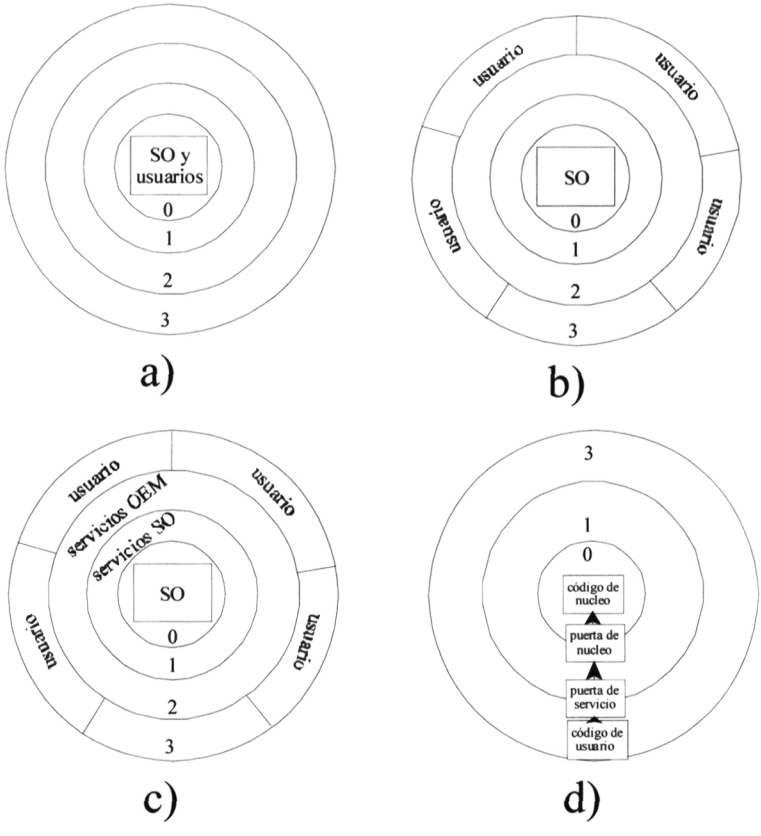


Figura 2.37: Ejemplos de sistemas de protección

El caso *b)* es un sistema multiusuario que establece dos niveles de privilegio: usuario y supervisor. El supervisor tiene máximo nivel de privilegio y los usuarios mínimo. Los usuarios también están protegidos unos de otros gracias a las tablas de descriptores locales[130].

El caso *c)* implementa todos los niveles de privilegio que proporciona el microprocesador: Nivel 0 para el núcleo del S.O., nivel 1 para rutinas del S.O que requieren actualizaciones, nivel 2 para librerías ajenas al S.O que solicitan sus servicios y nivel 3 para los programas de aplicaciones de usuario.

Finalmente, el ejemplo *d)* ilustra que el acceso a un programa de mayor nivel de privilegio debe hacerse a través de una puerta.

[130] Cada usuario se puede ver como una tarea con su particular tabla de descriptores locales.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

9.2.4. Multitarea

Para poder implementar multitarea se requiere gestión de memoria[131], protección de programas y datos y conmutación de tareas.

El microprocesador proporciona mecanismos para facilitar y acelerar la conmutación de tareas. Para ello, hace uso de un segmento especial denominado segmento de estado de la tarea (TSS, Task State Segment)(figura 2.38).

El TSS tiene dos campos:

- Un conjunto dinámico que el microprocesador actualiza en cada conmutación:
  - Registros generales, de segmento, de estado y puntero de instrucciones.
  - Unión al anterior TSS (Back link). Este campo junto con el flag NT permiten al microprocesador volver automáticamente a una tarea que a su vez llamó a otra, o que fue interrumpida por otra tarea.
- Un conjunto estático que el microprocesador lee pero no cambia:
  - El selector LDT de la tarea.
  - El registro PDBR (Page Directory Base Register) en CR3.
  - Punteros a la pilas de niveles 0-2. Usados cuando se cambia de privilegio.
  - El T-bit. Genera una excepción al depurador cuando se conmuta de tarea.
  - La base del mapa de permiso E/S. Contiene el desplazamiento del principio del mapa de permiso E/S. Cada bit de permiso del mapa se corresponde con una dirección de E/S.

La conmutación de tareas se puede hacer directa o indirectamente a través de una puerta. La puerta define el punto de entrada de un procedimiento y especifica el nivel de prioridad de ese punto de entrada (ver figura 2.38).

[131] Como la memoria principal tiene un tamaño limitado, hay dos restricciones a la hora de asignar memoria: el número máximo de programas a almacenar y el tamaño máximo de cada programa. Ambos se resuelven usando las técnicas de segmentación y paginación.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

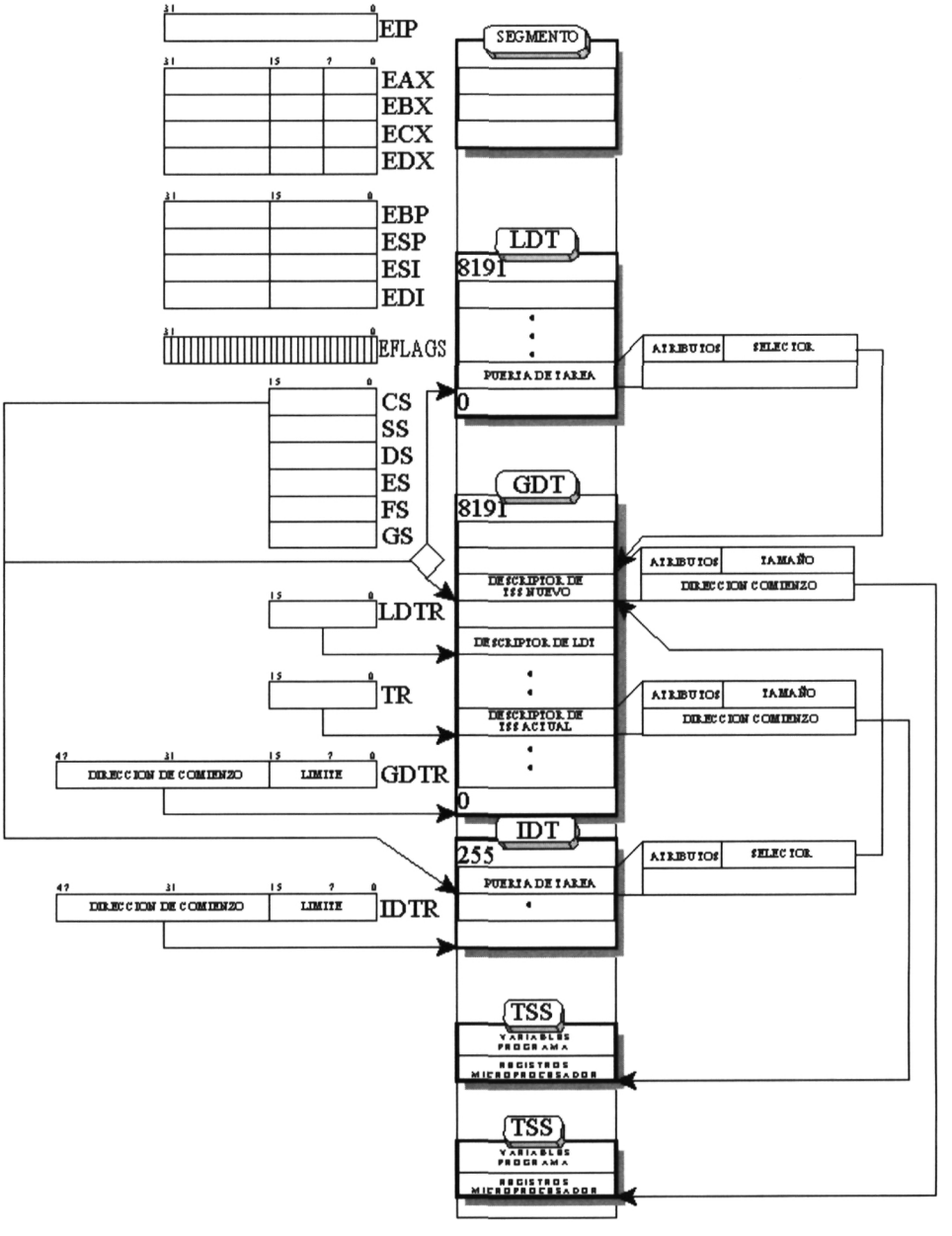


Figura 2.38: Ejemplo de procedimiento de multitarea

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

9.3. Ejemplos

9.3.1. Cálculo de dirección física

En este ejemplo se calcula la dirección física de la instrucción a la que apunta un microprocesador IA-32 en dos casos: Modo real y modo protegido. A continuación se muestran los contenidos de 3 registros y una dirección de memoria implicados en el cálculo:

```
(CS)=5000H (EIP)=A000H (GDTR)=01FE 3000 FFFFH

(01FE8000H) = AC0082AB0D00003FH
```

En modo real, la dirección física se calcula así (figura 2.39):

```
Dirección física =(CS)*10H +(IP) = 5000H*10H+A000H=5A000H
```

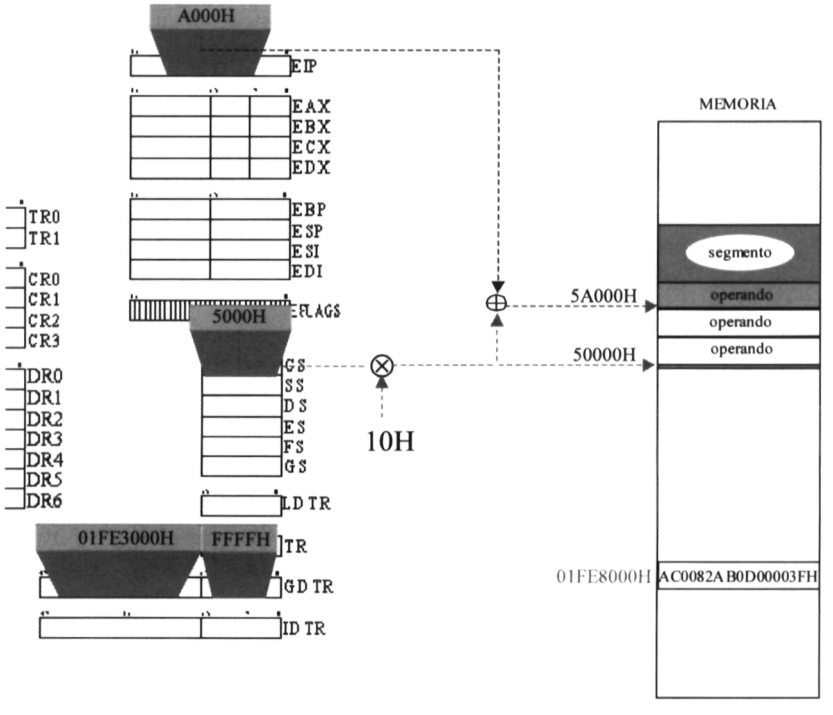


Figura 2.39: Ejemplo de acceso a memoria en modo real

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

En modo protegido, el cálculo requiere varios pasos (figura 2.40). En primer lugar, hay que examinar qué valores tienen los campos ÍNDICE, TI y RPL del selector de segmento.

El campo ÍNDICE vale 0101000000000B. TI vale 0 luego el descriptor está en la tabla de descriptores globales. RPL es 00B, así que el selector tiene máximo nivel de privilegio.

A continuación, se obtiene la dirección de comienzo de la tabla de descriptores globales, que son los 32 bits más significativos del registro GDTR: 01FE3000H

Seguidamente se calcula la dirección física del descriptor de segmento:

```
dirección descriptor = dirección comienzo segmento+INDICE*8 =
                     = 01FE3000H + 5000H = 01FE8000H
```

El contenido de la cuádruple palabra de dirección 01FE8000H (el descriptor) es:

```
(01FE8000H) = AC00 82AB 0D00 003FH
```

y ensamblando los campos correspondientes obtenemos la dirección de comienzo del segmento de código:

```
(01FE8000H) = AC00 82AB 0D00 003FH
       ______/  \___/       \_____
       dirección comienzo segmento = ACAB0D00H
```

Finalmente, la dirección física se calcula sumando a la dirección de comienzo del segmento el desplazamiento que indica el registro EIP:

```
dirección física = dirección comienzo segmento + (EIP)
                 = ACAB0D00H + A000H = ACABAD00H
```

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

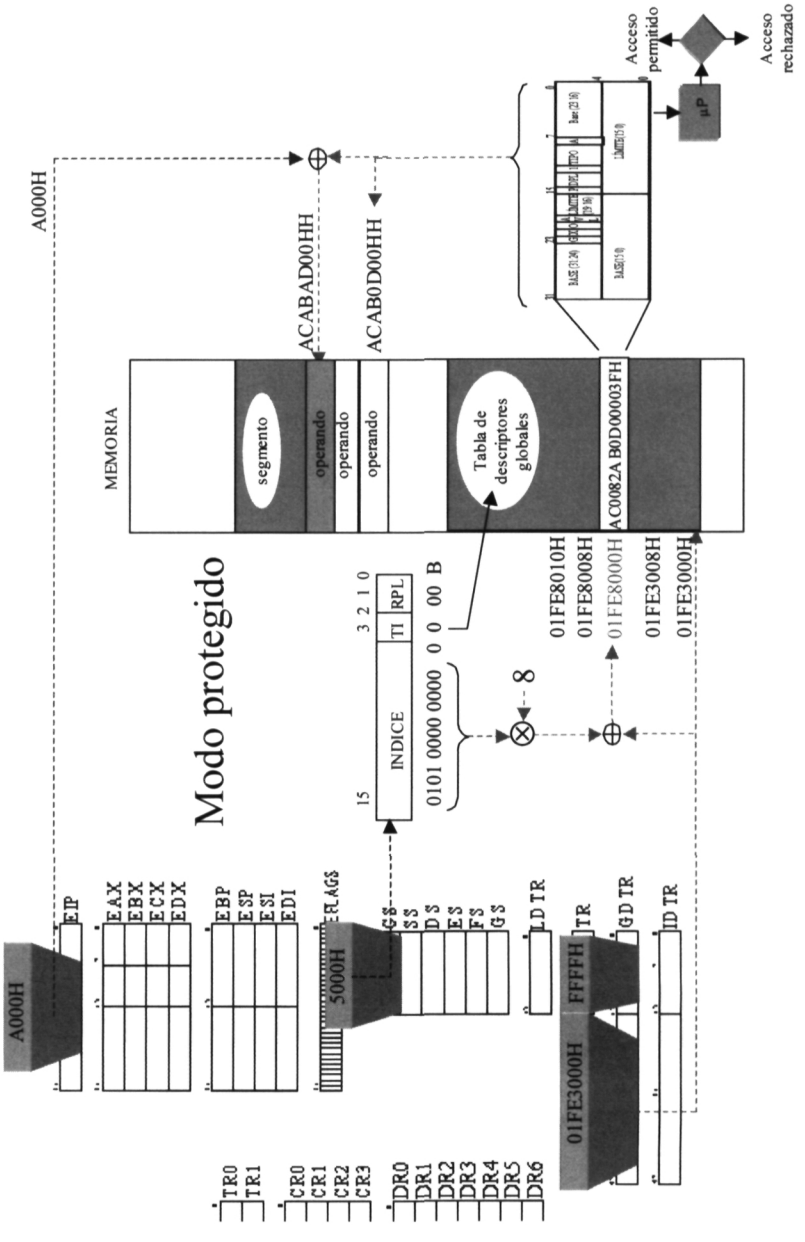


Figura 2.40: Ejemplo de de acceso a memoria en modo protegido

10. Interfaz del Pentium

Las señales del Pentium (ver figura 2.41) se agrupan en 7 interfaces: Memoria y E/S, Multiproceso, Interrupciones, Memoria caché, Punto flotante, SMM (System Management Mode) y Depurado y Testeo. En este capítulo sólo se estudian los tres primeros.

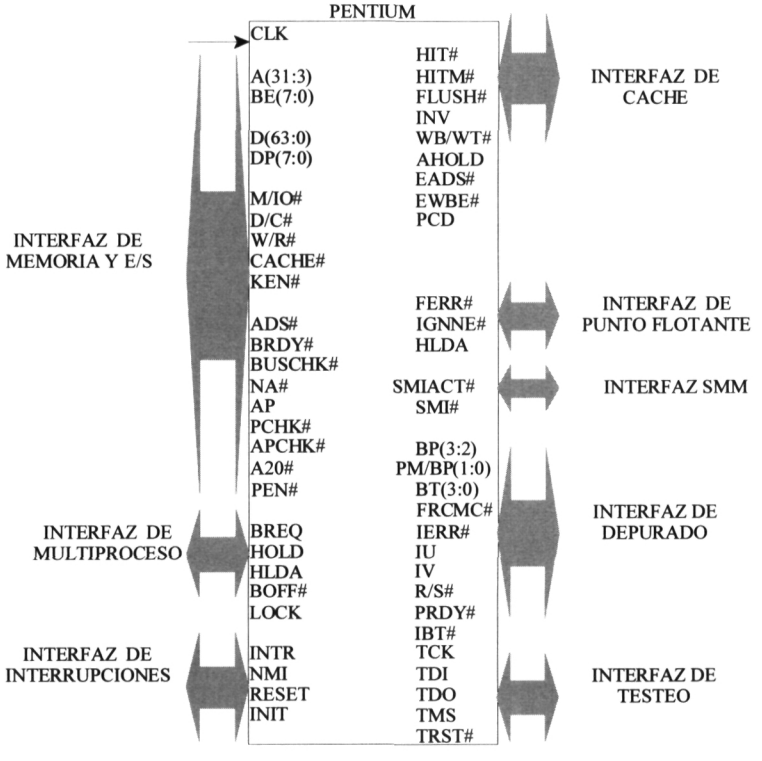


Figura 2.41: Interfaces del Pentium

10.1. Interfaz de memoria y E/S

En el Pentium se distingue entre operaciones de bus, ciclos de bus y ciclos de transferencia de datos (ver figura 2.42).

Las operaciones de bus son secuencias de ciclos de bus que llevan a cabo una función específica: Ciclo de escritura modifica lectura, ciclo de reconocimiento de interrupción, etc. A su vez, cada ciclo de bus es un conjunto de transferencias de datos que empieza cuando se activa la señal `ADS#` y termina cuando

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

se retorna la última activación de **BRDY#**. En la tabla 2.5 se presenta una descripción de todas las señales del interfaz.

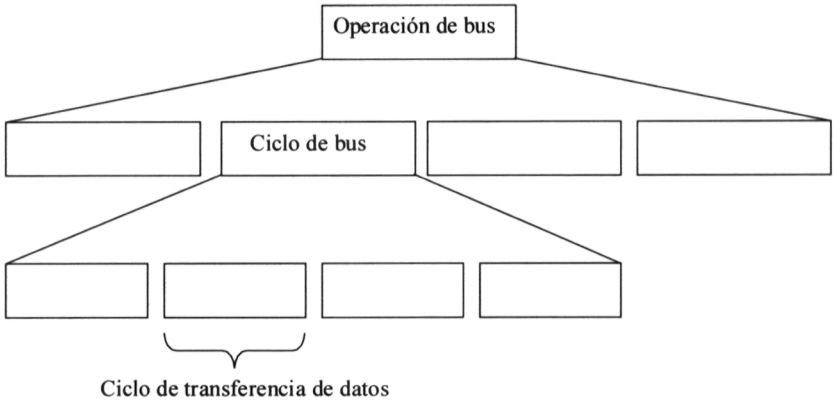


Figura 2.42: Operaciones de bus y ciclos de bus y de transferencia de datos en el Pentium

Nombre	Dirección	Descripción
A[31:3]	bidireccional	Bus de direcciones. Como entrada accede a la memoria caché de primer nivel.
BE[7:0]	salida	Habilitadores de bytes. Son A0, A1 y A2 decodificados.
D[63:0]	bidireccional	Bus de datos.
DP[7:0]	bidireccional	Byte de paridad. Hay un bit por cada byte de datos. Como salida indica paridad impar del dato a escribir. Como entrada, paridad impar del dato a leer. El Pentium lo usa si PEN# está activado. Si hay un error de paridad se genera una excepción y se activa la señal PCHK#.
M/IO#	salida	A nivel alto, indica ciclo de bus de memoria. A nivel bajo, indica ciclo de bus de E/S.
D/C#	salida	A nivel alto, indica ciclo de bus de datos. A nivel bajo, indica ciclo de bus de control.
W/R#	salida	A nivel alto, indica ciclo de escritura. A nivel bajo, indica ciclo de lectura.
CACHE#	salida	Indica "cacheabilidad" interna del ciclo (lectura de código o datos) o writeback (escritura).
KEN#	entrada	Indica al Pentium si el sistema puede soportar relleno de línea de caché en el ciclo en curso.
ADS#	salida	Indica que hay un nuevo ciclo de bus válido.
BRDY#	entrada	Indica al microprocesador que el ciclo de transferencia de datos puede terminar. Mientras no se active el Pentium añadirá estados de espera.
BUSCHK#	entrada	Indica que el ciclo no se completa bien.
NA#	entrada	Habilita los ciclos de transferencia de datos con pipeline. Indica al microprocesador que puede sacar las direcciones del siguiente ciclo de transferencia antes de que termine el que está en curso.
AP	bidireccional	Paridad de direcciones. Indica la paridad par de unos sobre el bus de direcciones y el pin AP.
PCHK#	salida	Indica si la paridad es correcta o no. Se activa dos ciclos de reloj después de BRDY# para lecturas con el resultado del chequeo de paridad.
A20#	entrada	Máscara de la dirección A20. Para compatibilidad con el modo real.
APCHK	salida	Chequeo de paridad del bus de direcciones para los ciclos de inquiry(acceso a memoria caché).
PEN#	entrada	Habilitación de paridad.

Tabla 2.5: Señales de interfaz de memoria y E/S del bus del Pentium

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Un ciclo de bus puede incluir de 1 a 4 transferencias de datos. Un ciclo de ráfaga es un ciclo de bus con 4 transferencias de datos. Finalmente, un ciclo de transferencia de datos es la transferencia de un operando de hasta 8 bytes leído/escrito por el microprocesador con la señal `BRDY#` activa.

En la tabla 2.6 se muestran los diferentes ciclos de bus que ejecuta el microprocesador en función de los valores de las señales de tipo de ciclo de bus: `M/IO#`, `D/C#`, `W/R#`, `CACHE#` y `KEN#`.

M/IO#	D/C#	W/R#	CACHE#	KEN#	Descripción	No. transf.
0	0	0	1	X	Reconocimiento de interrupción (2 ciclos locked)	1
0	0	1	1	X	Ciclo especial. Depende del valor de BE[7:0]#	
0	1	0	1	X	Lectura E/S, 32 bits o menos, no cacheable	1
0	1	1	1	X	Escritura E/S, 32 bits o menos, no cacheable	1
1	0	0	1	X	Lectura código, 64 bits, no cacheable	1
1	0	0	X	1	Lectura código, 64 bits, no cacheable	1
1	0	0	0	0	Lectura código, 256 bits, relleno de línea de caché	4
1	0	1	X	X	Reservado	no aplicable
1	1	0	1	X	Lectura memoria , 64 bits o menos, no cacheable	1
1	1	0	X	1	Lectura memoria, 64 bits o menos, no cacheable	1
1	1	0	0	0	Lectura memoria, relleno de línea de ráfaga de 256 bits	4
1	1	1	1	X	Escritura memoria, 64 bits o menos, no cacheable	1
1	1	1	0	X	Writeback ráfaga de 256 bits	4

Tabla 2.6: Tipos de ciclo de bus del Pentium

Hay ciclos de transferencia de datos de memoria, E/S y caché, ciclos de manejo de interrupciones y ciclos especiales.

Los ciclos de bus especiales son: shut down (error de paridad/excepción), flush (indica al sistema que la caché L1 se invalida), halt (`INT` AND (`NMI` OR `INI` OR `RESET`)), writeback (previo al borrado de la caché L1) y branch trace message (relacionado con el depurador).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

10.1.1. Operaciones de bus

A continuación se describen tres tipos de ciclos de transferencia de datos: transferencia simple, transferencia simple con estados de espera y transferencia de ráfaga.

10.1.1.1. Transferencia simple La figura 2.43 muestra dos ciclos de transferencia simple de datos: lectura y escritura sin pipeline[132].

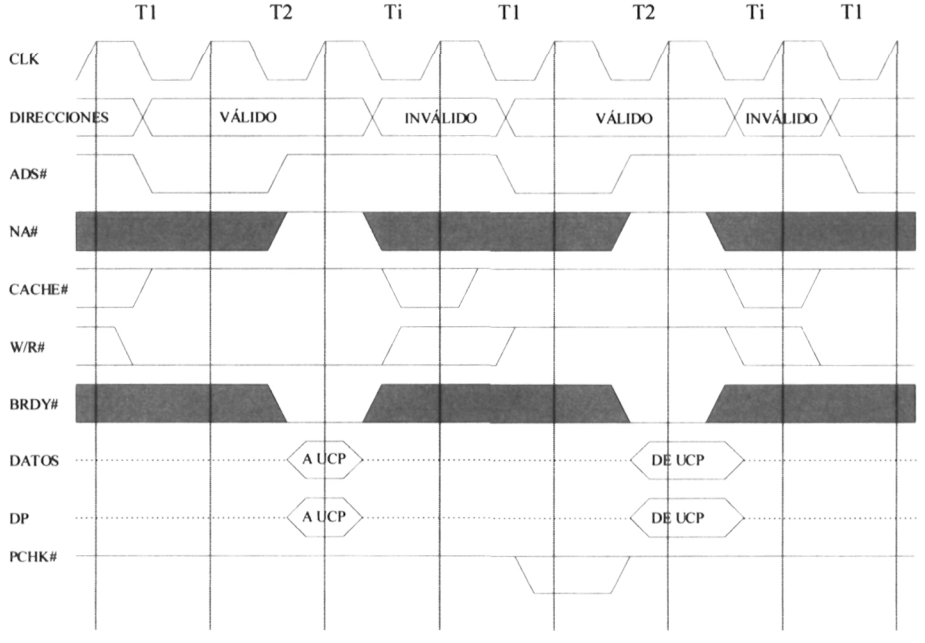


Figura 2.43: Ciclos de transferencia simple de datos del Pentium:lectura y escritura

El proceso que se lleva a cabo es el siguiente:

En el estado T1, tanto para lectura como para escritura, se generan las direcciones `A[31:3]`, los habilitadores de bytes `BE[7:0]#` y el tipo de ciclo de bus determinado por la señales `M/IO#`, `D/C#`, `W/R#`, `CACHE#` y `KEN#`. A continuación, se activa `ADS#`, indicando que todas las señales anteriores están estables.

En el estado T2, en lectura, se testea `BRDY#`. Como está activa, se termina el ciclo, leyendo los datos del bus de datos `D[63:0]` y el byte de paridad `DP[7:0]`.

[132] Los ciclos de transferencia de datos con pipeline permiten solapar dos transferencias consecutivas.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

En escritura, el microprocesador pone los datos y el byte de paridad sobre el bus de datos y, a continuación, testea **BRDY#**; cuando **BRDY#** se activa, termina el ciclo. Seguidamente, tanto en lectura como en escritura, el procesador retira del bus las señales de direcciones, habilitadores de bytes y tipo de ciclo de bus. Dos ciclos de reloj después de la activación de **BRDY#**, **PCHK#** indica, para lecturas del microprocesador, el resultado del chequeo de paridad.

10.1.1.2. Transferencia simple con estados de espera En la figura 2.44 se pueden ver ciclos de transferencia de datos con un estado de espera. El proceso que se lleva a cabo es similar al anterior con la diferencia de que en el estado T2 el microprocesador encuentra **BRDY#** inactiva y, en consecuencia, añade un estado de espera T2. En subsiguientes ciclos vuelve a consultar el valor de **BRDY#** hasta que ésta se encuentra activa.

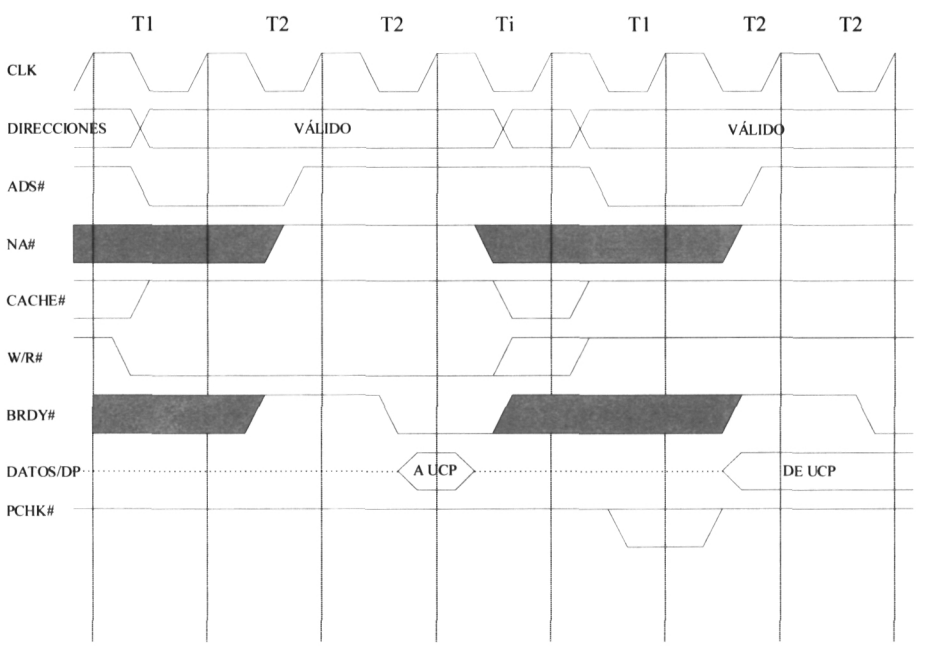


Figura 2.44: Ciclos de transferencia simple de datos del Pentium: lectura y escritura con estado de espera

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

10.1.1.3. Transferencia de ráfaga La figura 2.45 ilustra un ciclo de lectura en modo ráfaga (es una lectura de línea de la caché).

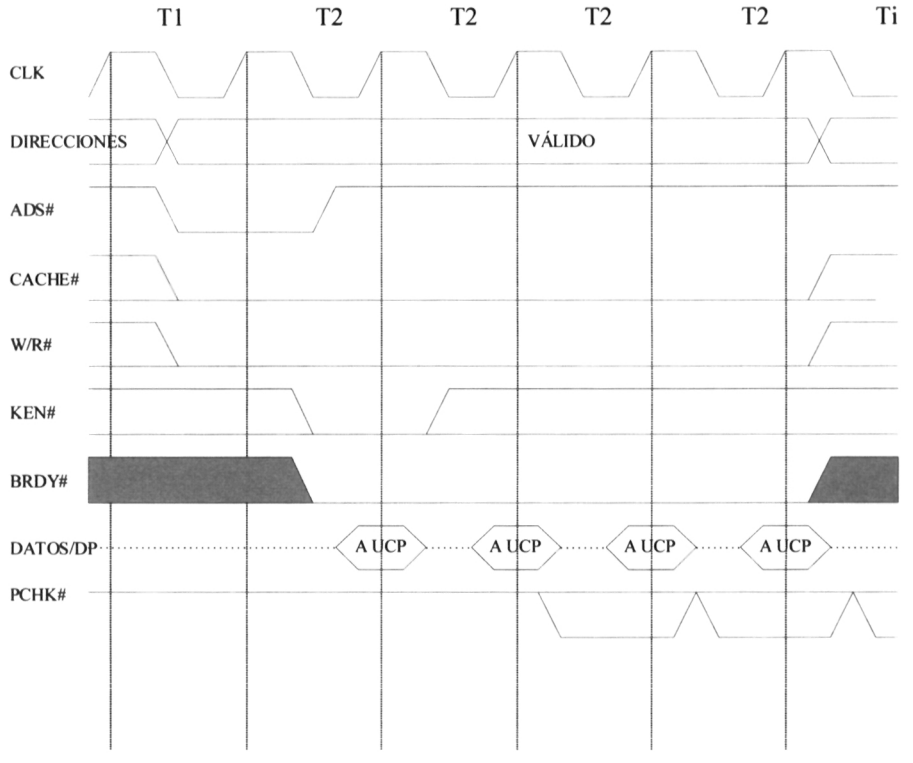


Figura 2.45: Ciclos de ráfaga del Pentium: lectura

El proceso que se lleva a cabo es el siguiente:

En el estado T1, se generan las direcciones `A[31:3]` de la primera transferencia[133], los habilitadores de bytes `BE[7:0]#` y el tipo de ciclo de bus determinado por la señales `M/IO#`, `D/C#`, `W/R#`, `CACHE#` y `KEN#`. A continuación, se activa `ADS#`, indicando que todas las señales anteriores están estables.

En el primer estado T2 se testea `KEN#` para saber si el sistema soporta transferencias en modo ráfaga. En cada estado T2 se testea `BRDY#`. Como está activa, el microprocesador lee los datos de `D[63:0]` y el byte de paridad `DP[7:0]` y termina el ciclo de transferencia de datos. Si no es la última transferencia de la ráfaga, espera hasta el siguiente ciclo de reloj y vuelve al estado

[133] Observe que el microprocesador mantiene la dirección de la primera transferencia durante todo el ciclo de ráfaga.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2. Si es la última transferencia, el microprocesador retira del bus las señales de direcciones, habilitadores de bytes y tipo de ciclo de bus. Dos ciclos de reloj después de la activación de `BRDY#`, `PCHK#` indica el resultado del chequeo de paridad.

10.2. Interfaz de interrupciones

El microprocesador testea la señal `INTR` al comienzo de cada instrucción (ver figura 2.46).

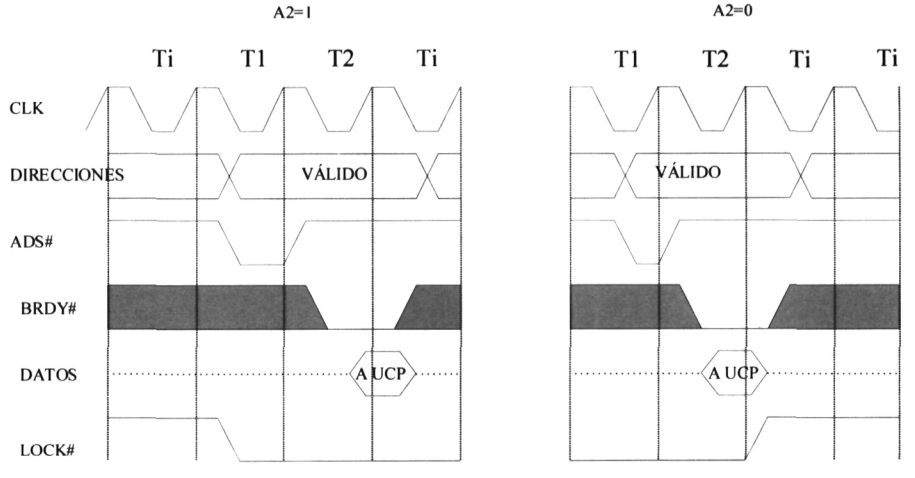


Figura 2.46: Ciclo de reconocimiento de interrupción del Pentium

Si `INTR` se activa, indica que algún dispositivo solicita servicio. Si el procesador acepta la solicitud de interrupción, genera un ciclo de reconocimiento de interrupción. Este ciclo consta de varios ciclos de transferencia de datos. Durante el primer ciclo de transferencia, el microprocesador indica que la interrupción ha sido aceptada y que el dispositivo externo solicitante debe prepararse para enviar su vector de interrupción. Durante el segundo ciclo (puede haber estados de espera entre el primer y el segundo ciclo), el microprocesador lee el vector de interrupción suministrado por el dispositivo externo sobre el bus `D[7:0]`. Este vector se corresponde con una dirección de memoria (tabla de vectores/descriptores de interrupción).

En la tabla 2.7 se muestran las señales del interfaz de interrupciones del Pentium.

El Pentium dispone de un controlador de interrupciones programable avanzado denominado APIC (Advanced Programmable Interrupt Controller) que

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Nombre	Dirección	Descripción
INTR	entrada	Indica al Pentium que un dispositivo externo requiere su atención. Es una interrupción enmascarable
NMI	entrada	Similar INTR excepto que es no enmascarable
RESET	entrada	Inicializa los registros internos del microprocesador
INIT	entrada	Es un reset que no afecta a las memorias cachés de primer nivel ni a los registros de punto flotante

Tabla 2.7: Interfaz de interrupciones del Pentium

recibe interrupciones procedentes de INTR y NMI, de fuentes internas y de APIC de E/S externos[134].

10.3. Interfaz de multiproceso

En la tabla 2.8 se muestran las señales del interfaz de multiproceso del Pentium. En multiproceso, el APIC transfiere mensajes de interrupción entre procesadores. Los mensajes se pueden usar para distribuir las interrupciones entre los procesadores del sistema o para ejecutar funciones generales del sistema (arranque de procesadores, trabajo distribuido de un grupo de procesadores, etc.).

Nombre	dirección	Descripción
BREQ	salida	Indica que el microprocesador tiene un ciclo de bus pendiente. Puede ser usado por lógica externa para arbitrar entre múltiples microprocesadores
HOLD	entrada	Indica que un microprocesador externo o DMA quiere usar el bus. El microprocesador acepta la petición terminando su ciclo de bus en curso y a continuación colocando sus buses en alta impedancia y activando la señal HLDA
HLDA	salida	Indica que el microprocesador cede el control del bus
BOFF#	entrada	Aborta el ciclo de bus en curso. Se utiliza para resolver bloqueos entre buses maestros
LOCK	salida	Indica que el bus está inhibido. Por tanto, ningún otro microprocesador o DMA podrá adueñarse del bus

Tabla 2.8: Interfaz de multiproceso del Pentium

[134] Los APICs de E/S forman parte de chipset de la placa base.

Publicaciones y Divulgación Científica
UMA UNIVERSIDAD DE MÁLAGA

11. Interfaz del Pentium II

Las señales del Pentium II se agrupan en 9 interfaces (ver figura 2.47): Arbitración, Solicitud, Error, Snoop, Respuesta, Datos, Control, Compatibilidad y Diagnóstico.

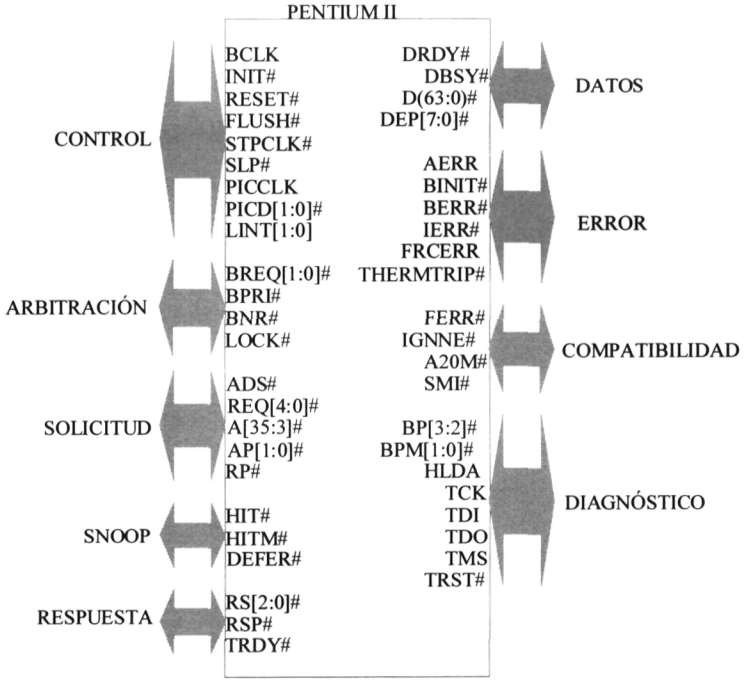


Figura 2.47: Señales de interfaz de Pentium II

El Pentium II usa señales de bajo voltaje del tipo AGTL+ (Assisted Gunning Transistor Logic) para poder soportar altas frecuencias. Todas las señales son open-drain activas a nivel bajo[135] incluyendo buses de datos y direcciones.

11.1. Transacciones

Las operaciones de bus del Pentium II se llaman transacciones y difieren bastante de las del Pentium. Los diferentes interfaces del Pentium II son grupos de señales independientes por lo que permiten aplicar la técnica del pipeline; es decir dividir la operación en etapas independientes que se pueden solapar.

[135] Por tanto requieren resistencias de pull-up.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Se distinguen 6 etapas o fases:

1. Arbitración
2. Solicitud
3. Error
4. Snoop
5. Respuesta
6. Datos

En la figura 2.48 se puede ver un ejemplo de transacciones[136].

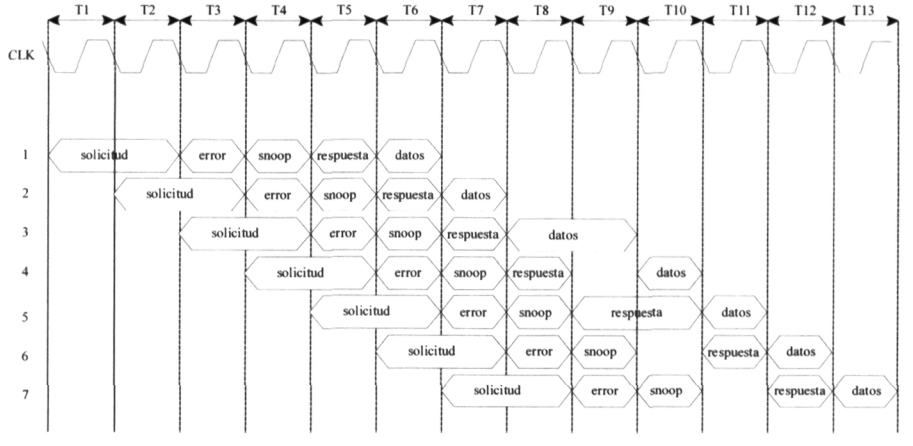


Figura 2.48: Transacciones en el PentiumII

1. La fase de arbitración se usa en sistema multiproceso para determinar qué agente (procesador) será el siguiente maestro del bus.

2. La fase de solicitud[137] con la activación de la señal **ADS#** dura dos ciclos de reloj. En el primero se identifica el tipo de solicitud (acceso a memoria, acceso a E/S, ciclo especial del sistema, lectura o escritura, etc..) y en el segundo se proveen algunos parámetros (habilitadores de byte, longitud de ráfaga, etc.).

[136] No se incluye la fase de arbitración.

[137] Las solicitudes del bus entran en una cola (cada agente mantiene su cola).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3. En la tercera fase (error), cualquier agente puede indicar que algo ha ido mal (por ejemplo, que la dirección tiene un error de paridad) y que la transacción debe ser cancelada o retrasada.

4. En la cuarta fase (snoop) se comprueban las cachés del sistema para evitar posibles incoherencias. Observando la transacción, `HIT` y `HIT#` se usan para indicar si la línea es válida o inválida dentro del agente snooping, si la línea ha sido modificada en el agente caching, o si la transacción necesita ser extendida. `HIT` y `HIT#` se usan para mantener la coherencia de caché a nivel de sistema.

5. La quinta fase (respuesta) puede durar varios ciclos de reloj y define las diferentes respuestas a la solicitud (terminada, postpuesta, suspendida, etc.). El agente respuesta[138] es el responsable de completar la transacción de la primera solicitud de la cola.

6. Finalmente, en la última fase (datos) se transfieren los datos[139].

No todas las fases son necesarias en todas las transacciones y, en algunos casos, la fase puede durar más de un ciclo lo que ralentiza el pipeline. En cualquier caso, esta técnica consigue mejorar el ancho de banda considerablemente.

Las tablas 2.9, 2.10, 2.11, 2.12, 2.13, 2.14, describen las señales de los interfaces de arbitración, solicitud, error, snoop, respuesta y datos, respectivamente.

Nombre	Tipo	Descripción
BREQ[3:0]#	bidireccional	Cuando se activa la senal BREQ[X] indica que el agente X solicita el bus
BPRI#	entrada	Bus Priority Request. La activa el agente prioritario para solicitar el bus
BNR#	salida	La activa un agente para bloquear posteriores transacciones
LOCK#	bidireccional	Indica que el agente está realizando una secuencia atómica de transacciones que no puede ser interrumpida. Útil para implementar semáforos basados en memoria

Tabla 2.9: Señales de arbitración del Pentium II

[138] El agente respuesta es el agente direccionado por la transacción.

[139] A veces se provocan escrituras no solicitadas debido al sistemas de coherencia de cachés.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Nombre	Tipo	Descripción
ADS#	bidireccional	Address Strobe. Define el comienzo de una transacción. Indica que las señales REQ[4:0]#, A[35:3]#, AP[1:0]# y RP# son válidas
A[35:3]#	bidireccional	Bus de direcciones. Las líneas de direcciones A[2:0] se sustituyen por los habilitadores de byte BE[7:0] que se envían en el segundo ciclo de reloj de la fase de solicitud sobre las líneas de A[15:8]. El orden en el que se envían los datos de una ráfaga también viene determinado por líneas de direcciones: A[4:3]#
AP[1:0]#	bidireccional	Paridad del bus de direcciones. AP[1]# cubre A[35:24]# y AP[0]# A[23:3]#.
RP#	bidireccional	Request parity. Paridad de ADS# y REQ[4:0]#
REQ[4:0]#	bidireccional	Son señales de comando de solicitud. En el primer ciclo de reloj definen el tipo de transacción a un nivel suficiente para poder empezar una solicitud de snoop. En el segundo ciclo de reloj, completan la información necesaria (por ejemplo la longitud de la ráfaga)

Tabla 2.10: Señales de solicitud del Pentium II

Nombre	Tipo	Descripción
AERR	bidireccional	Address Parity Error.
BINIT#	bidireccional	Bus Initialization.
BERR#	bidireccional	Bus Error.
IERR#	salida	Internal Error
FRCERR#	bidireccional	Functional Redundancy Checking
THERMTRIP#	salida	Se activa cuando el microprocesador alcanza una temperatura peligrosa

Tabla 2.11: Señales de error del Pentium II

Nombre	Tipo	Descripción
HIT#	bidireccional	Guardando una línea de caché no modificada
HITM#	bidireccional	Indica línea de caché modificada
DEFER#	entrada	Indica que una transacción tiene que repetirse. Normalmente es activada por el agente de memoria o E/S

Tabla 2.12: Señales de snoop del Pentium II

Nombre	Tipo	Descripción
RS[2:0]#	entrada	Response Status. Hay varios tipos de respuestas: transacción completa o normal (con o sin datos), transacción postpuesta y transacción write back implícita
RSP#	entrada	RS parity . Provee paridad para RS[:0]
TRDY#	entrada	Target Ready. Activada por el agente respuesta para indicar que está preparado para aceptar escritura o datos writeback

Tabla 2.13: Señales de respuesta del Pentium II

Nombre	Tipo	Descripción
DRDY#	bidireccional	Indica que hay un dato válido sobre el bus y que debe ser registrado (en un latch). Se desactiva para insertar estados de espera en la transferencia de datos
DBSY#	bidireccional	Indica que el bus de datos esta ocupado. La activa el agente responsable de transferir los datos
D[63:0]#	bidireccional	Bus de datos
DEP[7:0]#	bidireccional	Código de corrección de errores para el bus de datos

Tabla 2.14: Señales de datos del Pentium II

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

11.2. Arbitración

La familia P6 admite hasta 5 agentes simultáneos arbitrando el bus del sistema (figura 2.49): 1 agente prioritario (subsistema de memoria y E/S) que usa `BPRI#` y 4 agentes simétricos que usan `BREQ[1:0]#`. La arbitración simétrica sigue un esquema de prioridad rotatoria round-robin. Después de un `RESET` el agente 0 tiene máxima prioridad seguido del agente 1. Todos los agentes rastrean al propietario en curso del bus. En función de los valores muestreados y de quién ha sido el último que ha usado el bus, los agentes deciden quién será el siguiente dueño del bus.

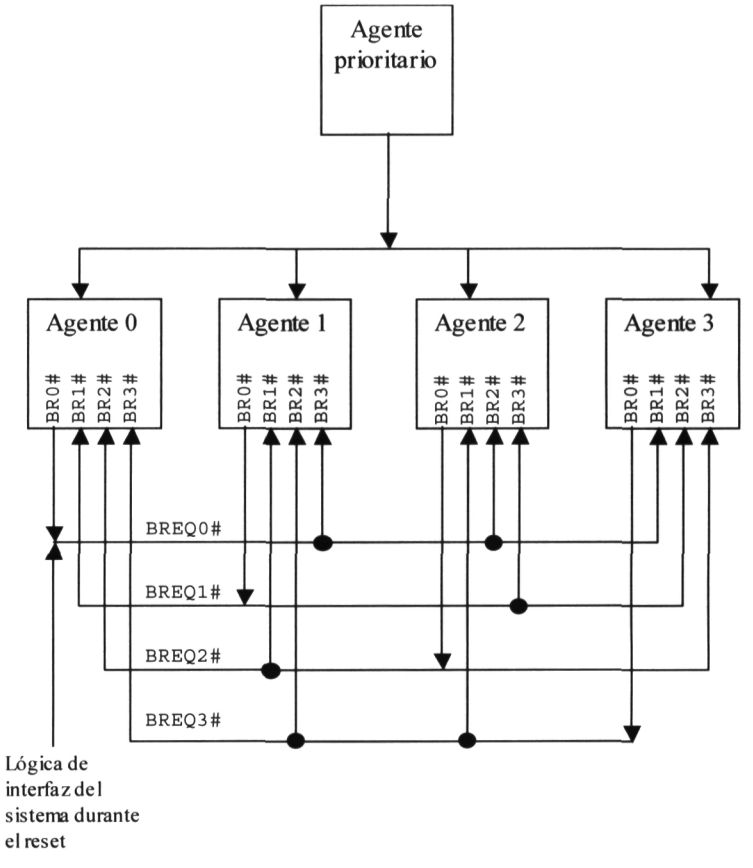


Figura 2.49: Arbitración del Pentium II

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

11.3. Interfaces de control, compatibilidad y diagnóstico

Nombre	Tipo	Descripción
BCLK	entrada	Reloj del bus. La frecuencia interna del microprocesador se obtiene de este reloj multiplicando su valor por un factor que proporcionan las señales LINT[1:0] A20# y IGNNE# cuando se inicializa el microprocesador
RESET#	entrada	Reset
INIT#	entrada	Similar a RESET sin afectar a la cachés L1 y L2 y los registros de punto flotante
FLUSH#	entrada	Hace un write back de todas las cachés internas e invalida todas las líneas de caché
STPCLK#	entrada	Pone al microprocesador en estado de bajo consumo (stop-grant)
SLP#	entrada	Pone en estado sleep al microprocesador que está en estado stop-grant.
PICCLK	entrada	Es un reloj de entrada al APIC(Advanced Processor Interrupt Controller) del microprocesador
PICD[1:0]#	bidireccional	Bus de intercambio de mensajes del bus APIC
LINT[1:0]	entrada	Señales de interrupción locales. Si el APIC está deshabilitado LINT0 es equivalente a INTR y LINT1 a NMI

Tabla 2.15: Señales de control de ejecución del Pentium II

Nombre	Tipo	Descripción
FERR#	salida	Floating Point Error
IGNNE#	entrada	Ignore Numeric Error
A20M#	entrada	Address 20 Mask. Sólo funciona en modo real
SMI#	entrada	System Management Interrupt. Hace que el microprocesador conmute al modo SMM

Tabla 2.16: Señales de compatibilidad del Pentium II

Nombre	Tipo	Descripción
BP[3:2]#	bidireccional	Break Point. Indica estado de break point
BPM[1:0]#	bidireccional	BreakPoint Monitor
TCK	entrada	Test Clock
TDI	entrada	Test Data In. Transfiere datos de test serie según especificación JTAG
TDO	salida	Test data Out. Transfiere datos de test serie según especificación JTAG
TMS	entrada	Test Mode Select. según especificación JTAG
TRST#	entrada	TAP reset
PRDY#	salida	Probe Ready. depurador
PREQ#	entrada	Probe Request.

Tabla 2.17: Señales de diagnóstico del Pentium II

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

12. Interfaz de Pentium 4

Las señales del Pentium 4 son bastante parecidas a las del Pentium II. Las diferencias más significativas están en que:

- Las señales `A[35:3]#` y `REQ[4:0]#` son *source synchronous* y utilizan una nueva señal `ADSTB[1:0]#` para sincronizarlas y registrarlas.
- Las señales del bus de datos son **quad pump** lo que permite realizar 4 transferencias en un ciclo de reloj (figura 2.50). Para ello se requieren nuevas señales de sincronización del bus de datos: `DSTBp[3:0]#` y `DSTBn[3:0]#`. Hay una por cada 16 bits.
- La señal `BPM[1:0]#` se extiende a `BPM[5:0]#`.

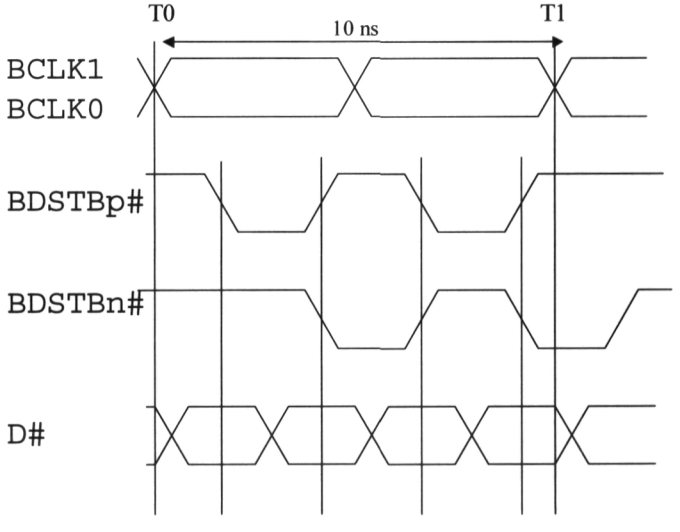


Figura 2.50: Transferencia quad pump del Pentium 4

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

13. Últimas tendencias

La evolución de los microprocesadores es realmente vertiginosa. Hace sólo dos décadas el 8086 corría a 4,77MHz e integraba menos de un millón de transistores, mientras que, hoy en día, las últimas versiones de Pentium superan los 3GHz e integran más de 100 millones de transistores.

El diseño superescalar empieza a ahogarse a partir de las 5 ó 6 instrucciones despachadas por ciclo. Esto se debe a que el hardware de planificación crece de forma geométrica con el número de unidades funcionales y absorbe una parte importante de la superficie del chip. Una alternativa denominada VLIW[140] (Very Large Instruction Word) es que el compilador resuelva toda esta planificación. El Itanium de Intel es una arquitectura de 64 bits que incorpora esta nueva filosofía de diseño (la llama EPIC: Explicitly Parallel Instruction Computing) e incluye nuevas mecanismos para aliviar las dependencias y las latencias de memoria como predicación, rotación de registros, eliminación de bucles, etc.

El Itanium cuenta con 325 millones de transistores (25 en la UCP y 300 en las cachés). Trabaja a 800 MHz y dispone de 3 niveles de caché que reducen drásticamente la latencia (2-4MB, 96KB y 32KB par las cachés de nivel L3, L2 y L1, respectivamente). Tiene un bus de datos de 2.1 GBps (266MHz) y dispone de un sistema avanzado de detección y corrección de errores y de gestión del sistema. Es compatible con la arquitectura IA-32 de Intel.

En 2004 varios microprocesadores de 64 bits destacaron en el mercado: El Itanium II alias Madison de Intel[141], el Athlon 64FX de AMD y el PowerPC G5 de IBM.

El Itanium II Madison a 1,5GHz y 410 Millones de transistores integra 3 niveles de caché[142] y dispone de un bus del sistema de 128 bits a 400MHz (6,4GBps). Soporta hasta 4 procesadores.

El Athlon 64FX de AMD integra dos niveles de caché[143], nuevos registros de propósito general y SSE/SSE2, instrucciones multimedia 3DNow Professional y SSE/SSE2, tecnología Hypertransport y un controlador de memoria de

[140] Un compilador inteligente es capaz de examinar un programa, encontrar todas las instrucciones que no tengan dependencias, agruparlas en procesos por lotes muy grandes y ejecutarlos concurrentemente en una matriz de unidades funcionales igualmente grande. Las instrucciones que se crean pueden llegar a tener entre 256 y 1024 bits.

[141] Intel lanzó un Pentium 4 con extensiones de 64 bits para competir con el Athlon 64FX de AMD: Es la denominada tecnología EM64T.

[142] L1:32KB(instrucciones+datos), L2:256KB y L3:6MB.

[143] L1:64KB(instrucciones)+64KB(datos) y L2:1MB.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

128 bits que soporta 6,4 GBps. El microprocesador es compatible con Windows XP.

El PowerPC G5 a 2GHz integra dos niveles de caché[144] y tiene un bus del sistema de 32+32 bits a 1GHz (8GBps). Las aplicaciones de 32 bits corren a la velocidad del microprocesador[145].

Una iniciativa de IBM, Sony y Toshiba, inicialmente pensada para la inminente PlayStation 3, es el procesador Cell, especializado en procesamiento distribuido y vectorial, que promete revolucionar el mundo de los microprocesadores.

[144] L1: 64KB(intrucciones)+32KB(datos) y L2: 512 KB.

[145] La arquitectura PowerPC fue diseñada desde el principio para 32 y 64 bits.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

14. Bibliografía

Los artículos recomendados estudian la microarquitectura de los últimos microprocesadores y descubren aspectos que no son fáciles de encontrar en los manuales. Por otro lado, los manuales son, sobre todo, la guía de referencia de este capítulo para la arquitectura software de la familia IA-32. Los libros propuestos no tienen la actualidad de las demás referencias pero si la experiencia y la didáctica de sus autores que empiezan a abordar ya la microarquitectura P6. También se proponen muchos títulos de revistas y páginas web; estas últimas son de mucha utilidad para estar al día en esta tecnología que deja obsoletos sus logros en plazos cada vez más cortos.

Artículos

Cohen, D.: On Holy Wars and a Plea for Peace, IEEE Computer Magazine, Vol. 14, pp.48-54,, October 1981.

Marr, T.D. et al.: Hyper-threading Technology Architecture and Microarchitecture, Intel Technology Journal, Vol. 06, No. 1, February 2002.

Ronen, R. et al.: Coming Challenges in Microarchitecture and Architecture, Proceedings of the IEEE, Vol. 89, No. 3, March 2001, pp. 325-340.

Sharangpani, H., and Arora, K.: Itanium Processor Microarchitecture, IEEE Micro, September-October 2000, pp. 24-43.

Valero, M. y González Colás, A.: Evolución de la arquitectura de los microprocesadores, Mundo Electrónico, nº 271, pp. 78-84.Noviembre, 1996.

Gwennap, L.: Intel´s MMX Speeds Multimedia, Microprocessor Report, Vol. 10 No. 3 March 1996.

Gwennap, L.: Intel´s P6 Uses Decoupled Superscalar Design, Microprocessor Report, Vol. 9 No. 2 March 1995.

Case, B.: Intel Reveals Pentium Implementation Details, Microprocessor Report, Vol. 7 No. 4 March 1993.

Manuales

IA-32 Intel Architecture Software Developer's Manual, Volume 1: Basic Architecture, 2006.

IA-32 Intel Architecture Software Developer's Manual, Volume 2: Instruction Set Reference Manual, 2006.

IA-32 Intel Architecture Software Developer's Manual, Volume 3: System Programming Guide, 2006.

P6 Family of Processors, Hardware Developer's Manual Order No: 244001-001, 1998.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Intel Pentium 4 Processor in the 423-pin Package at 1.30, 1.40, 1.50, 1.60, 1.70, 1.80, 1.90 and 2 GHz Datasheet, Order No.: 249198-005, 2001.

Libros

Angulo, J.S. and Funke, E.M.: Microprocesadores Avanzados 386 y 486. Introducción al Pentium y PentiumPro, Paraninfo, 1997. ISBN 84-283-2374-7.

Antonakos, J.L.: The Pentium Microprocessor, Prentice Hall Career&Technology, 1997. ISBN: 0-02-303614-1.

Mazidi, M.A. y Mazidi, J.G.: The 80x86 IBM PC and compatible Computers, Vol. II: Design and Interfacing of the IBM PC and compatible Computers, Prentice Hall Career&Technology, 1998. ISBN: 0-13-758491-1.

Triebel, W.A.: The 80386, 80486, and Pentium Microprocesssor: Hardware, Software, and interfacing, Prentice Hall Career&Technology, 1998. ISBN: 0-13-533225-7.

Barry, B.B.: INTEL Microprocessors 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, Pentium Pro Processor, Pentium II, III, 4, 7ª Edición, Prentice May, 2006.

Revistas

Intel Technology Journal, Intel. ISSN 1535766X.

Computer Design, PennWell Publishing Company, ISSN-0010-4566.

EDN, Cahners Publishing Company, ISSN 0012-7515.

Electronics Design, UNU Businesss Publications Inc., ISSN 0013-4872.

Mundo Electrónico,Boixareu Ed. ISSN: 0300-3787

Revista Española de Electrónica, Ediciones técnicas REDESA, ISSN: 0482-6396.

Computers and Digitals Techniques, IEE Proceedings-E., IEE Publishing. ISSN: 0143-7062.

Computer, IEEE Computer Society, mensual, ISSN 0018-9162.

Computers Surveys, ACM

IEEE J. Solid-State Circuits. IEEE Computer Society

Micro, IEEE Computer Society, bimensual. ISSN: 0272-1732.

Microproccesing and microprogramming, North Holland. ISSN: 0165-6074.

Microprocessor Report, MicroDesign Resources.

Spectrum, IEEE Computer Society

Transactions on Computers, IEEE Computer Society, ISSN 0018-9340.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Transactions on Computers System, ACM.

Direcciones de Internet

Guías

PCguide, www.pcguide.com

Museo de CPUs, www.cpu-museum.de

PC Webopaedia, www.pcwebopedia.com

Tom´hardware Guide, www2.tomshardware.com

Revistas

IEEE Micro, www.computer.org

EDN, www.edn.com

Centros de excelencia

UC Berkeley CPU Info Center, bwrc.eecs.berkeley.edu/

Fabricantes de microprocesadores

AMD. www.amd.com

ARM. www.arm.com

Digital. www.digital.com

Hewlett-Packard. www.hp.com

IBM Microelectronics. www.chips.ibm.com

Integrated Device Technology. www.idt.com

Intel. www.intel.com

Mips. www.mips.com

Motorola. www.mot.com,

Freescale. www.freescale.com

National Semiconductor. www.nsc.com

Oki. www.oki.com

Philips. www.nxp.com

Siemens. www.siemens.com

Texas Instrument. www.ti.com

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

15. Guía de estudio

1. Clasificar microprocesadores de la familia IA-32 de Intel atendiendo a criterios de frecuencia de trabajo, capacidad de direccionamiento, conjunto de instrucciones y integración de cachés.
2. Explicar las diferencias entre las microarquitecturas Pentium y P6.
3. Explicar las diferencias entre las microarquitecturas P6 y NetBurst.
4. Enumerar las características distintivas del microprocesador Core.
5. Explicar el funcionamiento del predictor de saltos del Pentium.
6. Describir el concepto de la caché de trazas.
7. Analizar la ejecución de una secuencia de instrucciones de microprocesadores Pentium y P6 (ver apartado 3.5.1).
8. Enumerar los modos de operación de IA-32.
9. Explicar la función de los registros de segmento de IA-32.
10. Enumerar los tres espacios lógicos de direcciones de IA-32 y distinguirlos.
11. Explicar las diferencias entre instrucciones MMX, SSE, SSE2 y SSE3.
12. Explicar el proceso de traducción de dirección lógica a dirección lineal en IA-32.
13. Analizar secuencias de código MMX sencillas (ver apartado 6.7).
14. Definir los conceptos de selector, descriptor, tabla de descriptores y puerta.
15. Explicar el proceso de traducción de dirección lineal a dirección física en IA-32 cuando hay paginación.
16. Enumerar los mecanismos de protección de IA-32.
17. Calcular una dirección física a partir de los valores de los registros y algunas direcciones de memoria en un microprocesador IA-32 (ver apartado 9.3).
18. Explicar las diferencias entre los buses del Pentium y del Pentium II.
19. Explicar la diferencia entre el bus de datos de un Pentium II y el de un Pentium 4.
20. Interpretar un cronograma de un ciclo de lectura de un Pentium.
21. Explicar las diferencias entre los microprocesadores G5, Athlon 64FX e Itanium 2.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD DE MÁLAGA
Publicaciones y Divulgación Científica

3 Memorias

1. Introducción

La memoria es un elemento fundamental en cualquier sistema basado en microprocesador. Desafortunadamente, no hay una única tecnología de memoria que cubra todos los requerimientos de cada sistema. Características tales como velocidad, espacio, coste y consumo obligan a usar diferentes tecnologías para cada caso.

Un ordenador personal usa SRAM para las memorias cache, DRAM para la memoria primaria, memorias magnéticas u ópticas para la memoria secundaria, memorias especializadas para el buffer de video, Flash para la memoria de arranque, etc.. Sin embargo, los teléfonos móviles, cámaras digitales o reproductores MP3 usan Flash para almacenar el código y ejecutarlo[146]. Otro grupo muy importante de sistemas son los pequeños microcontroladores que sólo disponen de memoria ROM para guardar el código y una pequeña memoria SRAM para las variables.

En este capítulo se describen las características básicas de las memorias semiconductoras, magnéticas y ópticas.

[146] Otros sistemas como los PDAs usan Flash sólo para almacenarlo volcándolo en memoria SDRAM cuando va a ser ejecutado. Las videoconsolas suelen usar RDRAM como memoria primaria. También hay sistemas específicos de muy alta velocidad como, por ejemplo, los routers que requieren memorias SRAM multipuerto muy veloces para sus buffers de datos.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2. Memorias semiconductoras

Todas las memorias semiconductoras tienen una organización similar. Disponen de un interfaz al bus, un decodificador de direcciones, una matriz de células de memoria y una unidad de control. La figura 3.1 muestra un diagrama de bloques de una arquitectura típica. La dirección se descompone en dirección de fila que selecciona una fila completa de la matriz de memoria y dirección de columna que selecciona una columna de la fila. El número de datos leídos o escritos en cada acceso depende del número de multiplexores del dispositivo. En la figura 3.1 se muestra un ejemplo de una memoria organizada en bit que sólo requiere un multiplexor.

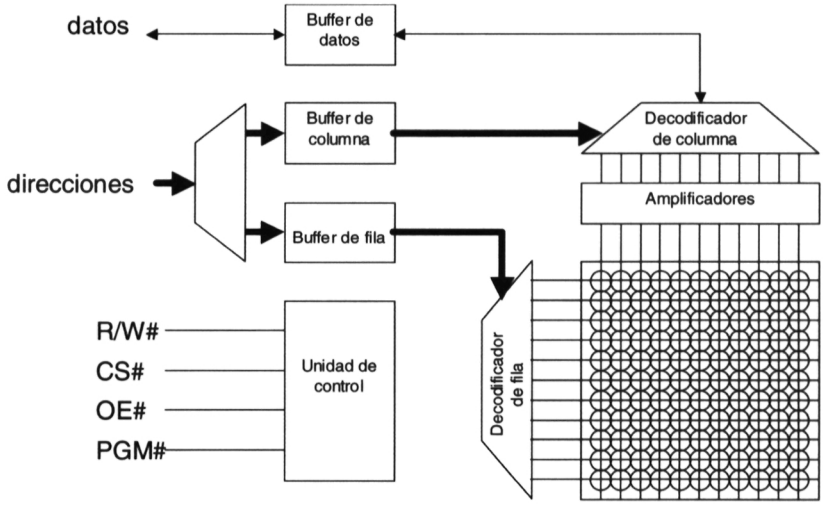


Figura 3.1: Arquitectura típica de una memoria semiconductora

Las memorias semiconductoras han evolucionado mucho desde su aparición en los años 60. En la figura 3.2 se muestra una clasificación general de las mismas donde se distingue entre memorias volátiles y memorias no volátiles.

La memoria no volátil conserva la información almacenada aunque se retire la alimentación del sistema. Hasta no hace muchos años, ha sido básicamente una memoria de lectura o ROM (Read Only Memory) con versiones que facilitaban su borrado y re-escritura[147]. Sin embargo, la tecnología Flash ha revolucionado estos dispositivos que, hoy en día, compiten con otros tipos de

[147] PROM (Programmable ROM), EPROM (Erasable EPROM) y EEPROM (Electrically EPROM).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

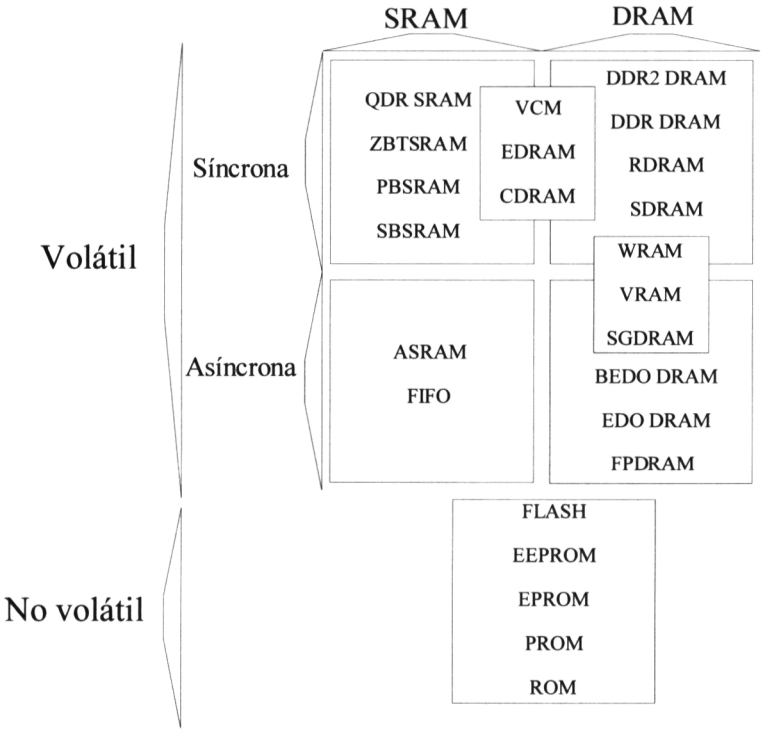


Figura 3.2: Clasificación de las memorias semiconductoras

memoria en muchos sistemas empotrados.

Las memorias volátiles pierden la información almacenada cuando se retira la alimentación. Son memorias muy rápidas de lectura y escritura. Se denominan memorias RAM (Random Access Memory) ya que son de acceso directo (no secuencial). Se clasifican en memorias DRAM (Dynamic RAM) y memorias SRAM (Static RAM).

Desde la década de los 80, los microprocesadores han sido mucho más rápidos ejecutando instrucciones que la memoria primaria DRAM proporcionándoselas. Este hecho, si no se toman medidas, hace al sistema ineficiente ya que el procesador permanece desocupado mientras está a la espera de recibir los datos de memoria. La solución ideal es usar memoria SRAM, pero esto no es viable para muchos sistemas que requieren gran cantidad de memoria y bajo coste. Así que las soluciones han venido por otro lado (figura 3.3): ampliar el tamaño del bus de datos, mejorar el ancho de banda de la memoria DRAM[148]

[148]Se han utilizado muchas técnicas: Modo página, pipeline, modo ráfaga, sincronismo, etc.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

o utilizar nuevas técnicas de acceso[149] o la memoria cache. Esta última ha sido determinante en la resolución del problema y es un elemento clave de la denominada jerarquía de memoria

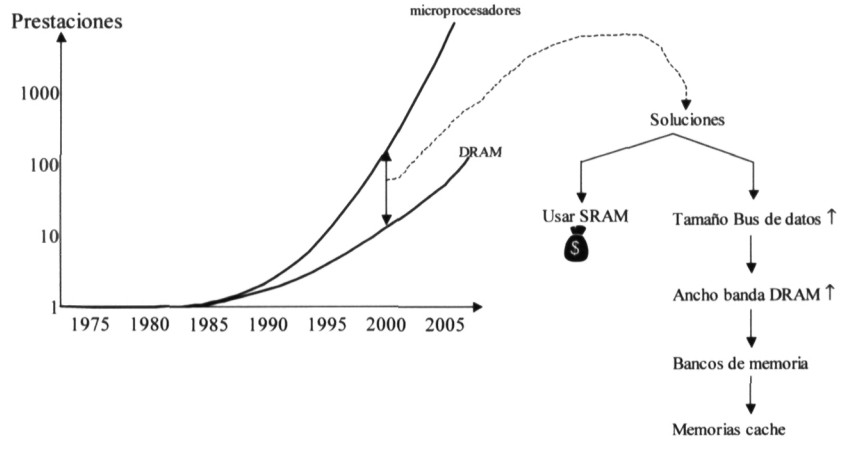


Figura 3.3: Desequilibrio entre prestaciones de microprocesadores y memorias DRAM. Soluciones propuestas

El aumento del ancho de banda de las memorias DRAM se consiguió inicialmente con pequeñas modificaciones arquitecturales que mejoraban el acceso secuencial: el modo página de FPDRAM (Fast Page DRAM), el pipeline de EDO DRAM (Extended Data Out DRAM) y el modo ráfaga de BEDO DRAM(Burst EDO DRAM). Posteriormente, se diseñaron arquitecturas síncronas multibanco: SDRAM (Synchronous DRAM), RDRAM (RAMbus DRAM), DDR SDRAM (Double Data Rate SDRAM), SLDRAM (Synchronous Link DRAM), etc..

Las memorias SRAM han sufrido una evolución parecida. Se han vuelto síncronas y han mejorado su ancho de banda mediante técnicas similares a las utilizadas en las DRAM: PBSRAM (Pipeline Burst SRAM), ZBTSRAM (Zero Bus Turnaround SRAM), DDR SRAM(Double Data Rate SRAM) y QDR SRAM (Quad Data Rate SRAM).

También hay memorias de propósito específico. Para el buffer de video hay memorias de doble puerto como VRAM (Video RAM) o WRAM (Windows RAM), o memorias DRAM con facilidades para el manejo de gráficos como SGRAM (Synchronous Graphics RAM). Muchos sistemas de conmutación de paquetes hacen uso de memorias FIFO (First In First Out). Hay memorias

[149] Como los sistemas multibanco o buses especializados.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

SRAM direccionables por el contenido (CAM, Content Addressable Memory). También hay memorias que no encajan en ninguna de las clasificaciones anteriores como las EDRAM(Enhanced DRAM), CDRAM (Cache DRAM) y VCM (Virtual Channel Memory) que incluyen tanto memoria SRAM como DRAM.

2.1. Memorias no volátiles

Las memorias no volátiles se pueden clasificar en memorias ROM, PROM, EPROM, EEPROM, FLASH y NVRAM.

La memoria ROM (Read Only Memory), también llamada ROM de máscara, es una memoria de sólo lectura que se programa en fábrica. Por tanto, el contenido de la memoria tiene que ser especificado antes de la producción del chip. Su principal ventaja es su bajo coste para un gran volumen de producción. Cada célula de memoria es un transistor que puede estar conectado (1 lógico) o no (0 lógico) a la matriz de células.

La memoria PROM (Programmable ROM) es una memoria ROM que permite ser grabada por el usuario, aunque sólo admite una grabación. Hoy en día se suele denominar OTP (Only Time Programmable). Cada célula de memoria es un transistor con un fusible asociado. Para desconectar un transistor se funde el fusible (0 lógico) aplicando una corriente eléctrica[150]. Si el fusible no se funde el transistor "almacena" un 1 lógico.

La memoria EPROM (Erasable PROM) es una PROM que se puede borrar y reprogramar varias veces. Se borra mediante rayos UltraVioleta, lo que exige extraerla del zócalo donde está ubicada. El número de veces que se puede borrar es limitado (100-1000). Son más caras que las PROMs pero muy utilizadas en el desarrollo de sistemas. Cada célula de memoria es un transistor con una puerta flotante que puede almacenar carga de forma permanente (1 lógico) o no (0 lógico). El borrado permite eliminar la carga de las puertas flotantes.

La memoria EEPROM es una EPROM que se puede programar y borrar eléctricamente (no requiere sacarla del circuito). Son más pequeñas y lentas[151] que las EPROM, además de ser más caras. Permiten la programación byte a byte. Su interfaz suele ser serie. La célula de la memoria EEPROM es similar a la de la memoria EPROM con la diferencia de que se añade una puerta de control para eliminar o almacenar carga en la puerta flotante.

[150] Se requiere una tensión de programación elevada: 12V-21V.

[151] Sus tiempos de acceso son del orden de 200-300ns mientras que en las EPROM los tiempos son de 30-40ns.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

La memoria Flash es similar a la EEPROM pero con considerables mejoras: Es más rápida, más barata y de menor consumo. Además, no requiere altas tensiones para su programación y permite más ciclos de reprogramación. A diferencia de la EEPROM, no se borra byte a byte sino por sectores. Hay varios tipos de memorias Flash dependiendo de la configuración de las células y de la manera de acumular la carga eléctrica (NOR, DiNOR, AND, NAND, etc..) aunque se podrían reducir a dos grandes familias: NOR y NAND. En la tabla 3.1 se muestra una comparativa de ambas.

Característica	**NOR**	**NAND**
Capacidad	1Mb-256Mb	16Mb-8Gb
Aplicaciones	almacenamiento y ejecución de código del S.O.	almacenamiento de ficheros (imágenes, datos, música)
Acceso	aleatorio	secuencial
Tiempo de lectura	rápido	rápido
Tiempo de borrado	lento	rápido
Tiempo de programación	lento (byte a byte)	rápido
Densidad	baja	alta
Coste/bit	cara	barata
Interfaz	directo	indirecto
Ciclos de borrado	$10^4 - 10^5$	$10^5 - 10^6$

Tabla 3.1: Comparativa de memorias Flash

La memoria NVRAM (No Volátil SRAM) usa una tecnología diferente a las anteriores. Es una memoria SRAM con batería. Es muy cara y su uso se restringe al almacenamiento de unos cientos de bytes de información crítica del sistema que no se puede guardar de otra forma.

2.2. Memorias volátiles dinámicas

En las memorias volátiles dinámicas o DRAM, cada célula de almacenamiento está representada por la carga de un pequeño condensador[152], que se implementa dentro de un sencillo transistor.

Las figuras 3.4 y 3.5 muestran la arquitectura y el cronograma, respectivamente, de una memoria DRAM.

La dirección se descompone en dirección de fila[153] y dirección de columna[154]. En primer lugar, se proporciona la dirección de fila que se valida con la activación de la señal **RAS#**. A continuación, se hace lo mismo con la columna de la fila seleccionada usándose, en este caso, la señal **CAS#**. Observe que las direcciones de filas y columnas no son necesarias al mismo tiempo, por lo que

[152] 30-50 femptofaradios. Un femtofaradio son 10^{-15} faradios.

[153] La fila también se llama página.

[154] La columna es la mínima cantidad direccionable de memoria. Puede tomar valores de 1, 4, 8 , 16 ó 32 bits.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

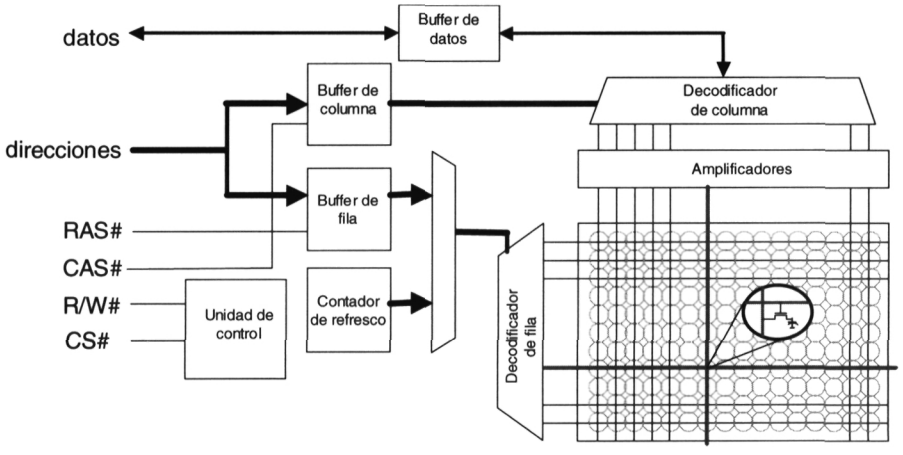


Figura 3.4: Arquitectura de una memoria DRAM

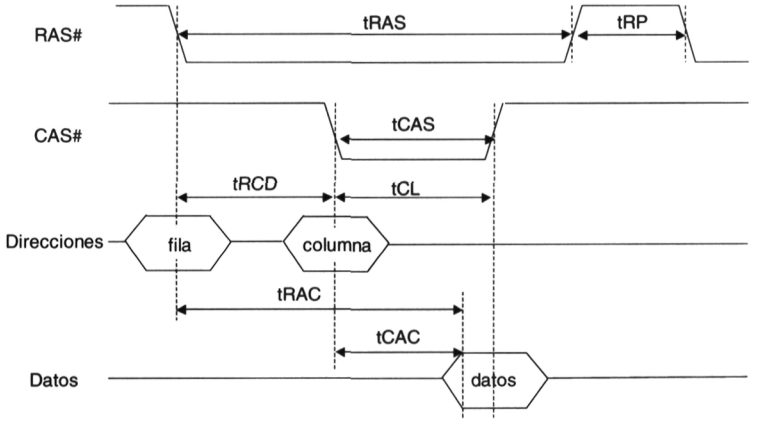


Figura 3.5: Ciclo básico de una memoria DRAM

comparten las mismas señales; esto hace que el empaquetado sea más pequeño y más barato que el de las SRAM, aunque complique la distribución de señales dentro del chip.

Una vez seleccionada la dirección del dato, la memoria tarda cierto tiempo en volcarlo sobre el bus de datos. Este tiempo se denomina latencia CAS o latencia CL.

La lectura de una fila es "destructiva" por lo que si queremos acceder a otra fila hay que esperar cierto tiempo, denominado tiempo de precarga

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

(tRP), para que la memoria pueda reestablecer los datos[155]. Una típica técnica para ocultar el tiempo de precarga es usar bancos de memorias con acceso intercalado (interleaving).

Las memorias dinámicas tiene que "refrescarse" periódicamente, ya que las células de almacenamiento son condensadores que se descargan con el tiempo. Esto se hace leyendo y escribiendo las células cada cierto tiempo. Esto ocurre automáticamente siempre que se accede a una fila.

En la figura 3.5 se pueden ver otros parámetros temporales de interés. tRCD es el intervalo entre la selección de fila y columna. El tiempo que tarda la memoria en leer una palabra desde que la dirección de fila es válida hasta que el dato está estable en el bus se denomina tiempo de acceso aleatorio (tRAC, Row Access Time). El tiempo que tarda la memoria en leer una palabra desde que la dirección de columna es válida hasta que el dato está estable en el bus se denomina tiempo de acceso en modo página (tCAC, Column Access Time).

La figura 3.6 muestra la evolución de la densidad de memoria DRAM. Se observa que sigue un crecimiento exponencial.

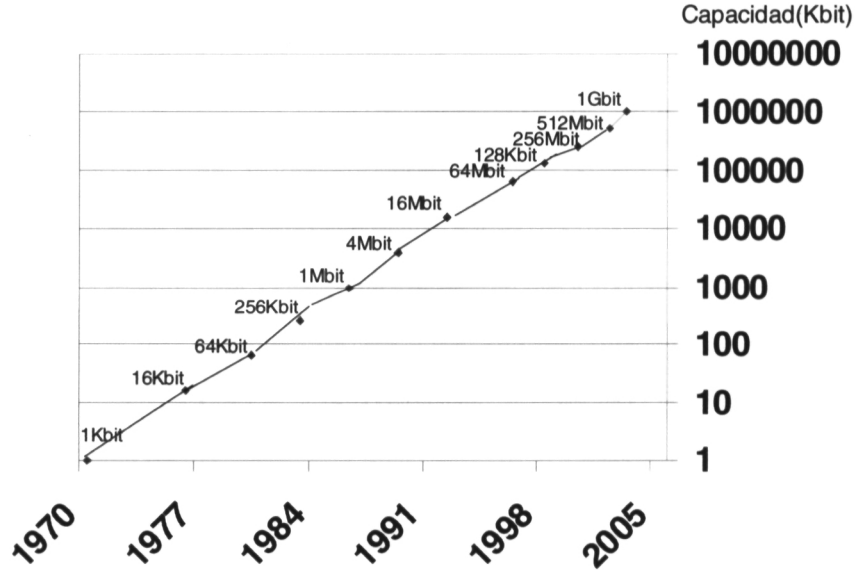


Figura 3.6: Evolución en densidad de la memoria DRAM

[155]Se hace mediante una lectura de la fila.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.2.1. Memorias asíncronas

La memorias DRAM asíncronas son las DRAM tradicionales, ya en desuso. Hay varios tipos: FPM DRAM, EDO DRAM y BEDO DRAM.

2.2.1.1. FPM DRAM La FPM DRAM (Fast Page Mode DRAM) es una DRAM convencional con un tiempo de acceso en el modo página más rápido. En la figura 3.7 se puede ver un ejemplo de lectura de este tipo de memoria.

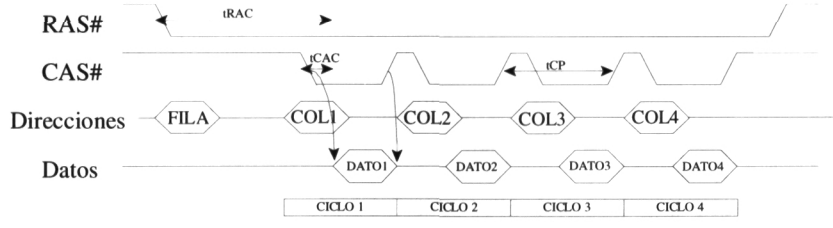


Figura 3.7: Ciclo de lectura de una memoria FPDRAM

Inicialmente el bus pone una dirección de fila y activa la señal `RAS#`. Esto selecciona una fila de la matriz. A continuación, se pueden hacer múltiples accesos a columnas (sin modificar la dirección de fila) poniendo una dirección de columna y activando la señal `CAS#`. Cuando los datos son válidos sobre el bus (después del tiempo `tCAC`), se desactiva `CAS#` para preparar el siguiente ciclo. Note que cuando `CAS#` se desactiva, se deshabilitan los datos de salida, por tanto, esto sólo puede ocurrir cuando los datos han sido capturados por el sistema. El tiempo entre accesos en modo página se denomina tiempo de ciclo de página (`tCP`).

2.2.1.2. EDO DRAM Las mejoras en la velocidad de las DRAM habían venido siendo de avances en el proceso de fabricación. Sin embargo, ligeros cambios en la arquitectura que no requerían aumentar el tamaño del chip consiguieron aumentar sustancialmente las prestaciones. Este fue el caso de las DRAM con etapas pipeline. El pipeline desacopla las diferentes tareas que lleva a cabo la lectura/escritura de memoria, permitiendo el solape de accesos consecutivos (ver figura 3.8).

La EDO (Extended Data Output) DRAM es una DRAM con una etapa de pipeline que se implementa incluyendo unos latches para guardar los datos

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

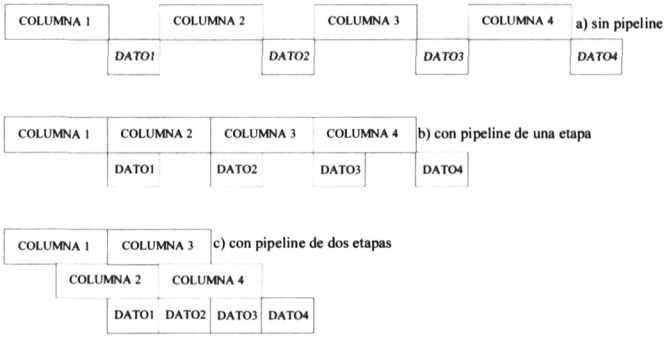


Figura 3.8: Pipeline de memoria

de salida. Así, cuando se presenta la dirección de página (fila), los datos seleccionados se amplifican y se guardan en estos latches al mismo tiempo que se presentan al bus. Esto permite al decodificador de direcciones y al circuito de camino de datos iniciar un acceso a la siguiente dirección de página, sin necesidad de deshabilitar los datos de salida. En la figura 3.9 se puede ver un ejemplo de lectura con este tipo de memorias.

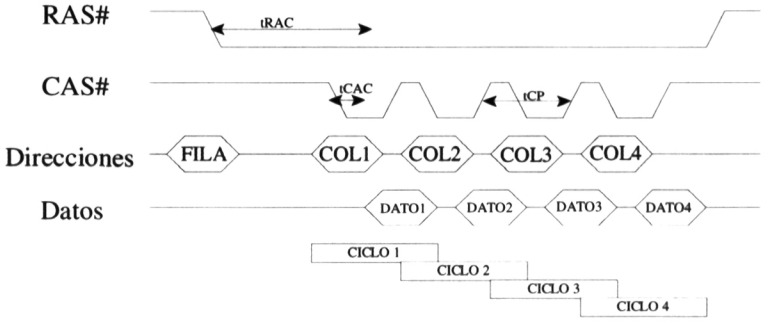


Figura 3.9: Ciclo de lectura de una memoria EDO DRAM

El acceso es muy similar al visto en las FPM DRAM, con la excepción de que cuando la señal **CAS#** se desactiva no se deshabilitan los datos de salida. Esto permite generar una nueva dirección sin corromper los datos del acceso anterior.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

El problema de las EDO DRAM es que no trabajan para velocidades de reloj superiores a 66MHz. Actualmente los buses del sistema superan los 100MHz por lo que este tipo de memorias ha quedado obsoleto.

2.2.1.3. BEDO DRAM La memoria BEDO DRAM (Burst EDO DRAM) incorpora un contador de direcciones interno de 2 bits, de manera que el sistema sólo tiene que enviar la dirección inicial de la ráfaga de datos que quiere obtener[156]. Esta técnica elimina los retardos asociados con la decodificación de direcciones. Las BEDO DRAM consiguen tiempos de acceso de ráfaga de 5-1-1-1 pero no pueden ir a más de 66MHz[157].

La memoria BEDO DRAM suele añadir una etapa de latch adicional, de manera que los datos no alcanzan[158] la salida en el primer ciclo **CAS#** (ver figura 3.10).

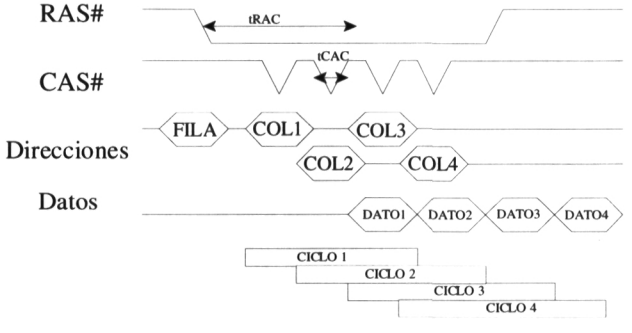


Figura 3.10: Ciclo de lectura de una memoria BEDO DRAM

Los beneficios de este pipeline interno de 2 etapas son que los datos aparecerán en menos tiempo desde la activación de la señal **CAS#** en el segundo ciclo (**tCP** es más pequeño).

[156]Una ráfaga de datos es una secuencia de datos que están ubicados en direcciones consecutivas de memoria. Es el típico modo de acceso a memoria principal en los sistemas de propósito general que hacen uso de memoria cache.

[157]La notación 5-1-1-1 significa que tarda 5 ciclos de reloj en el primer acceso y sólo un ciclo en cada uno de los tres restantes.

[158]Se puede ver que el primer ciclo de **CAS#** no causa retardo adicional en la recepción del primer elemento de datos. Esto es porque el acceso al primer elemento de datos está limitado por **tRAC**, el cual, en efecto, oculta el primer ciclo **CAS#**.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.2.2. Memorias síncronas

En las memorias síncronas un reloj sincroniza los accesos de memoria. Esta sincronización de la memoria con el procesador simplifica el diseño del interfaz, reduce tiempos de setup y hold y mejora, junto con un aumento de etapas del pipeline, el tiempo de acceso. Hay varios tipos de memorias DRAM síncronas: SDRAM, DDR SDRAM, DDR2 SDRAM y RDRAM.

2.2.2.1. SDRAM La memoria SDRAM (Synchronous DRAM) es una memoria DRAM síncrona que integra un registro de modo, contadores de refresco y de ráfaga y varios bancos de memoria (figura 3.11).

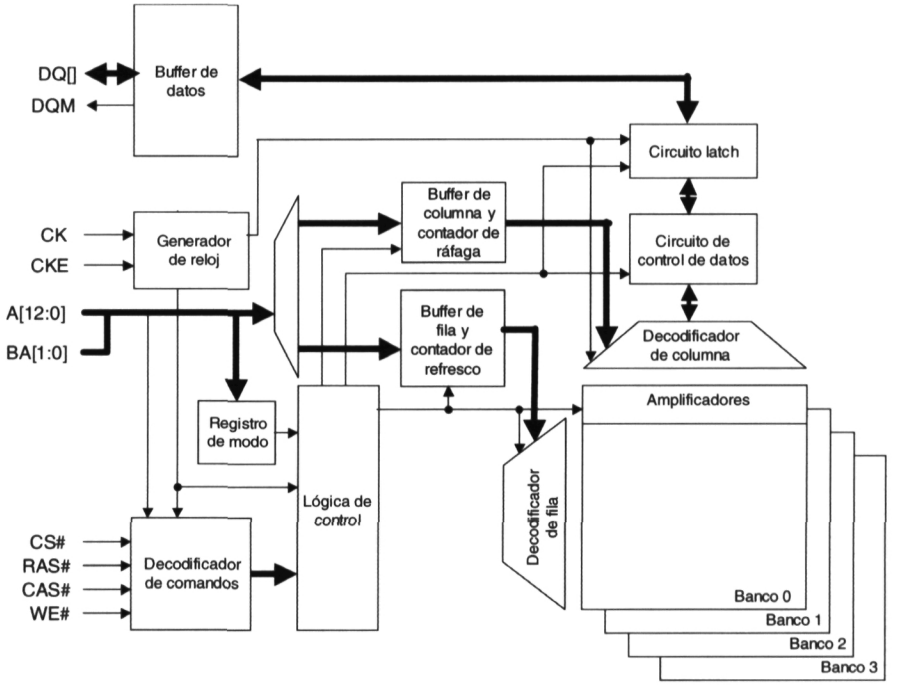


Figura 3.11: Arquitectura de una memoria SDRAM

El registro de modo permite programar la longitud y tipo de ráfaga y la latencia `CAS#`. Los bancos de memoria (2 ó 4) facilitan el acceso entre páginas (interleaving) y, en consecuencia, consiguen enmascarar (en muchos casos) el tiempo de precarga.

Las diferentes combinaciones de las señales de control (`CS#`, `RAS#`, `CAS#`, y `WE#`) determinan el conjunto de comandos de la SDRAM (tabla 3.2):

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

CS#	RAS#	CAS#	WE#	Comando
0	0	0	0	Configuración del registro de modo
0	0	0	1	Refresco
0	0	1	0	Precarga
0	0	1	1	Activación de banco y fila
0	1	0	0	Escritura
0	1	0	1	Lectura
0	1	1	0	Parada de ráfaga
0	1	1	1	No operación
1	X	X	X	No hay comando

Tabla 3.2: Conjunto de comandos de una SDRAM típica

La figura 3.12 muestra un acceso de lectura en modo ráfaga. Todas las señales están sincronizadas con el flanco de subida del reloj `CLK`. `CKE` es una señal de habilitación del reloj utilizada en modos especiales (bajo consumo y auto-refresco).

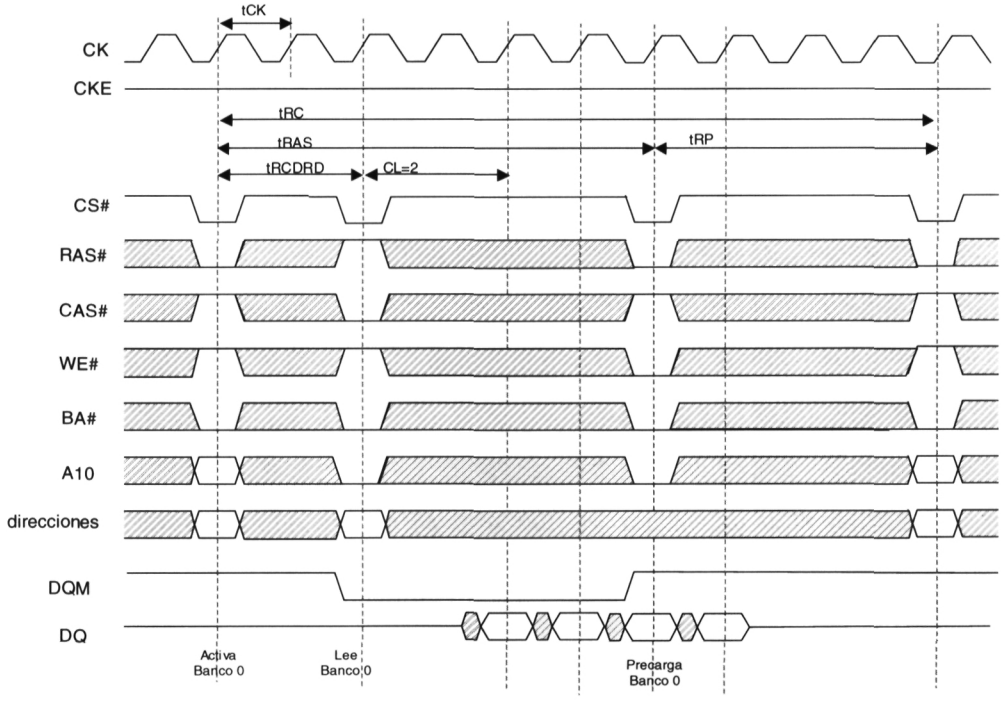


Figura 3.12: Ciclo de lectura de una memoria SDRAM

Inicialmente se genera un comando de activación de banco y fila. Las señales `BA#` determinan el banco seleccionado y `A10` el tipo de precarga. Debe

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

transcurrir un tiempo `tRCD` antes de emitir el comando de lectura. A continuación, hay que esperar `CL` ciclos de reloj (latencia `CAS#`) para que los datos estén disponibles. Las señales `DQM` son habilitadoras de bytes[159]. La precarga del banco se puede solicitar antes de que se acabe la transferencia de datos. SDRAM usa niveles LV TTL (Low Voltage TTL) de 3,3V.

2.2.2.2. DDR SDRAM La memoria DDR SDRAM (Double Data Rate DRAM) es una SDRAM con un mecanismo *2n-prefetch* que le permite leer/escribir dos datos en cada acceso. En la figura 3.13 se muestra un esquema comparativo que ilustra esta técnica.

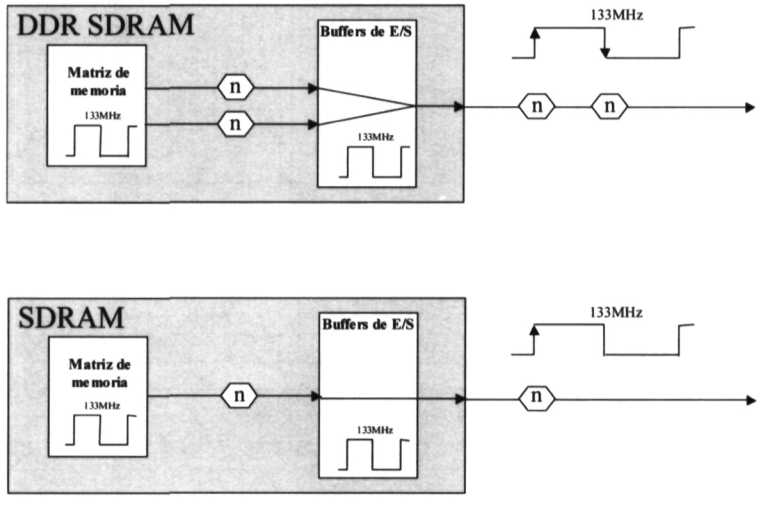


Figura 3.13: Comparativa entre SDRAM y DDR SDRAM

En una SDRAM, el bus de datos de la matriz de células proporciona n bits por cada ciclo de reloj (a 133MHz). El bus externo transfiere los n bits en el flanco de subida del reloj externo (también de 133MHz). En una DDR SDRAM, el bus de datos de la matriz de memoria duplica su tamaño lo que le permite transferir $2n$ bits por ciclo de reloj. El bus externo usa un reloj de la misma frecuencia (133MHz) pero transfiere los $2n$ bits en un ciclo de reloj, aprovechando tanto los flancos de subida como de bajada. El resultado es que el ancho de banda se duplica.

La figura 3.14 muestra un cronograma de una DDR SDRAM.

[159] Por ejemplo, en un bus de datos de 16 bits habrá dos `DQM`: `DQM[1]` para el byte alto de datos `DQ[15:8]` y `DQM[0]` para el byte bajo `DQ[7:0]`.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

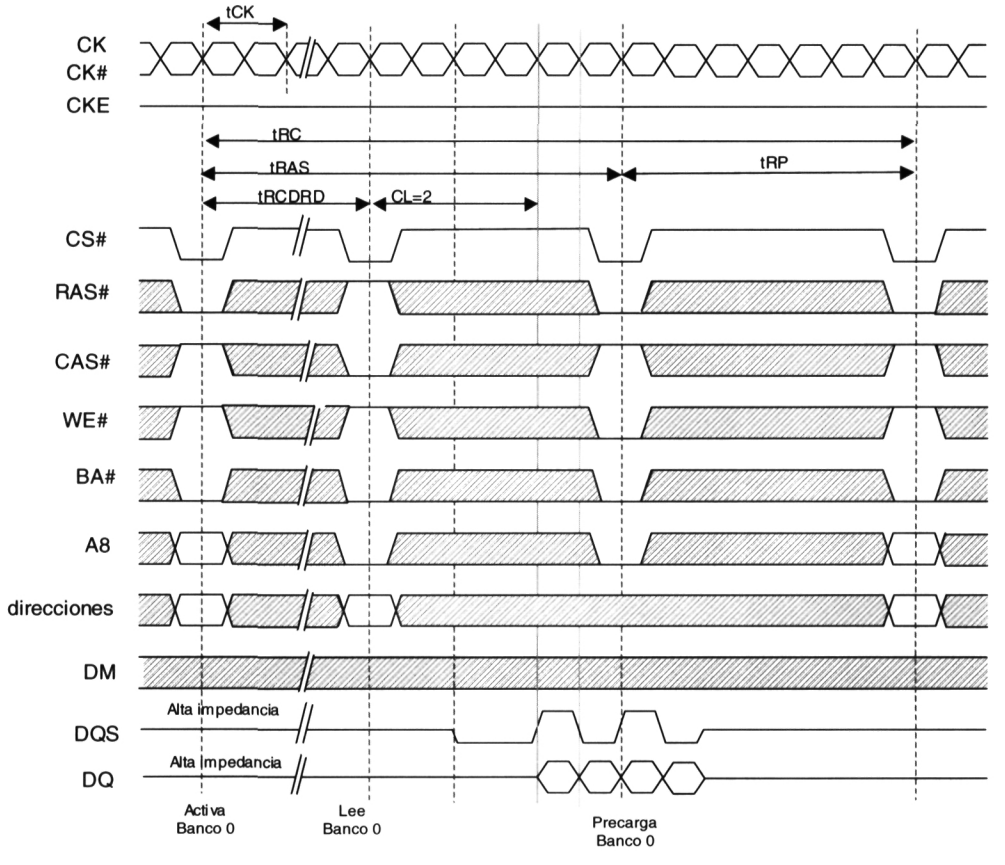


Figura 3.14: Ciclo de lectura de una memoria DDR DRAM

El reloj (`CK` y `CK#`) es diferencial para minimizar el ruido. Todas las señales de datos `DQ` están referenciadas a la señal `DQS` que es bidireccional[160]. Una DDR SDRAM permite alcanzar velocidades de 266 (133MHz×2), 333 (166MHz×2) y 400 (200MHz×2) MTps (MegaTransferencias por segundo). Los comandos se introducen en los flancos de subida de reloj, mientras que los datos se reciben tanto en los flancos de subida como en los de bajada. Se usan niveles SSTL-2 (Stub Series Terminated Logic for 2,5V) junto con una tensión de referencia.

2.2.2.3. DDR2 SDRAM La DDR2 SDRAM es una DDR que introduce nuevas funciones y características. El ancho del bus de datos interno tiene

[160] `DQS` es una salida en ciclos de lectura y una entrada en ciclos de escritura.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

cuádruple anchura que el externo (*4n-prefetch*) por lo que puede leer/escribir cuatro datos en cada acceso. En la figura 3.15 se muestra un esquema comparativo que ilustra esta técnica.

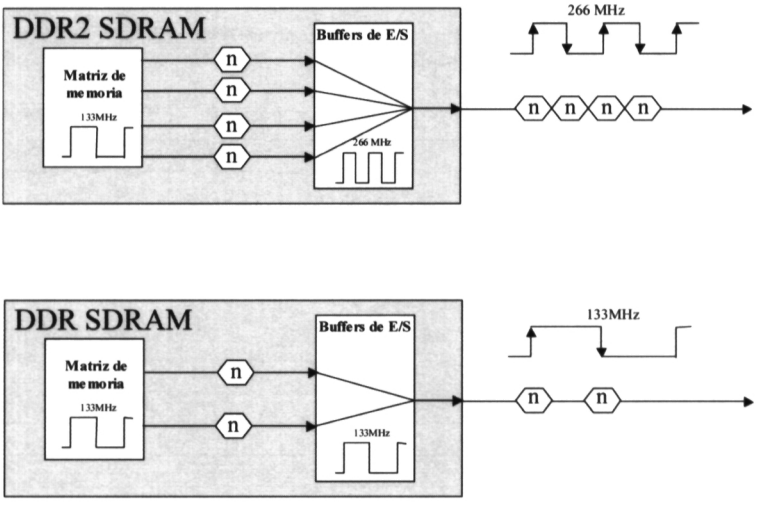


Figura 3.15: Comparativa entre DDR SDRAM y DDR2 SDRAM

El bus externo de la DDR2 trabaja a doble frecuencia (233 MHz) que la matriz de almacenamiento (133MHz) por lo que es capaz de transferir los $4n$ bits en un ciclo de reloj de 133MHz (o lo que es lo mismo, en dos de 233MHz)

Una DDR2 puede alcanzar velocidades[161] de 400 (200 MHz×2), 533 (266 MHz×2), 667 (333 MHz×2) y 800 (400 MHz×2)MTps.

DDR2 proporciona 8 bancos de memoria y tiene un tamaño de página más pequeño con el objetivo de reducir el consumo de potencia. Otras mejoras son el uso de resistencias de adaptación de impedancias dentro del chip (on-die termination) para reducir el ruido y la latencia adicional que mejora la eficiencia del planificador de comandos[162]. DDR2 trabaja con señales de 1,8V lo que permite reducir el consumo.

[161] Observe que la matriz de células sigue trabajando a 100 MHz, 133 MHz, 166 MHz y 200 MHz, respectivamente: 400 (100 MHz×4), 533 (133 MHz×4), 667 (166 MHz×4) y 800 (200 MHz×4)MTps.

[162] En una DDR hay que respetar el tiempo `tRCD` entre el comando de activación y el de lectura/escritura; Esto ocasiona conflictos entre comandos de diferentes accesos que reducen el ancho de banda de la memoria. DR2 resuelve este problema permitiendo que el comando de lectura/escritura se pueda emitir en cualquier momento después del comando de activación.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.2.2.4. RDRAM La RDRAM (Rambus DRAM) es una memoria DRAM síncrona de muy alta velocidad desarrollada por Rambus Inc. RDRAM requiere un diseño muy cuidadoso del sistema de memoria (ver figura 3.16) para poder ofrecer sus elevadas prestaciones.

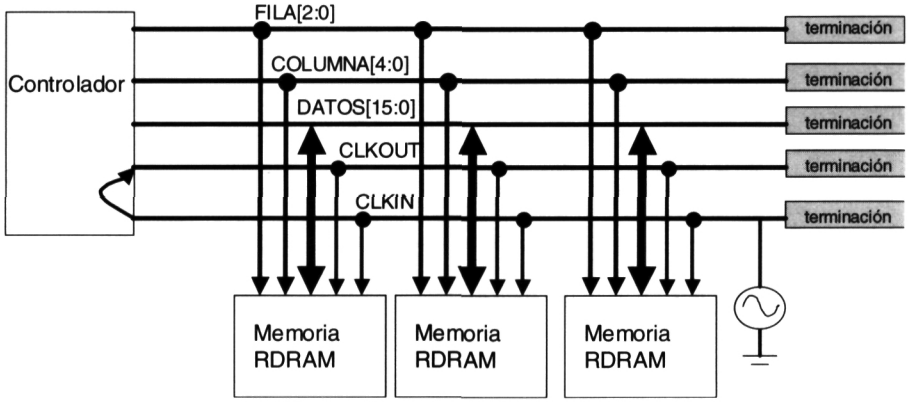


Figura 3.16: Bus de memoria RAMbus

El sistema incluye un controlador, un canal Rambus y los módulos de memoria. El canal Rambus usa 30 señales de muy alta velocidad que usan ambos flancos de un reloj para transferir datos a tasas que pueden llegar a las 1200MTps[163]. Las señales se dividen en bus de control de fila `FILA[2:0]`, bus de control de columna `COLUMNA[4:0]`, bus de datos `DATOS[15:0]` y 2 señales de reloj `CLKIN` y `CLKOUT`.

En la figura 3.17 se muestra la arquitectura de una memoria DRDRAM (Direct RDRAM) típica. Se puede ver el gran número de bancos que tiene comparado con las memorias SDRAM. Esta arquitectura reduce la probabilidad de que cuando se quiera acceder a un banco éste se encuentre en el periodo de precarga, lo que mejora sustancialmente el ancho de banda efectivo.

[163]RDRAM maneja relojes de 400, 533 y 800 MHz que proporcionan tasas de 800, 1066 y 1200 MTps, respectivamente.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

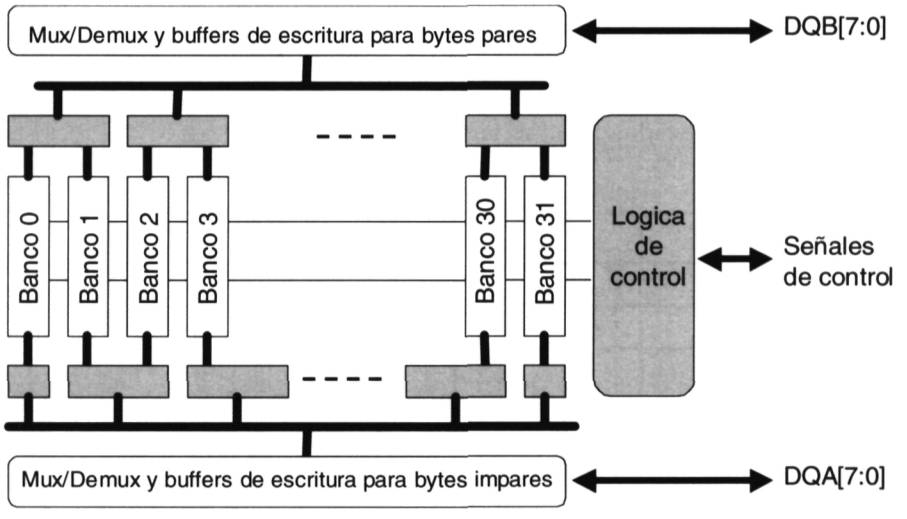


Figura 3.17: Arquitectura de memoria RAMbus

La figura 3.18 muestra un acceso de lectura de una memoria RDRAM donde se puede observar como se obtienen 16 bytes en tan sólo 10 ns (1.6 GBps).

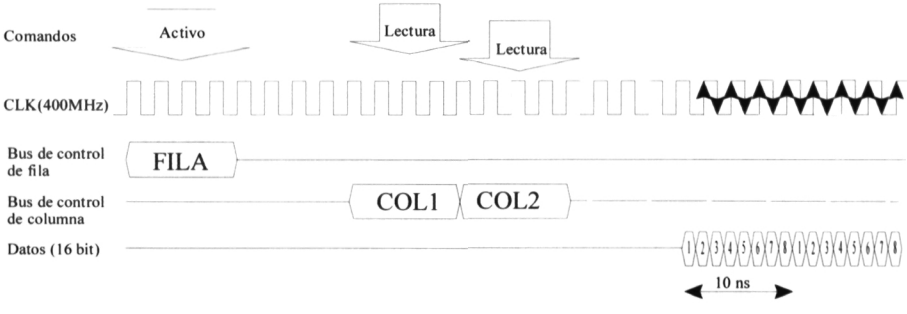


Figura 3.18: Ciclo de lectura de una memoria RDRAM

La información de control y datos se transfiere a través del canal mediante un protocolo orientado a paquete. Los paquetes duran 4 ciclos de reloj. Se usan niveles RTL (Rambus Signaling Level) de 0.8V de margen.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.2.3. Memorias especializadas

Hay muchos tipos de memorias DRAM de propósito especial. En este apartado se comentan algunas de las más significativas.

2.2.3.1. PSDRAM Las PSDRAM (PseudoStatic DRAM) usan el mecanismo de almacenamiento de una DRAM con lógica adicional para que el chip se comporte como una SRAM. El refresco se lleva a cabo internamente. Su interfaz es igual al de las memorias SRAM (No tiene entradas de direcciones multiplexadas). Son memorias que intentan reducir el consumo sacrificando velocidad.

2.2.3.2. Memorias de video Algunas memorias DRAM están especialmente diseñadas para su uso en los buffers de video de las tarjetas gráficas. Hay que distinguir entre memorias de un puerto y memorias de doble puerto.

Entre las memorias de un puerto destacan las SGRAM (Synchronous Graphics DRAM). Las SGRAM son memorias SDRAM que añaden lógica adicional para optimizar operaciones típicas de gráficos tales como doble buffering, 3D rendering y escritura de bloques. También destacan las DDR SGRAM, GDDR (Graphics DDR) y XRAM (eXtreme data Rate DRAM); estas últimas[164] pueden llegar a conseguir anchos de banda de 8 GBps.

Las memorias de doble puerto permiten dos accesos simultáneos a la memoria, uno aleatorio (para cargar la información) y otro secuencial para leerla y transferirla al monitor. Estas memorias permiten duplicar el ancho de banda, a costa, por supuesto, de mayor complejidad y coste[165].

La memoria de doble puerto más conocida es la VRAM (Video RAM) aunque hoy en día hay otras implementaciones más eficientes y económicas como la WRAM (Windows RAM). La WRAM es una VRAM que optimiza muchas de las operaciones típicas de un entorno gráfico: escritura de texto y relleno en documentos de alta resolución, copia de información de una parte del buffer a otra (8 veces más rápida que una VRAM), etc..

[164] Las XRAM se caracterizan por unos niveles de señal extremadamente bajos (200mv), 8 transferencias en un ciclo de reloj y adaptación de impedancias individualizada para cada linea de datos.

[165] Una ventaja esencial de esta arquitectura con respecto a la memorias de un sólo puerto es que es menos dependiente de parámetros tales como resolución, número de colores y tasa de refresco.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.2.4. Empaquetado

En los años 70, las memorias DRAM se empaquetaban en formato DIP (Dual In-Line) y los fabricantes de sistemas las soldaban en la placa o las enchufaban en zócalos dispuestos para tal fin[166]. A principios de los 80, los fabricantes de DRAM empezaron a ofrecer módulos de memorias DRAM sobre pequeñas placas de circuito impreso que se conectaban en zócalos especiales. A finales de los 80, estos módulos, denominados SIMM (Single In Line Memory Modules), se habían vuelto el empaquetado DRAM más popular.

Actualmente se usan módulos DIMMs (Dual In Line Memory Modules) (ver figura 3.19) que son similares a los SIMMs, pero con señales distintas por cada cara[167].

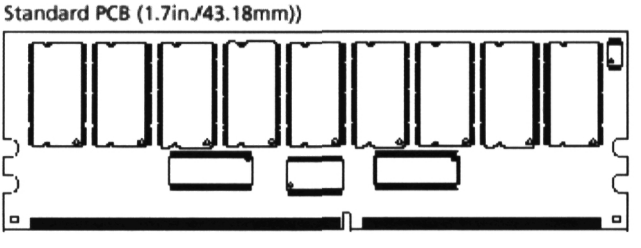


Figura 3.19: Módulo DIMM

Los módulos actuales integran una memoria EEPROM denominada SPD (Serial Presence Detect) que almacena los datos del fabricante, la configuración y los tiempos del módulo. Con la incorporación de SPD, un sistema de memoria puede tener módulos de diferentes características.

Los módulos pueden ser registrados o no. En la versión no registrada, el controlador de memoria se conecta directamente con la memoria. En la versión registrada, unos registros retrasan un ciclo de reloj el acceso a memoria para asegurar la captura simultánea de datos de todos los chips de memoria del módulo.

[166] El empaquetado DIP se volvió un problema cuando los usuarios empezaron a instalar su propia RAM, ya que las patillas eran demasiado delicadas. Otro factor muy importante fue que los zócalos para el empaquetado DIP ocupaban mucho espacio en placa.

[167] Este incremento de señales era necesario para implementar los buses de datos de 64 bits.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Los módulos se ajustan a un estándar JEDEC que usa la nomenclatura[168] PCXXXX, donde XXXX se refiere al ancho de banda de pico que soporta la memoria (ver tabla 3.3). Por ejemplo, PC2700 se refiere a una memoria DDR333 que trabaja a 166MHz y que tiene como ancho de banda 166x2x8 = 2700 MBps.

Nomenclatura	Tipo de memoria	Ancho de banda de pico
PC66	SDRAM a 66MHz	528 MBps
PC100	SDRAM a 100MHz	800 MBps
PC133	SDRAM a 133MHz	1,064GBps
PC800	RDRAM a 400MHz	1,6 GBps
PC1600/DDR200	DDR a 100MHz	1,6GBps
PC2100/DDR266	DDR a 133MHz	2,1GBps
PC2700/DDR333	DDR a 166MHz	2,76GBps
PC3200/DDR400	DDR a 200MHz	3,2GBps
PC3200/DDR2-400	DDR2 a 100MHz	3,2GBps
PC4300/DDR2-533	DDR2 a 133MHz	4,3GBps
PC5300/DDR2-667	DDR2 a 166MHz	5,3GBps
PC6400/DDR2-800	DDR2 a 200MHz	6,4GBps

Tabla 3.3: Ejemplos de nomenclatura PCXXXX para algunos módulos de memoria DRAM

Una nomenclatura muy usada en los módulos para definir sus características es $N1 - N2 - N3$ donde[169] $N1 = \lceil \frac{tRP}{tCK} \rceil$, $N2 = \lceil \frac{tRCD}{tCK} \rceil$ y $N3 = \lceil \frac{tCL}{tCK} \rceil$. tCK es el periodo del reloj. En la figura 3.5 se definen estos tiempos.

En cuanto a la capacidad de almacenamiento, en la tabla 3.4 se pueden ver algunos de los valores actuales.

Tipo de memoria	Tipo de módulo	Velocidad(MHz)	Organización	No. bancos	Densidad
RDRAM	RIMM/184	400x2	256Mbx8x2	32/512	512MB
RDRAM	RIMM/184	400x2	128Mbx8x2	256	256MB
DDR2	DIMM/240	133x4	64Mbx8x9	4	512MB
DDR	DIMM/184	133x2	128Mbx8x9	4	1GB
DDR	DIMM/184	133x2	64Mbx8x9	4	512MB
SDRAM	DIMM/168	133	256Mbx8x9	2	2GB
SDRAM	DIMM/168	133	128Mbx8x9	2	1GB
EDO	DIMM/168	66	32Mbx8x9	1	256MB

Tabla 3.4: Características de algunos módulos de memoria DRAM

2.2.5. Detección y corrección de errores de memoria

El sistema de memoria no está libre de errores y éstos pueden obedecer a muchas causas:

[168] En las primeras versiones de SDRAM el significado era distinto. Observe las primeras filas de la tabla 3.3.

[169] $\lceil x \rceil$ es el entero inmediatamente superior a x

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Problemas hardware como malas soldaduras, falsos contactos, chips defectuosos, etc. Son fáciles de diagnosticar y, por tanto, de resolver.

- Ruidos en la placa o en el propio chip, debidos a partículas alfa[170], rayos cósmicos[171], ruido de alta frecuencia de alimentación[172], radiointerferencias[173]. Son difíciles de diagnosticar y, a veces, requieren el rediseño del sistema.

En la tabla 3.5 se puede ver una antigua estadística de IBM sobre la probabilidad de error en sistemas que no usan control de errores.

Entorno	Probabilidad de error
casero/móvil 4MB y 100horas de uso/mes	1 error no detectado cada 39 años
Oficina/desktop 8MB de memoria y 300 horas de uso/mes	1 error no detectado cada 8 años
uso intensivo 16MB y 600 horas/mes	1 error no detectado cada 3 años

Tabla 3.5: Probabilidad de error de memoria. Estudio de IBM del año 1980

Hay varios métodos de detección y corrección de errores de memoria. El más sencillo se basa en añadir por cada byte de datos un bit de paridad. Esta técnica permite detectar errores de bit sencillos y parar el sistema (para evitar daños mayores). Otros métodos más complejos como el ECC (Error Checking and Correcting) chequean y corrigen un bit automáticamente, sin parar el sistema. El sistema se para cuando se detectan 2, 3 ó 4 errores de bits, dependiendo de la configuración. El ECC requiere más memoria que la técnica de paridad. En la tabla 3.6 se puede ver una antigua estadística de IBM que compara el número de fallos de un sistema que usa paridad o ECC.

Cantidad de memoria	Fallos con paridad	Fallos con ECC
8MB	0,16	0,00008
16MB	0,32	0,00016
64MB	1,26	0,00064
128MB	2,52	0,00128

Tabla 3.6: Probabilidad de fallos de un sistema en 5 años usando paridad o ECC. Estadística de IBM del año 1980

[170] El empaquetado de los chips de memoria usa material radioactivo. Fue un grave problema en los 80, aunque hoy en día parece resuelto.

[171] Son inevitables, salvo que blindemos el sistema. La probabilidad de que hagan bailar un bit es muy baja pero mayor que cero. Son un problema en los sistemas ubicados en estaciones espaciales.

[172] Se resuelven con filtros de red.

[173] Se evitan con un diseño cuidadoso del sistema que aisle pistas y evite acoplos.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.3. Memorias volátiles estáticas

En las memorias volátiles estáticas o SRAM (figura 3.20), cada bit se representa por el estado de un biestable que se implementa con 4 o 6 transistores[174].

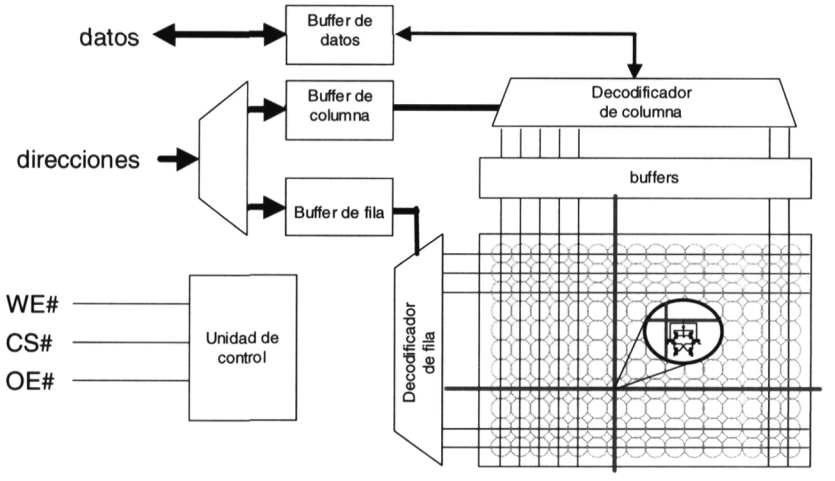


Figura 3.20: Arquitectura de una memoria SRAM

A diferencia de las DRAM, las SRAM tienen un pin por cada dirección de línea. Todo esto hace que las memorias sean más caras y ocupen mucho más espacio que las DRAM. Su ventaja es que son más rápidas y que no pierden la información almacenada mientras no se corte la alimentación (a diferencia de las memorias DRAM que deben ser refrescadas periódicamente).

La evolución de las memorias SRAM ha sido similar a la de las memorias DRAM. La necesidad de más ancho de banda en muchos sistemas ha llevado al desarrollo de arquitecturas síncronas con pipeline y mecanismos para multiplicar la velocidad de acceso.

No obstante, las tradicionales memorias SRAM (ASRAM, Asynchronous SRAM) siguen existiendo para sistemas de bajas prestaciones o bajo consumo.

A continuación, se describen las memorias SRAM mas representativas.

[174] 4 para densidad máxima y 6 para mayor velocidad y menor consumo.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.3.1. Memorias asíncronas

La memoria SRAM asíncrona no requiere ningún tipo de sincronismo. Su interfaz al sistema es muy sencillo. En la figura 3.21 se muestra un típico ciclo de lectura. El ciclo dura un tiempo `tRC` (Read Cycle). En primer lugar, se ponen las direcciones y, a continuación, se activan las señales de habilitación del chip (`CS#`) y de habilitación de salida del bus de datos (`OE#`). Después de un tiempo `tOE` (Output Enable to Valid Output) los datos correspondientes a la dirección aplicada aparecen estables sobre el bus de datos.

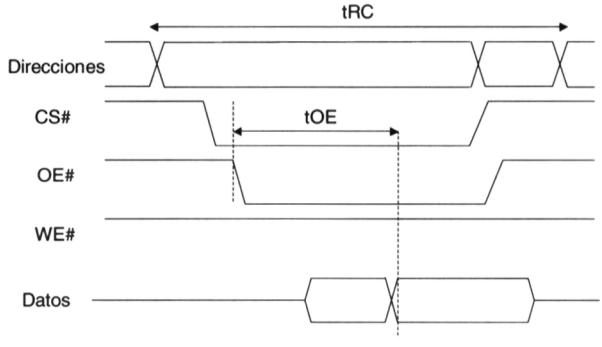


Figura 3.21: Ciclo de lectura de una memoria SRAM asíncrona

La figura 3.22 muestra un ciclo de escritura.

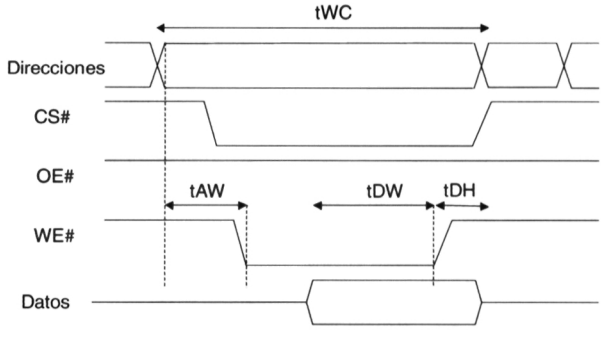


Figura 3.22: Ciclo de escritura de una memoria SRAM asíncrona

El ciclo dura un tiempo `tWC` (Write Cycle). Al igual que en lectura, se aplican las direcciones para posteriormente activar las señales de habilitación del chip (`CS#`) y de habilitación de entrada del bus de datos (`WE#`). A continuación, se aplican los datos a escribir sobre el bus de datos. Estos datos deben

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

permanecer estables cierto tiempo antes (`tDW`: Data to Write Time Overlap) y después (`tDH`: Data Hold from Write Time) de desactivar la señal `WE#`.

Se puede distinguir entre memorias de alta velocidad y memorias de bajo consumo. Las primeras tienen tiempos de acceso del orden de 10 ns y se usan normalmente en buffers de comunicaciones y memorias caches de prestaciones medias. Las memorias ASRAM de bajo consumo son muy útiles en muchos sistemas empotrados de actualidad (por ejemplo, los teléfonos móviles) o como memorias cache de disco; su tiempo de acceso suele ser superior a 50 ns.

2.3.2. Memorias síncronas

Las memorias SRAM síncronas se utilizan en sistemas de altas prestaciones que requieren elevados anchos de banda. Para ello, usan un reloj que sincroniza sus transferencias con el controlador de memoria, reduciendo así tiempos de acceso. Su arquitectura integra registros de entrada que capturan las señales de control, direcciones y datos en los flancos de reloj. También dispone de un contador para realizar transferencias de ráfaga.

En la figura 3.23 se muestran dos ciclos de lectura de una memoria SRAM síncrona: una lectura sencilla y otra en modo ráfaga.

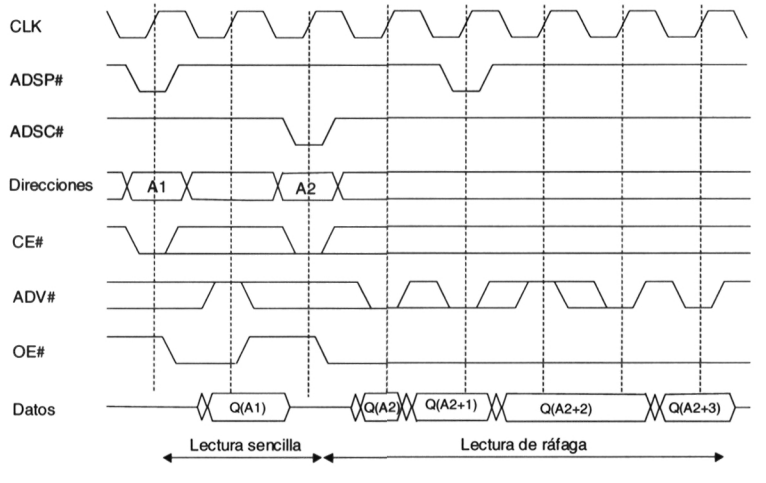


Figura 3.23: Ciclo de lectura de memoria SBSRAM

Las señales `ADSC#` (Synchronous Address Status Controller) y `ADSP#` (Synchronous Address Status Processor) controlan la duración de la ráfaga y el

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

comienzo de una nueva ráfaga[175]. Cuando **ADSP#** se activa, cualquier ráfaga entrante se interrumpe y se ejecuta una lectura usando las nuevas direcciones. De forma similar, cuando **ADSC#** se activa (y **ADSP#** está inactiva) cualquier ráfaga se interrumpe y una lectura o escritura se ejecuta usando la nueva dirección externa. Después del primer ciclo de la ráfaga, la señal **ADV#**(Advance) controla los siguientes ciclos de ráfaga. Cuando **ADV#** se activa, la dirección interna se incrementa antes de la operación. Cuando no está activa, la dirección interna no se incrementa y se inserta un estado de espera.

Se distinguen dos tipos básicos de memorias SRAM síncronas: Memorias flowthrough o SBSRAM (Sync Burst SRAM) y memorias con pipeline o PB-SRAM (Pipeline Burst SRAM).

La memoria PBSRAM se diferencia de la SBSRAM en que incluye un registro de salida para implementar un pipeline de dos etapas. Esto hace que mejore el ancho de banda a costa de una mayor latencia. Las PBSRAM requieren dos ciclos de reloj para un acceso de lectura.

En la tabla 3.7 se puede ver una comparación de prestaciones entre varios tipos de memorias SRAM. Se observa como cada memoria tiene un margen de frecuencias de trabajo en el que supera en prestaciones a las restantes.

CLK bus	33	50	60	66	75	83	100	125
ASRAM	2-1-1-1	3-2-2-2	3-2-2-2	3-2-2-2	3-2-2-2	3-2-2-2	3-2-2-2	3-2-2-2
SBSRAM	2-1-1-1	2-1-1-1	2-1-1-1	2-1-1-1	3-2-2-2	3-2-2-2	3-2-2-2	3-2-2-2
PBSRAM	3-1-1-1	3-1-1-1	3-1-1-1	3-1-1-1	3-1-1-1	3-1-1-1	3-1-1-1	3-1-1-1

Tabla 3.7: Comparativa de prestaciones de memorias SRAM

Una mejora posterior de estas memorias ha sido, a semejanza de las memorias DDR DRAM, el uso de los flancos de subida y bajada del reloj para transferir los datos. Son las memorias DDR SRAM. Esta técnica permite duplicar el ancho de banda de memoria.

2.3.3. Memorias especializadas

La memoria síncrona no puede hacer una escritura inmediatamente después de una lectura ya que se produce una colisión en el bus. Este hecho le obliga a insertar un estado de espera. Esta ineficiencia es intolerable en aplicaciones de red (conmutadores ATM, routers, etc.) donde hay muchas lecturas y escrituras intercaladas. Las memorias NoBL (No bus Latency) o ZBT (Zero Bus Turnaround) son memorias SRAM síncronas que resuelven este problema.

[175] Estas dos señales se refieren al controlador de cache y microprocesador, respectivamente.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Otra memoria especializada, muy útil en sistemas de conmutación de redes, es QDR (Quad Data Rate). Como su nombre indica es capaz de transferir 4 datos en un ciclo de reloj. Para ello implementa dos puertos independientes (uno para lectura y otro para escritura) que trabajan simultáneamente. Además, cada puerto transfiere datos tanto en los flancos de subida como de bajada del reloj.

Las memorias FIFO(First In First Out) son otro tipo de memorias muy utilizadas para ayudar en la temporización de eventos no sincronizados. Hay memorias FIFO asíncronas y síncronas. También hay memorias especializadas para memoria cache como la Cache Tag RAM. Esta memoria integra SRAM y lógica de comparación.

2.4. Ejemplos

2.4.1. Tiempos de acceso en modo ráfaga

Una nomenclatura muy utilizada para representar el tiempo de acceso de una memoria en modo ráfaga es: N_1-N_2-N_3-...-N_B, donde N_i es el número de ciclos de reloj que tarda en hacerse el acceso i, y B es el número de accesos de la ráfaga.

Como ejemplo, para una ráfaga de 4 datos, memorias FPM DRAMs de `tRAC`=60ns y `tCP`=35ns y reloj del bus del sistema de 66MHz, el tiempo de acceso del primer dato se lleva a cabo en 5 ciclos de reloj mientras que los tres restantes sólo necesitan 3:

1. 66MHz $\rightarrow 10^6/66 \approx 15$ ns
2. $T_{primer_acceso_bus}$=15ns/ciclo $\times N_1 >$ `tRAC` = 60ns $\quad N_1 > 4; N_1 = 5$
3. $T_{siguientes_accesos_bus}$=15ns/ciclo $\times N_i >$ `tCP`=35ns $\quad N_i > 2; N_i = 3 \forall i > 1$

Por tanto, el tiempo de acceso en modo ráfaga sería de 5-3-3-3.

Si la memoria hubiera sido EDO DRAMs, el valor de `tCP` sería más pequeño. Para `tRAC`=60ns, `tCP`=25ns y reloj del bus del sistema de 66MHz, se consiguen tiempos de acceso de ráfaga de 5-2-2-2.

2.4.2. Mapeo de memoria

Un módulo DIMM está formado por hileras de chips (ranks). Cada hilera responde a un comando sencillo transfiriendo datos de todos sus chips. Cada chip distribuye internamente su memoria en bancos (por ejemplo, una memoria

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

DDR tiene 4 bancos). Cuando se accede a los bancos se selecciona una fila y una columna. La columna es la mínima cantidad direccionable del chip.

Direcciones	16Mb×16
fila	8K (A0-A12)
banco	4 (BA0-BA1)
columna	512(A0-A8)

Tabla 3.8: Memoria DDR DRAM de 16Mb×16

Sea un módulo de memoria de 4 chips de memoria 16Mb×16. Los cuatro chips permiten configurar un bus de 16×4 = 64 bits. En la tabla 3.8 se muestra la capacidad de direccionamiento del chip para acceso a fila, banco y columna. Si queremos acceder a la dirección `0AC75C38H`, tendremos que acceder a la hilera `1`, banco `1`, fila `0B1DH` y columna `187H`:

```
0AC75C38H = 0000 1010 1100 0111 0101 1100 0011 1000B

                0000 1010 1100 0111 0101 1100 0011 1000B
                  /   \             /  \           \
                 /     \           /    \           \
                | rank  |   fila   |banco |  columna  |
```

2.4.3. Granularidad

La granularidad `G` es la mínima densidad del sistema que es posible obtener para una configuración dada. Se define como:

$$G = \frac{D \times \frac{S_s}{S_m}}{8}(MB) \tag{3.1}$$

donde D es la densidad de la memoria en Mb y S_s y S_m los tamaños de los buses del sistema y de memoria, respectivamente. La tabla 3.9 muestra algunos valores de G para memorias SDRAM y RDDRAM.

Tipo	D	S_m	S_s	G
SDRAM	64Mbits	16 bits	64 bits	32 MBytes
SDRAM	512Mbits	16 bits	64 bits	256 MBytes
RDRAM	512Mbits	16 bits	16 bits	64 MBytes

Tabla 3.9: Ejemplos de granularidad de algunos sistemas

Se puede observar cómo la memoria RDRAM admite menores granularidades para la misma configuración.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.4.4. Ancho de banda efectivo

El ancho de banda efectivo B_e es un valor medio del ancho de banda del sistema. Algunos autores lo definen como:

$$B_e = B_p \times F \tag{3.2}$$

donde F es un factor de utilización que depende de las aplicaciones que corren en el sistema y B_p el ancho de banda de pico. En la tabla 3.10 se dan algunos valores significativos.

Nomenclatura	Tipo de memoria	B_p	F	B_e
PC100	SDRAM a 100MHz	800 MBps	62 %	494 MBps
PC133	SDRAM a 133MHz	1,064GBps	59 %	631 MBps
PC800	RDRAM a 400MHz	1,6 GBps	74 %	1190MBps
PC2100/DDR266	DDR a 133MHz	2,1GBps	42 %	897 MBps

Tabla 3.10: Ejemplos de ancho de banda efectivo. Fuente: Toshiba

2.4.5. Manejo de nomenclatura

La tabla 3.11 muestra valores de la nomenclatura explicada en el apartado 2.2.4

tCL	tRP	tRCD	fCLK	Nomenclatura
20ns	20ns	20ns	100MHz	2-2-2
25ns	20ns	20ns	100MHz	3-2-2
20ns	20ns	20ns	133MHz	4-3-3

Tabla 3.11: Ejemplos de valores de parámetros de la nomenclatura N1-N2-N3 de los módulos de memoria

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3. Memorias magnéticas

La memoria magnética almacena información grabando cambios de flujo magnético sobre una superficie magnética. Podemos distinguir entre unidades de disco duro, unidades de disco flexible, unidades de cinta y memoria semiconductora magnética.

3.1. Discos duros

Los discos duros han sufrido una tremenda evolución desde su aparición (3.24).

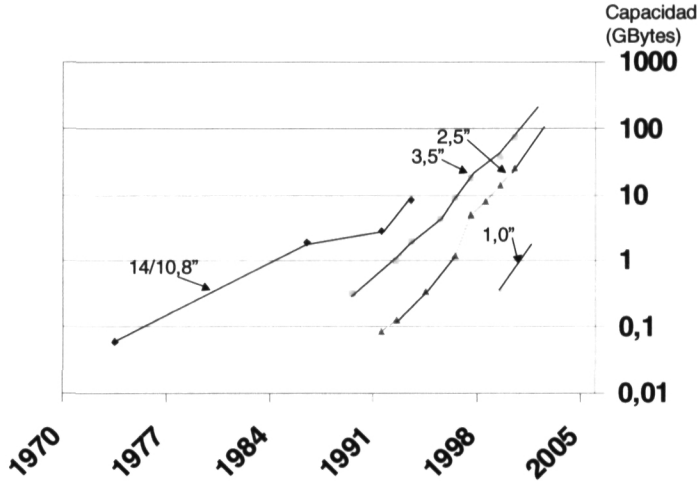


Figura 3.24: Evolución en la capacidad de los discos duros. Fuente: IBM

El primer disco duro fue el RAMAC[176] de IBM (1956) de tan sólo 5MB distribuidos en 50 discos de 24". Otro disco importante, considerado el padre de los discos modernos, fue el famoso disco Winchester de 60MB, también de IBM, del año 1973. Hoy en día, los discos duros han reducido considerablemente su tamaño al mismo tiempo que han incrementado espectacularmente tanto su capacidad como su velocidad de acceso.

Una unidad de disco duro está formada por uno o varios discos[177], cabezales

[176] El IBM 305 RAMAC(Random Access Method of Accounting and Control) tenía una densidad de área de 2Kb/pulgada^2 y un ancho de banda de 8800bps.

[177] de 1 a 5 para PC de sobremesa y hasta 12 para servidores.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

de lectura/escritura , motor y mecanismo de control asociado (ver figura 3.25).

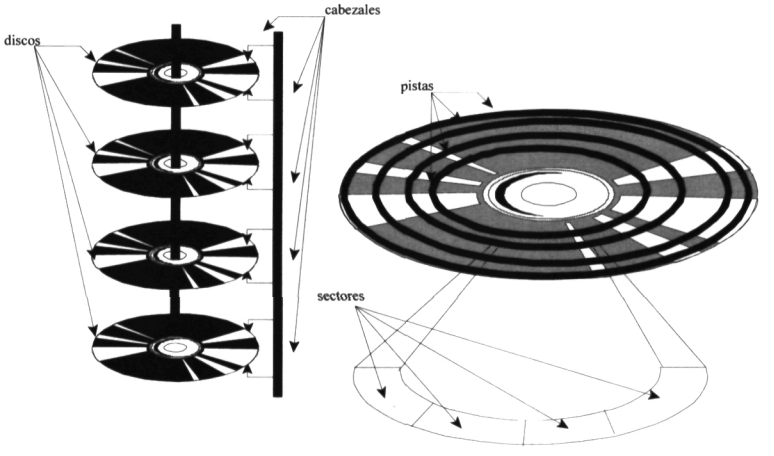


Figura 3.25: Arquitectura de un disco duro

Todos los cabezales, que acceden a las dos caras de los discos, están unidos entre sí. El disco duro o disco winchester está cerrado herméticamente para evitar la contaminación por polvo. Las cabezas de esta unidad tienen forma aerodinámica y flotan en un colchón de aire generado por los platos al girar[178].

El disco es un plato de aluminio, u otro material base (vidrio o cerámica), revestido por ambas caras por un material magnético. Cada cara del disco consiste en un conjunto de anillos concéntricos llamados pistas (tracks). Al conjunto de pistas que ocupan la misma posición en cada disco se le denomina cilindro. Cada pista se divide en bloques llamados sectores. En estos sectores se almacenan los datos (512 bytes) y cierta información de control, de forma codificada. La información se graba como inversiones de flujo magnético.

Para llevar a cabo una transferencia de información, el S.O. consulta la FAT (File Allocation Table) que le indica la dirección física del sector (número de cilindro, de cabezal y de sector). El cambio de pista dentro de un cilindro es rápido, ya que el cambio de cabezal es electrónico. Sin embargo, el cambio de cilindro, obliga a mover los cabezales y esto lleva de 1-3ms entre pistas adyacentes hasta 15ms para ir de un extremo a otro del disco (es el denominado tiempo de búsqueda tB). Además, una vez se llega a la nueva pista hay que esperar a que el sector solicitado se aproxime a ella, antes de empezar la

[178] La figura 3.26 ilustra la importancia de este cierre hermético para el correcto funcionamiento del disco duro.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

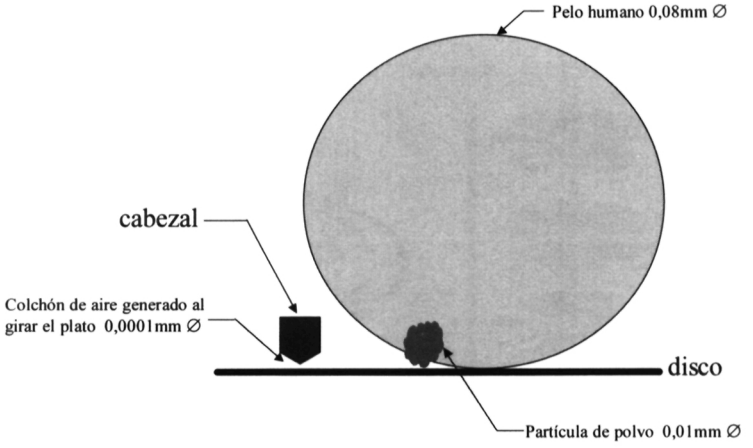


Figura 3.26: Importancia del cierre hermético en un disco duro

transferencia (esto puede llevar de 0s a un tiempo de rotación completo y se denomina latencia de rotación tR). Los discos giran a una velocidad V que oscila entre 5200 y 15000 r.p.m.(revoluciones por minuto). La información se transfiere a un ritmo de una pista por periodo de rotación. Por tanto, el tiempo total de acceso tA será:

$$tA = tB + tR + C/V \tag{3.3}$$

donde C es la cantidad de información transferida.

En la figura 3.27 se puede ver un esquema simplificado de la lectura/escritura en el disco duro. Para escribir en el disco duro, los datos son codificados mediante diversas técnicas (FM, MFM, RLL, etc.). La señal codificada incluye información de reloj para facilitar la recuperación de los datos. A continuación, el flujo de datos codificados se usa para generar una secuencia de pulsos de polaridad alternada que servirá para excitar el cabezal de escritura y conseguir los cambios de flujo magnético sobre la superficie del disco. Cuando se quieren leer datos del disco, hay que amplificar la señal y detectar los pulsos de grabación. Después, un separador de datos decodifica la señal separando los datos del reloj (se requiere un PLL para sincronizar los datos correctamente). Los primeros interfaces a disco duro tenían que implementar tanto el decodificador como el separador (ST-506), sin embargo, actualmente toda esta circuitería forma parte del disco duro.

Hay discos duros extraíbles (ZIP, Jaz, etc.) que pueden incluir la unidad

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

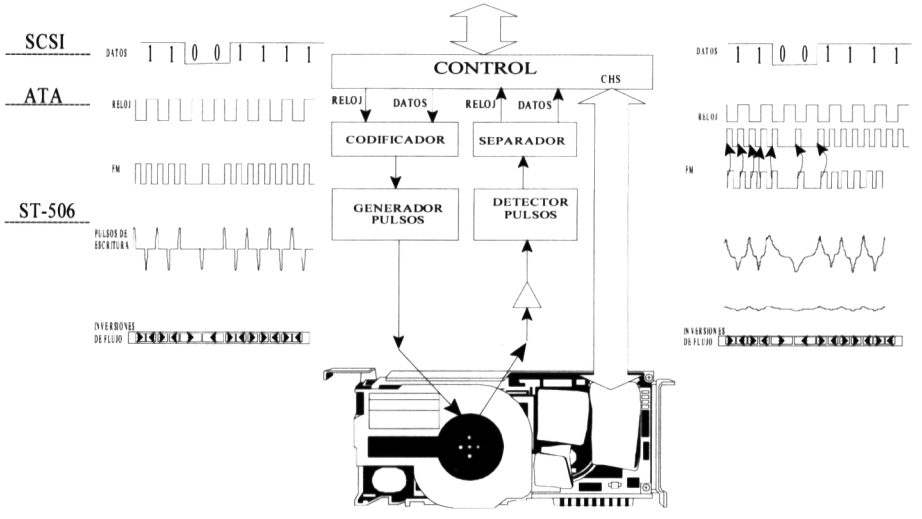


Figura 3.27: Ejemplo simplificado de lectura y escritura en un disco duro

completa o sólo el disco. Su utilidad se centra en la seguridad de la información o para sustituir un disco duro que está fallando.

3.2. Discos flexibles

El disquete o disco flexible fue, en realidad, inventado por IBM para guardar información sobre el mantenimiento de sus computadores, aunque se adoptó muy rápidamente por los fabricantes de ordenadores para distribuir software comercial. Su ventaja esencial es que es transportable.

A diferencia del disco duro las cabezas están en contacto con la superficie, lo que hace que se desgasten más rápido. Para reducir el desgaste, los PCs retiran las cabezas y detienen la rotación cuando el dispositivo no está leyendo o escribiendo. En consecuencia, cuando se da el siguiente comando de lectura/escritura existe un tiempo de espera de alrededor de medio segundo mientras el motor alcanza la velocidad requerida.

Las unidades de disquete pueden ser internas o externas. Los disquetes están siendo desbancados por las memorias Flash con interfaz USB. En la tabla 3.12 se muestra una comparativa de este tipo de disquetes.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Tamaño (pulgadas)	5,25	5,25	3,5	3,5	3,5
capacidad (bytes)	360K 2S,2D	1.2M 2S,HD	720K 2S,2D	1.44M 2S,HD	Floptical 25MB
pistas	40	80	80	80	
sectores/pista	9	15	9	18	27
cabezales	2	2	2	2	
rotaciones/mim	300	360	300	300	720
velocidad (kbps)	250	500	250	500	
tipo	flexible	flexible	rígido	rígido	rígido

Tabla 3.12: Comparativa de disquetes flexibles

3.3. Cintas

La cinta magnética o tape fue el primer tipo de memoria secundaria. Es similar a una cinta de casete y consiste en una cinta que se va desenrollando del carrete de alimentación y se va enrollando en la bobina, pasando por un cabezal de grabación/reproducción. Su acceso secuencial hace que la latencia sea muy elevada. Actualmente se suele usar para realizar copias de seguridad (backup) de los datos almacenados en el disco duro.

Hay varios tipos de cintas. Destacan el cartucho de un cuarto de pulgada (QIC,Quarter-inch Cartridge), la cinta digital de audio (DAT, Digital Audio Tape), la cinta de 8mm y la cinta digital lineal (DLT, Digital Lineal Tape). El QIC es el más barato, tiene el aspecto de un casete de audio, suele almacenar de 80MB a 4GB y tiene una velocidad de 100 pulgadas/s. El DAT permite mejores prestaciones de velocidad y almacenamiento (12 GB) pero es más caro. La cinta de 8mm es similar al DAT pero alcanza los 25GB de capacidad. La DLT puede llegar a los 500GB.

3.4. RAID

Un RAID(Redundant Array of Independent[179] Disks) es un conjunto de unidades de disco organizadas de manera que mejoran las prestaciones y la fiabilidad del sistema de almacenamiento. El sistema ve al RAID como un único disco virtual. Se distinguen 5 niveles RAID correspondientes a 5 organizaciones distintas. Por ejemplo, el nivel 0 (RAID 0) accede a todos los discos en paralelo mejorando sustancialmente la velocidad de acceso. Otros niveles añaden redundancia de datos (RAID 1) o mecanismos de protección adicionales (por ejemplo, RAID 4 usa uno de los discos para almacenar la paridad de los datos).

[179] Inicialmente el acrónimo decía Inexpensive en lugar de Independent, lo que muestra la intención de sus creadores.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.5. Ejemplos

3.5.1. Cálculo de parámetros de un disco duro

Se calcula la capacidad y el tiempo máximo y mínimo que tarda en leer una pista entera un disco duro con las siguientes características:

Características	Valor
Número de cilindros	2000
Número de cabezales	16
Número de sectores	63
Tamaño del sector	512 bytes
Tiempo máximo de búsqueda tB_{max}	20 ms
Velocidad de giro	3600 r.p.m.

La capacidad del disco duro es:

$$Capacidad = C \times H \times S \times 512 = 2000 \times 16 \times 63 \times 512 = 1032192000 bytes \quad (3.4)$$

El tiempo que tarda en leer una pista entera viene dado por la ecuación 3.3 y su valor máximo será:

$$t_{max} = tB_{max} + tR_{max} + tT = 20ms + 16,67ms + 16,67ms = 53,34ms \quad (3.5)$$

en este caso el valor de tR_{max} y tT es el mismo: el inverso de 3600 r.p.m. En cuanto al valor mínimo, suponiendo que no hay que cambiar de pista y que estamos en el sector adecuado antes de comenzar la transferencia, será:

$$t_{min} = tB_{min} + tR_{min} + tT = 0ms + 0ms + 16,67ms = 16,67ms \quad (3.6)$$

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4. Memorias ópticas

En estas unidades la grabación binaria se lleva a cabo mediante pequeños "agujeritos" realizados sobre una superficie plástica que son "leídos" por un láser. El almacenamiento óptico es una tecnología que permite elevadas capacidades sobre un soporte robusto y de bajo coste. Se basa en la misma tecnología que utilizan los discos compactos de audio (CD-DA, Compact Disk Digital Audio).

4.1. CD-DA

La unidad de CD-DA está formada por un rayo láser de infrarrojos de baja potencia, un fotodiodo y el mecanismo de control asociado (ver figura 3.28).

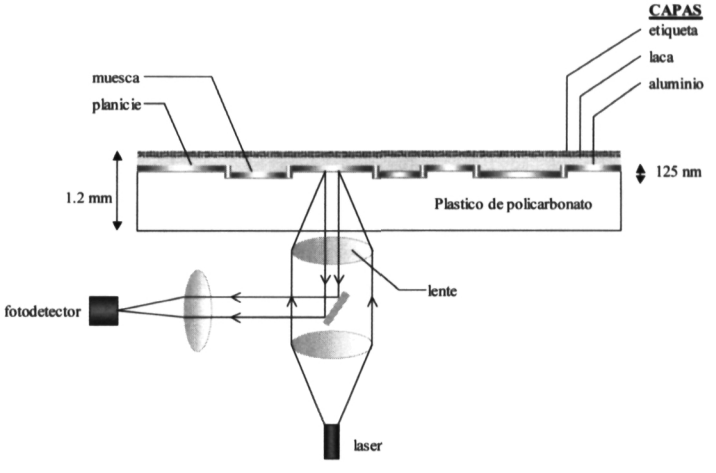


Figura 3.28: Sección de un CD y ejemplo de lectura

El CD-DA es básicamente un disco de plástico con una fina capa de aluminio que hace de espejo reflector para el rayo láser. Los datos se graban en forma de agujeros microscópicos denominados muescas (pits). Un potente láser hace estas muescas sobre un disco matriz. Después se hace un molde que se usa para imprimir copias en discos plásticos, del mismo modo que se elaboraban los discos musicales de vinilo.

Para leer la información, se utiliza un láser de infrarrojos de baja potencia y un fotodiodo. El rayo láser incide sobre las muescas y las áreas sin laserizar (planicies o lands) que tienen diferente reflectividad, lo que hace posible que el fotodiodo distinga entre ambos.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

La forma de almacenar la información difiere bastante de la utilizada en discos duros. Hay una única pista en espiral de 6Km en lugar de anillos concéntricos (ver figura 3.29).

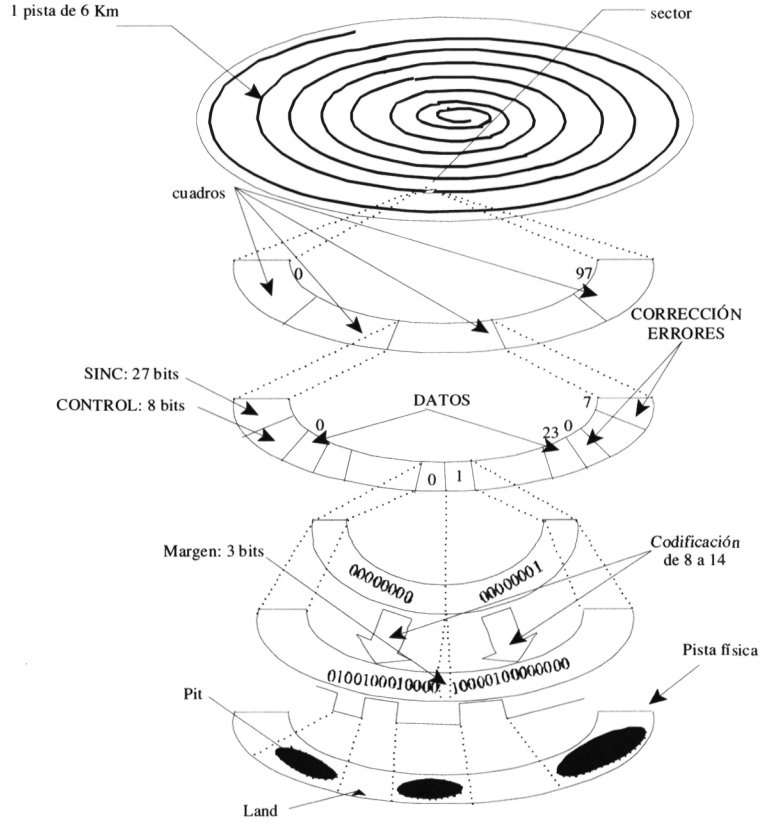


Figura 3.29: Formato de un CD-DA

La pista se divide en sectores de 2352 bytes. El CD-DA permite almacenar 74 minutos[180] de música estéreo de alta fidelidad por lo que su capacidad es de 747MB.

El sector es la mínima unidad direccionable. Cada sector se compone de 98 cuadros. Cada cuadro contiene 24 bytes de datos, 27 bits de sincronismo, un byte de control y 8 bytes de detección y corrección de errores. Los bytes se codifican mediante el código Reed-Solomon (que extiende los 8 bits de 8 a 14+3), obteniendo un flujo de datos listos para su grabación en el CD-DA.

[180]Este curioso número parece proceder de la duración de la novena sinfonía de Beethoven.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Sólo se graban las transiciones de 0 a 1 del flujo de datos.

Los CD-DA tienen una velocidad lineal constante de 75 sectores/s. Esto significa que su velocidad angular debe ser variable (a diferencia de lo que ocurre con las unidades de disco duro) y que, por tanto, el motor de giro de la unidad de CD irá más deprisa cuanto más cerca estén los datos leídos del centro de disco.

La enorme capacidad de los CD-DA no tardó en aprovecharse para el almacenamiento de datos. Esto requería cierta adaptación y nuevos formatos. A continuación se describen algunos de ellos.

4.2. CD para datos

Hay varias especificaciones de los CD-DA para su uso como almacenamiento de datos.

En el libro amarillo[181] se establece un formato similar al del CD-DA pero mejorando la protección contra errores e incluyendo información básica para seleccionar los sectores. Para ello, el campo de datos de 2352 bytes del CD-DA se descompone en 4 nuevos campos (ver figura 3.30): Comienzo de sector (12 bytes), número y modo de sector (4 bytes), campo de datos (2048 bytes para el modo 1 y 2336 para el modo 2) y campo adicional de detección y corrección de errores (288 bytes). Por tanto, la capacidad del CD-ROM queda algo reducida (650MB en el modo 1).

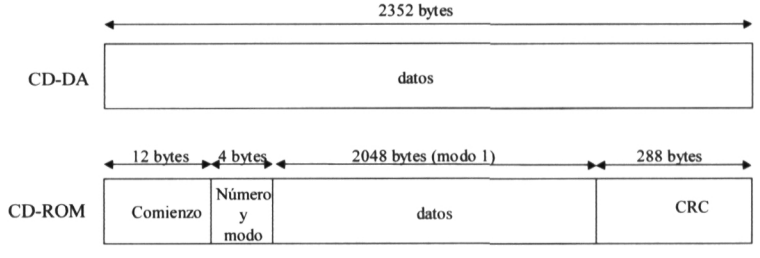


Figura 3.30: Formato de un CD-ROM

Los primeros CD-ROM usaban velocidad lineal constante, lo que requería cambiar constantemente la velocidad del motor conforme el lector se desplazaba entre las pistas. Esta era una de las razones por la que los CD-ROM eran más lentos que las unidades de disco duro, ya que la variación de velocidad

[181] Cada especificación se identifica por el color de la carátula de la documentación correspondiente.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

consume cierto tiempo. Además, es mucho más complicado encontrar un sector a lo largo de una espiral de 6Km que encontrarlo en un conjunto de anillos concéntricos. El tiempo de acceso de las primeras unidades de CD-ROM era de unos 500 ms y su velocidad básica (×1) de 75 sectores/s (150KBps para el modo 1). Las unidades de CD-ROM actuales alcanzan velocidades mucho más elevadas[182] que pueden llegar hasta los 6 MBps (50×).

En el libro verde se define el CD interactivo: un formato que añade gráficos y que permite intercalar audio, video y datos en el mismo sector, característica esencial para las aplicaciones multimedia.

En el libro naranja se especifican varios formatos: CD-R(CD-Recordable), CD RW (CD ReWritable) y CD-MO(CD Magneto-Optical).

En el CD-R, el láser de escritura incide sobre una pista de tinte orgánico que, al quemarse por efecto del láser, cambia de color modificando su reflectividad. Esta alteración permite al lector distinguir entre muescas (zona quemada) y planicies (zona no quemada).

Los CD RW(CD ReWritable) usan un material especial que admite dos estados. Dependiendo de la temperatura a la que se caliente con el láser de escritura, cuando se enfría se vuelve cristalino o amorfo creando así las dos reflectividades que se requieren para grabar la información. Los datos se pueden borrar ya que estos estados son reversibles.

En los CD-MOs, el disco se recubre de una aleación de metales (trebium y gladolinium). A bajas temperaturas estos metales son insensibles a los campos magnéticos, pero a altas su estructura molecular se alinea con cualquier campo magnético. Las cabezas tienen un láser y un magneto. El láser calienta la superficie y el magneto polariza en alguno de los dos posibles sentidos. Con un láser de baja potencia se puede leer los datos. El tiempo de búsqueda y la velocidad de datos es de un orden de magnitud más lenta que los discos winchester[183]. El O-ROM es una versión magneto-óptica de sólo lectura 10 veces más rápida que un CD-ROM convencional.

Hay más especificaciones, como el libro azul que define un formato de CD mejorado o el libro blanco que propone un formato de video (VCD).

[182] Una unidad 24× gira a 5000 r.p.m y consigue 1.8 MBps en las pistas internas y 3.6 MBps en las externas. Una unidad 48x gira a 12000 r.p.m. Estas velocidades tan elevadas aumentan la vibración y el ruido de las unidades, sobre todo en las pistas más externas. Nuevas técnicas hacen uso de unidades más lentas pero con varias lecturas simultáneas o unidades con velocidad angular constante.

[183] Se fabrican en 5,25" (325MB/cara) y 3,5" (128MB) con un tiempo de acceso de unos 30 ms.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4.3. DVD

El DVD (Digital Versatile Disk) es similar al CD-ROM pero con un diseño muy optimizado y una capacidad muy superior. En la tabla 3.13 se muestran algunas de las diferencias más significativas.

TIPO	CD-ROM	DVD
Diámetro del disco (cm)	1.2	1.2
Tamaño de los pits (micras)	0.8	0.4
Espacio entre pistas (micras)	1.6	0.74
Longitud de onda del láser(micras)	0.78	0.65
Velocidad rotacional (x1)(r.p.m.)	200-500	570-1600
Ancho de banda (x1)	1.41 Mbps	11.08Mbps
Corrección de errores	CIRC	RS-PC
Overhead de corrección de errores	23-24 %	13 %
Overhead del formato	252 %	136 %

Tabla 3.13: Comparativa CD y DVD

Además permite grabación por ambas caras y varias capas por cara a las que puede acceder cambiando el enfoque del rayo láser. En la tabla 3.14 se muestran las capacidades que se pueden llegar a alcanzar haciendo uso de esta técnica[184].

Cara/capa	una cara	2 caras
Una capa	4,7GB	9,4GB
Dos capas	8,5GB	17GB

Tabla 3.14: Capacidad de almacenamiento de un DVD

En DVD también se han establecido un conjunto de especificaciones que se describen en los libros A(DVD-ROM), B (DVD-Video), C(DVD-Audio), D(DVD-R) y E(DVD-RAM). Las especificaciones se dividen en tres niveles: 1) Propiedades físicas del disco, 2) Formato lógico y estructura de ficheros y 3) Aplicación (tipos de video, audio, fotos y comandos de navegación y presentación). Todos los formatos de sólo lectura (DVD-ROM, DVD-Video y DVD Audio) comparten los mismos niveles 1 y 2. DVD-ROM no tiene nivel de aplicación.

Como ejemplo, el DVD-Video define en su nivel de aplicación el tipo de decodificador de video, la compresión de audio, los comandos de navegación y algunos ficheros auxiliares. DVD-Video permite distribuir una película doblada en 5 idiomas y subtitulada en más de 30.

[184]Se puede observar que no llega a cuadruplicarse la capacidad. Esto es debido a que los DVD multicapa tienen las muescas un poco más grandes para reducir los errores y diafonías (las capas están demasiado próximas y el disco no es perfectamente plano).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

En la figura 3.31 se muestra la sección de un DVD de doble capa y un ejemplo de lectura.

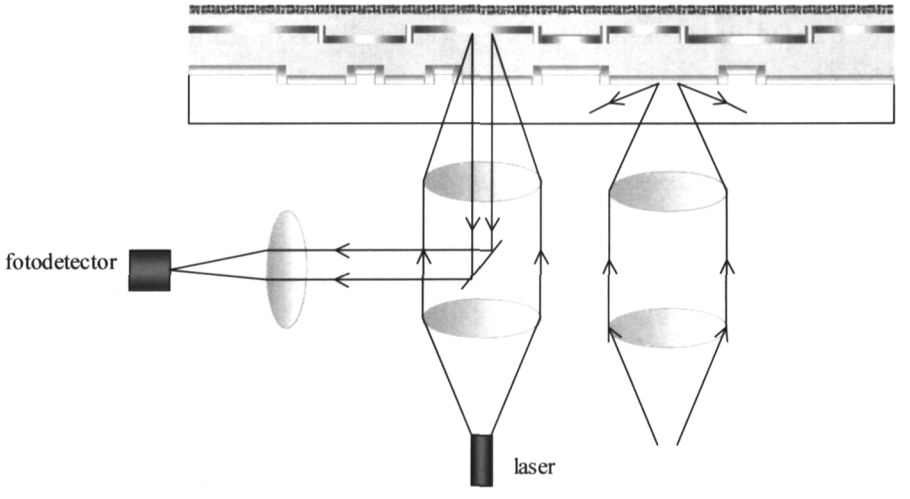


Figura 3.31: Sección de un DVD de doble capa y ejemplo de lectura

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4.4. Ejemplos

4.4.1. Cálculo de capacidad de un CD-DA

El CD-DA es el sustituto de los antiguos discos de vinilo en su formato LP (Long Play).

Considerando un sistema estéreo (2 canales), una duración de 74 minutos y alta alta fidelidad (frecuencia de muestreo[185] de 44.1 KHz y cuantificación de 16 bits), el ancho de banda requerido B será de:

$$B = 44100 muestras/s \times 2 bytes/muestra \times 2 canales = 176400 bytes/s \quad (3.7)$$

Por tanto, la capacidad requerida será:

$$C = 176400 bytes/s \times 74 minutos \times 60 s/minuto = 783216000 bytes \quad (3.8)$$

que son:

$$\frac{783216000 bytes}{1048576 bytes/Mbyte} \approx 747 MB \quad (3.9)$$

Otro dato de interés que se deduce de estos cálculos es la velocidad de lectura del CD-DA. Como el tamaño de un sector es de 2352 bytes, la velocidad en sectores/s es de:

$$\frac{176400 bytes/s}{2352 bytes/sector} = 75 sectores/s \quad (3.10)$$

[185] El margen de frecuencias que escucha el oído humano está comprendido entre 20Hz - 20KHz. Esa es la razón por la que la frecuencia de muestreo para sonido de alta fidelidad sea de $f_m \approx 44$KHz.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5. Últimas tendencias

A pesar de los grandes esfuerzos de los fabricantes, la memoria primaria de los sistemas de propósito general sigue sin alcanzar la velocidad del procesador. Nuevas tecnologías como MRAM (Magneto resistive RAM), podrían modificar esta tendencia. MRAM es una memoria no volátil de lectura /escritura de alta velocidad e integración que podría estar disponible en pocos años.

Un nuevo mercado de sistemas empotrados hace uso intensivo de memorias no volátiles. Esto ha llevado al desarrollo de memorias Flash, PSRAM (Pseudo SRAM) y SRAM de bajo consumo,

En cuanto a la memoria secundaria, los discos duros parecen superar el límite superparamagnético[186] de 20-40 Gb/pulgadas^2. Se supone que la memoria magnética seguirá dominando en los sistemas que requieran capacidades muy elevadas con bajo coste por Mbit. Sin embargo, aplicaciones con otros condicionantes como robustez, consumo y tamaño, serán terreno abonado para las memorias Flash NAND. Una CompactFlash basada en Flash NAND está proporcionando capacidades mayores de 1GByte.

En cuanto a las memorias ópticas, también se incrementará su capacidad. Ya hay varias propuestas para el DVD de alta definición (HD-DVD-9, AOD, Blu-Ray, etc.) que plantean mejoras en capacidad y codificación. A más largo plazo, se prevén memorias semiconductoras ópticas (OUM, Ovonic Unified Memory) y memorias holográficas.

[186] Cuando las regiones magnéticas son demasiado pequeñas no pueden retener la orientación del flujo y, en consecuencia, el almacenamiento de los datos.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

6. Bibliografía

En esta bibliografía recomendada se proponen algunos libros, artículos de interés, guías y, sobre todo, direcciones de fabricantes de memorias y organismos de estandarización. En las páginas de los fabricantes hay infinidad de artículos (white papers) y notas de aplicación sobre todos y cada uno de los aspectos aquí tratados. Además, la visita a las páginas de los fabricantes es obligada para estar actualizado y conocer las novedades.

Libros

Floyd, T.L.: Fundamentos de Sistemas Digitales 7ª Ed., Prentice Hall. 2000. ISBN: 84-205-2994-X.

Sharma, A.K.: Advanced Semiconductor Memories: Architectures, Designs and Applications, IEEE Press Wiley Interscience, New York, 2003.

Mazidi, M.A. y Mazidi, J.G.: The 80x86 IBM PC and compatible Computers, Vol. II: Design and Interfacing of the IBM PC and compatible Computers, Prentice Hall Career&Technology, 1998. ISBN: 0-13-758491-1.

Prince, B.: High Performance Memories: New Architecture DRAMs and SRAMs, Evolution and Function, John Wiley&Sons, 1996.

Sharma, A.K.: Semiconductor Memories: Technology, Testing and Realibility, IEEE Press, 1997.

Taylor, J.: DVD Demystified, McGraw Hill, New York, 1997

Artículos

Dipert, B.: SRAM strive to specialize, EDN, November 1998.

Parker, D.: Defining DVD, IEEE Multimedia, January-March 1999.

Revistas

IEEE Multimedia

IEEE Computer

EDN. www.edn.com

Emedia Professional www.emediapro.net

Algunas páginas WWW

Guías

Guías Pcguide. www.pcguide.com

PC Webopaedia. www.pcwebopedia.com

Tom´hardware Guide. www2.tomhardware.com

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

www.pctechguide.com

Organismos

Solid State Technology Association (JEDEC,Joint Electron Device Engineering Council), www.jedec.org

CFA (CompactFlash Association), www.compactflash.org

Forum DVD www.dvdforum.org

Fabricantes de memoria semiconductora

Fujitsu. www.fujitsu.com

Hitachi. www.hitachi.com

IBM Microelectronics. www.chips.ibm.com

Intel. ww.intel.com

Micron Technology. www.micron.com

NEC. www.nec.com

Rambus. www.rambus.com

Samsung. www.samsung.com

Texas Instrument.www.ti.com

Toshiba. www.toshiba.com

Fabricantes de memoria masiva

Maxtor. www.maxtor.com

Seagate Technology. www.seagate.com

Quantum. www.quantum.com

IBM. www.storage.ibm.com

LSI Logic Corporation. www.lsilogic.com

Verbatim. www.verbatimcorp.com

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

7. Guía de estudio

1. Distinguir entre memoria cache, memoria primaria, memoria secundaria y buffer de video.
2. Describir el tipo de memorias que contiene un ordenador personal, un teléfono móvil y un microcontrolador.
3. Distinguir entre memorias ROM, PROM, EPROM, EEPROM y Flash.
4. Explicar diferencias entre Flash NOR y Flash NAND.
5. Distinguir entre SRAM y DRAM.
6. Enumerar las técnicas que han permitido aliviar el desequilibrio entre el ancho de banda del procesador y de la memoria.
7. Explicar el concepto de precarga en una DRAM.
8. Explicar cómo afectan las técnicas de pipeline, modo ráfaga e interleaving en las prestaciones de las memorias semiconductoras (latencia, ancho de banda de pico y ancho de banda efectivo).
9. Reconocer un cronograma de lectura de una FPM DRAM, EDO DRAM, BEDO DRAM, SDRAM, DDR DRAM y RDRAM.
10. Explicar las diferencias entre SDRAM, DDR DRAM y DDR2 SDRAM.
11. Explicar las diferencias entre DDR DRAM y RDRAM.
12. Explicar las diferencias entre ASRAM, SBSRAM, PBSRAM, ZBT y QDR.
13. Calcular los tiempos de acceso en modo ráfaga de una memoria DRAM (ver apartado 2.4).
14. Explicar la función de los registros y la memoria EEPROM (SPD) de los módulos de memoria DIMM.
15. Explicar diferencias entre las técnicas de detección y corrección de errores de paridad y ECC.
16. Calcular un mapeo de memoria (ver ejemplo del apartado 2.4).
17. Describir de qué depende el tiempo de acceso a un disco duro.
18. Definir los conceptos de cilindro, cabezal y sector de un disco duro.
19. Enumerar diferencias entre un disco duro y un CD-ROM (arquitectura, formato, velocidad, capacidad, latencia, etc.).
20. Enumerar diferencias entre un CD-ROM y un DVD.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4 Entrada/Salida

1. Introducción

La Unidad de Entrada/Salida (E/S) permite conectar el sistema con el mundo exterior. La entrada al sistema debe ser transformada para convertirse en una señal digital compatible con el sistema y codificada de forma que pueda ser interpretada por el programa; esto requiere adaptadores más o menos complejos dependiendo de la naturaleza de los periféricos: desde un simple conmutador hasta un complejo interfaz de reconocimiento de voz[187]. Con la salida ocurre algo parecido, en ocasiones casi no se requerirá adaptación (por ejemplo, para encender un LED), mientras que en otros casos la adaptación puede llegar a ser más compleja que el propio sistema (por ejemplo, crear una imagen tridimensional interactiva en la pantalla del ordenador). Además, el número de periféricos que puede llegar a gestionar un sistema de propósito general puede ser muy grande lo que obligará a usar buses de E/S y controladores que liberen al microprocesador de tareas rutinarias.

En este capítulo se estudian controladores, puertos y buses de E/S y un nutrido conjunto de ejemplos de interfaces a periféricos de ordenadores personales.

[187] *El interfaz de reconocimiento de voz requiere acondicionadores, conversores y procesadores de señal, detectores de actividad, extractores de características y reconocedores de patrones lingüísticos.*

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2. Controladores de E/S

En este apartado se describen dos controladores básicos de E/S: El controlador de interrupciones 8259 y el controlador de DMA 8237, ambos de Intel[188].

2.1. Controlador de interrupciones 8259 de Intel

El controlador programable de interrupciones 8259 de Intel puede manejar hasta 8 interrupciones vectorizadas. Además, puede conectarse en cascada a otros 8259 (hasta un máximo de 8). El 8259 está especialmente diseñado para los microprocesadores 8080/8085 y 8086 de Intel. La figura 4.1 muestra su diagrama de bloques.

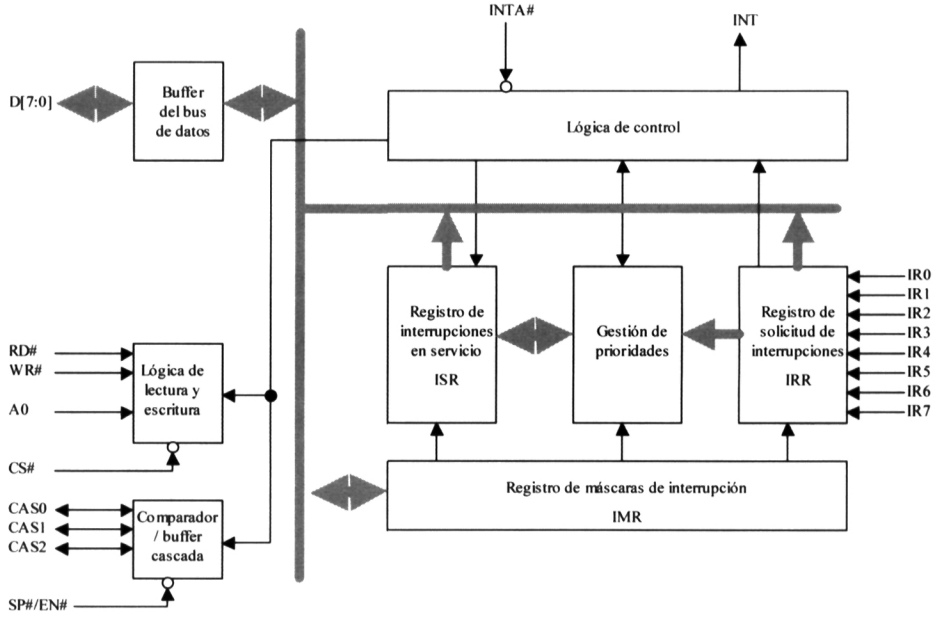


Figura 4.1: Diagrama de bloques del controlador de interrupciones 8259 de Intel

A la derecha están las 8 entradas de solicitud de interrupción (`IR[7:0]`). El resto de señales implementan el interfaz con el microprocesador y otros controladores.

[188]La información que se proporciona es sólo una introducción a estos dispositivos y no pretende ser una guía completa para su programación.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

El interfaz al microprocesador está formado por:

- El bus de datos `D[7:0]`.
- El interfaz de E/S: línea de direcciones `A0` para acceso a los registros del 8259, señales de lectura `RD#` y escritura `WR#` y señal de selección de chip `CS#`.
- El interfaz de control que añade dos señales más: `INT` e `INTA#`. `INT` es la salida de solicitud de interrupción hacia el microprocesador e `INTA#` la entrada de reconocimiento de interrupción procedente del microprocesador.
- El interfaz a otros controladores. Hay un buffer de cascada/comparador para sistemas que hagan uso de varios 8259 conectados en cascada. En esta configuración hay un 8259 que actúa como maestro y el resto actúan como esclavos. El maestro selecciona a los esclavos a través de las líneas `CAS[2:0]`. `SP#/EN#` tiene dos funciones: En modo buffered mode (`EN#`) habilita los buffers del bus y en el modo normal (`SP#`) indica si el 8259 es maestro o esclavo.

Se distinguen tres registros: `IRR` (Interrupt Request Register) que almacena todas las peticiones de interrupción `IR[7:0]` pendientes, `ISR` (In Service Register) que almacena todas las interrupciones que están siendo atendidas en un momento dado e `IMR` (Interrupt Mask Register) que permite enmascarar la interrupciones. También hay una lógica de gestión de prioridad de interrupciones que determina qué interrupción, de las solicitadas en el `IRR`, debe ser atendida primero.

El funcionamiento básico del 8259 es el siguiente:

1. Un periférico solicita una interrupción activando la línea `IR` a la que está conectado, lo que hace que se ponga a 1 el correspondiente bit del `IRR`. Si se producen varias solicitudes simultáneas se activarán varios bits de `IRR`.
2. El 8259 evalúa las solicitudes y, si procede, solicita una interrupción al microprocesador a través de su línea `INT`.
3. Cuando el microprocesador reconoce la interrupción, genera un primer ciclo de reconocimiento de interrupción que activa la señal de entrada `INTA#`.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4. Cuando el 8259 detecta la activación de la señal **INTA#**, activa el bit correspondiente a la interrupción de mayor prioridad (la que va a ser procesada) en el **ISR** y lo borra del **IRR**[189].

5. Cuando el microprocesador genera el segundo ciclo de reconocimiento de interrupción, activando de nuevo la señal **INTA#**, el 8259 envía al microprocesador, a través de su bus de datos, el número de vector de interrupción correspondiente al periférico que solicitó servicio.

6. Cuando termina la rutina de servicio de interrupción, el microprocesador envía el comando EOI(End Of Interrupt)[190].

El 8259 acepta dos tipos de comandos: ICW (Inicialization Command Word) y OCW (Operation Command Word)(ver tabla 4.1). Para inicializar el 8259 hay que enviarle varios comandos ICW consecutivos[191]. Una vez inicializado, el 8259 está listo para procesar las interrupciones que se produzcan.

Dirección	Acceso	Función
0	escritura	Initialization Command Word 1 (ICW1)
	escritura	Operation Command Word 2 (OCW2)
	escritura	Operation Command Word 3 (OCW3)
	lectura	Read Interrupt Request Register (IRR)
	lectura	In-Service Register (ISR)
1	escritura	Initialization Command Word 2(ICW2)
	escritura	Initialization Command Word 3 (ICW3)
	escritura	Initialization Command Word 4 (ICW4)
	lectura/escritura	Interrupt Mask Register (IMR)

Tabla 4.1: Registros del controlador de interrupciones 8259 de Intel

El 8259 tiene cuatro modos de trabajo:

- Completamente anidado, donde las prioridades de las interrupciones son fijas (**IR0** es la de mayor prioridad, le sigue **IR1**, y así sucesivamente hasta **IR7**).

- Prioridad rotatoria, útil para sistemas donde todos los periféricos tienen la misma prioridad.

[189] Mientras el bit del **ISR** esté activo, todas las interrupciones de igual o menor prioridad que llegan permanecen inhibidas; sin embargo, las de mayor prioridad pueden interrumpir. Si la petición de interrupción ya no está presente, el 8259 envía una interrupción de nivel 7.

[190] En el modo AEOI del 8259, el bit de la interrupción en el **ISR** se borra al terminar el segundo pulso **INTA#**; en caso contrario, permanece activo hasta que el microprocesador envía el comando EOI.

[191] Son 2 ó 4 bytes sincronizados por pulsos de la señal de escritura **WR#**. Observe que el 8259 sólo tiene dos direcciones de E/S para acceder a sus registros.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Máscara especial, que permite a las rutinas de servicio de interrupción alterar dinámicamente la estructura de prioridades del sistema durante su ejecución bajo control software.
- Sondeo, que no usa la salida `INT` sino un comando (POLL) para sondear las solicitudes de interrupción.

2.2. Controlador de DMA 8237 de Intel

El 8237 de Intel es un *controlador programable de acceso directo a memoria* con 4 canales independientes y posibilidades de expansión a cualquier número de canales mediante la conexión de otros 8237 en cascada. Tiene registros de 16 bits que le permiten manejar un espacio de 2^{16}direcciones. Admite transferencias memoria-memoria y alcanza anchos de banda de 1.6MBps para frecuencias de reloj de 5MHz. La figura 4.2 y la tabla 4.2 muestran el diagrama de bloques y la descripción de señales del controlador, respectivamente.

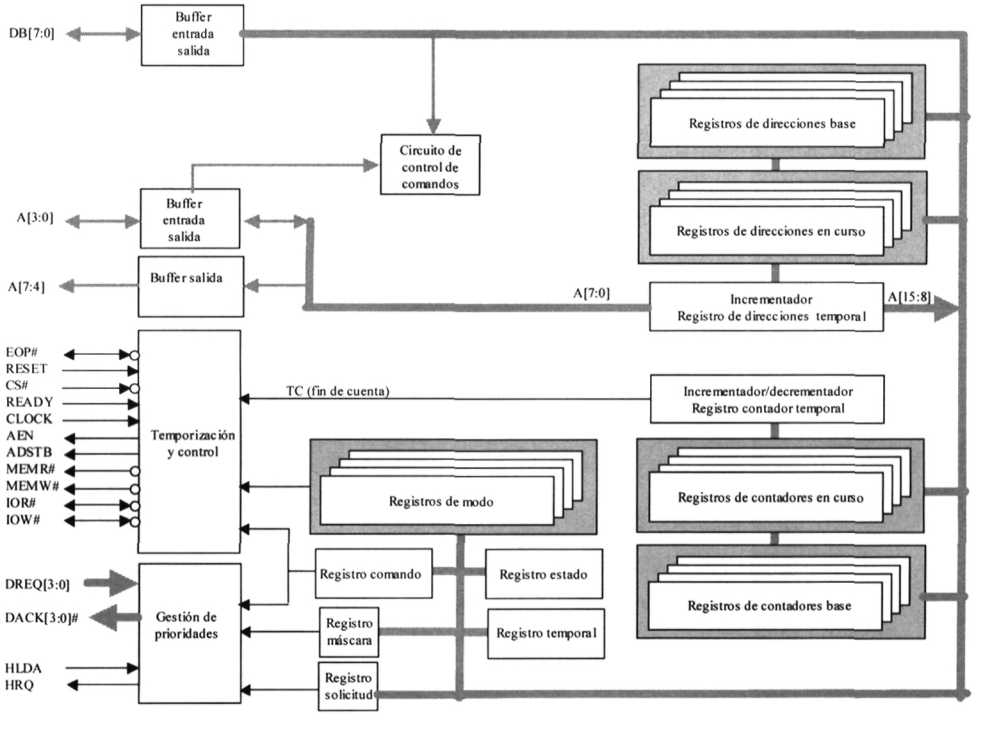


Figura 4.2: Diagrama de bloques del controlador de DMA 8237 de Intel

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Señal	Descripción	dirección
EOP#	End Of Process. Indica, como salida, que ha terminado la transferencia de un canal. Un dispositivo externo, puede usarla como entrada para terminar la transferencia	bidireccional
RESET	Señal de reinicialización	entrada
CS#	Chip select	entrada
READY	Señal de READY para insertar estados de espera en el ciclo de bus	entrada
CLK	Reloj de 5MHz	entrada
AEN	Address Enable. Habilita el latch de 8 bits que guarda la parte alta de la dirección	salida
ADSTB	Address Strobe. Indica que la parte alta de la dirección puede ser capturada en el latch externo	salida
MEMR#	Memory Read. Señal de control de lectura a memoria	salida
MEMW#	Memory Write. Señal de control de escritura a memoria	salida
IOR#	I/O Read. Señal de control de lectura de E/S. En el ciclo inactivo lee los registros del controlador(entrada). En el ciclo activo lee los puertos de E/S del periférico (salida)	bidireccional
IOW#	I/O Write. Señal de control de escritura de E/S. En el ciclo inactivo escribe en los registros del controlador(entrada). En el ciclo activo escribe en los puertos de E/S del periférico (salida)	bidireccional
DREQ[3:0]	DMA Request. Solicitudes de DMA. Son señales asíncronas con polaridad programable	entrada
HLDA	Hold Acknowledge. El microprocesador indica que cede el bus al controlador de DMA	entrada
HRQ	Hold Request. Solicitud de DMA al microprocesador	salida
DACK[3:0]	DMA Acknowledge. Indica al periférico de E/S que su petición ha sido atendida	salida
DB[7:0]	Bus de datos y líneas de direcciones altas (A[15:8])	bidireccional
A[7:0]	Líneas de direcciones. En el ciclo inactivo A[3:0] son líneas de direcciones de los registros del 8237 (entrada). En el ciclo activo, A[7:0] son líneas de direcciones de memoria (salida)	bidireccional

Tabla 4.2: Señales del controlador de DMA 8237 de Intel

El 8237 tiene 344 bits de memoria interna en forma de registros: Registros de direcciones, registros contadores, registro de estado, de modo, de comando, de máscara, de solicitud y temporal. Hay tres bloques básicos de lógica de control: El bloque de temporización y control que genera la mayoría de las señales de interfaz, el bloque de control de comandos y el bloque de gestión de prioridades, que resuelve los conflictos de prioridad cuando varios canales de DMA son accedidos a la vez y que genera las señales de interfaz específicas del controlador: solicitudes y reconocimientos de DMA. La tabla 4.3 muestra el conjunto de registros y comandos del controlador.

La operación de DMA tiene dos fases de trabajo llamadas: ciclo inactivo y ciclo activo.

- En el ciclo inactivo el controlador puede ser programado por el microprocesador a través del interfaz `A[3:0]`, `D[7:0]`, `IOR#` e `IOW#`. Como el bus de datos es de sólo 8 bits, para poder acceder a los registros de 16 bits, se hace uso de un flip-flop interno que determina si el acceso es al byte alto o al byte bajo del registro. Una vez programado, el controlador DMA permanece inactivo a la espera de una petición DMA (por software o una solicitud `DRQx` del canal `x`).

- Cuando llega la petición DMA, el controlador solicita el bus al microprocesador a través de la línea `HRQ` y entra en ciclo activo. Cuando el microprocesador responde activando la señal `HLDA`, el controlador pasa al

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

A[3:0]	Comando o registros	Acceso	Descripción
00	Registro dirección canal 0	lectura y escritura	Almacena la dirección de memoria de la transferencia DMA. Su contenido se incrementa/decrementa después de cada transferencia. Su valor inicial se copia en el registro de direcciones base. En modo autoinicialización, al terminar la transferencia el valor del registro de direcciones base se carga en el registro de direcciones en curso de forma automática.
01	Registro de cuenta canal 0	lectura y escritura	Almacena el número de bytes a transferir en la operación menos uno. Después de cada transferencia se decrementa. Cuando se desborda (paso de 0 a 0FFFFH) se activa la señal TC (Terminal Count). Su valor inicial se copia en el registro contador base. En modo autoinicialización, al terminar la transferencia el valor del registro de cuenta se carga en los registros de cuenta en curso de forma automática.
02	Registro dirección canal 1	lectura y escritura	análogo a canal 0.
03	Registro de cuenta canal 1	lectura y escritura	análogo a canal 0.
04	Registro dirección canal 2	lectura y escritura	análogo a canal 0.
05	Registro de cuenta canal 2	lectura y escritura	análogo a canal 0.
06	Registro dirección canal 3	lectura y escritura	análogo a canal 0.
07	Registro de cuenta canal 3	lectura y escritura	análogo a canal 0.
08	Registro de estado	lectura	Los bits 0,1,2 y 3 señalan si los canales 0,1, 2 y 3 respectivamente, han terminado la transferencia (activación de la señal TC (Terminal Count)) o si se les ha aplicado una señal EOP# externa. Los bits 4,5,6 y 7 los canales que están solicitando servicio (0,1,2 y 3 respectivamente), independientemente de que estén enmascarados o no.
08	Registro de comando	escritura	
09	Registro de solicitud de DMA	escritura	Es un registro con tres bits activos. Los dos bits menos significativos seleccionan un canal y el bit 3 activa/desactiva la petición software.
0A	Registro de comando	lectura	Controla la operación del 8237: polaridad de señal DREQ y DACK, tipo de prioridad, transferencia memoria a memoria, etc.
0A	Single Mask Bit	escritura	Escribe un bit de máscara de DMA. Con este comando se puede seleccionar el bit de máscara que se desea modificar (activándolo o borrándolo.).
0B	Registro de modo	escritura	ver tabla 4.4.
0C	Clear Byte Pointer F/F	escritura	Borra el flip-flop para asegurar que el primer byte enviado a un registro de 16 bits será el menos significativo y, el segundo, el más significativo.
0C	Set Byte Pointer F/F	lectura	Similar al anterior. En este caso, el primer byte enviado será el más significativo y, el segundo, el menos significativo.
0D	Registro temporal	lectura	Se emplea para contener los bytes que se transfieren en las operaciones memoria-memoria.
0D	Master Clear	escritura	Comando que reinicializa los registros
0E	Clear Mode Register Counter	lectura	Comando que limpia el contador de registro de modo.
0E	Registro de habilitación de máscara	escritura	Es un registro con tres bits activos. Los dos bits menos significativos seleccionan un canal y el bit 3 habilita/deshabilita el canal. Ver tabla 4.5.
0F	Read/Write All Mask bits	escritura	Comando que lee/escribe todos los bits de máscara de DMA. Permite consultar o establecer el estado de todos los bits de máscara de DMA a la vez en los 4 canales.

Tabla 4.3: Registros y comandos del controlador de DMA 8237 de Intel

ciclo activo. A continuación, el controlador activa la señal **DACKx** correspondiente para indicarle al periférico **x** que se ha reconocido la solicitud y comienza la transferencia. Las transferencias pueden ser de lectura (de memoria a puerto de E/S, activando **IOW#** y **MEMR#**), de escritura (de

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

puerto de E/S a memoria, activando IOR# y MEMW#) y de verificación[192].

Hay cuatro modos de trabajo:

- Modo de transferencia única (single) que sólo realiza una transferencia. El registro contador de palabras se decrementa y el de direcciones se incrementa/decrementa según haya sido programado. Cuando el registro contador de palabras se desborda (pasa de 0 a 0FFFFH) se activa el bit Terminal Count (fin de cuenta) en el registro de estado y la señal EOP# genera un pulso.

- Modo de transferencia de bloque que transfiere palabras hasta que el registro contador de palabras se desborda. La señal DREQx sólo tiene que estar activada hasta que se active la señal DACKx.

- Modo de transferencia por demanda. Similar al anterior, con la particularidad de que la transferencia sólo se realiza si DREQx está activo.

- Modo cascada. Permite expandir el número de canales de DMA conectando varios 8253 en cascada.

La transferencia puede configurarse en modo de autoinicialización. En este caso, cuando la transferencia termina, los registros de direcciones y contadores se recargan con los valores iniciales para arrancar otra transferencia (cuando se solicite a través de la línea DREQx o por software).

Hay dos tipos de prioridades:

- Prioridad fija donde el canal 0 el más prioritario, seguido del 1 y así sucesivamente hasta el 3.

- Prioridad rotatoria donde el último canal servido pasa a tener la menor prioridad y el que le sigue la máxima.

[192] Es importante recordar que la transferencia es directa entre periféricos de E/S y memoria. Por eso, las líneas IOR# y MEMW#, o IOW# y MEMR#, son activadas al mismo tiempo.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Bits/Valor	Función
Bits 7:6	**bits de selección de modo**
00	Modo demanda
01	Modo single
10	Modo bloque
11	Modo cascada
Bit 5	**bit de incremento/decremento de direcciones**
1	Decremento de direcciones
0	Incremento de direcciones
Bit 4	**bit de habilitación de autoinicialización**
1	DMA autoinicializado
0	DMA de ciclo single
Bits 3:2	**Bits de transferencia**
00	Transferencia de verificación
01	Transferencia de escritura (a memoria)
10	Transferencia de lectura (de memoria)
11	Ilegal
**	Ignorado si bits bits 7:6 = 11
Bits 1:0	**bits de selección de canal**
00	canal 0
01	canal 1
10	canal 2
11	canal 3

Tabla 4.4: Registro de modo del controlador de DMA 8237 de Intel

Bits/Valor	Función
Bits 7:3	**No usados (puestos a cero)**
Bit 2	**Pone/quita máscara**
1	Pone máscara (deshabilita canal)
0	Quita máscara (habilita canal)
Bits 1:0	**bits de selección de canal**
00	canal 0
01	canal 1
10	canal 2
11	canal 3

Tabla 4.5: Registro de máscara del controlador de DMA 8237 de Intel

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.3. Ejemplos

2.3.1. Acceso a puertos de E/S

El depurador del PC nos permite leer y escribir fácilmente en los puertos de E/S del PC. Por ejemplo, para leer el valor del puerto de E/S de dirección `378H` se ejecuta el comando `i`:

```
c:\>debug
- i 378
AA
-
```

y para escribir el comando `o`:

```
- o 378 FF
```

2.3.2. Programando una RSI

El código que se muestra a continuación es una plantilla de programa[193] que ilustra como implementar una nueva rutina de servicio de interrupción (RSI) llamada `mi_rutina` que hace uso de la línea de interrupción hardware 3 (vector de interrupción `0BH`):

```
#include <dos.h>
#define INTNO 0x0B
void interrupt(*oldhandler)();
void main(void){
 oldhandler=getvect(INTNO);
 setvect(INTNO,mi_rutina);
 outportb(0x21,(inportb(0x21) & 0xF7));
 /*Habilita al periférico para generar interrupciones*/
 /*Cuerpo del programa principal*/
 /*Deshabilita al periférico para generar interrupciones*/
 outportb(0x21,(inportb(0x21) | 0x08));
 setvect(INTNO, oldhandler);
}
```

Lo primero que hace el programa es salvar el vector de interrupción `BH` correspondiente a la interrupción hardware 3. Para ello usa la sentencia `olhandler`

[193] Procede de *Using interrupts V1.0* de Craig Peacock. www.beyondlogic.org

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

= getvect(INTNO); donde INTNO es el número de vector de interrupción (BH). Antes de que oldhandler pueda ser usado hay que declararlo usando: void interrupt (*oldhandler)();

Una vez el antiguo vector de interrupción ha sido guardado, procedemos a instalar el nuevo vector de interrupción correspondiente a la nueva RSI. Para ello se ejecuta la sentencia: setvect(INTNO, mi_rutina);.

A continuación se habilita la interrupción en el controlador de interrupciones poniendo a 0 el bit 3 del puerto de E/S de dirección 21H. Se hace mediante la sentencia: outportb(0x21,(inportb(0x21) & 0xF7));

Seguidamente, hay que habilitar la generación de interrupciones en el periférico que va a interrumpir. El cuerpo del programa normalmente consiste de un bucle infinito de tareas rutinarias: Chequear actualizaciones de reloj, entradas de teclado, ratón, puerto serie, etc.

Antes de salir del programa hay que poner a 1 el bit 3 del puerto 21H (outportb(0x21,(inportb(0x21) | 0x08));) y restablecer el antiguo vector de interrupción para evitar que el ordenador se vuelva inestable. Esto se hace usando setvect(INTNO, oldhandler);

A continuación, se muestra el código correspondiente a la RSI:

```
void interrupt mi_rutina() {
 disable();
 /* Aquí se escribe la RSI */
 oldhandler();
 outportb(0x20,0x20);
 enable();
}
```

La RSI es la función especial void interrupt mi_rutina() donde mi_rutina es un puntero lejano (far pointer) que apunta a la dirección de comienzo de la RSI. Esta dirección es la que se ubica en la tabla de vectores de interrupción.

Al entrar en la RSI se pueden deshabilitar el resto de interrupciones enmascarables mediante la función disable(). disable() borra el flag de interrupción, así que ninguna otra interrupción hardware, excepto NMI (Non-Maskable Interrupt) podrá atenderse. Si no se usa esta función, interrupciones de mayor prioridad podrán interrumpir la ejecución de mi_rutina.

El cuerpo de la RSI incluirá el código que se quiera ejecutar cuando se active la solicitud de interrupción. Muchos periféricos interrumpen al microprocesador por varias razones(byte recibido, time-outs, buffer FIFO vacío, desbordamiento, etc.), de manera que lo primero que hay que hacer es saber la

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

causa de la interrupción; esto se consigue leyendo los registros de estado del periférico. Es práctica común hacer sólo lo imprescindible dentro de la RSI (con el objeto de no bloquear al sistema) y transferir la información correspondiente (dato recibido, flag de estado, etc.) a memoria para que sea el programa principal el que la procese (véase como ejemplo el apartado 5.1 referente al interfaz de teclado).

En algunos casos puede ser apropiado encadenar la vieja RSI a la nueva. Esto se debe a que el S.O. o ciertos programas residentes podrían hacer uso de la antigua RSI y si ésta se elimina el sistema podría volverse inestable o incluso dejar de funcionar. Para encadenar la vieja RSI a la nueva, se la invoca con la función `oldhandler();` donde `oldhandler` apunta a la vieja RSI.

Antes de retornar de la interrupción, hay que decirle al controlador de interrupciones que la interrupción ha terminado. Para ello le envía un comando EOI (End of Interrupt `0x10`)[194].

2.3.3. Guía básica de programación de DMA

Se presenta una guía básica de programación del controlador 8237 para llevar a cabo transferencias DMA.

1. Deshabilitar el canal de DMA.
2. Borrar el byte pointer flip-flop.
3. Programar el registro de modo (ver tabla 4.4).
4. Programar el offset del buffer, byte bajo seguido de byte alto. Para datos de 16 bits, el offset debería ser en palabras desde el comienzo de una página de 128KB. El método más fácil para calcular los parámetros de 16 bits es dividir la dirección física por dos antes de calcular el offset.
5. Configurar la longitud de la transferencia, byte bajo seguido de byte alto. Para una transferencia de 8 bits/16 bits, escribir el número de bytes/palabras menos 1.
6. Configurar el registro de página del canal DMA con la página del buffer[195].
7. Habilitar el canal de DMA.

[194] Si la interrupción corresponde al PIC1(`IRQ[7:0]` se escribe `20H` en el puerto `20H`: `outportb(0x20,0x20);`. Si es el PIC2 (`IRQ[15:8]`) la escritura se hace en la dirección `A0H`: `outportb(0xA0,0x20);`. Si se usa el PIC2 hay que enviar el comando EOI a ambos PIC.

[195] El espacio de direcciones del microprocesador IA-32 en modo real es de 1MB (2^{20}) y está formado por 16 páginas de 64KB (2^{16}) cada una.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3. Puertos de E/S

Los puertos de E/S más característicos de un ordenador personal compatible IBM son el puerto serie RS-232C y el puerto paralelo CENTRONICS. A continuación se describen sus características.

3.1. Puerto serie RS-232C

El puerto serie del PC sigue el estándar RS-232C de la EIA (Electronics Industry Association) que especifica la manera de establecer una comunicación entre un equipo terminal de datos (DTE, Data Terminal Equipment), por ejemplo un PC, y un equipo de comunicación de datos (DCE, Data Communications Equipment), por ejemplo, un modem.

RS-232C especifica 25 señales con su conector correspondiente de 25 patillas. En ocasiones, los PCs usan un conector de 9 patillas ya que en muchas aplicaciones no se requieren la mayoría de las señales[196](ver tabla 4.6).

9pines	25pines	Señal	Significado	Descripción
1			Protective ground	
2	3	RD	Receive Data	Recepción de datos
3	2	TD	Transmit Data	Transmisión de datos
4	20	DTR	Data Terminal Ready	El DTE indica al DCE que está preparado para conectar
5	7	GND	Signal Ground	Masa
6	6	DSR	Data Set Ready	El DCE indica al DTE que está preparado para conectar
7	4	RTS	Request To Send	El DTE indica al DCE que está preparado para intercambiar datos
8	5	CTS	Clear To Send	El DCE indica al DTE que está preparado para intercambiar datos
9	22	RI	Ring Indicator	El DCE indica al DTE que se ha detectado una llamada entrante

Tabla 4.6: Conectores de puerto serie RS-232C

Según la norma RS-232C, la longitud máxima del cable es de 150m para velocidades de 20Kbaudios[197]. Sin embargo, es posible el uso de cables más largos a velocidades más bajas o velocidades mayores para cables más cortos.

Los niveles de tensión del puerto serie son completamente distintos a los utilizados en los sistemas digitales, por lo que se requieren conversores de nivel (figura 4.3). Un "0"lógico es una señal comprendida entre +3v y +12v, mientras que un "1"lógico es una señal comprendida entre -3v y -12v.

[196]En muchos casos, sólo serán necesarias tres señales: transmisión de datos, recepción de datos y masa.

[197]La unidad baudio se utiliza en modems. Se entiende por baudio el número de cambios de la señal por segundo. Es importante saber que no siempre coincide el número de baudios con el de bits por segundo ya que en ciertas modulaciones se transfieren varios bits en un

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

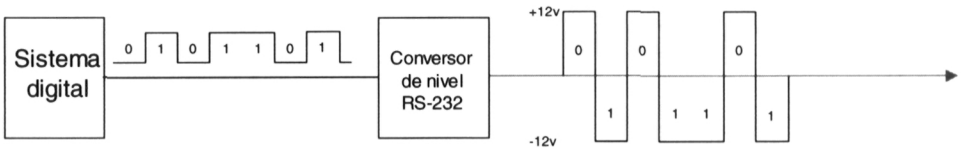


Figura 4.3: Niveles de tensión de las señales RS-232

3.1.1. Modem nulo

El modem nulo permite conectar dos DTEs entre sí. Se usa mucho en los sistemas de desarrollo para monitorización y carga del programa y en los protocolos como Zmodem o Xmodem para la transferencia de ficheros entre PCs. La idea es hacer creer a un DTE que al otro lado del interfaz hay un DCE (ver figura 4.4).

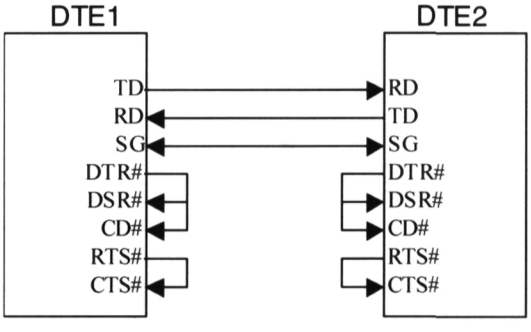


Figura 4.4: Modem nulo

3.1.2. Control de flujo

La velocidad entre el DTE y el DCE puede ser superior a la velocidad entre DCEs; este es, por ejemplo, el caso de los modems donde se usan técnicas de compresión de datos. Esto puede provocar un desbordamiento de los buffers de transmisión y recepción que se corrige con las técnicas de control de flujo software o hardware.

El control de flujo software, a veces llamado control Xon/Xoff, usa dos caracteres especiales: ASCII 17 (Xon) y ASCII 19 (Xoff). Cuando el buffer del

cambio de señal.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

modem se llena, el modem envía un carácter Xoff al PC para que éste deje de transmitir. Cuando el modem considera que puede admitir más datos envía un carácter Xon. Es una técnica muy simple que sólo requiere 3 hilos (RD,TD y GND), pero puede ser lenta en algunas aplicaciones.

El control de flujo hardware usa dos líneas de control: RTS y CTS. Cuando el PC quiere enviar datos activa la línea RTS. Si el modem puede aceptar los datos activa la línea CTS y comienza la transmisión. En caso contrario, no activa CTS y el PC queda a la espera.

3.1.3. UART 16550 de National

Una UART (Universal Asynchronous Receiver/Transmitter) es un chip que facilita la transmisión serie asíncrona de datos. Uno de los primeros fue el ACE8250 (Asynchronous Communication Element) de National que integra una UART y un generador de reloj. Posteriormente han ido apareciendo versiones compatibles con diferentes mejoras. Destacan el 16450 usado en los PC-AT, el 16550 que añade un buffer de 16 bytes, el 16650 que incorpora un buffer de 32 bytes, caracteres Xon,Xoff programables y gestión de potencia y el 16750 que amplía el buffer a 64 bytes; los tres primeros son de National y el último de Texas Instruments. Actualmente estos chips están integrados en los chipsets de la placa base de los PCs.

La figura 4.5 muestra el diagrama de bloques de una UART 16550 de National en el que se pueden distinguir los interfaces al microprocesador y al modem, los bloques de transmisión y recepción, el generador de reloj y el bloque de gestión de interrupciones.

3.1.3.1. Interfaz al microprocesador El interfaz al microprocesador está formado por el bus de datos `D[7:0]`, el bus de direcciones `A[2:0]` para acceso a los 11 registros del 16550 (ver tabla 4.7), las señales[198] de lectura `RD` y `RD#` y de escritura `WR` y `WR#`, las señales de selección[199] del chip `CS2#` y `CS[1:0]`, la señal de reset `MR` y las señales de READY `RXRDY#` y `TXRDY#`.

Otras señales opcionales son `DDIS` para deshabilitar el transceiver del bus de datos y `ADS#` para indicar cuando son estables los valores de las señales `A[2:0]`, `CS2#` y `CS[1:0]`. Este interfaz también incluye las entradas del oscilador de cristal del chip `XIN` y `XOUT`. Si se usa un generador de reloj externo se conecta a `XIN` dejando `XOUT` en abierto.

[198] Se proporcionan señales activas a nivel alto y a nivel bajo para ofrecer un interfaz más flexible.

[199] El disponer de varios chip select simplifica la lógica de interfaz.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

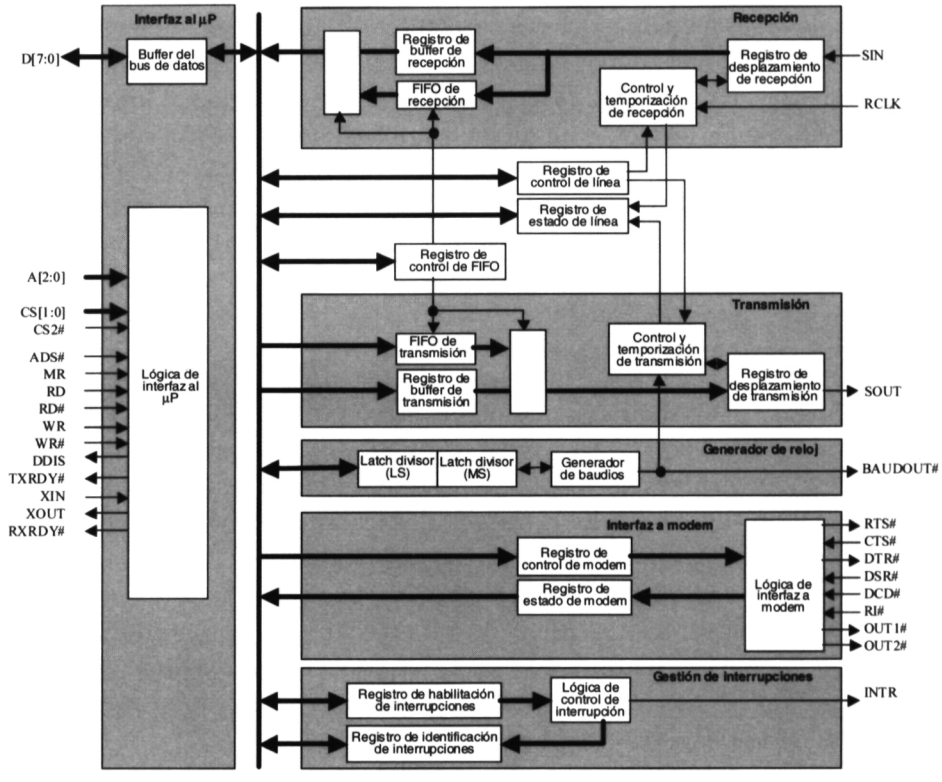


Figura 4.5: Diagrama de bloques de la UART 16550 de National

A2	A1	A0	DLAB	Modo	Nombre	Descripción
0	0	0	0	lectura	RBR	Receiver Buffer Register
0	0	0	0	escritura	THR	Transmitter Holding Register
0	0	0	1	lectura/escritura	DLL	Divisor Latch LSB
0	0	1	1	lectura/escritura	DLM	Divisor latch MSB
0	0	1	0	lectura/escritura	IER	Interrupt Enable Register
0	1	0	x	lectura	IIR	Interrupt Identification Register
0	1	0	x	escritura	FCR	FIFO Control Register. Sólo 16550
0	1	1	x	lectura/escritura	LCR	Line Control Register. El bit 7 es DLAB.
1	0	0	x	lectura/escritura	MCR	Modem Control Register
1	0	1	x	lectura/escritura	LSR	Line Status Register
1	1	0	x	lectura/escritura	MSR	Modem Status Register
1	1	1	x	lectura/escritura	SCR	Scratch Register

Tabla 4.7: Registros de la UART 16550 de National. DLAB es el bit 7 del registro LCR

3.1.3.2. Generador de reloj El generador programable de reloj proporciona las diferentes frecuencias del reloj de transmisión baudibles[200]. Para ello,

[200] El reloj de recepción se deriva del de transmisión.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

hay que cargar los latches divisores LS y MS con los valores apropiados (ver tabla 4.8) que se obtienen de la siguiente fórmula:

$$frecuencia\ reloj\ transmision = \frac{frecuencia\ generador\ reloj}{16 \times velocidad\ deseada} \tag{4.1}$$

Velocidad (bps)	Divisor	DLM	DLL
50	2304	09	00
300	384	01	80
600	192	00	C0
1200	96	00	60
2400	48	00	30
4800	24	00	18
9600	12	00	0C
19200	6	00	06
38400	3	00	03
57600	2	00	02
115200	1	00	01

Tabla 4.8: Programación de la tasa de baudios de la UART para un reloj de 1,8432 MHz

La salida del generador de reloj es `BAUDOUT`.

3.1.3.3. Transmisión El bloque de transmisión está formado por un registro de desplazamiento, un buffer, una FIFO y un módulo de control y temporización de transmisión. El reloj de trasmisión es `BAUDOUT` y lo proporciona el generador de reloj. La programación del transmisor se hace a través de los registros de control y estado de línea y de control de FIFO.

3.1.3.4. Recepción El bloque de recepción es similar al de transmisión. Está formado por un registro de desplazamiento, un buffer, una FIFO y un módulo de control y temporización de recepción. El reloj de recepción es `RCLK` y se suele conectar a la salida `BAUDOUT#`. La programación del receptor se hace a través de los registros de control y estado de línea y de control de FIFO.

3.1.3.5. Registros de control y estado de linea y FIFOS Son los registros LCR (Line Control Register), LSR (Line Status Register) y FCR (FIFO Control Register). Ver tablas 4.9, 4.10 y 4.11, respectivamente.

3.1.3.6. Interfaz al modem El interfaz a modem implementa algunas de las señales de la norma RS-232C: `DTR#`, `DSR#`, `DCD#`, `RTS#`, `CTS#` y `RI#`. También incluye dos entradas de propósito general `OUT1#` y `OUT2#`. La programación del interfaz se hace a través de los registros de control y estado de modem. Ver tablas 4.12 y 4.13.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Bit	Descripción			
7	DLAB(Divisor Latch Access). Permite el acceso a los latches divisores del generador de reloj. Ver tabla 4.7			
6	Break Control. Fuerza la salida del transmisor a "0" (espacio)			
5:3	bit5	bit4	bit3	paridad
	x	x	0	no hay paridad
	0	0	1	paridad impar
	0	1	1	paridad par
	1	0	1	paridad a "1"
	1	1	1	paridad a "0"
2	bit2	n^o bits de STOP		
	0	1		
	1	1 si T=6, 7 u 8 bits y 1,5 si T = 5 bits.		
1:0	bit1	bit0	tamaño dato (T)	
	0	0	5 bits	
	0	1	6 bits	
	1	0	7 bits	
	1	1	8 bits	

Tabla 4.9: Registro LCR: Line Control Register del 16550 de National

Bit	Nombre	Significado	Descripción
7			Error en buffer de recepción
6	TEMT	Transmitter Empty	Buffer y registro de desplazamiento del transmisor vacíos
5	THRE	Transmitter Holding Register Empty	Buffer del transmisor vacío
4	BI	Break Interrupt	Detección de un "0"(espacio) en recepción durante una palabra (START,datos, paridad y STOP).
3	FE	Framing Error	Error en la recepción del bit de STOP
2	PE	Parity Error	Error de paridad
1	OE	Overrun Error	Error de desbordamiento en el buffer de recepción
0	DR	Data Ready	Dato recibido y preparado para su lectura

Tabla 4.10: Registro LSR: Line Status Register del 16550 de National

Bit	Descripción		
7:6	bit7	bit6	Nivel de disparo de interrupción
	0	0	1 byte
	0	1	4 bytes
	1	0	8 bytes
	1	1	14 bytes
5	Habilita FIFO de 64 bytes (16750)		
4	Reservado		
3	Selecciona modo DMA		
2	Borra FIFO de transmisión		
1	Borra FIFO de recepción		
0	Habilita FIFOS		

Tabla 4.11: Registro FCR: FIFO Control Register del 16550

3.1.3.7. Gestión de interrupciones El bloque de gestión de interrupciones genera la señal `INTR` cuando se produce alguna de las 4 condiciones de interrupción: Recepción de flag de error, dato recibido disponible, registro de retención de transmisión vacío y estado del modem. Las interrupciones tiene un orden de prioridad. La gestión de interrupciones hace uso de dos registros: el registro de habilitación de interrupciones (tabla 4.14) y el registro de identificación de interrupciones (tabla 4.15).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Bit	Nombre	Significado	Descripción
7	0		Reservado
6	0		Reservado
5	0		Control de autoflujo habilitado (sólo 16750)
4	LOOP	Bucle local	Conecta internamente la transmisión con la recepción; es una función muy útil para chequear el chip. El transmisor se pone en estado "1" (marca).
3	OUT2		Salida de propósito general
2	OUT1		Salida de propósito general
1	RTS	Request to Send	Informa al modem que la UART está preparada para intercambiar datos
0	DTR	Data Terminal Ready	Informa al modem que la UART está preparada para conectar

Tabla 4.12: Registro MCR: Modem Control Register del 16550 de National

Bit	Nombre	Significado	Descripción
7	DCD	Data Carrier Detect	Indica que el modem ha detectado portadora
6	RI	Ring Indicator	Indica que el modem ha detectado una llamada entrante
5	DSR	Data Set Ready	Indica que el modem está preparado para conectar
4	CTS	Clear To Send	Indica que el modem está preparado para intercambiar datos
3	DDCD	Delta Data Carrier Detect	Indica que DCD ha sido modificado desde la última vez que se leyó
2	TERI	Trailing Edge Of Ring Indicator	Indica que ha habido un flanco de subida en la linea RI
1	DDSR	Delta Data Set Ready	Indica que DSR ha sido modificado desde la última vez que se leyó
0	DCTS	Delta Clear To Send	Indica que CTS ha sido modificado desde la última vez que se leyó

Tabla 4.13: Registro MSR: Modem Status Register del 16550 de National

Bit	Descripción
7:6	Reservados
5	Habilita modo de baja potencia (16750)
4	Habilita modo sleep (16750)
3	Habilita interrupción de estado del modem
2	Habilita interrupción de estado de linea de recepción
1	Habilita interrupción de buffer de transmisión vacío
0	Habilita interrupción de dato de recepción disponible

Tabla 4.14: Registro IER: Interrupt Enable Register del 16550 de National

Bit	Descripción		
7:6	bit7	bit6	uso de la FIFO
	0	0	no hay FIFO
	0	1	FIFO habilitada pero o utilizable
	1	1	FIFO habilitada
5	FIFO de 64 bytes habilitada (sólo 16750)		
4	Reservado		
3	Interrupción pendiente de time out (16550)		
2:1	bit2	bit1	tipo
	0	0	estado del modem
	0	1	buffer de transmisión vacío
	1	0	dato recibido
	1	1	estado de linea de recepción
0	Cuando está a 0 indica interrupción pendiente		

Tabla 4.15: Registro IIR: Interrupt Identification Register del 16550 de National

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.2. Puerto paralelo IEEE 1284

El puerto paralelo del IBM PC, también llamado puerto CENTRONICS, se utilizaba para conectar la impresora (figura 4.6).

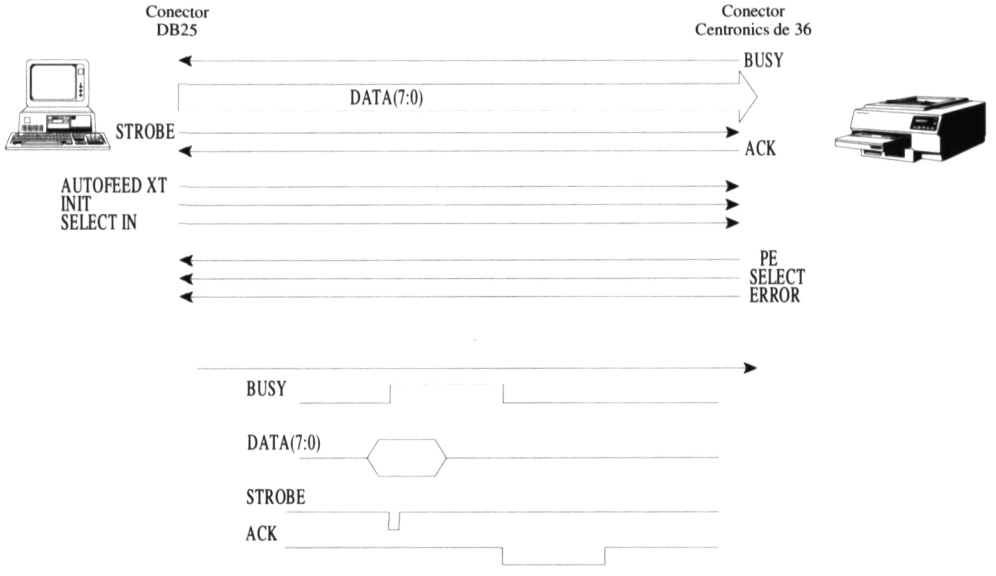


Figura 4.6: Interfaz de impresora CENTRONICS

En el cronograma adjunto se muestra el protocolo utilizado. El PC envía a través del puerto `DATA[7:0]` un dato a la impresora y lo valida con la señal de control `STROBE`. Cuando la impresora recibe el dato, activa su señal `BUSY` que mantiene activa mientras procesa la información. La señal `ACK` de la impresora indica al PC que ya puede enviar otro dato. El resto de señales son de estado de la impresora (ver tabla 4.16).

Señal	Descripción
BUSY	Ocupado. Indica que la recepción de datos no está operativa
DATA[7:0]	Bus de datos
ACK#	Indica que el dato ha sido reconocido por la impresora
PE	Paper Empty. Indica que no hay papel
SELECT	Indica estado de la impresora: activada/desactivada
IRQ ENABLE	Habilita solicitud de interrupción
ERROR#	Error
SELECT_IN#	Selección de entrada. Aceptación de la transmisión de datos
IRQ#	Interrupción activa a nivel bajo
INIT	Ordenador o impresora reinicializado
AUTOFEED#	Indica avanzar línea después de imprimir cada línea
STROBE#	Pulso que indica dato válido sobre D[7:0]

Tabla 4.16: Descripción de señales del puerto paralelo

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

El interfaz CENTRONICS se implementa con tres puertos de E/S de 8 bits del PC: El puerto de datos, el puerto de estado y el puerto de control (ver tabla 4.17). Los puertos se ubican en direcciones consecutivas. Si la dirección del puerto de datos es BASE, la dirección del puerto de estado es BASE+1 y la dirección del puerto de control BASE+2.

Bit	Estado	Control	Datos
7	BUSY	Reservado	DATA[7]
6	ACK#	Reservado	DATA[6]
5	PE	Dirección	DATA[5]
4	SELECT	IRQ ENABLE	DATA[4]
3	ERROR#	SELECT_IN#	DATA[3]
2	IRQ#	INIT	DATA[2]
1	reservado	AUTOFEED#	DATA[1]
0	reservado	STROBE#	DATA[0]

Tabla 4.17: Puerto paralelo: puertos de E/S de datos, estado y control

Las tablas 4.18 y 4.19 muestran la distribución de las señales del puerto paralelo en los conectores del PC y de la impresora, respectivamente.

Pin	Señal	Pin	Señal
1	STROBE#	2	DATA[0]
3	DATA[1]	4	DATA[2]
5	DATA[3]	6	DATA[4]
7	DATA[5]	8	DATA[6]
9	DATA[7]	10	ACK#
11	BUSY	12	PE
13	SELECT	14	AUTOFEED#
15	ERROR#	16	INIT#
17	SELECT_IN#	18	GND
19	GND	20	GND
21	GND	22	GND
23	GND	24	GND
25	GND		

Tabla 4.18: Conector impresora DB-25

Pin	Señal	Pin	Señal
1	STROBE#	2	DATA[0]
3	DATA[1]	4	DATA[2]
5	DATA[3]	6	DATA[4]
7	DATA[5]	8	DATA[6]
9	DATA[7]	10	ACK#
11	BUSY	12	PE
13	SELECT	14	AUTOFEED#
15	OSCXT	16	SIGNAL GND
17	FrameGND	18	+5v
19	GND	20	GND
21	GND	22	GND
23	GND	24	GND
25	GND	26	GND
27	GND	28	GND
29	GND	30	GND
31	PRIME#	32	ERROR#
33	SIGNAL GND	34	N/C
35	N/C	36	N/C

Tabla 4.19: Conector Centronics

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.2.1. Modos 1284

Aunque el adaptador de puerto paralelo del PC se diseñó específicamente para conectar impresoras, hoy en día es un puerto paralelo de propósito general. En 1994, se aprobó el estándar IEEE1284 que añade al modo de operación CENTRONICS cuatro modos más:

1. CENTRONICS. Es unidireccional. El protocolo `BUSY/STROBE/ACK` se hace por software. El ancho de banda oscila entre 50 y 150KBps.

2. Modo Nibble. Es bidireccional con canal de retorno de 4 bits. El protocolo `BUSY/STROBE/ACK` se hace por software.

3. Modo Byte. Es bidireccional con canal de retorno de 8 bits. El protocolo `BUSY/STROBE/ACK` se hace por software.

4. Modo EPP (Enhanced Parallel Port). Es bidireccional y usa hardware específico para generar y manejar el protocolo `BUSY/STROBE/ACK`. Su ancho de banda oscila entre 500KBps y 2MBps.

5. Modo ECP (Extended Capabilities Port). Similar al modo EPP, incorpora transferencia DMA, buffers FIFO, compresión de datos en tiempo real.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.3. Ejemplos

3.3.1. Acceso a impresora

Si se quiere implementar un sistema sencillo controlado por el PC que no tenga requerimientos muy exigentes de velocidad se puede hacer uso del puerto paralelo del PC. La siguiente secuencia del depurador del PC nos permite averiguar cuántos puertos paralelos tiene el PC y cuáles son sus direcciones.

```
c:>debug <CR> -D 40:08 L8 <CR> 0040:0008 78 03 00 00 00 00 00 00
```

El depurador accede a las direcciones de memoria `0040:0008 - 0040:000F` donde el PC almacena las direcciones de E/S de hasta cuatro puertos paralelo (tabla 4.20).

Puerto	Direcciones	Contenido
LPT1	0040:0008-0040:0009	78 03
LPT2	0040:000A-0040:000B	00 00
LPT3	0040:000C-0040:000D	00 00
LPT4	0040:000E-0040:000F	00 00

Tabla 4.20: Direcciones de los puertos paralelos del ejemplo

La primera dirección se corresponde con la dirección `BASE` de `LPT1: 378H`. No hay más puertos paralelos ya que el resto de direcciones contienen el valor 0.

Es importante hacer notar que la dirección de los puertos puede no ser siempre la misma. En la inicialización, el PC lee ciertas direcciones de E/S para comprobar si pertenecen a puertos paralelos implementados. Para ello sigue la siguiente secuencia: 1) Para $i = 1$, si en la dirección 3BCH hay un puerto paralelo le asigna `LPT`i e incrementa i. 2) Si en 378H hay un puerto paralelo le asigna `LPT`i e incrementa i. 3) Si en 278H hay un puerto paralelo le asigna `LPT`i e incrementa i, etc..

La dirección `378H` es la dirección del puerto de datos y `379H(BASE+1)` y `37AH(BASE+2)` las direcciones del puerto de estado y de control respectivamente.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Una vez sabemos la dirección de LPT1, podemos acceder a sus puertos. La siguiente secuencia pretende ejecutar la rutina de impresora `INT 17H` de la BIOS (Basic Input Output System):

```
c:>debug <CR>
  -A <CR>
  16B7:0100 MOV AH,2 <CR>
  16B7:0102 MOV DX,0<CR>
  16B7:0105 INT 17 <CR>
  16B7:0107 INT 3 <CR>
  16B7:0108
  -G <CR>
```

Esta rutina tiene dos parámetros de entrada que se pasan en los registros `AX` y `DX` del procesador. En `AX` se escribe el servicio deseado: `AH=0` imprime un carácter, `AH=1` inicializa el puerto y `AH=2` lee el estado de la impresora. En `DX` se escribe la dirección del puerto paralelo de la impresora: `LPT1=0`, `LPT2=1`, etc.

Se puede observar como el bit 7 de `AX`, que ahora almacena el contenido del registro de estado de la impresora, vale 1 si la impresora está encendida y 0 si está apagada. El bit 7 es la señal de `BUSY` de la impresora.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4. Buses estándar de E/S

Los buses estándar de E/S de un ordenador personal cumplen con unas especificaciones mecánicas, eléctricas, electrónicas y de protocolo que facilitan el uso de elementos de terceros fabricantes y la ampliación del sistema.

Se puede distinguir entre buses internos y buses externos. El bus interno de E/S interconecta elementos del sistema y accede al exterior a través de ranuras de expansión en las que se conectan tarjetas controladoras de todo tipo de periféricos; destacan los buses paralelo ISA[201], PCI y VME y el bus serie PCI Express. Los buses externos proporcionan una conexión directa a los periféricos; destacan los buses serie USB y Firewire y los buses paralelo SCSI, ATA y GPIB. Otro bus de interés que podría encajar en esta categoría es el I2C que ofrece una conexión directa entre chips.

4.1. Evolución de buses internos de E/S

El primer PC de IBM (1981) disponía de una arquitectura abierta que permitía ampliar el sistema: el denominado bus de expansión del PC. IBM divulgó sus especificaciones con la idea de forzar la expansión del PC. Posteriormente, con la llegada del PC-AT (1984), este bus fue mejorado, ampliándose el bus de datos a 16 bits y aumentando la frecuencia de reloj a 8MHz (ver figura 4.7).

En 1987, IBM introdujo en su PS/2 un nuevo sistema de bus más rápido y sencillo pero incompatible con el bus AT: el MCA (Micro Channel Architecture). IBM sólo proporcionaba la especificación del bus MCA a cambio del pago de una licencia[202]. El resto de fabricantes de PCs desarrollaron una arquitectura de bus alternativa denominada bus EISA (Extended Industry Standard Architecture) que era compatible con el bus AT. Al final, ninguno de los dos buses cosechó el éxito que esperaba. Fue la arquitectura AT la que se mantuvo vigente durante muchos años. No obstante, su reducida velocidad para transferencia de datos en gráficos o discos duros obligó al desarrollo de buses locales. El VL-bus (Video Electronics Standards Association bus) era un bus local que se acoplaba directamente al microprocesador trabajando a su frecuencia de reloj y usando sus señales de control. Esta dependencia fue la causa de su fracaso frente al bus PCI (Peripheral Component Interconnect) de Intel (1990) que es independiente del microprocesador y que consiguió convertirse en el estándar de hecho de los buses de sistema para PCs. Sin embargo, el

[201] Se estudia en el capítulo de microcomputadores.

[202] Hoy en día se considera que esto fue un error.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

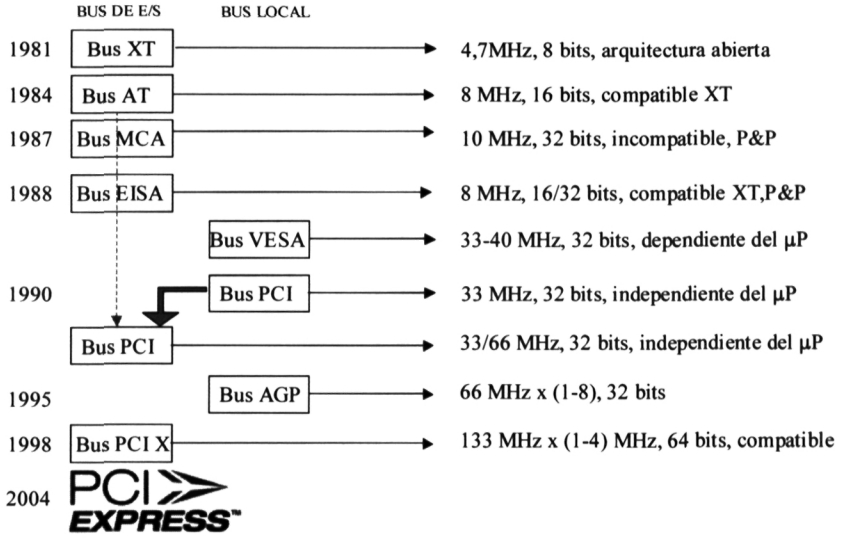


Figura 4.7: Evolución de los buses del PC

elevado ancho de banda del bus PCI no fue capaz de seguir a los gráficos 3D, que requerían tasas mucho más elevadas. Esto llevó a Intel a desarrollar un nuevo bus local específico para gráficos y que denominó bus AGP (Accelerated Graphics Port).

Posteriores versiones de PCI, todas compatibles con las anteriores, han mejorado sustancialmente su ancho de banda. PCI 2.1 (1995) puede trabajar a 66MHz con un bus de datos de 64 bits (533MBps). PCI-X (1998) alcanza 1 GBps gracias a su reloj de 133 MHz. La versión PCI-X-2 alcanza anchos de banda de 2 y 4 GBps respectivamente. Finalmente, el sucesor de PCI es PCI Express , un bus serie que ha revolucionado los buses de E/S y que puede llegar hasta los 16 GBps. En la tabla 4.21 se resumen las prestaciones de algunos buses de E/S.

Bus	Bus de datos	Frecuencia de reloj	Ancho de banda de pico
XT	8	4.77	1,59MBps
AT	8/16	8-8.33	5.55 MBps
MCA	16/32	10	40 MBps
EISA	32	8-8.33	32MBps
VLB	32	25-40	130MBps
PCI	32/64	33-66	133/533 MBps
PCI-X	32/64	133	1GBps

Tabla 4.21: Comparativa de buses de E/S

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4.2. Bus PCI

El bus PCI es un bus de E/S síncrono, independiente del procesador, con direcciones y datos multiplexados[203], configuración automática (plug&play) y compatibilidad con los buses ISA, EISA y MCA.

La figura 4.8 muestra las señales más representativas del bus PCI agrupadas por interfaces: Interfaz de memoria y E/S, errores, arbitración, extensión a 64 bits, memoria cache, interrupciones y testeo.

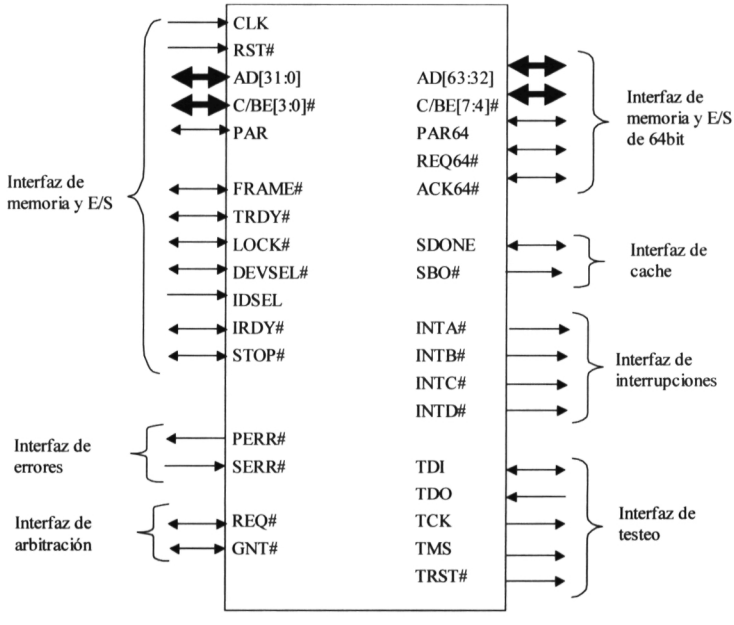


Figura 4.8: Señales del bus PCI

Las operaciones PCI se llevan a cabo entre un maestro (initiator) y un esclavo (target)(figura 4.9). Una transferencia PCI consta de una fase de direcciones y de una o varias fases de datos. Las operaciones de E/S que acceden a registros de los dispositivos PCI suelen tener sólo una fase de datos, mientras que en los accesos a memoria son comunes las fases de datos múltiples (acceso en modo ráfaga).

Durante la fase de direcciones, el maestro pone la dirección sobre el bus e indica el tipo de comando a ejecutar (lectura de memoria, escritura de memoria,

[203] El bus multiplexado reduce el número de señales en el conector, redundando este hecho en un menor coste y en componentes PCI más pequeños.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

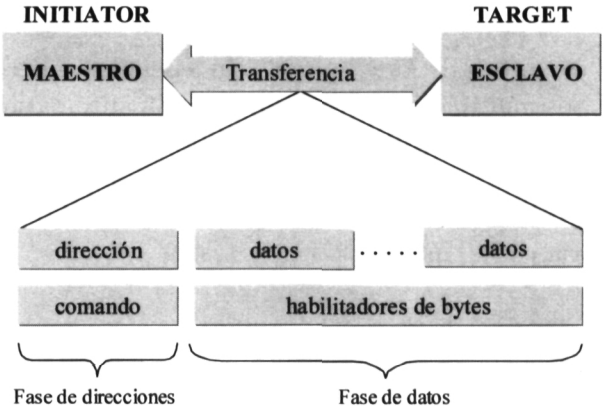


Figura 4.9: Transferencia PCI

lectura de E/S, etc.). A continuación, elimina la dirección (si es una escritura sustituye la dirección por los datos) y activa los habilitadores de byte. Seguidamente se pasa a la fase de datos donde se llevan a cabo las transferencias de datos.

Tanto el maestro como el esclavo pueden retrasar, parar o abortar una transferencia. Por ejemplo, un esclavo que necesite mucho tiempo para responder a una transacción, puede liberar el bus (parar su transferencia) mientras tanto.

La transferencia se puede suspender mediante reintentos o desconexiones. Se produce un reintento cuando el esclavo termina una transacción sin ejecutar ninguna fase de datos. Se produce una desconexión cuando el esclavo ha completado una o más fases de datos. En ambos casos, señalan al maestro que tiene que volver más tarde para intentar ejecutar la transacción de nuevo.

La transferencia también se puede abortar. Si hay un error fatal, tal como un problema hardware, el esclavo puede abortar la transferencia (aborto de esclavo) indicando así al maestro que informe al S.O. de que un error fatal ha sido detectado.

4.2.1. Interfaz de memoria y E/S

Las tablas 4.22, 4.23 y 4.24 describen las señales del interfaz de memoria y E/S.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Nombre	Descripción
CLK	Reloj. Todas las señales excepto reset e interrupciones se muestrean con el flanco de subida de la señal CLK. Las frecuencias de trabajo son 33MHz y 66MHz para las versiones 2.0 y 2.1, respectivamente.
RST#	Reset. Un reset pone los registros de configuración, máquina de estados y señales de salida a su estado inicial.
AD[31:0]	Direcciones y datos multiplexados. AD[31:0] transfiere la dirección física de 32 bits durante la fase de direcciones y información de datos de 32 bits durante la fase de datos. La fase de dirección ocurre durante el reloj siguiente al flanco de bajada de la señal FRAME#.
C/BE[3:0]#	Comando del bus (ver tabla 4.23) y habilitadores de byte multiplexados.
PAR	Paridad par sobre las señales AD[31:0] y C/BE[3:0]#. Es decir, hay un número par de unos entre las señales AD[31:0], C/BE[3:0]# y PAR. La señal PAR tiene la misma temporización que AD[31:0] pero retrasada un ciclo de reloj para dar tiempo al cálculo de la paridad.
FRAME#	Se activa por el maestro para señalar el comienzo de una nueva transacción de bus.
IRDY#	Initiator Ready. Se activa por el maestro como una indicación de que está preparado para concluir la fase de datos de una transacción en curso. Durante escrituras indica que el maestro ha colocado datos válidos sobre AD[31:0]. Durante lecturas indica que el maestro está preparado para aceptar datos sobre AD[31:0]. Una vez activado, el maestro mantiene activa IRDY# hasta que TRDY# se activa para concluir la transferencia, o hasta que el esclavo usa la señal STOP# para suspenderla. IRDY# permite al maestro insertar estados de espera.
TRDY#	Target Ready. Se activa por el esclavo como una indicación de que está preparado para concluir la fase de datos en curso de la transacción. Durante escrituras indica que el esclavo está preparado para aceptar datos sobre AD[31:0]. Durante lecturas indica que el esclavo ha colocado datos válidos sobre AD[31:0]. Una vez activada, el esclavo mantiene TRDY# activada hasta que IRDY# se activa para completar la transferencia. TRDY# permite al esclavo insertar estados de espera.
STOP#	Se activa por el esclavo para solicitar al maestro que termine la transacción en curso.
DEVSEL#	Device Select. Se activa por un esclavo cuando detecta su dirección sobre el bus. DEVSEL# puede ser conducida una, dos o tres ciclos de reloj después de la fase de direcciones. Una vez DEVSEL# ha sido activada, no puede ser desactivada hasta que la última fase de datos no haya concluido,o hasta que el esclavo aborte la transferencia (aborto de esclavo). Si el maestro nunca recibe una señal DEVSEL# activa aborta la transferencia (aborto de maestro).
IDSEL	Initialization Device Select. Se usa como un chipselect durante transacciones de lectura y escritura de configuración. Un dispositivo PCI es seleccionado por un ciclo de configuración sólo si IDSEL está activo (1), AD[1:0] está a 00 (indicando un ciclo de configuración de tipo 0) y el comando colocado sobre C/BE[3:0]# durante la fase de direcciones es una lectura o escritura de configuración. AD[10:8] puede ser usado para seleccionar una de hasta 8 funciones dentro del dispositivo PCI. AD[7:2] seleccionan registros de configuración individuales dentro de un dispositivo y función.
LOCK#	LOCK# es activada por un maestro para solicitar acceso exclusivó y ejecutar múltiples transacciones con un esclavo. Evita que otros maestros puedan modificar las direcciones bloqueadas mientras se lleva a cabo la transferencia. Sólo se bloquea una región específica (un mínimo de 16 bytes) de las direcciones del esclavo para acceso exclusivo. Mientras LOCK# está activado, otras transacciones no exclusivas pueden acceder a direcciones no bloqueadas.

Tabla 4.22: Interfaz de memoria y E/S del bus PCI

C/BE[3:0]#	Tipo de comando
0000	Reconocimiento de interrupción
0001	Ciclo especial
0010	Lectura de E/S
0011	Escritura de E/S
0100	Reservado
0101	Reservado
0110	Lectura de memoria
0111	Escritura de memoria
1000	Reservado
1001	Reservado
1010	Lectura de configuración
1011	Escritura de configuración
1100	Multiple lectura de memoria
1101	Ciclo de acceso dual
1110	Línea de lectura de memoria
1111	Escritura de memoria e invalidación

Tabla 4.23: Tipos de comandos PCI

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Nombre	Descripción
AD[63:32]	Direcciones y datos multiplexados. Provee 32 bits adicionales cuando opera en un bus de 64 bits. Durante las fases de datos estos bits transfieren 32 bits adicionales de datos cuando tanto REQ64# como ACK64# están activas. Durante las fases de direcciones, cuando un ciclo de direcciones dual está siendo emitido y la señal REQ64# está activa, estos bits transfieren los 32 bits más significativos de la dirección.
C/BE[7:4]#	Comando de bus y habilitadores de byte multiplexados. Provee 4 bits adicionales cuando opera en un entorno de bus de 64 bits. Durante las fases de datos estos bits transfieren los habilitadores de byte para los 32 bits más significativos del bus de datos (AD[63:32]) cuando tanto REQ64# y ACK64# están activos. Durante la fase de direcciones, cuando el ciclo de direcciones dual está siendo emitido y la señal REQ64# está activa, estos bits transfieren el comando del bus.
REQ64#	Request 64-bit Transfer. Se activa por el maestro para indicar que desea una transferencia de 64 bits. Esta señal tiene la misma temporización que FRAME#.
ACK64#	Acknowledge 64-bit Transfer. Se activa por un esclavo como una indicación de que ha decodificado su dirección como el esclavo del acceso en curso y que es capaz de ejecutar una transferencia de 64 bits.
PAR64	Parity Upper DWORD. Es el bit de paridad par que protege AD[63:32] y C/BE[7:4]#.

Tabla 4.24: Extensión a 64 bits del bus PCI

4.2.2. Transacciones

La figura 4.10 ilustra una transacción de lectura del bus PCI.

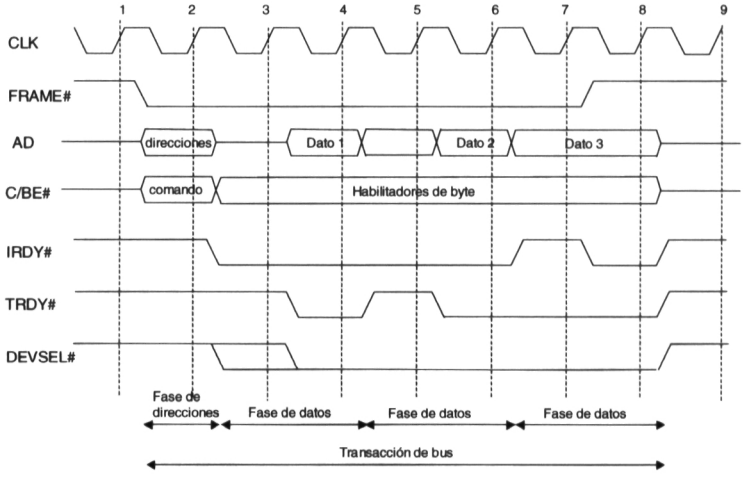


Figura 4.10: Transacción de lectura de un bus PCI

A continuación se describe la transacción ciclo a ciclo:

1. El bus está desocupado
2. El maestro pone una dirección válida sobre el bus y un comando de lectura sobre las señales `C/BE#`. Es la fase de direcciones.
3. El maestro pone en triestado las direcciones. El maestro pone los habilitadores de byte sobre `C/BE#`. El maestro activa `IRDY#` indicando

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

que está preparado para captura los datos de lectura. El esclavo activa DEVSEL# (en este ciclo o en el siguiente) como reconocimiento de que ha decodificado correctamente la dirección. El esclavo mantiene desactivado TRDY# indicando que todavía no ha proporcionado datos de lectura válidos.

4. El esclavo provee datos válidos y activa TRDY#. Como tanto IRDY# y TRDY# están activos, la transferencia tiene lugar y el maestro captura los datos. Esta es la primera fase de datos.
5. El esclavo desactiva TRDY# indicando que necesita más tiempo para preparar la siguiente transferencia de datos.
6. La segunda transferencia de datos ocurre cuando se activan IRDY# y TRDY#. El maestro captura los datos proporcionados por el esclavo.
7. El esclavo provee datos válidos para la tercera fase de datos, pero el maestro indica que no está preparado, desactivando IRDY#.
8. El maestro reactiva IRDY# para completar la tercera fase de datos. El maestro captura los datos provistos por el esclavo. El maestro desactiva FRAME# indicando que es la fase final de datos (terminación del maestro).
9. FRAME#, AD, y C/BE# se ponen en triestado. IRDY#, TRDY#, y DEVSEL# se desactivan durante un ciclo antes de ir a triestado.

En la figura 4.11 se ilustra una transacción similar, pero en este caso de escritura.

A continuación se describe la transacción ciclo a ciclo:

1. El bus está desocupado
2. El maestro pone una dirección válida sobre el bus y un comando de escritura sobre las señales C/BE#. Es la fase de direcciones.
3. El maestro pone sobre el bus los datos de escritura y las señales de habilitación de byte correspondientes. El maestro activa IRDY# indicando que hay datos de escritura válidos sobre el bus. El esclavo activa DEVSEL# como reconocimiento de que ha decodificado correctamente la dirección. El esclavo activa TRDY# indicando que está preparado para capturar los datos. La primera fase de datos ocurre ya que tanto IRDY# como TRDY# están activos. El esclavo captura los datos de escritura.
4. El maestro provee nuevos datos y habilitadores de byte. Se produce la segunda fase de datos y el esclavo captura los datos de escritura.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

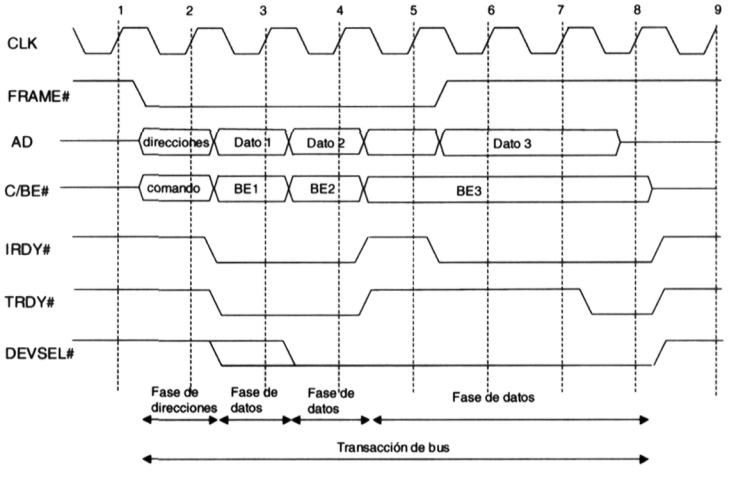


Figura 4.11: Transacción de escritura de un bus PCI

5. El maestro desactiva `IRDY#` indicando que no está preparado para proveer el siguiente dato. El esclavo desactiva `TRDY#` indicando que no está preparado para capturar el siguiente dato.
6. El maestro provee el siguiente dato válido y activa `IRDY#`. El maestro desactiva `FRAME#` indicando así el final de la fase de datos (terminación del maestro). El esclavo no está todavía preparado y mantiene `TRDY#` desactivada.
7. El esclavo no está todavía preparado y mantiene `TRDY#` desactivada.
8. El esclavo ya está preparado y activa `TRDY#`. Finalmente se produce la tercera fase de datos y el esclavo captura los datos de escritura.
9. `FRAME#`, `AD` y `C/BE#` son triestado. `IRDY#`, `TRDY#`, y `DEVSEL#` se desactivan un ciclo antes de ir a triestado.

4.2.3. Interfaz de interrupciones

El interfaz de interrupciones usa 4 señales : `INTA#`, `INTB#`, `INTC#`, `INTD#`. Estas señales se activan por dispositivos PCI para solicitar la atención de sus controladores (device driver software). Se disparan por nivel y son activadas como una señal open drain. Una vez activada, la señal `INTx#` continuará siendo activada por el dispositivo PCI hasta que el controlador borre la solicitud pendiente. Un dispositivo PCI que sólo contiene una función sencilla usa

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

INTA#. Dispositivos multifunción (tales como, por ejemplo una tarjeta híbrida LAN/modem) pueden usar[204] múltiples líneas INTx# . Todos los controladores PCI tienen que ser capaces de compartir un nivel de interrupción por encadenamiento con otros dispositivos usando el vector de interrupción.

4.2.4. Interfaz de arbitración

En un sistema PCI hay un árbitro del bus. Cada dispositivo PCI tiene dos líneas de control de arbitración del bus: REQ# y GNT# (figura 4.12).

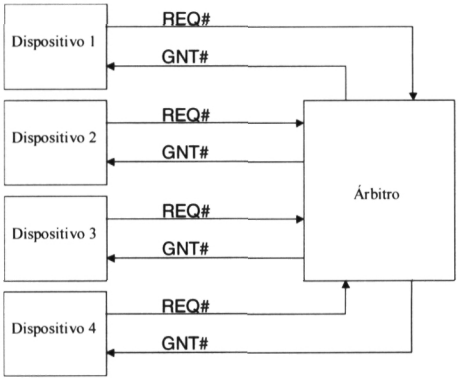


Figura 4.12: Arbitración PCI

Para conseguir el bus, el esclavo (incluido el procesador) activa REQ# y espera a recibir GNT#. Si GNT# se activa puede usar el bus en el siguiente ciclo de reloj. El algoritmo de arbitración no está especificado en la norma PCI (puede ser arbitración round robin, prioritaria, etc.).

La concesión del bus es para una transferencia, aunque ésta puede ser ilimitada. La arbitración en PCI es transparente en el sentido de que no consume ciclos de reloj. Las transferencias de bus del maestro en curso se solapan con el proceso de arbitración que determina el siguiente dueño del bus.

La tabla 4.25 describe las señales del interfaz de arbitración del bus PCI.

4.2.5. Otros interfaces

Las tablas 4.26 y 4.27 describen otras señales entre las que se encuentran las correspondientes a los interfaces de errores, memoria cache y testeo, respec-

[204] Un dispositivo de función sencilla usa INTA#, un dispositivo de dos funciones usa INTA# y INTB#, y así sucesivamente.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Nombre	Descripción
REQ#	Request. Se usa por un dispositivo PCI para solicitar el uso del bus. Cada dispositivo PCI tiene su propia y única señal REQ#. El árbitro en el sistema PCI recibe las señales REQ# de cada dispositivo. Es importante que esta señal sea triestado mientras RST# se activa para prevenir una caída del sistema. Esta señal es implementada sólo para maestros.
GNT#	Grant. Indica que una solicitud REQ# de un dispositivo PCI ha sido concedida. Cada dispositivo PCI tiene su propia y única señal GNT# desde el árbitro del sistema PCI. Si una señal GNT# de un dispositivo está activa durante un ciclo de reloj, entonces el dispositivo puede empezar una transacción en el siguiente ciclo de reloj activando la señal FRAME#. Esta señal sólo se implementa para maestros.

Tabla 4.25: Interfaz de arbitración del bus PCI

tivamente. El interfaz de testeo implementa el estándar JTAG/Boundary Scan (IEEE 1149.1, Test Access Port and Boundary Scan Architecture). JTAG permite testear exhaustivamente los componentes instalados en una tarjeta PCI.

Nombre	Descripción
PERR#	Parity Error. Se usa para reportar errores de paridad de datos durante todas las transacciones PCI excepto un ciclo especial. PERR# se activa dos ciclos de reloj después de la fase de datos con paridad incorrecta. Se activa por un mínimo de un periodo de reloj. PERR# es compartida entre todos los dispositivos PCI y es conducida con un driver triestado. Una resistencia de pull-up asegura que la señal está inactiva cuando ningún dispositivo la conduce. Después de ser activada, PERR# tiene que desactivarse un ciclo de reloj antes de volverse triestado para restablecer la señal a su estado inactivo. Esto asegura que la señal no permanece activada en el siguiente ciclo debido al tiempo de subida lento del pull-up.
SERR#	System Error. Reporta errores de paridad de direcciones, errores de paridad de datos durante un ciclo especial, o cualquier otro error fatal de sistema. SERR# es compartida entre todos los dispositivos PCI y es conducida sólo como una señal open drain (es conducida a cero o triestado por dispositivos PCI, pero nunca conducida a nivel alto).

Tabla 4.26: Señales de reporte de errores del bus PCI

Nombre	Descripción
SBO#	Snoop Backoff. Cuando se activa indica un acierto a una línea modificada. Cuando SBO# se desactiva y SDONE se activa, indica un resultado de *snoop clean*.
SDONE	Snoop Done. Indica el estado del snoop para el acceso en curso. Cuando se desactiva, indica el resultado del snoop está todavía pendiente. Cuanto activado, indica que el snoop ha concluido.
PRSNT[1:2]#	Present. Se usan para dos propósitos: 1) indicar que una tarjeta esta físicamente presente y 2) indicar los requerimientos de potencia de una tarjeta. Son señales estáticas que están puestas a 0 a en circuito abierto.
CLKRUN#	Clock Running. Es una señal opcional usada para deshabilitar el reloj y así reducir el consumo. CLKRUN# está pensaba para el mercado de PCs móviles.
M66EN	66MHz Enable. Indica si una tarjeta puede trabajar a 66MHz (en abierto o desconectada) o sólo a 33 MHz (conectada a masa).

Tabla 4.27: Otras señales del bus PCI

Nombre	Descripción
TCK	Test Clock de JTAG/Boundary Scan
TDI	Test Data Input de JTAG/Boundary Scan
TDO	Test Output de JTAG/Boundary Scan
TMS	Test Mode Select de estándar JTAG/Boundary Scan
TRST#	Test Reset de estándar JTAG/Boundary Scan

Tabla 4.28: Interfaz de testeo del bus PCI JTAG

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4.2.6. Conector PCI

Las tablas 4.29 y 4.30 ilustran las señales del conector PCI básico de 32 bits y de su extensión de 64 bits, respectivamente.

Se pueden implementar tres tipos de tarjetas de expansión: tarjetas de +5V que incluyen una *key notch* en los pines 50 y 51 para permitirles enchufar sólo conectores de esa alimentación, tarjetas de +3.3V con un *key notch* en los pines 12 y 13 y tarjetas universales que incluyen ambos *key notches* para permitirles ambas alimentaciones.

La alta velocidad de PCI limita el número de ranuras (slots) de expansión (3 ó 4) sobre el mismo bus. Para ampliar el número de ranuras se hace uso de un puente PCI-PCI. El puente PCI-PCI es un ASIC (Application Specific Integrated Circuit) que aísla eléctricamente dos buses PCI mientras que permite transferencias entre ellos. Cada puente tiene un bus PCI primario y otro secundario. Puede haber múltiples puentes en cascada para crear un sistema con muchos buses.

4.2.7. Autoconfiguración

Cuando el sistema arranca, el sistema operativo detecta, testea y configura los dispositivos PCI usando la información residente en el espacio de configuración PCI sobre cada módulo. Este espacio contiene campos fijos que indican el fabricante, modelo y parámetros operacionales del chip y campos variables que indican los parámetros opcionales y grabación de resultados de test.

Se asignan recursos del sistema a cada dispositivo PCI, construyéndose un mapa de recursos del sistema. Los controladores de dispositivos PCI, suministrados por el fabricante con cada módulo PCI, tienen que ser capaces de usar cualquier recurso del sistema asignado y tienen que soportar interrupciones hardware compartidas.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Pin	Cara B	Cara A
1	-12V	TRST#
2	TCK	+12V
3	GND	TMS
4	TDO	TDI
5	+5V	+5V
6	+5V	INTA#
7	INTB#	INTC#
8	INTD#	+5V
9	PRSNT1#	Reservado
10	Reservado	+3.3V (I/O)
11	PRSNT2#	Reservado
12	Connector Key	3.3V key
13	Connector Key	3.3V key
14	Reservado	Reservado
15	GND	RST#
16	CLK	+3.3V (I/O)
17	GND	GNT#
18	REQ#	GND
19	+3.3V (I/O)	Reservado
20	AD[31]	AD[30]
21	AD[29]	+3.3V
22	GND	AD[28]
23	AD[27]	AD[26]
24	AD[25]	GND
25	+3.3V	AD[24]
26	C/BE[3]#	IDSEL
27	AD[23]	+3.3V
28	GND	AD[22]
29	AD[21]	AD[20]
30	AD[19]	GND
31	+3.3V	AD[18]
32	AD[17]	AD[16]
33	C/BE[2]#	+3.3V
34	GND	FRAME#
35	IRDY#	GND
36	+3.3V	TRDY#
37	DEVSEL#	GND
38	GND	STOP#
39	LOCK#	3.3V
40	PERR#	SDONE
41	+3.3V	SBO#
42	SERR#	GND
43	+3.3V	PAR
44	C/BE[1]#	AD[15]
45	AD[14]	+3.3V
46	GND	AD[13]
47	AD[12]	AD[11]
48	AD[10]	GND
49	M66EN	AD[09]
50	GND	GND
51	GND	GND
52	AD[08]	C/BE[0]#
53	AD[07]	+3.3V
54	+3.3V	AD[06]
55	AD[05]	AD[04]
56	AD[03]	GND
57	GND	AD[02]
58	AD[01]	AD[00]
59	+3.3V (I/O)	+3.3V (I/O
60	ACK64#	REQ64#
61	+5V	+5V
62	+5V +5V 32-bit end	
	Connector Key	Connector Key
	Connector Key	Connector Key
63-92		

Tabla 4.29: Conector PCI

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Pin	Cara B	Cara A
63	Reservado	GND
64	GND	C/BE[7]#
65	C/BE[6]#	C/BE[5]#
66	C/BE[4]#	+3.3V (I/O)
67	GND	PAR64
68	AD[63]	AD[62]
69	AD[61]	GND
70	+3.3V	(I/O) AD[60]
71	AD[59]	AD[58]
72	AD[57]	GND
73	GND	AD[56]
74	AD[55]	AD[54]
75	AD[53]	+3.3V (I/O)
76	GND	AD[52]
77	AD[51]	AD[50]
78	AD[49]	GND
79	+3.3V (I/O)	AD[48]
80	AD[47]	AD[46]
81	AD[45]	GND
82	GND	AD[44]
83	AD[43]	AD[42]
84	AD[41]	+3.3V (I/O)
85	GND	AD[40]
86	AD[39]	AD[38]
87	AD[37]	GND
88	+3.3V	(I/O) AD[36]
89	AD[35]	AD[34]
90	AD[33]	GND
91	GND	AD[32]
92	Reservado	Reservado
93	Reservado	GND
94	GND	Reservado

Tabla 4.30: Extensión del conector PCI para bus de 64 bits

4.3. Bus PCI Express

PCI-Express es la evolución de PCI. Se caracteriza porque es un bus serie escalable de alta velocidad que mantiene compatibilidad software con PCI. Permite conexiones punto a punto entre dispositivos, garantiza calidad de servicio para aplicaciones de tiempo real y admite conexiones en caliente. Su conector es de 36 pines.

Un enlace PCI Express (ver figura 4.13) consiste en un canal simplex dual. Cada canal tiene dos pares diferenciales de bajo voltaje, uno para transmisión y otro para recepción.

Los datos se codifican usando un esquema 8b/10b que convierte cada 8 bits de datos en un carácter de transmisión de 10 bits. La codificación 8b/10b facilita la sincronización de los bits, simplifica los transmisores y receptores y mejora la detección de errores.

El ancho de banda (tabla 4.31) de un enlace puede ser escalado añadiendo pares de señal para formar múltiples camino (lanes) entre los dos dispositivos. La especificación soporta anchuras de caminos de x1, x4, x8, and x16. PCI

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

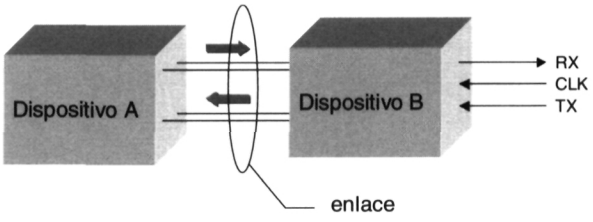


Figura 4.13: Enlace de PCI Express

Implementación	Velocidad datos codificados	Velocidad datos no codificados
x 1	5 Gbps	4 Gbps (500 MBps)
x 4	20 Gbps	16 Gbps (2 MBps)
x 8	40 Gbps	32 Gbps (4 MBps)
x 16	80 Gbps 64 Gbps (8 MBps)	

Tabla 4.31: Ancho de banda de PCI Express

Express ha sustituido a AGP[205].

En PCI Express es necesario que el bus esté siempre activo para que los trasmisores y receptores se mantengan sincronizados. Este proceso consume potencia y, por tanto, afecta negativamente a la vida de la batería de los sistemas portátiles. Para resolver esto, la especificación de PCI Express crea dos estados de enlace de baja potencia y un protocolo de gestión de potencia de estado activo (ASPM, Active-State Power Management). Cuando el enlace está desocupado, pasa a uno de los dos estados de baja potencia. Estos estados permiten reducir el consumo cuando el enlace está desocupado, pero requieren un tiempo de recuperación para resincronizar los transmisores y receptores cuando se necesita transmitir datos de nuevo. Es por tanto una solución de compromiso.

A diferencia de PCI, PCI Express incluye soporte nativo para trasferencia de datos isócronos (dependientes del tiempo) y varios niveles de calidad de servicio. Estas características son implementadas mediante el uso de canales virtuales que garantizan que los paquetes de datos llegan a su destino en el tiempo especificado.

[205] Ofrece el doble del ancho de banda de AGP×8.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4.4. Interfaz SCSI

El interfaz SCSI (Small Computer Systems Interface) se ideó en 1986 para conectar el ordenador (Macintosh, estaciones de trabajo y algunos PCs) a varios periféricos: discos duros, streamer, CD-ROM, scanner, etc.. El interfaz básico es un conector de 50 pines, 25 de los cuales son de masa para proveer inmunidad al ruido y facilitar altas velocidades. El bus de datos puede ser de 8 ó 16 bits.

SCSI es un sistema inteligente con bus propio que permite a un periférico comunicarse directamente con otro y mediante un protocolo realizar la transferencia correspondiente. Soporta 8 ó 16 unidades de periféricos. Además, cada unidad puede tener varias subunidades.

En la tabla 4.32 se pueden ver las características más importantes de las diferentes versiones SCSI que existen hasta la fecha. La longitud del cable depende del método de señalización.

Versión	Estándar	Bus de datos (bits)	Frecuencia (MHz)	Ancho de banda (MBps)	Longitud cable(m)	N. max. dispos.	nº pines
SCSI-1	SCSI-1	8	5	5	6	8	50
Wide SCSI	SCSI-1	16	5	10	1,5-3	16	68
Fast SCSI	SCSI-2	8	10	10	1,5-3	8	50
FastWideSCSI	SCSI-2	16	10	20	1,5-3	16	68
UltraSCSI	SCSI-3/SPI	8	20	20	1,5-3	8	50
WideUltraSCSI	SCSI-3/SPI	16	20	40	1,5-3	16	68
Ultra2SCSI	SCSI-3/SPI-2	8	40	40	12	8	50
WideUltra2SCSI	SCSI-3/SPI-2	8	40	80	12	16	68
Ultra3SCSI	SCSI-3/SPI-3	16	40	160	12	16	68
Ultra320SCSI	SCSI-3/SPI-4	16	80	320	12	16	68
Serial Attached SCSI				3-12GBps			

Tabla 4.32: Especificaciones SCSI

4.5. Interfaz ATA

En 1983, Compaq quería integrar una controladora ST-506 dentro de la unidad de disco, así que desarrolló el disco duro IDE (Integrated Drive Electronics). El interfaz del disco duro IDE se denominó ATA (AT Attachment).

El interfaz ATA es un puerto paralelo de 16 bits que permite conectar varias unidades de almacenamiento. En la tabla 4.33 se muestran las señales del conector ATA de 40 patillas. ATA tiene dos modos de transferencia: PIO (Programmed Input/Output) que es un acceso al espacio de direcciones de

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

E/S (hoy en día ya obsoleto) y DMA (Direct Access Memory).

Pin	Señal	Pin	Señal
1	RESET#	2	GND
3	D7	4	D8
5	D6	6	D9
7	D5	8	D10
9	D4	10	D11
11	D3	12	D12
13	D2	14	D13
15	D1	16	D14
17	D0	18	D15
19	GND	20	Key
21	reservado	22	GND
23	IOW#	24	GND
25	IOR#	26	GND
27	IOChrdy	28	ALE
29	reservado	30	GND
31	INTRQ	32	IOCS16#
33	A1	34	PDIAG#
35	A0	36	A2
37	CS1FX#	38	CS3FX#
39	DASP#	40	GND

Tabla 4.33: Conector ATA de 40 patillas

Las tablas 4.34, 4.35 y 4.36 muestran una descripción de las señales y registros del interfaz ATA.

Señal	Dirección	Descripción
RESET#	salida	Señal de reinicialización
D[15:0]	bidireccional	Bus de datos
IOW#	salida	Señal de control de escritura
IOR#	salida	Señal de control de lectura
IOCHRDY#	entrada	Se activa a nivel bajo para indicar que los registros no están preparados para realizar la transferencia
SPSYNC/ALE	bidireccional	Spindle syncronization. Es entrada como maestro y salida como esclavo. Antes la introducción del estándar era ALE(Address Latch Enable) y se utilizaba para validar la dirección
IRTRQ	entrada	Solicitud de interrupción
IOCS16#	entrada	Indica que la transferencia será de 16 bits
A[2:0]	salida	Bus de direcciones
CS1FX#/CS3FX#	salida	Chip selects procedentes de la decodificación de las direcciones A[9:3]. Seleccionan los registros del disco duro
DASP#		Señal multiplexada. Indica que una unidad está activa o que la unidad 1 está presente.
PDIAG#		Señal de diagnóstico

Tabla 4.34: Señales de interfaz ATA

Dirección	CS1	CS3	A2	A1	A0	Lectura	Escritura
	0	0	X	X	X	Ilegal	Ilegal
	1	1	X	X	X	Z	No usado
3FX	1	0	0	X	X	Z	No usado
3FX	1	0	1	0	X	Z	No usado
3F6	1	0	1	1	0	estado alterno	Control de dispositivo
3F7	1	0	1	1	1	dirección de unidad	no usado

Tabla 4.35: Registros de bloque de control del interfaz ATA

El interfaz ATA también ha sufrido una evolución considerable que le ha

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Dirección	CS1	CS3	A2	A1	A0	Lectura	Escritura
1F0	0	1	0	0	0	Puerto de datos	Puerto de datos
1F1	0	1	0	0	1	Registro de error	precompensación
1F2	0	1	0	1	0	cuenta de sector	cuenta de sector
1F3	0	1	0	1	1	número de sector	número de sector
1F4	0	1	1	0	0	cilindro bajo	cilindro bajo
1F5	0	1	1	0	1	cilindro alto	cilindro alto
1F6	0	1	1	1	0	unidad/cabezal	unidad/cabezal
1F7	0	1	1	1	1	estado	comando

Tabla 4.36: Registros de bloque de comandos del interfaz ATA

llevado a competir con SCSI en el mercado de los PCs. La primera versión ATA-1 solo admitía dos unidades de disco y un ancho de banda muy reducido (4 MBps).

ATA-2 duplicó el número de unidades a controlar, aumentó el ancho de banda (16,6 MBps) e incorporó direccionamiento LBA[206] (Logical Block Addressing).

ATA3 se limitó a mejorar la fiabilidad y el consumo de las unidades. ATAPI (ATA Packet Interface) era un estándar diseñado para CD-ROMs y unidades de cinta que se podía conectar a un puerto ATA[207]. Las ventajas de esta aproximación fueron economía, facilidad de conexión a PCs y mejor rendimiento y fiabilidad que otros interfaces específicos.

ATA-4 definió nuevos modos DMA de alta velocidad (hasta 33MBps), estableció un conector de 80 patillas necesario para reducir las interferencias entre señales de alta velocidad e incluyó CRC (Cyclical Redundancy Checking). En ATA-5, la mejora más significativa fue la duplicación del ancho de banda (modo DMA hasta 66MBps). ATA-6, el último de la saga, tiene un ancho de banda de 100MBps.

El interfaz paralelo ATA (PATA) parece haber llegado a su fin con la especificación Serial ATA (SATA). SATA es un interfaz serie de sólo 4 hilos compatible software con PATA y de un coste similar. Usa dos pares de señales diferenciales de bajo voltaje[208] para la transferencia de datos y comandos en una configuración punto a punto. Su ancho de banda es de 150MBps escalable a 300MBps y 600MBps. Además, admite conexión plug and play on line, está preparado para direccionamiento de unidades mayores de 137GB y facilita

[206] La manera tradicional de direccionar un disco duro es indicándole el número de cilindro C, el número de cabezal H y el número de sector S al que se quiere acceder. Diferentes restricciones impuestas por el S.O. de los PCs y por la estructura de los discos IDE fijaron un tamaño máximo para los discos duros de 504MB. Esta capacidad se hizo insuficiente hace algunos años y condujo a modificar el BIOS (Basic Input Output System) o a usar una nueva forma de direccionamiento: LBA.

[207] El denominado interfaz EIDE cumplía con las especificaciones de ATA-2 y ATAPI.

[208] 0.5V pico-pico a diferencia de PATA que necesita 5V.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

la implementación de RAIDs más flexibles y robustos.

La tabla 4.37 resume los anchos de banda de todas las versiones.

Versión	Ancho de banda (MBps)
ATA-1	4
ATA2	16
ATA3	16
ATA/ATAPI4	33
ATA/ATAPI5	44,66
ATA/ATAPI6	100
ATA/ATAPI7	133
SerialATA	150-300

Tabla 4.37: Especificaciones ATA

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4.6. USB (Universal Serial Bus)

El estándar USB surgió como una necesidad de simplificar la conexión e instalación de periféricos al PC. Un periférico con interfaz USB se puede conectar/desconectar con el PC encendido (plug & play on-line). El PC, de forma transparente al usuario, detecta la presencia del periférico y procede a la instalación/desinstalación del controlador correspondiente sin necesidad de reinicialización.

La figura 4.14 muestra un ejemplo de conexión USB. El PC es el nodo raíz con dos conexiones USB, una va a un teléfono y otra a un monitor que actúa además como nodo de tres conexiones: altavoces, impresora y un nodo auxiliar que permite conectar un teclado y un ratón. La impresora también se comporta como un nodo que facilita la conexión de un escáner.

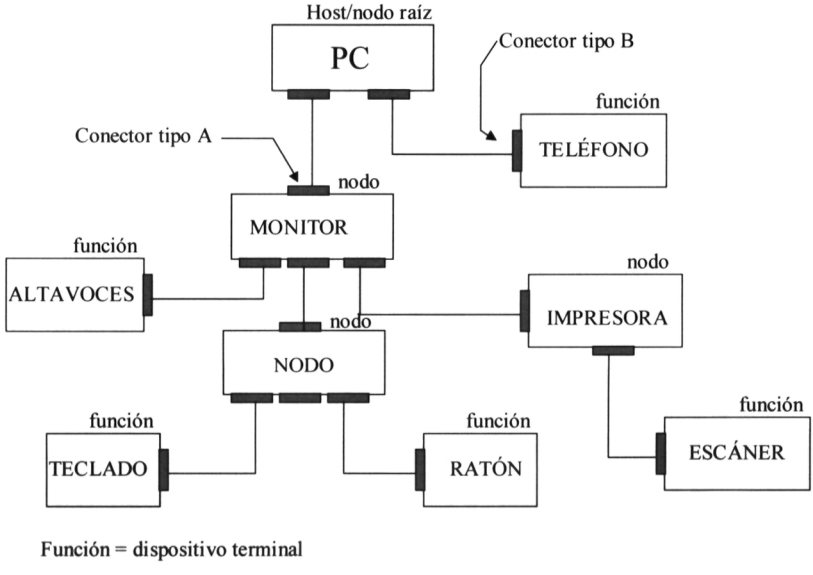


Figura 4.14: Ejemplo de conexión USB

USB es un bus serie telealimentado que permite la conexión de 127 periféricos con diferentes requerimientos de calidad de servicio[209].

USB tiene tres velocidades y admite el cambio automático entre ellas[210]:

[209] Hay una reciente versión de USB inalámbrica: WUSB que hace uso de UWB y que promete ser toda una revolución en la interconexión de periféricos.

[210] Slow and full speed se corresponden con la versión V1.1(1998). High speed es la versión V2.0 (2001) compatible con la versión 1.1. y con funcionalidad mejorada.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Slow speed (1.5Mbps) para periféricos lentos como teclado, ratón, joystick, etc..
- Full speed (12Mbps) para periféricos de media velocidad tales como audio digital, modems, impresoras, etc.
- High speed (480Mbps) para periféricos que requieren gran ancho de banda como el video.

4.6.1. Topología del bus USB

USB establece una topología en estrella entre el PC (host) y los dispositivos USB (ver figura 4.15).

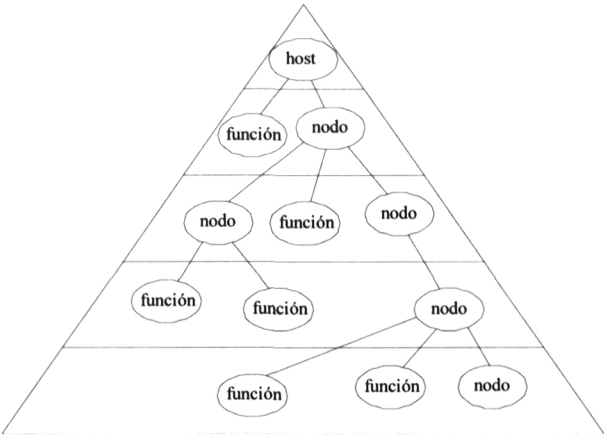


Figura 4.15: Topología de USB

El PC controla y planifica todas las actividades en el sistema e integra un nodo (hub) raíz que provee uno o más puntos de enganche. Hay dos tipos de dispositivos USB: nodos USB que proveen puntos de enganche extra) y funciones USB que son dispositivos terminales.

4.6.2. Modelo de flujo de datos

El bus utiliza una técnica de división en el tiempo (ver figura 4.16) con tramas (frames) de 1ms para la versión 1.1 y 125 μs para la versión 2.0 (estas últimas se denominan microtramas). El host establece conexiones lógicas punto a punto con los periféricos mediante la trasmisión y recepción de paquetes de información.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

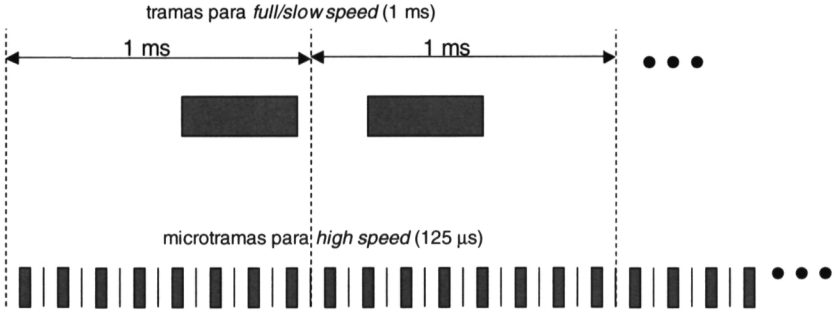


Figura 4.16: Tramas USB

USB proporciona 127 circuitos virtuales, uno para cada dispositivo (ver figura 4.17). A su vez, el circuito virtual se descompone en 16 canales o endpoints. Cada canal puede tener diferentes requerimientos de calidad de servicio y necesitar un tratamiento distinto[211].

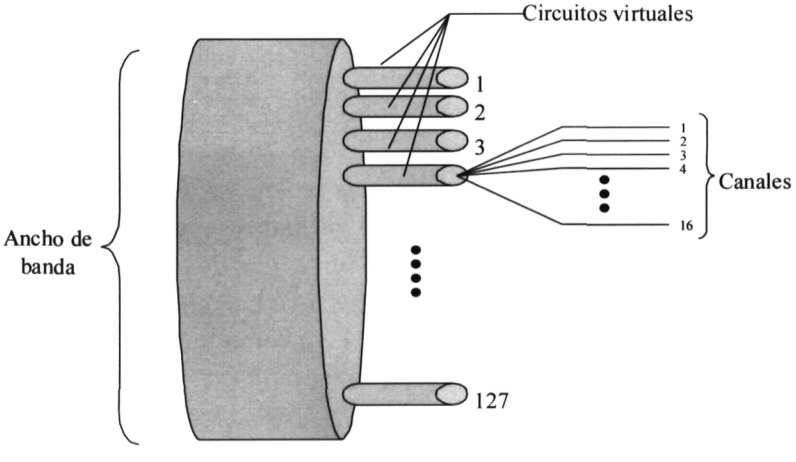


Figura 4.17: Canales USB

[211] Por ejemplo, una transferencia de un fichero de datos a una impresora requiere una tasa de error de bit (BER, Bit Error Rate) de 1×10^{-10}. Sin embargo, una transmisión de voz puede tolerar un BER de 1×10^{-3} aunque no admite un retardo excesivo (menos de 20ms).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4.6.3. Tipos de transferencias

Hay cuatro tipos de transferencias USB: control, isócrona, interrupción y bulk.

La transferencia de control es bidireccional y se usa para comunicaciones de estado, comandos o configuración entre el host y las funciones.

La transferencia isócrona puede ser unidireccional o bidireccional. Se usa para transmitir datos que requieren una velocidad constante y pueden tolerar errores[212]. La transferencia isócrona garantiza el acceso a USB con una latencia limitada y no retransmite los datos en caso de fallos en el envío. El tamaño máximo de paquete está limitado por trama[213].

La transferencia de interrupción es unidireccional (hacia el host). Se usa para transferencias de pequeño tamaño y poco frecuentes. Hace uso de polling con frecuencia de muestreo máxima de 1KHz (el periodo puede variar de 1ms a 255ms). Si hay errores se retransmite en el siguiente periodo de polling. Un ejemplo típico de transferencia de interrupción es el teclado.

La transferencia bulk puede ser unidireccional o bidireccional . Se usa para grandes transferencias de datos que no tienen requerimientos críticos de retardo. Estas transferencias usan el ancho de banda sobrante y admiten retransmisiones.

4.6.4. Interfaz USB

USB usa dos tipos de conectores: serie A y serie B. El conector serie B conecta dispositivos downstream (salidas de nodos y entradas de funciones). La serie A conecta dispositivos upstream (entradas de nodos).

El interfaz físico tiene 4 hilos (ver figura 4.18): Un par diferencial para datos (D+ y D-), alimentación (+5V) y GND. El consumo máximo es de 500mA[214]. Los cables tienen terminadores[215] para detectar la conexión/desconexión de periféricos y si son rápidos o lentos.

---

[212] Por ejemplo, la transmisión de voz PCM(Pulse Code Modulation) que exige 64 Kbps y un BER de 1×10^{-3}.

[213] 1023 bytes (full/slow speed) y 3072 bytes (high speed).

[214] Si cuelga del raíz o de un nodo alimentado.

[215] Las resistencias de pull-up del conector hacen que las líneas de datos cambien de nivel detectando así la conexión. Un dispositivo V1.1 rápido/lento tiene una resistencia de pull-up (1.5KΩ y tensión de 3.0V-3.5V) en la línea D+/D-. Un dispositivo V2.0 no lleva resistencias de pull-up.También hay resistencias serie (29-44 Ω) en las líneas D+ y D- para adaptación de impedancias.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

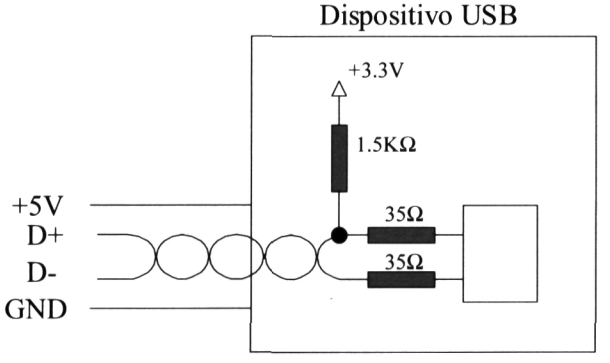


Figura 4.18: Interfaz USB de un dispositivo rápido

4.6.5. Formato de paquetes

La figura 4.19 muestra el aspecto de un paquete USB. El comienzo del paquete se señala con una transición de estado (de desocupado a activo). Las siguientes transiciones[216] marcan el campo de sincronismo(SYNC) que es usado por el circuito de entrada para alinear los datos entrantes con el reloj local.

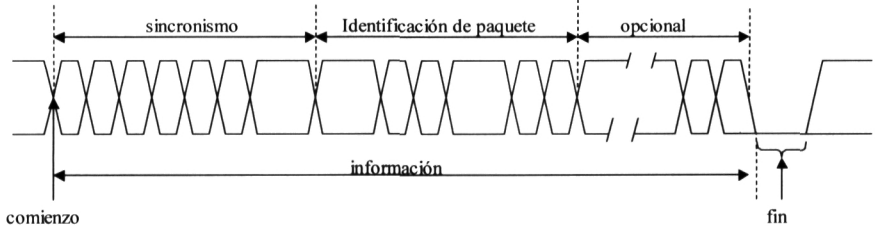


Figura 4.19: Paquete USB

A continuación, sigue un campo de identificación de paquete (PID) de 8 bits dividido en dos subcampos de 4 bits: El primero (PID_x) indica el tipo de paquete mientras que el segundo($PID_x\#$) es el valor complementado del anterior y se usa para chequeo de errores (tabla 4.38).

Seguidamente, hay un campo opcional de datos y finalmente un identificador de fin de paquete [217].

[216] 8 bits para V1.1. y 32 bits para V2.0.

[217] Consiste en mantener el estado desocupado durante dos bits para V1.1 y 40 bits para

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

PID_0	PID_1	PID_2	PID_3	$PID_0\#$	$PID_1\#$	$PID_2\#$	$PID_3\#$

Tabla 4.38: Campo PID de un paquete USB

Hay 4 tipos de paquetes USB: token, data, handshake y special. (ver tabla 4.39).

Tipo	PID	PID_3	PID_2	PID_1	PID_0
token	SOF	0	1	0	1
token	SETUP	1	1	0	1
token	IN	1	0	0	1
token	OUT	0	0	0	1
datos	DATA0	0	0	1	1
datos	DATA1	1	0	1	1
datos*	DATA2	0	1	1	1
datos*	MDATA	1	1	1	1
handshake	ACK	0	0	1	0
handshake	NACK	1	0	1	0
handshake	STALL	1	1	1	0
handshake	NYET	0	1	1	0
especial	PRE	1	1	0	0
especial*	ERR	1	1	0	0
especial*	SPLIT	1	0	0	0
especial*	PING	0	1	0	0
reservado	reservado	0	0	0	0

Tabla 4.39: Tipos de paquetes en USB. Los marcados con * son exclusivos de V2.0

4.6.5.1. Paquetes de token Hay cuatro paquetes de token: SOF, SETUP, IN y OUT.

El host envía un paquete SOF (Start Of Frame) cada comienzo de trama. El paquete tiene un campo de número de trama de 11 bits y un campo CRC (Cyclic Redundancy Check) de chequeo de errores de 5 bits. La tabla 4.40 muestra su formato.

SYNC	PID	Número de trama	CRC

Tabla 4.40: Formato de paquete SOF

En USB el host tiene el control y la gestión de bus y los dispositivos responden a las solicitudes del host. Si el host solicita información de un canal de dispositivo le enviará un token IN y el dispositivo responderá con un paquete de datos. Por otro lado, si el host desea transferir datos al dispositivo enviará un token OUT seguido de un paquete de datos. El token SETUP es un caso especial de paquete OUT.

V2.0.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Los paquetes IN, OUT y SETUP tienen el mismo formato (tabla 4.41).

SYNC	PID	ADDR	ENDP	CRC

Tabla 4.41: Formato de los paquetes IN, OUT y SETUP

El campo ADDR de 7 bits es el campo de dirección y se usa para seleccionar un dispositivo. El host asigna una dirección única a cada dispositivo durante el proceso de enumeración USB que se lleva a cabo cuando el dispositivo se conecta por primera vez al sistema.

El campo ENDP de 4 bits selecciona un canal (endpoint) del dispositivo. Finalmente, el campo CRC (Cyclic Redundancy Check) provee chequeo de errores para los campos no PID[218].

4.6.5.2. Paquetes de datos Hay varios tipos de paquetes de datos para proveer detección de errores: DATA0, DATA1, DATA2 y MDATA. En la versión 1.1, el transmisor suele alternar los paquetes DATA0 y DATA1. DATA2 y MDATA son específicos de la versión 2.0.

En la tabla 4.42 se muestra el formato de los paquetes de datos. El campo de datos varía de 0 a 1024 bytes y el campo CRC es de 16 bits.

SYNC	PID	Datos	CRC

Tabla 4.42: Paquete de datos de USB

4.6.5.3. Paquetes de handshake Hay cuatro tipos de paquetes de handshake: ACK, NACK, STALL y NYET.

Si el tipo de transferencia es bulk, control o interrupción, el host responderá con un paquete de handshake si ha recibido los datos sin errores. Si la transferencia es isócrona no hay paquete de handshake.

En una transferencia bulk (figura 4.20) se transmite: a) un paquete ACK si el dato recibido por el host o el dispositivo llega sin errores, b) Un paquete NAK si el dispositivo es incapaz en ese momento de atender la solicitud del host y c) Un paquete STALL si hay una condición de error en el dispositivo que requiere la intervención del host.

Los paquetes NYET se usan en la versión 2.0 en conjunción con el paquete especial PING. El paquete PING pregunta si el receptor tiene espacio en el

[218] El campo PID tiene sus propios bits de chequeo.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

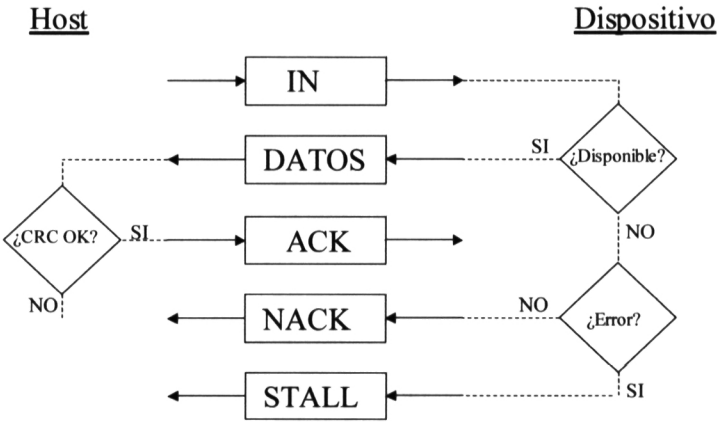


Figura 4.20: Protocolo de una transferencia bulk

buffer para aceptar una transacción OUT. En caso negativo, el receptor retorna un paquete NYET.

En una transferencia de interrupción, el dispositivo puede retornar un paquete NAK cuando no tiene datos que enviar. El host envía un paquete ACK si recibió el dato sin errores y no envía nada en caso contrario.

La tabla 4.43 muestra el formato de un paquete de handshake.

SYNC	PID

Tabla 4.43: Paquete de handshake de USB

4.6.5.4. Paquetes especiales El host envía un paquete de preámbulo especial PRE cuando se quiere comunicar con dispositivos lentos.

4.6.6. Enumeración USB

Cuando un dispositivo USB se conecta al sistema, se arranca un proceso de enumeración USB. El nodo al que se ha conectado el dispositivo detecta su presencia e informa al host. Se establece una comunicación entre el nodo y el host para habilitar y reinicializar el dispositivo[219]. En esta fase, los dispositivos tienen por defecto la dirección 0 y el canal 0. A continuación, el host se

[219] Un dispositivo USB se reinicializa llevando sus líneas de datos D+ y D- a masa durante 2.5 μs.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

comunica con el dispositivo para configurar la transferencia y para asignarle una dirección.

4.7. Bus I2C

El bus I2C (Inter-Integrated Circuit), desarrollado por Philips Semiconductors en 1982[220], provee un enlace de comunicaciones serie multimaestro entre circuitos integrados. El bus I2C simplifica enormemente la circuitería de los sistemas digitales ya que ahorra líneas de direcciones, datos y control, decodificadores y lógica de interfaz, además de aumentar la inmunidad al ruido.

Cada chip conectado al bus tiene su propia dirección de 7 bits y puede actuar en modo maestro[221] o en modo esclavo, dependiendo de su funcionalidad.

El bus I2C está formado por dos señales bidireccionales: datos (`SDA`) y reloj (`SCL`). Ambas señales son del tipo open collector[222] lo que permite crear una estructura wired-AND que facilita la arbitración, la implementación de estados de espera o la generación de condiciones de error. Para evitar efectos capacitivos, reflexiones, etc. la línea de bus se termina con una resistencia dinámica (twin charge pump).

La figura 4.21 muestra una secuencia I2C. Los datos son válidos durante el semiciclo alto del reloj.

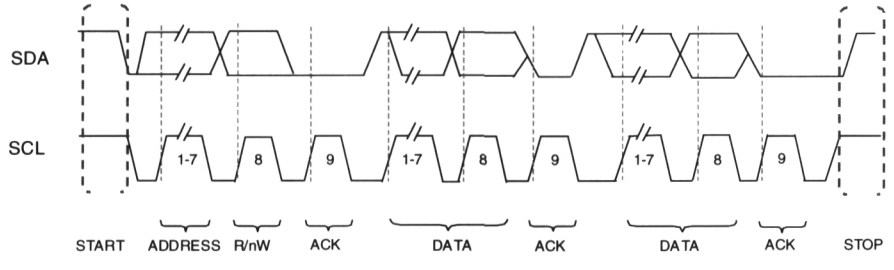


Figura 4.21: Ejemplo de transferencia en un interfaz I2C

Cuando un chip quiere hacer uso del bus se comporta como maestro y emite una condición START que es escuchada por el resto de chips conectados al bus que actúan como esclavos. A continuación, el maestro envía la dirección ADDRESS del chip al que quiere acceder junto con una indicación de si

[220] Su propósito original fue conseguir una manera fácil de interconectar los chips de un aparato de televisión.

[221] I2C establece que el dispositivo que inicia una transferencia de datos sobre el bus se considera el maestro del bus convirtiéndose el resto en esclavos.

[222] O open drain, dependiendo de la tecnología.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

el acceso es de lectura o escritura. Cuando los esclavos reciben la dirección verifican si coincide con la suya; en caso negativo quedan a la espera de que el bus se libere con una condición STOP. Sin embargo, el esclavo seleccionado responderá con una condición ACK. Una vez el maestro recibe la condición ACK comienza la transferencia de un byte de datos DATA. Cuando el receptor recibe el byte (maestro o esclavo) responde con una condición ACK. Cuando termina la transferencia el maestro emite una condición STOP indicando que el bus se ha liberado.

La arbitración del bus se consigue gracias a la naturaleza open collector de las señales de reloj y datos que permite detectar si el bus está ocupado o no. Todos los maestros generan sus propios relojes sobre la línea `SCL` para transferir mensajes sobre el bus. Un maestro pierde el control de bus si intenta poner `SDA` a "1"mientras otro maestro fuerza `SDA` a "0".

Cuando un esclavo es lento y necesita que el maestro le espere para que pueda completar la transferencia pone la línea `SCL` a "0"lo que provoca la inserción de estados de espera.

En 1992 apareció el bus I2C mejorado (Fast Mode), una nueva versión que disponía de un modo rápido de 400 Kbps, capacidad de direccionamiento de 10 bits y mayor inmunidad al ruido. Posteriormente se propuso el modo de alta velocidad (High Speed Mode o Hs-mode) que consigue velocidades de hasta 3.4 Mbps y es compatible con los modos Fast y estándar con la excepción de que no dispone de arbitración y sincronización de reloj.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5. Interfaz a periféricos

En este apartado se describen el funcionamiento e interfaz de los periféricos más característicos de un ordenador personal.

5.1. Teclado

El teclado es un dispositivo de entrada del PC para introducir datos de información y control. Se basa en una matriz de conmutadores (teclas) y un microcontrolador asociado. El microcontrolador[223] detecta las pulsaciones de cada tecla y les asocia un código que envía al PC través del interfaz de teclado. El S.O. decodifica la información recibida y la deja disponible para su uso por los programas de aplicación.

La figura 4.22 muestra el interfaz de teclado de un PC-AT. Es un interfaz serie síncrono telealimentado que tiene cuatro señales básicas: datos, reloj, alimentación y masa (`DATA`, `CLK`, `+5V` y `GND`, respectivamente). Las señales de datos y reloj son open-collector. La frecuencia del reloj es de 10-16,7KHz.

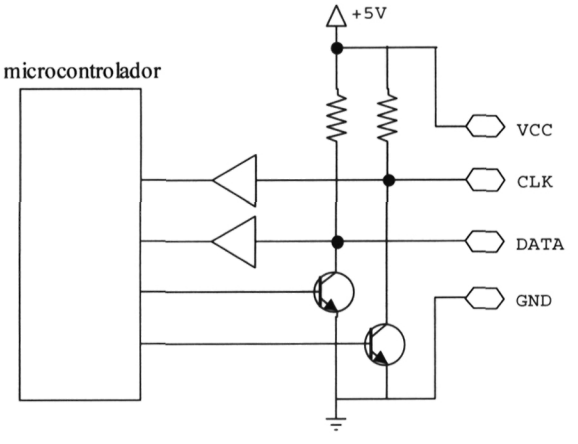


Figura 4.22: Interfaz de teclado AT

La transmisión de datos consiste en el envío a través de la linea `DATA` de 11 bits: un bit de START, 8 bits de datos, donde el bit 0 es el menos significativo, un bit de paridad impar[224] y un bit de STOP. La figura 4.23

[223] El microcontrolador también dispone de capacidad de diagnóstico, chequeo de errores y buffer para almacenar acciones de tecla (de unos 16 bytes).

[224] Los bits de datos junto con el de paridad tienen un número impar de unos.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

muestra una secuencia de transmisión de código del teclado al PC.

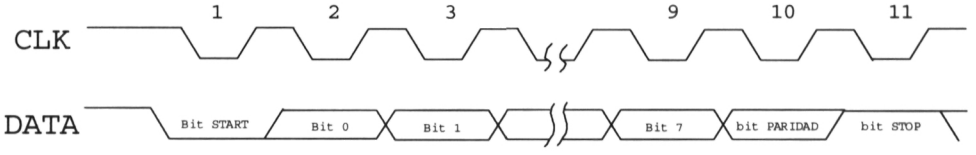


Figura 4.23: Transmisión de código del teclado al PC

En condición de reposo, las señales `DATA` y `CLK` están 1 lógico. Cuando el teclado quiere enviar un código tiene que chequear previamente la señal `CLK` ya que podría estar forzada a 0 por el PC (es una señal open-collector) si éste no está preparado para recibir datos[225].

El teclado escribe un bit en la línea de datos cuando el reloj está a 1 lógico y el PC lo lee cuando el reloj está a 0 lógico.

Cuando el PC quiere enviar datos al teclado ejecuta la siguiente secuencia (figura 4.24): Primero fuerza a 0 la señal `CLK` durante al menos 100 microsegundos y a continuación fuerza a 0 la señal `DATA` y libera la señal `CLK`. Cuando el teclado detecta esta secuencia, empieza a generar la señal de reloj y a leer los datos que envía el PC. Cuando el teclado termina de recibir el byte pone a 0 la señal `DATA` y genera un pulso de reloj más.

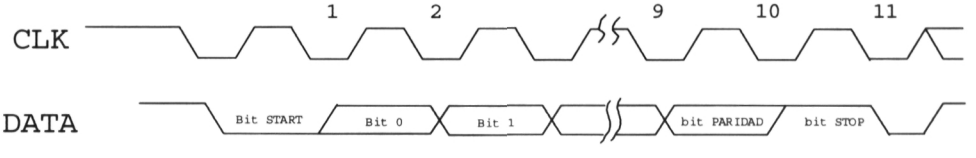


Figura 4.24: Transmisión de código del PC al teclado

La tabla 4.44 muestra una comparativa de los teclados del PC-XT, PC-AT y PS/2. El interfaz de teclado del PC-XT está obsoleto y su protocolo serie es incompatible con el del PC-AT. Por otro lado, los interfaces de teclado del PC-AT y PS/2 son compatibles si exceptuamos el conector y la distribución de pines. La tabla 4.45 muestra una comparativa de los interfaces de teclados del PC-XT, PC-AT y PS/2.

[225] En este caso, el teclado tendrá que retener los códigos generados hasta que la señal CLK se libere.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Característica	PC-XT	PC-AT	PS/2
Número de teclas	84	101	101
Microcontrolador	8048	8042	
Conector	DIN 5 pines	DIN 5 pines	miniDIN 6 pines
Interfaz	serie síncrono unidireccional	serie síncrono bidireccional	serie síncrono bidireccional
Códigos	set1	set 2	set2 y set3
Comandos PC→Teclado		8	12

Tabla 4.44: Comparativa de teclados de PC

Pin	PC-XT	PC-AT	PS/2
1	CLK	CLK	DATA
2	DATA	DATA	-
3	-	-	GND
4	GND	GND	+5V
5	+5V	+5V	DATA
6			-

Tabla 4.45: Tipos de conectores de teclado

5.2. Ratón

El ratón es el dispositivo de entrada del PC que maneja el cursor del interfaz gráfico de usuario (GUI, Graphics User Interface). El ratón dispone de unos sensores que detectan sus movimientos sobre una superficie y la pulsación de sus botones. Un microcontrolador codifica esta información y la envía al sistema cada vez que se produce una variación. El S.O. decodifica la información recibida y la deja disponible para su uso por los programas de aplicación.

El clásico ratón es optomecánico (ver figura 4.25) y se basa en una bola que cuando se mueve hace girar dos ejes perpendiculares(X e Y).

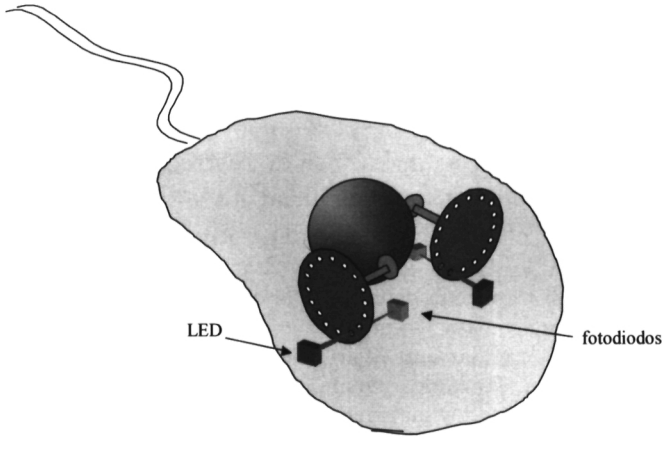


Figura 4.25: Arquitectura de ratón optomecánico

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Cada eje se conecta a una rueda dentada. A la altura de los dientes de cada rueda se sitúan los sensores: Un LED (Light Emitting Diode) y dos fototransistores, situados a ambos lados de la rueda[226]. La luz emitida por el LED sólo alcanza los fototransistores cuando la rueda al girar pasa entre dos dientes. Esto permite a los fototransistores detectar el movimiento y la dirección de las ruedas[227].

La figura 4.26 representa el movimiento de una de las ruedas.

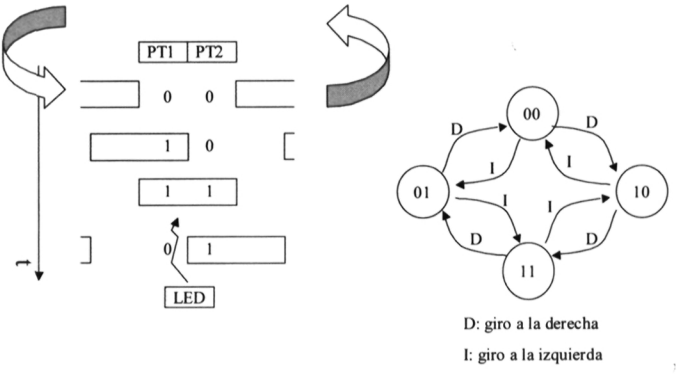


Figura 4.26: Codificación de los movimientos del ratón optomecánico en uno de los ejes

Inicialmente, los dos fototransistores (PT1 y PT2) son excitados por el LED (estado 00). Cuando la rueda empieza a girar hacia la derecha el fototransistor PT1 deja de recibir luz (estado 10) ya que un diente se interpone entre PT1 y el LED. Si la rueda sigue girando, el diente obstaculiza el paso de la luz a los dos fototransistores (estado 11). Los estados siguientes son 01 (cuando PT2 vuelve a recibir luz) y, de nuevo, 00. Hay, por tanto, sólo cuatro estados posibles. La cantidad de desplazamiento producida es registrada por un contador de pasos de la rueda. La tasa de rastreo no suele ser constante ya que el movimiento real del ratón no es lineal[228].

En el ratón óptico, el sensor es un LED y un fototransistor. Se usa sobre

[226] Se usan dos fototransistores para determinar la dirección del movimiento.

[227] Esta no es la única implementación posible.

[228] El ratón se mueve muy rápidamente al área deseada y luego, una vez en ella, busca con más detalle el punto exacto. Estos dos movimientos tienen requerimientos contradictorios por lo que la monitorización de los movimientos del ratón suele tener dos valores de CPI (cuentas por pulgada). El valor de CPI oscila entre 100 y 400 CPI pero puede llegar a 1000 CPI. La distancia se mide en la unidad mickey que según la resolución corresponde a una distancia de 1/200 a 1/400 pulgadas (moderno).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

una superficie cuadriculada. Conforme el ratón se mueve sobre la cuadrícula el fotodetector detecta los cruces de líneas (en cada dirección) viendo los cambios en la cantidad de luz reflejada desde el LED. Hoy en día, los ratones ópticos no necesitan la cuadrícula ya que se basan en otro principio; un LED genera patrones sobre la superficie que son leídos por una microcámara y un DSP(Digital Signal Processor) detecta los cambios de patrones conforme se mueve el ratón por la superficie y calcula el movimiento producido.

Cada vez que el ratón cambia de estado debido a un movimiento o a una pulsación o liberación de botones se transmite un paquete de información de 3 bytes[229]. En la figura 4.27 se puede ver el formato[230] de un ratón PS/2. El primer byte indica pulsación de botones y signo y desbordamiento de los contadores. El segundo y tercer byte señala el valor del contador en el eje X e Y, respectivamente. El valor que se envía de los contadores corresponde a un desplazamiento relativo al valor que se envió en el paquete anterior.

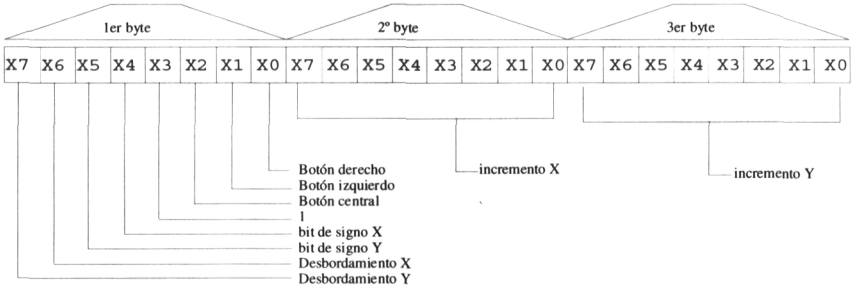


Figura 4.27: Paquete de datos de un ratón PS/2

El interfaz de ratón ha ido variando con el tiempo. Desde el bus mouse[231], el puerto serie RS-232-C, el puerto de teclado PS/2 y finalmente el bus USB.

En el antiguo ratón con interfaz serie RS-232 la transmisión se efectuaba a 1200 baudios con datos de 7 bits sin paridad. Para alimentar los circuitos del ratón (microcontrolador, LEDs, etc..) se usaban algunas de las señales de control del interfaz RS-232 (DTR, RTS, TD). Su consumo era de unos 10mA con un rango de 6 a 15V.

[229] En los ratones modernos hay un cuarto byte que indica el movimiento en el eje Z correspondiente a la rueda scroll.

[230] Los formatos varían ligeramente de unos ratones a otros.

[231] Era una tarjeta de expansión que contenía toda la circuitería del ratón exceptuando los sensores.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5.3. Joystick

El joystick es un dispositivo apuntador basado en un mango (stick) en posición vertical (eje Z) que admite cierto giro (del orden de $\pm 30^{o}$) en torno a los ejes X e Y. Esta configuración le permite transmitir órdenes de movimiento en varias direcciones del plano XY. También dispone de botones. Su principal aplicación ha sido en el campo de los videojuegos. Hay joysticks digitales[232] y analógicos.

El clásico joystick analógico de IBM se basa en dos potenciómetros **AX** y **AY** que varían su valor en función del ángulo de giro en los ejes X e Y, respectivamente. Los potenciómetros tienen su valor mínimo cuando el joystick está en la posición superior izquierda[233]. Conforme se va desplazando a lo largo del eje X o Y va incrementando su valor alcanzando el máximo en la posición inferior derecha. El joystick también dispone de dos botones (A1 y A2).

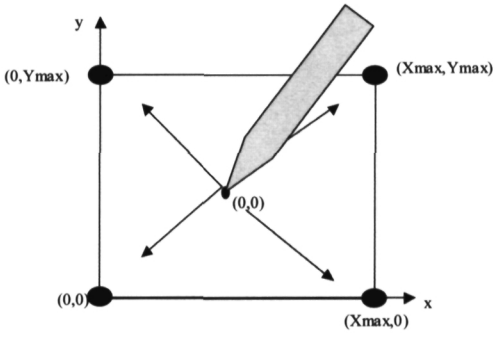


Figura 4.28: Joystick analógico

Cada uno de estos potenciómetros forma parte del circuito RC de un monoestable (figura 4.29).

El monoestable se reinicializa periódicamente para que genere pulsos que serán de una amplitud proporcional a la constante RC y, por tanto, proporcionales al valor del potenciómetro que marca el ángulo de giro del joystick. El sistema calcula la anchura del pulso leyendo mediante polling su duración[234].

El interfaz del joystick analógico del IBM-PC es simplemente un puerto de E/S del PC. Una escritura en el puerto reinicializa los monoestables. Una

[232] El joystick digital se basa en una serie de contactos eléctricos que se cierran cuando el joystick alcanza ciertas posiciones.

[233] Entendemos que superior/inferior hace referencia al eje Y e izquierda/derecha al eje X. Con el joystick en reposo los dos potenciómetros estarían a su valor medio.

[234] Hoy en día, muchos joysticks analógicos se basan en conversores analógico-digitales.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

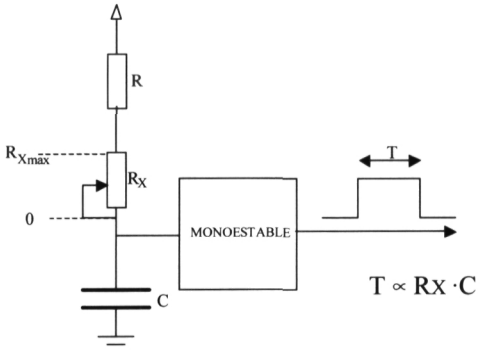


Figura 4.29: Fundamento del joystick analógico de IBM

lectura captura el valor de la salida de los monoestables y del estado de los botones. En la tabla 4.46 se pueden ver las señales de este interfaz que daba soporte a dos joysticks (A y B).

Pin	Señal	Pin	Señal
1	+5V	2	A1
3	AX	4	GND
5	GND	6	AY
7	A2	8	+5V
9	+5V	10	B1
11	BX	12	GND
13	BY	14	B2
15	+5V		

Tabla 4.46: Señales de interfaz de joystick de IBM

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5.4. Sistema Gráfico

El sistema gráfico es el encargado de representar la información de texto y/o gráficos que ha sido introducida, almacenada y/o procesada por el sistema digital. Consta de tres componentes básicos: Buffer de video, controlador gráfico y pantalla de visualización (figura 4.30).

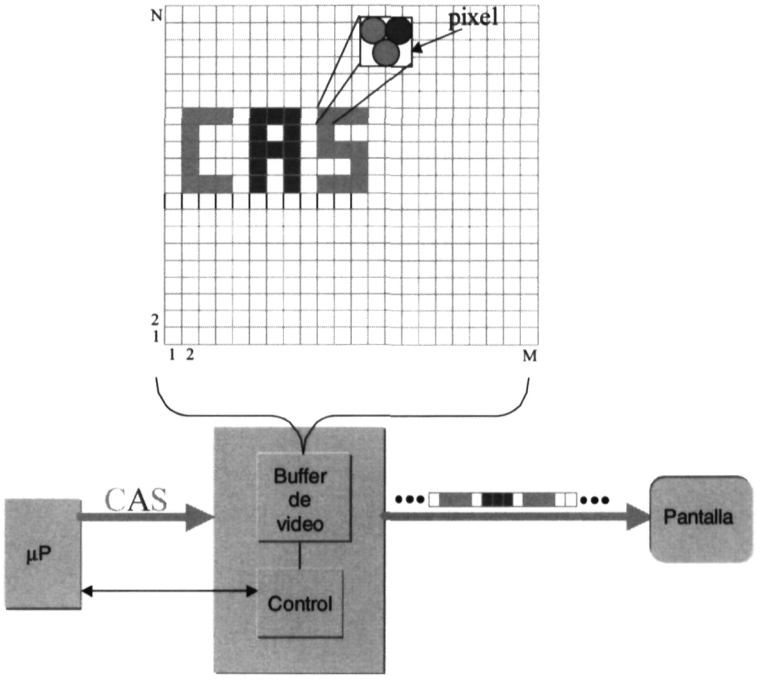


Figura 4.30: Pantalla de visualización y sistema gráfico

La información a representar hay que codificarla convirtiéndola en una matriz bidimensional de puntos gráficos[235] o pixels. Esta matriz de pixels se almacena en el buffer de video y es actualizada periódicamente por el sistema digital.

En una pantalla en color, cada pixel tiene tres componentes: rojo, verde y azul[236]. Si cada pixel se codifica con tres bytes, uno para cada componente

[235]En sistemas muy sencillos se pueden encontrar otros formatos no basados en puntos, como por ejemplo los displays de 7 segmentos.

[236]En inglés RGB (Red, Green, Blue). Según la teoría tricromática, a partir de los tres colores primarios RGB se puede obtener cualquier color.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

de color, se pueden obtener una gama de $C = 2^{8\times3} = 16777216$ colores. Es el denominado color verdadero.

Se define la resolución R de la pantalla como:

$$R = N \times M \tag{4.2}$$

donde N y M son el número de filas y columnas de pixels, respectivamente, de la matriz.

El tamaño del buffer de video depende de la resolución R y del número de colores C que puede tomar el pixel. La tabla 4.47 muestra la memoria requerida para diferentes valores de R y C. Se puede observar que 4MB son suficientes para una resolución de 1280×1024 y color verdadero (16,7 Millones de colores).

Colores/resolución	256	65536	16,7M
640×480	512KB	1MB	1MB
800×600	512KB	1MB	2MB
1024×768	1MB	2MB	3MB
1280×1024	2MB	3MB	4MB

Tabla 4.47: Resolución de pantalla

La pantalla de visualización es una matriz de elementos luminosos asociada a la matriz de pixels. El controlador lee periódicamente el buffer a una frecuencia denominada de refresco y asigna a cada punto de la pantalla el valor de un pixel.

En la figura 4.31 se ilustra como se reparte el tiempo entre el refresco de la pantalla y la actualización del buffer. Para una tasa de refresco f fija, conforme aumenta la resolución R y el número de colores C, el tiempo de actualización del buffer se va reduciendo llegando un momento en que será necesario, o bien reducir f, o bien usar técnicas aceleradoras como la memoria de doble puerto o los procesadores gráficos.

La actualización de buffer requiere un ancho de banda considerable que se podría estimar de la siguiente manera:

$$B = R \times \lg_2 C \times f \tag{4.3}$$

Por ejemplo, para una resolución de 1024×1024, color verdadero (3 bytes/pixel) y 25 imágenes/s, el ancho de banda asciende a 75MBps.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

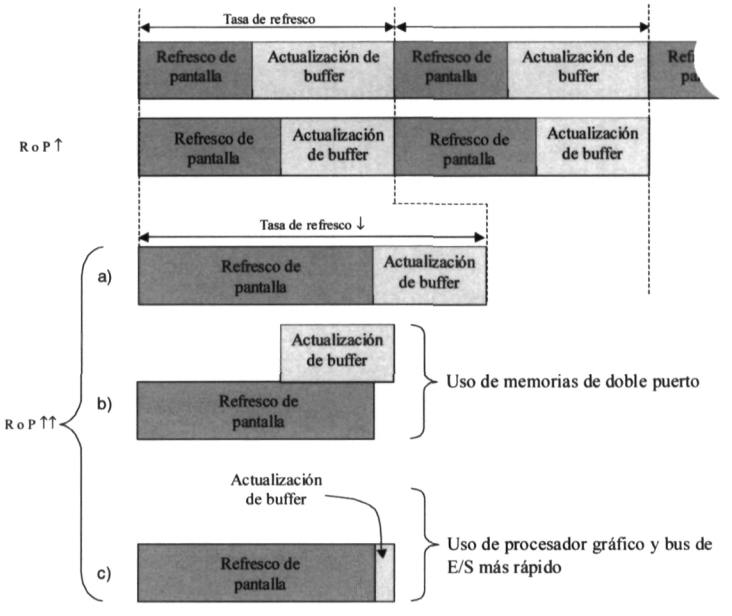


Figura 4.31: Estrategias de refresco de la pantalla

5.4.1. Pantallas de visualización

En función de la tecnología empleada, las pantallas de visualización se pueden clasificar en dos grandes grupos: pantallas de tubos de rayos catódicos (CRT, Cathode Ray Tube) y pantallas planas (FPD, Flat Panel Displays).

Hay dos tipos de FPD: emisivas y transmisivas. Las FPD emisivas emiten luz, siendo la pantalla de plasma (PDP, Plasma Display Panel) una de las tecnologías más representativas de este tipo de pantallas. Las FPD transmisivas o pantallas de cristal liquido (LCD, Liquid Crystal Display) requieren iluminación y son las utilizadas habitualmente en ordenadores personales. Las LCD pueden ser de matriz pasiva (STN) o activa (TFT, Thin Film Transistor).

A continuación se describen con más detalle las pantallas CRT y TFT.

5.4.1.1. Monitor CRT El tubo de rayos catódicos se inventó en el siglo XIX y desde los años 40 fue la tecnología usada en los aparatos de televisión. Posteriormente fue adoptado por los ordenadores personales donde se convirtió en el medio de visualización estándar hasta la llegada de las pantallas planas. Básicamente, el CRT consta de un cañón de electrones, bobinas deflectoras y una pantalla de material fosforescente (figura 4.32).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

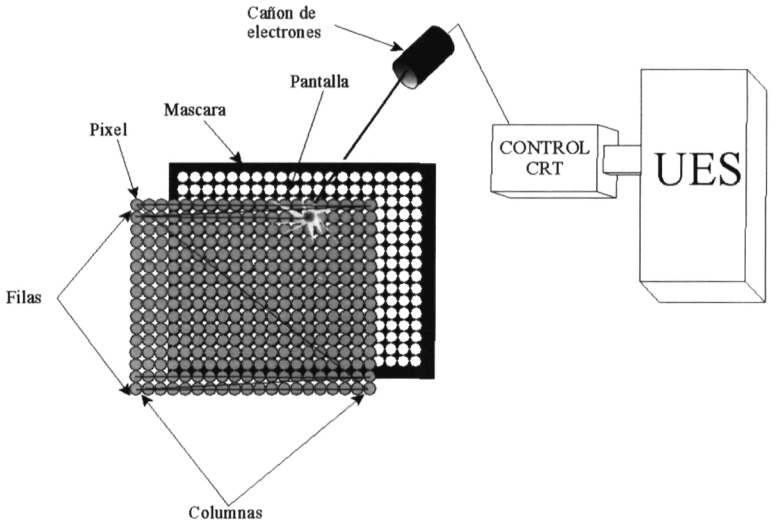


Figura 4.32: Monitor CRT

El cañón genera una haz modulado en intensidad que las bobinas deflectoras desvían para barrer la pantalla de visualización de izquierda a derecha y de arriba a abajo. La iluminación de cada punto de la pantalla depende de la intensidad del haz de electrones. Entre la pantalla y el cañón hay una máscara que permite separar los puntos para mejorar el contraste.

El fósforo de la pantalla brilla durante poco tiempo. Esto obliga a recorrer la pantalla una y otra vez a una velocidad que se denomina frecuencia de refresco que varía entre 50Hz y 85Hz. Se considera que a partir de 70Hz la imagen no parpadea, aunque algunos estándares como VESA superan este valor[237].

Hay dos formas de barrer la pantalla: en modo entrelazado y en modo no entrelazado. En el modo entrelazado, el cañón de electrones recorre primero las líneas impares y, a continuación, las pares. En este modo, el parpadeo es muy fuerte y suele provocar fatiga visual y hasta problemas de visión, por lo que no es muy recomendable. Su ventaja es que la frecuencia de refresco puede ser muy baja, reduciendo costes.

En los monitores en color hay tres haces de electrones correspondientes a

[237]Se recomiendan valores de 72Hz (14" y 15") y 75-85Hz (17").

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

cada uno de los colores primarios RGB. A su vez, cada pixel de la pantalla tiene las tres componentes de color, de manera que se puede conseguir cualquier color excitando adecuadamente cada componente con diferentes intensidades de los haces de electrones.

El manejo del CRT lo lleva a cabo el controlador CRT que se ubica en la denominada tarjeta adaptadora, aceleradora o controladora gráfica. La tarjeta adaptadora no procesa la información (es tarea exclusiva del microprocesador), sólo se limita a acondicionarla para su transferencia al monitor. Sin embargo, las aceleradoras y controladoras gráficas tiene una considerable capacidad de proceso que permite liberar al microprocesador de muchas tareas gráficas rutinarias.

5.4.1.2. Pantalla TFT La pantalla TFT se basa en un material especial denominado cristal líquido que tienen la capacidad de orientar la luz a su paso.

Las pantallas TFT están formadas por: una luz trasera (uno o más tubos de neón) para iluminar la pantalla, dos polarizadores con direcciones ortogonales entre sí y una matriz de celdas de cristal líquido contenida entre dos capas de vidrio (ver figura 4.33).

La luz procedente de los tubos de neón se polariza al atravesar el primer polarizador y penetra en el cristal líquido. Si no hay tensión aplicada a las celdas, éstas "tuercen" 90^o el plano de oscilación de la luz polarizada que, de esta forma, puede atravesar el segundo polarizador y llegar hasta el observador. Si se aplica tensión a las celdas de cristal líquido, éstas se ordenan de forma que no alteran el plano de oscilación de la luz. El resultado es que la luz no puede atravesar el segundo polarizador (ya que tiene orientación ortogonal) y la luz no llega al observador.

5.4.2. Adaptadora gráfica

La figura 4.34 muestra el diagrama de bloques de una tarjeta adaptadora típica donde se pueden distinguir los siguientes módulos: buffer de video, decodificador de atributos, generador de caracteres alfanuméricos, generador de la señal de video, controlador de CRT y control de modo.

El buffer de video es una memoria DRAM donde se almacenan los datos que se van a presentar en pantalla. Hay dos formas de representación: modo texto y modo gráfico. En el modo gráfico cada dato es un pixel de la pantalla. En el modo texto, cada dato es un carácter ASCII con atributos; El generador de caracteres alfanuméricos y el decodificador de atributos convierten los

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

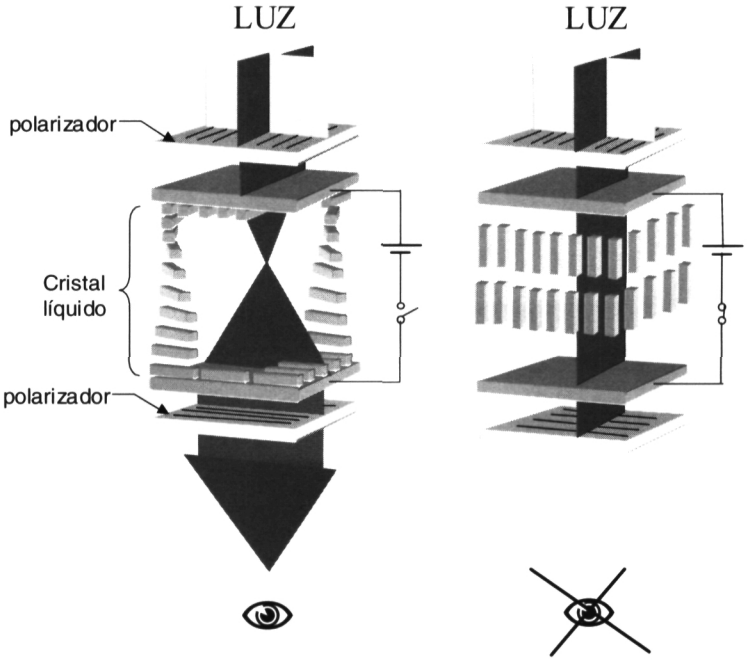


Figura 4.33: Fundamento de las pantallas de cristal líquido

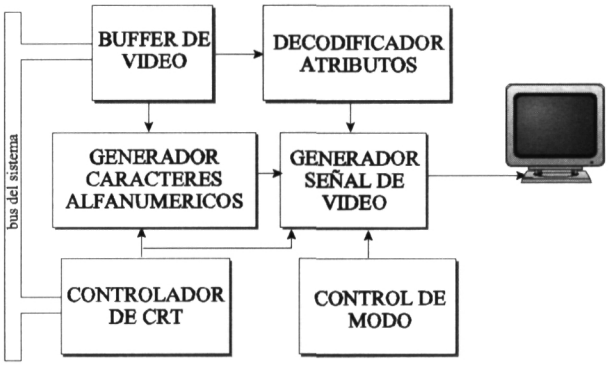


Figura 4.34: Diagrama de bloques de una adaptadora gráfica

caracteres en pixels y les asocian diversos atributos (color, intensidad, subrayado, etc..), respectivamente. En la figura 4.35 se puede ver un ejemplo del modo texto. El buffer puede contener más de una pantalla de video.

El controlador de CRT tiene varias funciones: Genera las señales de sin-

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

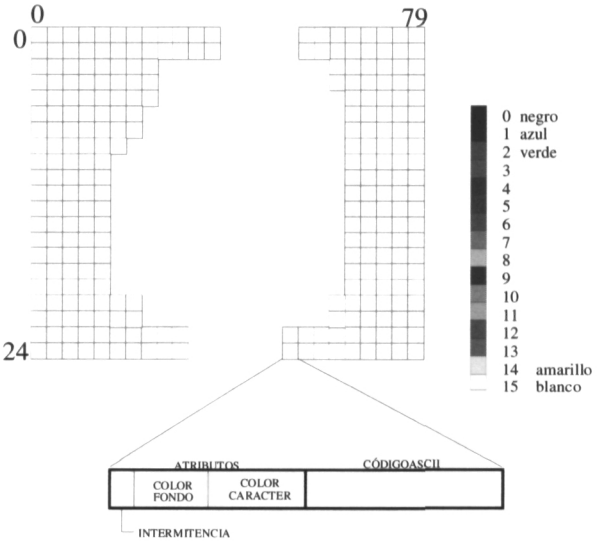


Figura 4.35: Modo texto de una adaptadora gráfica CGA

cronismo vertical y horizontal, sincroniza las señales de barrido en la pantalla con la lectura de datos del buffer de video, determina el tamaño y la posición del cursor y selecciona la parte del buffer de video a presentar en pantalla.

En 1981, IBM lanzó un par de tarjetas adaptadoras: MDA y CGA. MDA era monocromo[238] y sólo tenía modo texto. CGA tenía modo texto y modo gráficos aunque de baja resolución. En 1982, otra empresa comercializó el tarjeta adaptadora Hercules; era monocromo, compatible con MDA y de mejor resolución. Sin embargo la frecuencia de refresco de todas estas tarjetas dejaba mucho que desear (50Hz) provocando un molesto parpadeo.

En 1985, IBM lanzó el adaptador EGA que mejoraba sustancialmente tanto la resolución como el número de colores. Sin embargo, esto obligaba a las tarjetas a disponer de memoria propia, que solía oscilar entre 16 y 256KB.

En 1987, cuando IBM lanzó su modelo PS/2 presentó dos nuevos adaptadores gráficos: VGA, usado inicialmente sólo en los modelos de la gama alta 60, 70 y 80 y MCGA (MultiColor Graphic Array), usado para los ordenadores pequeños y que era un modo ampliado del CGA. Una característica diferenciadora de VGA es que requería la conversión de las señales de video digitales a analógicas. Para ello hacia uso de un chip denominado RAMDAC que inte-

[238] Al principio, el color del tubo de la pantalla era verde y negro y luego pasó a amarillo (ámbar y negro).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

graba memoria y conversores digitales analógicos. La memoria RAM de 256 palabras de 18 bits actuaba como una tabla de lookup[239]. Estas tarjetas trajeron un nuevo tipo de monitor: El monitor analógico RGB. En la tabla 4.48 se describen las señales del conector VGA.

Pin	Señal	Descripción
1	RED	Señal de video: componente rojo
2	GREEN	Señal de video: componente verde
3	BLUE	Señal de video: componente azul
4	N/C	No conectado
5	GND	Masa
6	RED RTN	Señal de retorno de video: componente rojo
7	GREEN RTN	Señal de retorno de video: componente verde
8	BLUE RTN	Señal de retorno de video: componente azul
9	N/C	No conectado
10	GND	Masa
11	GND	Masa
12	SDA	I2C data
13	HSYNC	Señal de sincronización horizontal
14	VSYNC	Señal de sincronización vertical

Tabla 4.48: Conector VGA DB-9/15

En 1989, aparecieron las SVGA. Estas tarjetas superaban el estándar VGA pero sus resoluciones no estaban normalizadas lo que hacía obligado el uso de un driver especial. VESA (Video Electronics Standards Association) definió unas especificaciones que evitaron utilizar un driver para cada programa. En la tabla 4.49 se muestra un resumen de características de las tarjetas adaptadoras más representativas.

Año	Nombre	Modos	Resolución	Otros
1981	MDA (Monocrome Display Adapter)	Texto monocromo	24×80 caracteres matriz de 9×16	Interfaz digital
1981	CGA (Color Graphics Adapter)	Texto y gráficos en color	320×200 (4 colores) 640×200 (2 colores) Matriz de 8×8	6845 de Motorola Interfaz digital
1982	Hercules	texto y gráficos monocromo	720×348	6845 de Motorola Compatible MDA 50Hz de refresco Interfaz digital
1985	EGA (Enhanced Graphics Adapter)	Texto y gráficos en color	640×350 (16 colores de paleta de 64) 80×43 caracteres matriz de 8×14	chip LSI diseñado por IBM Compatible con MDA y CGA Interfaz digital
1987	VGA (Video Graphics Array)	Texto y gráficos en color	640×480 (16 colores de paleta de 262144)	controlador VGA de IBM Compatible con MCGA, Hercules, CGA y EGA Interfaz analógico
1989	SVGA (Super VGA)	Texto y gráficos en color	SXGA de 1024×768	Incompatible IBM

Tabla 4.49: Evolución de las adaptadoras gráficas

A finales de la década de los 90, la llegada de las pantallas planas motivó la aparición de interfaces gráficos digitales como DVI (Digital Visual Interface).

[239] 256x(3x6). La RAM también se llamaba LUT (Look-Up Table), CLUT (Color LUT) o tabla intermedia.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

DVI usa el protocolo de señal TMDS (Transition Minimized Differential Signalling) y soporta 1280x1024 a 85Hz.

5.4.3. Aceleradora gráfica

Con la llegada de los interfaces gráficos de usuario, las tarjetas adaptadoras empezaron a quedarse cortas[240]. Esto provocó el desarrollo de las aceleradoras gráficas.

Los elementos básicos de una tarjeta aceleradora son (ver figura 4.36): interfaz a bus de alto ancho de banda[241] , buffer de video con memorias DRAM de altas prestaciones (DDR, GDR, etc..), RAMDAC de alta velocidad (200MHz-300MHz) y procesador gráfico GPU (Graphics Processor Unit). La aceleradora puede tener también funciones adicionales como captura de video, MPEG y sintonización de TV.

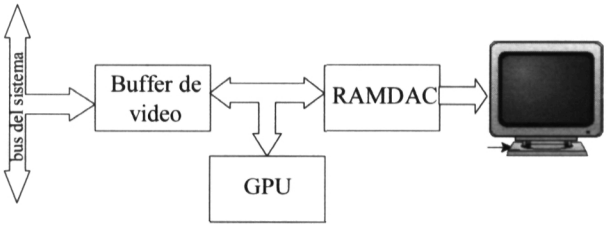


Figura 4.36: Diagrama de bloques básico de una aceleradora gráfica

El GPU incluye el generador de reloj, la controladora VGA, funciones aceleradoras y, en algunas ocasiones, el RAMDAC. El GPU implementa poderosas funciones gráficas[242] que descargan al procesador de muchas tareas rutinarias[243].

[240] La técnica de arrastrar y soltar con el ratón exige el movimiento de amplias áreas de memoria en las que se almacena la imagen de cada icono que está presente en la pantalla.

[241] PCI, AGP (Accelerated Graphics Port) y, últimamente, PCI Express.

[242] El programador accede a estas funciones a través de potentes librerías gráficas estándar como OpenGL o DirectX.

[243] Antes, si el procesador quería pintar un circulo en pantalla, tenía que calcular el valor de cada uno de los pixels implicados y enviarlo al buffer de video. Ahora el procesador se limita a indicar al procesador gráfico que quiere pintar un circulo de color H y radio R en la coordenada X.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5.4.3.1. Aceleración 3D Hoy en día, la mayoría de los videojuegos y las aplicaciones comerciales usan objetos 3D. Un mundo geométrico[244] 3D está compuesto de fuentes de luz, objetos 3D y el punto de vista del usuario (ver figura 4.37).

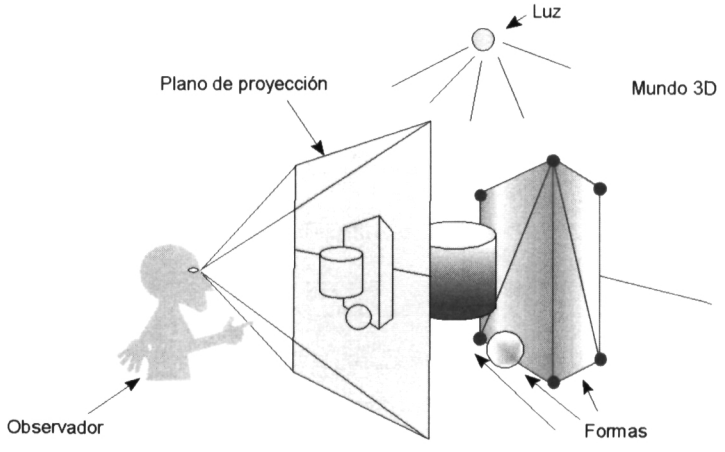


Figura 4.37: Mundo 3D y su proyección 2D sobre la pantalla del usuario

Los objetos 3D están formados por polígonos, normalmente triángulos, que se definen por sus vértices. Los vértices, las fuentes de luz y el punto de vista del usuario tienen unas coordenadas 3D. La aceleradora 3D tiene que calcular para cada fotograma (frame) que es lo que verá el usuario en una pantalla 2D en función de su punto de vista y las variaciones que se produzcan en los diferentes elementos del mundo 3D. Esto requiere dos fases (figura 4.38): Cálculo geométrico y representación (rendering).

En la primera fase, se recalculan la posición de todas las coordenadas 3D visibles y la iluminación de los objetos. En la segunda fase se proyecta lo que ve el usuario en un plano 2D (la pantalla) y se calculan el valor de los pixels. La figura 4.39 muestra estas dos fases en lo que se denomina el pipeline 3D, ya que todo el proceso se descompone en muchas etapas independientes.

En la fase geométrica:

1. Se ubican los diferentes objetos 3D en el mundo virtual modificando sus coordenadas (traslación).

[244] Hoy en día también se consiguen mundos 3D basados en imágenes, véase como ejemplo el visor QuicktimeVR de Apple.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

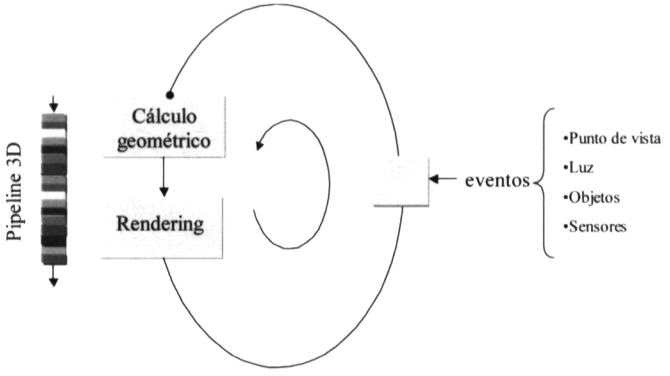


Figura 4.38: Bucle 3D

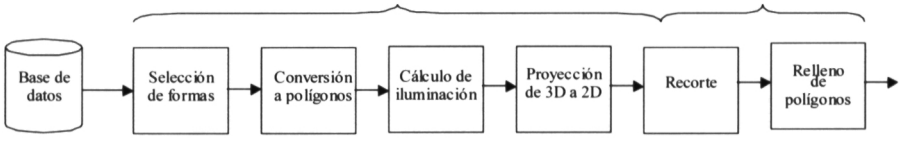


Figura 4.39: Pipeline 3D

2. Dependiendo del punto de vista del observador, se eliminan (clipping) todos los objetos del mundo que no se verán en pantalla.

3. Se giran, deforman, trasladan los objetos pertinentes (transformación).

4. En función de la ubicación de los focos de luz, se calcula la iluminación de cada objeto. Para ello, se usan fórmulas matemáticas que determinan el color del objeto en función sus propiedades reflectivas (difuso, reflectante, etc..) y del tipo de luz incidente (ambiente, direccional, etc..).

5. Se divide la superficie de los objetos irregulares en objetos regulares (polígonos, que suelen ser triángulos) fácilmente manipulables por el PC. Los polígonos se definen mediante sus vértices.

Todos los procesos anteriores requieren computación intensiva en punto flotante que puede ser llevada a cabo por el microprocesador o por procesadores especializados ubicados en la aceleradora 3D.

En la fase de representación, se llevan cabo las siguientes tareas:

1. Sombreado de los polígonos (shading). Hay varios tipos: flat shading, gouraud shading, phong shading, dithering, etc..

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2. Z-buffering. Es la técnica más utilizada para determinar qué objetos se verán en pantalla y cuáles quedarán ocultos por otros que se interponen entre ellos y el observador (figura 4.40). Consiste en almacenar el valor de profundidad para cada pixel en el display. Cuando se representan los objetos, se almacena la coordenada Z de cada pixel (X,Y,Z) en el Z-buffer. Cuanto más pequeña sea Z, más próximo estará el pixel del objeto al espectador. Cuando otro objeto tiene las mismas coordenadas (X,Y) se compara su coordenada Z con la almacenada en el Z-buffer y si es menor se sustituye.

3. Mapeo de texturas. Hay dos formas de añadir complejidad a la escena. Aumentando el número de polígonos (más geometría) o mediante mapeo de texturas. Una textura es un dibujo (mapa de bits) que se "pega" al polígono. La corrección de perspectiva para dar profundidad a la textura mapeada en un polígono también es un aspecto importante a tener en cuenta. Las texturas sufren de aliasing por lo que se usan diversas técnicas para aliviarlo: filtro bilineal, MIP mapping, filtro trilineal, etc..

4. Alpha blending controla la transparencia de un objeto (agua, vidrio, efectos atmosféricos, niebla).

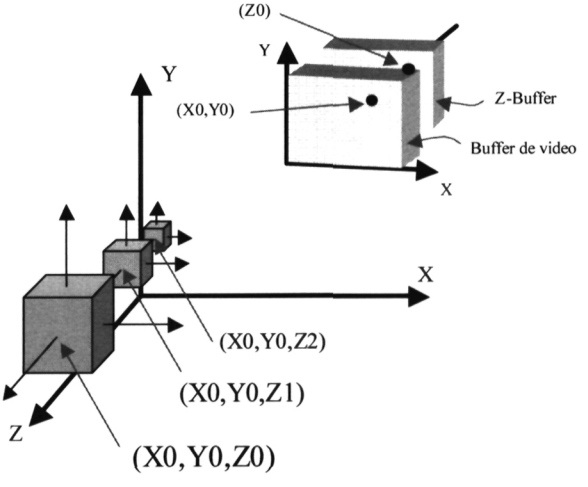


Figura 4.40: Z buffer

Las técnicas anteriores necesitan memoria y gran ancho de banda. Por ejemplo, la técnica Z-buffer requiere un tamaño de memoria que depende de

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

la resolución y de los bits utilizados para codificar la profundidad del eje Z (16 ó 32 bits). Las texturas pueden estar almacenadas en la memoria principal (es el caso, de AGP) o en el buffer de video. Hay algoritmos de caching o swapping que determinan qué texturas se almacenan en el buffer de video y cuáles en memoria principal. Es muy común encontrar doble o triple buffer dentro del buffer de video para acelerar la representación gráfica. La figura 4.41 muestra la evolución de las aceleradoras gráficas 3D con respecto a tres parámetros: Millones de vertices por segundo, millones de pixels por segundo y millones de transistores.

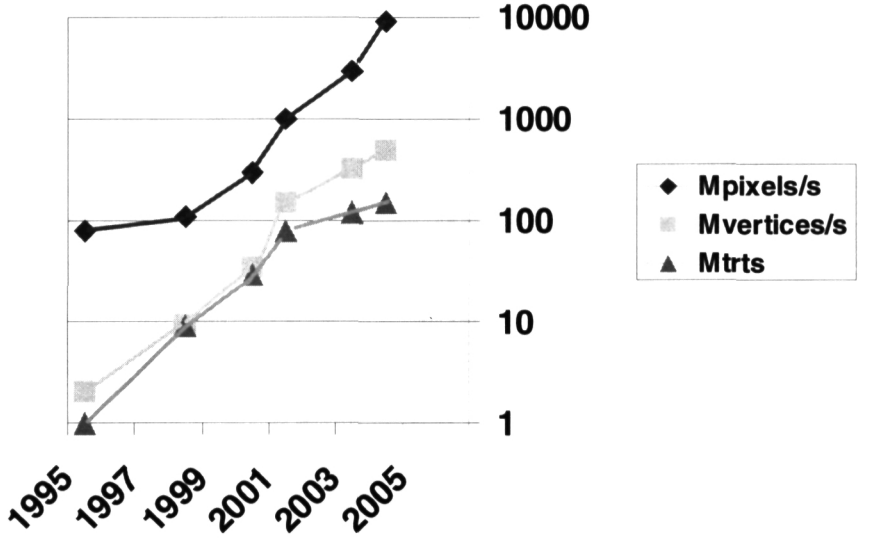


Figura 4.41: Evolución de las aceleradoras gráficas 3D. NVidia 2000

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5.4.4. AGP (Accelerated Graphics Port)

La avalancha de aplicaciones 3D, el video y el constante aumento de resolución y colores en los gráficos saturaron al bus PCI. Éste no daba abasto para transferir texturas de memoria principal a la tarjeta gráfica y, además, atender a otros dispositivos[245]. Una solución propuesta por Intel fue el bus local AGP (Accelerated Graphics Port). Este bus, a diferencia del PCI, es específico de gráficos y permite mejorar sustancialmente los gráficos 3D y el video.

AGP es un bus similar a PCI con las siguientes mejoras:

- Tiene un ancho de banda superior (ver tabla 4.50).

Versión	Ancho de banda
AGP×1	266MBps = 66MHz×4 bytes
AGP×2 (1998)	533MBps = 66MHz×2×4 bytes
AGP×4(1999)	1GBps = 66MHz×4×4 bytes
AGP×8(2000)	2GBps = 66MHz×8×4 bytes

Tabla 4.50: Ancho de banda del bus AGP

- Dispone de 8 líneas de dirección adicionales (sideband addressing) para poder emitir nuevas direcciones y solicitudes simultáneamente mientras los datos de solicitudes previas ocupan el bus multiplexado de datos/direccione
- Usa técnicas pipeline. En PCI, la solicitud $n+1$ no empieza hasta que la transferencia de datos de la solicitud n termina. En AGP, se pueden enviar muchas solicitudes de transferencia sin tener que esperar a que terminen.
- Mapeo de memoria. Algunos segmentos del sistema de memoria pueden ser dinámicamente reservados por el S.O. para su uso por el procesador gráfico (memoria AGP o memoria de video no local). Estos segmentos son pequeños y discontinuos en la memoria principal y asignados en tiempo de ejecución. Gracias a un traductor de direcciones (GART, Graphics Address Remapping Table), esta memoria es vista por el controlador como un espacio continuo de bytes, permitiendo el acceso a grandes estructuras de datos (1KB a 128KB) de forma sencilla.

La figura 4.42 muestra un ciclo de lectura AGP×8. Se puede observar que el reloj se mantiene a 66MHz y que se llevan a cabo 8 transferencias en un ciclo de reloj. Esto se consigue utilizando las señales `AD_STB` que acompañan al bus de datos de origen a destino y marcan cuando los datos son válidos[246].

[245] Controladores de disco duro, controladores de red de área local, etc.

[246] Es la técnica source synchronous.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

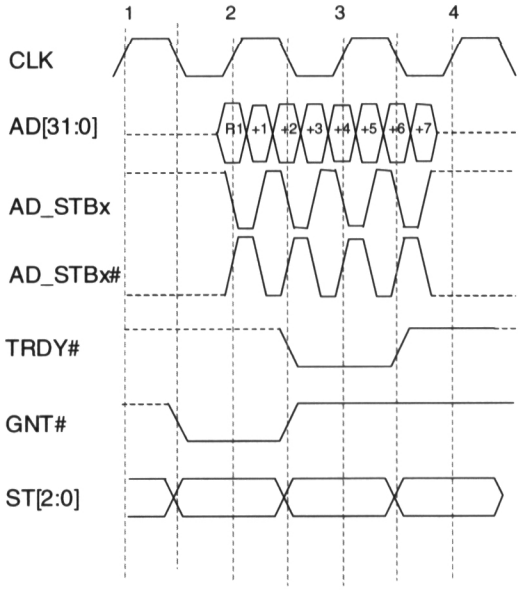


Figura 4.42: Ejemplo de ciclo de lectura de AGP×8

5.4.5. Ejemplos

5.4.5.1. Escritura en el buffer de video Este curioso ejemplo escribe en el buffer de video de PC haciendo uso del depurador. Abra una ventana MSDOS y configúrela a pantalla completa[247]. A continuación limpie la pantalla con el comando `cls`, entre en el depurador y escriba lo siguiente:

```
c:\>cls <CR> c:\>debug <CR> - e b800:0100 48 9f 4f 9f 4c 9f 41 9f
<CR>
```

Repase el apartado de tarjeta adaptadora básica para descubrir el formato de datos que se usa y repita la experiencia cambiando el texto y los colores.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

[247] Para ello, pulse Alt Enter.

5.5. Sistema de audio

En este apartado se describen la SoundBlaster16 de Creative Technology Ltd., la especificación AC´97 y el interfaz MIDI.

5.5.1. SoundBlaster 16

La SoundBlaster16 (SB16) de Creative Labs. es capaz de generar sonidos sintéticos FM y de reproducir y grabar sonidos digitalizados (figura 4.43). La capacidad de digitalizar sonidos va desde 8 bits, modo mono y frecuencia de muestreo a 5000 Hz hasta alta fidelidad (16 bits estéreo a 44KHz).

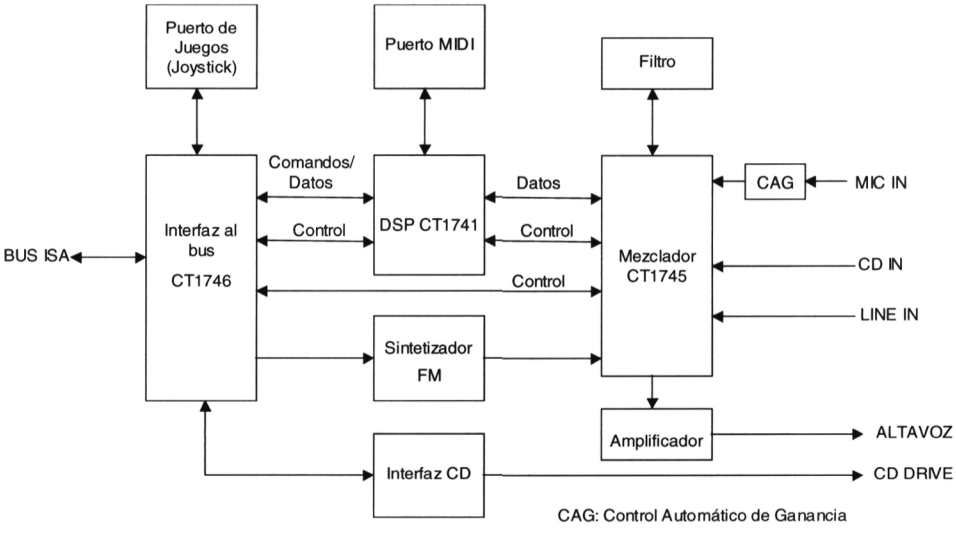


Figura 4.43: Diagrama de bloques de SoundBlaster 16

En el diagrama de bloques de la figura 4.43 se distingue:

- Procesador de Sonido Digital CT1741 (DSP, Digital Sound Processor). Está encargado de la reproducción y grabación de sonido digital mono/estéreo de 8/16 bits, descompresión ADPCM (4:1, 3:1, 2:1), control de la tasa de muestreo, interfaz MIDI y MPU-401 UART, control de altavoz y manejo de transferencias DMA.
- Mezclador CT1745. Mezcla señales de diferentes fuentes de sonido y proporciona facilidades de control de volumen. También selecciona la fuente de grabación.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Sintetizador FM (Frecuency Modulation). Genera sonidos de instrumentos musicales. Hace uso del chip OPL3 de Yamaha.

- Interfaz al bus ISA CT1746. Provee modo MPU-401 UART compatible MIDI, FIFOs para grabación y reproducción de audio digital y canal de interrupción compartido por tres procesos diferentes (audio digital de 8 bits, audio digital de 16 bits y MIDI).

La síntesis de sonidos FM se obtiene combinando generadores (operadores) de ondas senoidales que permiten obtener el timbre de la señal. FM es una solución adecuada para aplicaciones multimedia con pocas exigencias. Sin embargo, para calidad profesional se usan tablas de ondas. Las tablas de ondas son una librería de muestras digitalizadas de instrumentos reales. Mediante procesado de la señal las muestras se pueden modificar para representar diferentes tonos o combinar para generar todo tipo de sonidos musicales. Para conseguir las variaciones de amplitud que tiene los sonidos reales se usan funciones de envolvente.

5.5.1.1. Espacio de direcciones de E/S La SB16 tiene 16 puertos de E/S que se utilizan para síntesis FM, configuración del mezclador, programación del DSP y acceso al CD-ROM. En la tabla 4.51 se muestra una relación de los mismos referidos a su dirección BASE[248]. El mezclador tiene muchos registros. Para programarlo primero se escribe la dirección del registro en el puerto de direcciones y luego se escribe el dato en el puerto de datos.

5.5.1.2. Modos de transferencia La SB16 tiene 5 modos de transferencia: directo, DMA de ciclo único, DMA con autoinicialización, DMA de alta velocidad y DMA ADPCM.

En el modo directo, el programa de aplicación maneja la transferencia[249]. Sólo trabaja con datos PCM sin signo de 8 bits mono.

En el modo DMA de ciclo único (single), el DSP se programa para hacer sólo una transferencia. Al final de la misma, el DSP genera una interrupción y el programa de aplicación reprograma el controlador de DMA y el DSP con

[248] La dirección BASE puede ser `220H`, `240H`, `260H` ó `280H`. Los registros de música FM también puede ser accedidos a través de las direcciones `388H` y `389H`. Para el soporte del modo MU-401 UART se usa la dirección `BASE= 300H` ó `330H`. `BASE+0H` es el puerto de datos de lectura y escritura y `BASE +1H` el puerto de estado (lectura) y puerto de comando (escritura).

[249] Se suele usar la rutina de interrupción del timer del PC modificada para marcar la frecuencia de muestreo.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Dirección	Registro	Acceso
BASE+0H	Puerto de estado de música FM	lectura
BASE+0H	Puerto de direcciones de registro de música FM	escritura
BASE+1H	Puerto de datos de música FM	escritura
BASE+2H	Puerto de estado de música FM avanzado	lectura
BASE+2H	Puerto de direcciones de registro de música FM avanzado	escritura
BASE+3H	Puerto de datos de música FM avanzado	lectura
BASE+4H	Puerto de direcciones de registro del mezclador	escritura
BASE+5H	Puerto de datos del mezclador	lectura y escritura
BASE+6H	Reset del DSP	escritura
BASE+8H	Puerto de estado de música FM	lectura
BASE+8H	Puerto de registro de música FM	escritura
BASE+9H	Puerto de datos de música FM	escritura
BASE+AH	Puerto de datos de lectura del DSP	lectura
BASE+CH	Puerto de escritura de comando y datos de DSP	escritura
BASE+CH	Estado del buffer de escritura del DSP (bit 7)	lectura
BASE+EH	Estado del buffer de lectura del DSP (bit 7)	lectura
BASE+10H	Registro de datos o comandos de CDROM	lectura y escritura
BASE+11H	Registro de estado de CDROM	lectura
BASE+12H	Registro de reset de CDROM	escritura
BASE+13H	Registro de habilitación de CDROM	lectura

Tabla 4.51: Puertos de E/S de la SoundBlaster 16

el tamaño del siguiente bloque. Trabaja con datos PCM sin signo de 8 bits, 16 bits con signo y transferencias comprimidas ADPCM.

En el modo DMA con autoinicialización, el controlador de DMA y el DSP sólo se programan una vez con el tamaño de la transferencia de bloque. El DSP genera interrupciones a intervalos constantes de acuerdo al tamaño de transferencia de bloque programado. Se suele usar el método de doble buffering[250].

5.5.1.3. Programación En primer lugar hay que detectar la presencia de la tarjeta y su configuración[251]. A continuación, se inicializa la tarjeta con el siguiente procedimiento:

1. Escribir un 1 en el puerto de reset.
2. Esperar 3 microsegundos.

[250]El tamaño de transferencia de bloque de DSP se pone a la mitad del tamaño del buffer de almacenamiento DMA. Después de cada transferencia de bloque del DSP, el DSP genera una interrupción que el programa de aplicación atiende para transferir datos a/desde (dependiendo de reproducción o grabación) a esa mitad de buffer DMA cuyos datos acaban de ser transferidos.

[251]Para ello se suele acceder a las denominadas variables de entorno `BLASTER` del archivo `AUTOEXEC.BAT` del PC. Son una línea con el aspecto siguiente: `SET BLASTER=A220 I5 D1 H5 M220 P330 T6`. Los programas que acceden a la SoundBlaster suelen leer esta línea para conocer la dirección base de E/S de la tarjeta (parámetro `A`), el número de interrupción y canal de DMA de 8 bits utilizado (parámetros `I` y `D`, respectivamente), el canal de DMA de 16 bits (parámetro `H`), la dirección base de E/S del mezclador y del puerto MIDI (parámetros `M` y `P`, respectivamente) y la versión de la tarjeta (parámetro `T`).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3. Escribir un 0 en el puerto de reset.
4. Sondear el bit 7 del puerto de estado del buffer de lectura hasta que se ponga a 1.
5. Leer el puerto de datos de lectura y recibir `AAH`.

El DSP requiere unos 100 microsegundos para autoinicializarse. El reset habrá fallado si después de este periodo de tiempo, la tarjeta no devuelve ningún dato o devuelve un valor distinto de `AAH`[252].

Para escribir en el DSP se hace uso de una técnica de polling:

1. Sondear el bit 7 del puerto de estado del buffer de escritura hasta que se ponga a 0.
2. Escribir el valor en el puerto de escritura.

y para la lectura:

1. Sondear el bit 7 del puerto de estado del buffer de lectura hasta que se ponga a 1.
2. Leer el valor del puerto de lectura.

5.5.1.4. Interrupciones y DMA La SB16 permite la configuración software de líneas de interrupción y canales de DMA. Para ello, hace uso de dos registros sitos en el mezclador: El registro `80H` es el registro de configuración de interrupciones y el `81H` el de configuración de canal de DMA[253].

La interrupción del DSP puede deberse a 4 causas distintas: DMA 16 bits, DMA 8 bits, SB-MIDI y MPU401 MIDI. Por tanto, la rutina de servicio de interrupción tendrá que preguntarle a la tarjeta cuál ha sido la fuente de interrupción. Para ello, consulta el registro de estado de interrupción que es el registro `82H` del mezclador (ver tabla 4.52).

bits(7:3)	bit2	bit1	bit0
reservados	*MPU-401*	*DMA 16 bits*	*DMA 8 bits y SB-MIDI*

Tabla 4.52: Registro 82H del mezclador de la SoundBlaster

[252] En este caso, es probable que la tarjeta no esté instalada o que la dirección de E/S no sea la correcta.

[253] Ver manual de la SB16 para más información.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5.5.1.5. Guía de programación de la SoundBlaster 16 El procedimiento general para programar la SB16 para una transferencia de DMA es el que se muestra a continuación (figura 4.44):

1. Inicializar controlador de interrupciones y de DMA (se hace al arrancar el sistema).
2. Detectar e inicializar la SB16.
3. Crear un buffer de memoria.
4. Abrir fichero de audio.
5. Instalar la rutina de servicio de interrupción del DSP.
6. Habilitar la interrupción de la soundblaster en el controlador de interrupciones.
7. Habilitar la interrupción de la soundblaster en el DSP.
8. Programar la transferencia en el controlador de DMA.
9. Programar la tasa de muestreo, modo de transferencia DMA y longitud en el DSP. Se iniciará la transferencia y cuando termine, la soundblaster generará una interrupción.
10. Atender las interrupciones del DSP.
11. Enviar el comando `EOI` al controlador de interrupciones.
12. Restablecer la rutina de servicio de interrupción original

La rutina de servicio de interrupción del DSP debe seguir los siguientes pasos:

1. Ir a 3 si no hay más bloques de datos que transferir.
2. Si está configurado en modo de transferencia único (single), programar el controlador de DMA y el DSP para el siguiente bloque. Si está configurado en modo autoinicialización, transferir datos entre el buffer de DMA y el buffer de almacenamiento y si desea salir enviar el comando `exit` o conmutar a modo single.[254]
3. Reconocer la interrupción del DSP.
4. Enviar el comando `EOI` al controlador de interrupciones.

[254] Este paso puede ejecutarse en el programa principal si aquí se activa una variable global que le comunique que se ha producido la interrupción

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

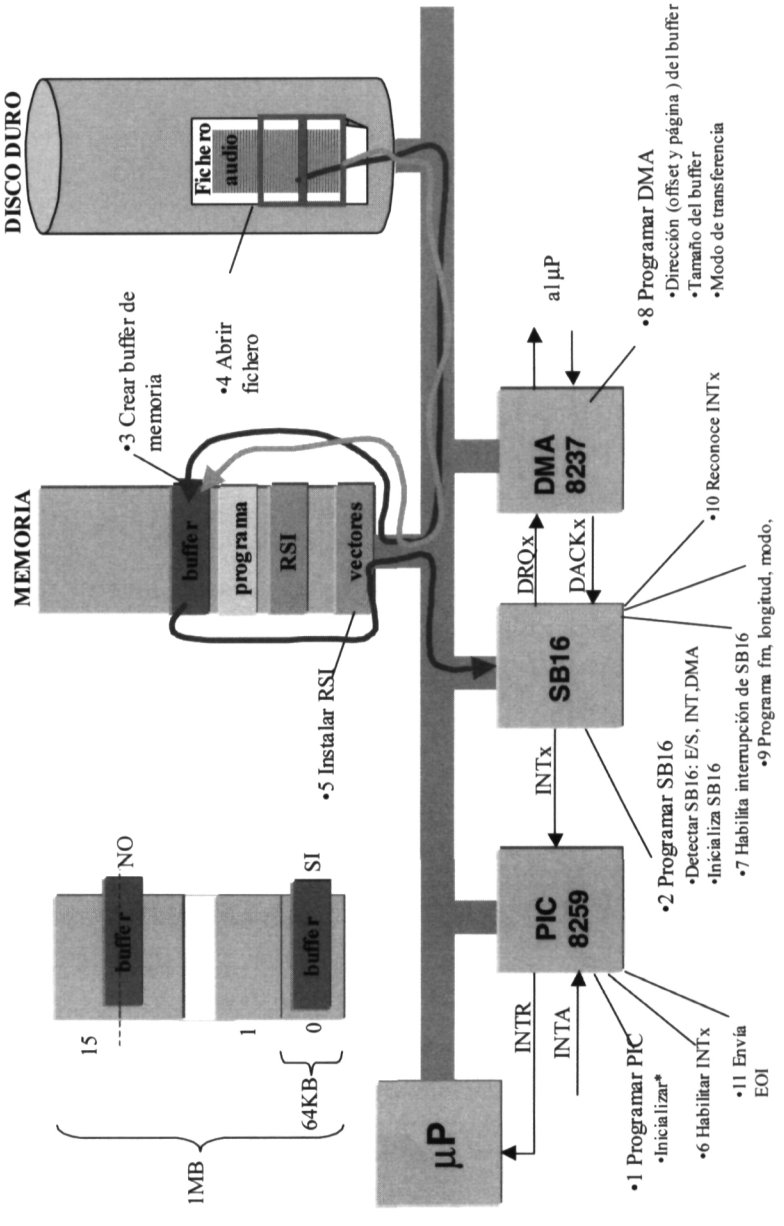


Figura 4.44: Proceso de reproducción de un fichero de audio

A continuación, se muestra un ejemplo de programación de la SB16 para reproducción estéreo con cuantificación de 16 bits, tasa de muestreo de 22050Hz y modo DMA con autoinicialización.

1. Crear un buffer en memoria que esté dentro de una página (no debe cruzar los límites de una página).
2. Instalar la rutina de servicio de interrupción del DSP.
3. Habilitar la interrupción usada.
4. Programar el controlador de DMA para transferencias de 16 bits modo autoinicialización.
5. Programar la tasa de muestreo del DSP. Se hacen tres escrituras en el puerto `2XCH`: 1) comando de salida (`41H`) 2) byte alto (`56H` para 22050 Hz) y 3) byte bajo (`22H` para 22050 Hz).
6. Programar modo de transferencia y tamaño[255] (número de transferencias menos 1). Se hacen cuatro escrituras en el puerto `2xC`: 1) comando de E/S (`BEH`), 2) comando de modo(`30H`), 3) byte alto de tamaño y 4) byte bajo de tamaño.

[255] El tamaño de la transferencia será la mitad del tamaño del buffer de almacenamiento.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5.5.2. Sistema AC´97

La especificación Audio Codec'97 (AC'97) define una arquitectura digital para audio y modems. La arquitectura consta de un controlador AC´97 y uno o varios codec AC´97 que se interconectan mediante un interfaz serie digital denominado AC-link. (figura 4.47).

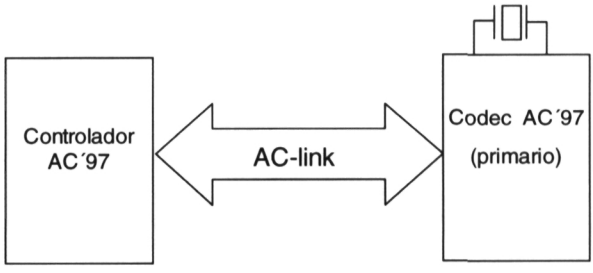


Figura 4.45: Arquitectura AC´97

AC-link es una interfaz serie bidireccional de velocidad fija que maneja múltiples flujos de audio PCM (Pulse Code Modulation) de entrada y salida así como acceso a registros de control. AC-link utiliza un esquema de multiplexación por division en el tiempo (TDM, Time Division Multiplexed) para configurar una trama bidireccional de 12 canales (slots) de audio de 20 bits (figura 4.46).

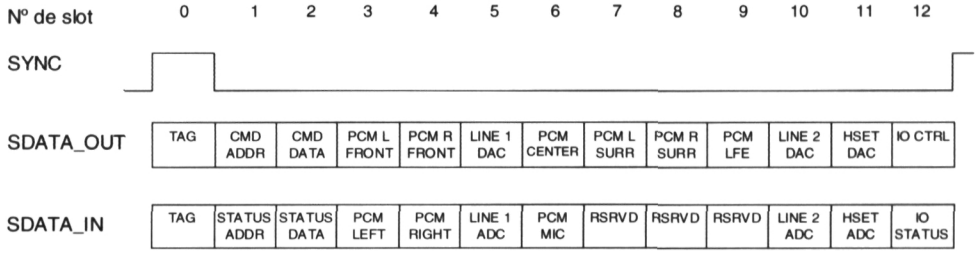


Figura 4.46: Trama AC´97

Hay 12 flujos de entrada (SDATA_IN) hacía el controlador y 12 de salida (SDATA_OUT) hacía el codec. Las tablas 4.53 y 4.54 muestran la descripción de los canales de salida y entrada, respectivamente.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Slot	Nombre	Descripción
0	SDATA_OUT TAG	MSBs indica qué canales contienen datos válidos; LSBs transporta la identificación del codec (Codec ID)
1	Control CMD ADDR write port	Dirección de registro del codec de 7 bits más bit de comando de lectura/escritura
2	Control DATA write port	Datos de escritura de registro de comandos de 16 bits.
3,4	PCM L&R DAC playback	Datos PCM de 20, 18 y 16 bits para canales derecho e izquierdo
5	Modem Line 1 DAC	Datos de modem de 16 bits para salida de línea 1 de modem
6,7,8,9	PCM Center, surround L&R, LFE	Datos PCM de 20, 18 y 16 bits para canales central, surround L&R y LFE
10	Modem Line 2 DAC	Datos de modem de 16 bits para salida de linea 2 de modem
11	Modem handset DAC	Datos de modem 16 bits para salida handset de modem
12	Modem IO control	Puerto de escritura GPIO para control de modem
10-11	SPDIF Out	Ancho de banda AC-link opcional para salida SPDIF
6-12	Double rate audio	Ancho de banda AC-link opcional para canales L, C y R de 88.2 or 96 KHz. Los canales usados son controlados por los bits.

Tabla 4.53: Trama AC´97: Canales de salida

Slot	Nombre	Descripción
0	SDATA_IN TAG	MSBs indica qué canales contienen datos válidos
1	STATUS ADDR read port	MSBs es la dirección de registro de eco. LSBs indica qué canales solicitan datos
2	STATUS DATA read port	Datos de lectura del registro de comando de 16 bits
3,4	PCM L&R ADC record	Datos PCM de 20, 18 ó 16 bits desde las entradas derecha e izquierda
5	Modem Line 1 ADC	Datos de modem de 16 bits desde la entrada de línea 1 de modem.
6	Dedicated Microphone ADC	Datos PCM de 20, 18 ó 16 bits desde entrada tercer ADC opcional
7,8,9	Vendor reserved	Específico del vendedor (entrada mejorada para docking, array de micrófonos, etc..)
10	Modem Line 2 ADC	Datos de modem de 16 bits procedentes de entrada de línea 2 de modem.
11	Modem handset input ADC	Datos de modem de 16 bits para entrada de handset de modem.
12	Modem IO status	Puerto de lectura GPIO para estado del modem

Tabla 4.54: Trama AC´97: Canales de entrada

5.5.2.1. Interfaz AC-link La figura 4.47 y la tabla 4.55 muestran las señales del interfaz AC-link.

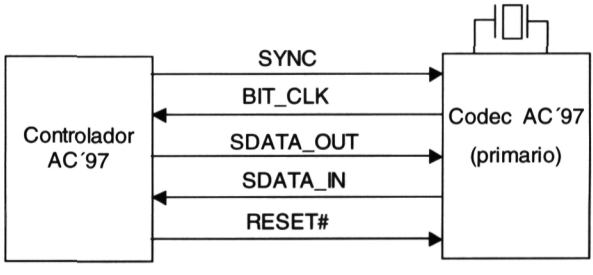


Figura 4.47: Arquitectura AC´97

La señal de reloj BIT_CLK puede ser generada por el codec, por el controlador o por un generador externo. En el primer caso, se dice que el codec actúa como fuente del reloj y en el resto que actúa como consumidor del reloj.

El codec se configura como consumidor si, con la señal RESET# activada,

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Señal	Dirección	Descripción
XTL_IN	entrada	Cristal de 24.576 MHz u oscilador de 24.576 MHz ó 14.318MHz.
XTL_OUT	salida	Cristal de 24.576 MHz si se usa cristal como fuente de reloj. En caso contrario reservado.
SDATA_OUT	entrada	Señal de datos. Es un flujo TDM de entrada del codec AC´97 procedente del controlador AC´97.
BIT_CLK	bidireccional	Codecs primarios: Reloj de 12.288 MHz. Es de salida para codecs primarios y de entrada para codecs secundarios
SDATA_IN	salida	Señal de salida. Es un flujo TDM de salida del codec AC´97 Codec hacia el controlador AC´97.
SYNC	entrada	Señal de sincronismo de trama de 48 KHz.
RESET#	entrada	Señal de reset hardware del maestro AC´97.

Tabla 4.55: Interfaz AC-link

el codec detecta sobre `BIT_CLK` un reloj (procedente del controlador o de un generador externo) durante al menos 5 ciclos de reloj antes de que **`RESET#`** se desactive. Si no detecta nada, entonces el codec se configura como fuente. La detección se hace cada vez que la señal **`RESET#`** se activa.

La figura 4.48 muestra el protocolo AC-link. Los datos se transmiten en el flanco de subida de la señal `BIT_CLK`, y se muestrean en recepción en el inmediato flanco de bajada. La trama comienza con la activación por parte del controlador de la señal de sincronismo[256] **`SYNC`** que es un ciclo de reloj de 48 KHz. Dentro de la trama hay una fase de etiquetas y otra de datos. En la fase de etiquetas se envía información de identificación y validación de canales y en la fase de datos los canales propiamente dichos.

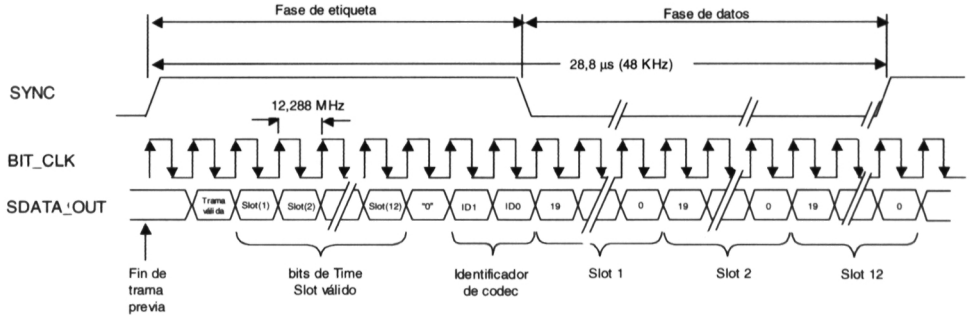


Figura 4.48: Trama AC´97 de salida

5.5.2.2. Codec AC´97 La figura 4.49 muestra el diagrama de bloques del codec AC´97 donde se distinguen como bloques básicos: el interfaz digital,

[256] El controlador genera `SYNC` dividiendo la señal `BIT_CLK` por 256 y aplicando algunas condiciones para configurar su ciclo de trabajo.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

los conversores analógico digitales y el módulo de mezcla de fuentes y control de ganancia. El resto de bloques (marcados en gris) son opcionales.

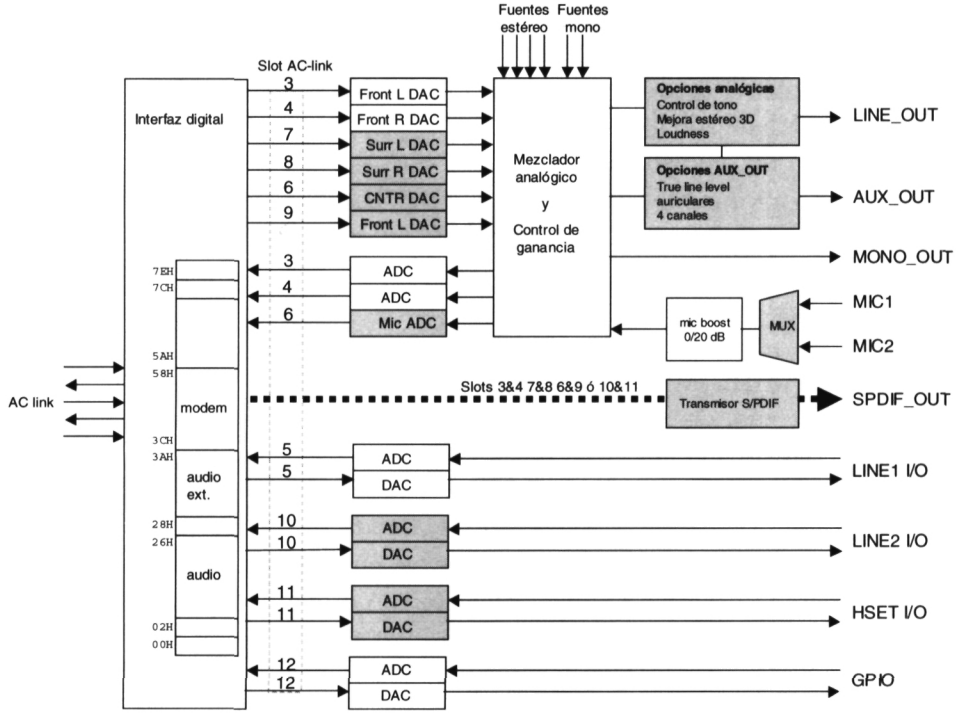


Figura 4.49: Diagrama de bloques de Codec AC´97

El interfaz digital genera el interfaz AC-link y gestiona el conjunto de registros de control y configuración del codec. El codec es un esclavo del controlador.

Los conversores digital analógicos (DACs) soportan una salida PCM estéreo (y opcionalmente multicanal) que contiene una mezcla generada en el controlador AC´97 de todas las fuentes de audio digitales. El mezclador combina la salida de los DACs con fuentes analógicas; la señal resultante, que puede ser opcionalmente procesada para obtener mejoras en el control de volumen, tonos y estéreo 3D, se envía a las salidas `LINE_OUT` y `AUX_OUT`. Esta última se puede configurar como salida de auriculares o como salida adicional para un sistema de 4 canales.

Diferentes configuraciones de las salidas mencionadas permiten a AC´97 soportar una amplia variedad de opciones de salida de audio, incluyendo:

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

estéreo ó 6 canales analógicos para cadena de altavoces, estéreo para auriculares, Dolby ProLogic, A/V y sistemas digitales 5.1 como Dolby Digital AC-3. Esta última hace uso de una salida digital SPDIF: `SPDIF_OUT`.

La salida `MONO_OUT` fue diseñada originalmente para conexiones de teléfono analógico pero puede ser configurada tanto para micrófono como para una mezcla de fuentes. Hay dos entradas de micrófono: `MIC1` y `MIC2`. Hay un tercer ADC opcional dedicado a entradas de voz que extiende el rango de capacidades de cancelación de eco acústico.

En cuanto a los modems hay un par ADC/DAC para la línea 1 de modem, siendo opcionales dos pares más: línea 2 de modem y handset de modem.

5.5.3. MIDI

La norma MIDI (Musical Instrument Digital Interface) de 1983 especifica una forma eficiente de comunicar instrumentos musicales digitales. Para ello, en lugar de enviar las muestras musicales digitalizadas, envía la información de la partitura musical y de los instrumentos que la interpretarán. Se consigue con esto reducir drásticamente el ancho de banda requerido[257].

Los instrumentos musicales MIDI suelen tener tres puertos: IN, OUT y THRU. IN es una entrada de información procedente de otro instrumento o del ordenador, OUT es la la salida de información del instrumento hacia otro instrumento o hacia el ordenador y THRU es una salida que reenvía la información que llega al puerto IN; se usa para conectar en cadena muchos instrumentos.

En la figura 4.50 se muestra una típica configuración MIDI. Hay un maestro y 4 esclavos. El maestro es un programa secuenciador que corre sobre un PC con interfaz MIDI. Los esclavos son sintetizadores de música con interfaz MIDI, algunos disponen de teclado y otros no. El puerto OUT del secuenciador se conecta al puerto IN del primer esclavo, el puerto THRU del primer esclavo se conecta al puerto IN del segundo esclavo, y así sucesivamente.

El secuenciador lee el fichero MIDI que contiene la partitura de la música que se quiere interpretar. En la partitura MIDI hay 4 canales (instrumentos) que se asocian a cada uno de los 4 esclavos. El secuenciador interpreta la música enviando mensajes a través de su puerto OUT a los esclavos.

Los mensajes MIDI (figura 4.51) consisten en una secuencia de bytes que dan información del canal seleccionado (el instrumento musical), de la acti-

[257] Se pasa de 10MB/minuto para ficheros tipo WAV a 10KB/minuto para un fichero MID (MIDI).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

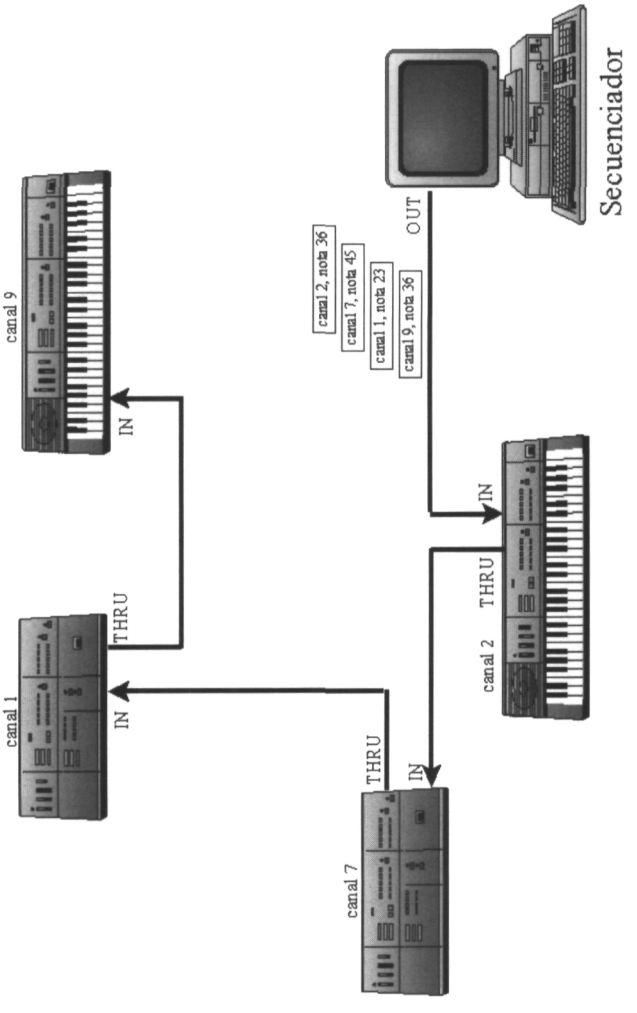


Figura 4.50: Ejemplo de sistema MIDI

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

vación de la nota (NOTA ON/OFF), del tono de la nota, de la fuerza de pulsación de la nota (velocidad), etc..

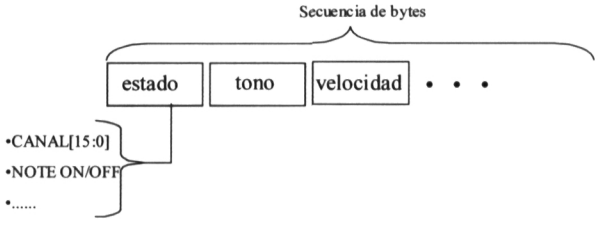


Figura 4.51: Protocolo MIDI

Los esclavos están a la escucha de todos los mensajes a través de su puerto IN. Cuando un esclavo recibe un mensaje con su dirección (canal) lo traduce y genera el sonido correspondiente.

MIDI utiliza un interfaz serie asíncrono a 31.25 Kbaudios. El circuito es un bucle de corriente de 5 mA. Usa pares trenzados y apantallados y un conector DIN (ver figura 4.52).

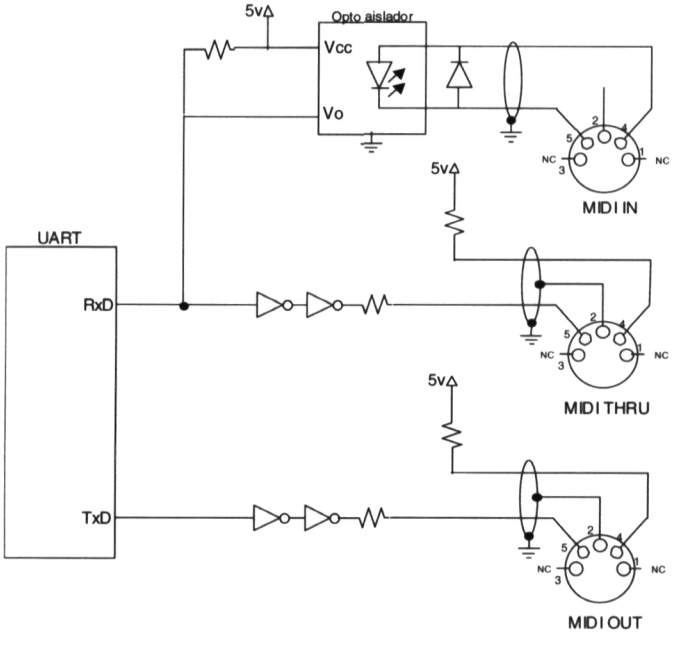


Figura 4.52: Interfaz MIDI

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Las entradas han de estar aisladas por un optoacoplador con tiempos muy rápidos de subida y bajada para evitar retardos y degradación de la señal en el puerto THRU. Las resistencias son de 220 Ω y regulan la corriente. El diodo que hay a la entrada del optoacoplador previene la polarización inversa.

5.5.4. Ejemplos

5.5.4.1. Cálculo de ancho de banda de audio Dependiendo de la frecuencia de muestreo, nivel de cuantificación y tipo de canal (mono o estéreo), la cantidad de información generada varía sustancialmente (ver tabla 4.56).

bit/muestra/f_m	11KHz	22KHz	44,1KHz
8 bits mono	644,53 KB	1.289,06 KB	2,583,98 KB
8 bits estéreo	1.289,06 KB	2,583,98 KB	5,156,24KB
16 bits mono	1.289,06 KB	2,583,98 KB	5,156,24KB
16 bits estéreo	2,583,98 KB	5,156,24KB	10.335,93KB

Tabla 4.56: Cantidad de información generada en un minuto en una grabación de audio

Por ejemplo, el ancho de banda requerido para almacenar sonido estéreo de alta fidelidad es de 10MB/minuto. Esto es demasiada información por lo que se usan técnicas de compresión como por ejemplo, ADPCM o la codificación subbanda.

El codificador subbanda transforma las muestras de sonido al dominio de la frecuencia y decide qué frecuencias se pueden despreciar (lo hace codificándolas con menos bits) sin pérdida audible apreciable[258]. Dentro de los estándares de codificación destacan dolby AC3 optimizado para codificación multicanal de pista de películas y MPEG-Audio MPEG1 nivel 3 (MP3).

[258] El oído humano tiene un gran rango dinámico (96dB) que le permite oír la caída de una hoja y el despegue de un avión. Sin embargo no es capaz de oír los dos a la vez. El sonido débil queda enmascarado por los sonidos fuertes. Este efecto es aprovechado por la psicoacústica para comprimir el sonido.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

6. Bibliografía

Libros

Doyle, L.F.: Computers peripherals, Prentice Hall Career&Technology, 1998. ISBN: 0-13-779463-0.

Joseph Di Giacomo: Digital bus handbook , McGraw-Hill, 1990, ISBN: 0-07-016923-3.

McDowell, S. and Seyer, M.D.: USB Explained, Prentice Hall Professional Tecnical Reference, 1998. ISBN 0-13-081153-X.

W. Ming Tan; Developing USB PC peripherals. 2ª Ed. Annabooks, San Diego CA USA 1999. ISBN: 0-929392-64-7

Hide, John.: USB Design by Example: A practical Guide to building I/O devices, 2ª Ed. Intel Press, USA, 2001.ISBN 0-9702846-5-9.

R.Jourdain: Solucionario del programador para IBM PC, XT, AT y compatibles, Anaya multimedia, 1986. ISBN: 84-7614-155-6.

LoBue, M.T.: Surveying Today´s Most Popular Storage Interfaces, IEEE Computer, December 2002.

Mazidi, M.A. y Mazidi, J.G.: The 80x86 IBM PC and compatible Computers, Vol. II: Design and Interfacing of the IBM PC and compatible Computers, Prentice Hall Career&Technology, 1998. ISBN: 0-13-758491-1.

Ferraro, R.F.: Programmer´s Guide to the EGA, VGA, and Super VGA Cards, 3 Ed, Addison-Wesley Publishing Company, 1994.

Wilton, R.: Sistemas de video, Anaya Multimedia, 1990. ISBN: 84-7614-251-X.

Oguic, P.: Control electrónico con el PC, Paraninfo, 1996.

Seyer, M.D.: Conexiones en el IBM PC/XT/AT, Teoría y práctica de periféricos, comunicaciones y configuraciones, Anaya multimedia, 1985, ISBN: 84-7614-119-X.

Ridge,P.M., Golden, D.M., Luk, I. y Sindorf, S.E.: Guía Oficial de Sound Blaster 2ª Ed., McGrawHill, 1997. ISBN: 84-481-1876-X.

Tisher, M. y Jennrich, B: PC interno 5, Marcombo Boixareu Editores, 1998. ISBN: 84-267-1081-6.

Varios autores:: Interfacing Sensors to the IBM PC. (Tompkings, W.J. and Webster, J., Eds.) Prentice Hall, 1988. ISBN: 0-13-469081-8.

Zoller, B.: Circuitos Electrónicos con el PC, Marcombo Importación, 1998. ISBN: 84-267-1125-1.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Manuales

SoundBlaster Series Hardware Programming Guide, Creative Technology Ltd. 1996.

Hacking the IBM PC de Vicent Himpe, Course manual `Digging int it`, Alcatel Microelectronics, 1998.

Especificaciones

Especificación AC´97 Rev. 2.3 2002, Intel

Data Sheets

- 8237A, High Performance Programmable DMA Controller, Intel, September 1993, Order Number: 231466-005.
- 8259A, High Performance Programmable Interrupt Controller, Intel, December 1988, Order Number: 231468-003.

Algunas páginas WWW

Pcguide. www.pcguide.com

PC Webopaedia. www.pcwebopedia.com

PC Techguide. www.pctechguide.com

Tom hardware Guide. www2.tomhardware.com

PCI Industrial Computer Manufacturers Group. www.pcisig.com

Universal Serial Bus. www.usb.org

Beyond Logic. www.beyondlogic.org

Comité Técnico T10 para interfaces de almacenamiento SCSI. www.t10.org

STA(SCSI Trade Association). www.scsita.org

Comité Técnico T10 para interfaces de almacenamiento ATA. www.t13.org

February 2004 Jim Brewer, J. and Sekel, J., PCI EXPRESS TECHNOLOGY, white paper, February 2004. www.dell.com/r&d

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

7. Guía de estudio

1. Describir el funcionamiento básico de los controladores 8259 y 8237 de Intel
2. Describir el funcionamiento básico de la UART 16550 de National.
3. Explicar la utilidad del modem nulo
4. Analizar un cronograma de escritura del puerto CENTRONICS.
5. Enumerar los modos de trabajo del interfaz IEEE1284 e indicar las diferencias entre ellos.
6. Clasificar los buses de E/S (XT,AT,MCA,EISA,PCI,PCI-X y PCIexpress) en función de su ancho de banda, bus de datos, compatibilidad, autoconfiguración y multiproceso.
7. Enumerar las características más significativas de los buses PCI y PCI Express.
8. Enumerar diferencias entre SCSI y ATA.
9. Enumerar las características básicas del bus USB.
10. Enumerar los 4 tipos de transferencias del bus USB y destacar las diferencias entre ellos (direccionalidad, velocidad, sincronismo, retransmisión, etc.).
11. Describir el interfaz físico de teclado del PC-XT indicando la función de las señales open-collector.
12. Describir el funcionamiento de un ratón optomecánico.
13. Describir el funcionamiento del joystick analógico de IBM. Aplicar este diseño para el control de un parámetro físico (temperatura, presión, humedad, etc.).
14. Señalar las diferencias entre monitores CRT y TFT.
15. Calcular el tamaño de un buffer de video para una resolución y profundidad de color determinada (ver tabla 4.47).
16. Enumerar las etapas de proceso de una aceleradora 3D.
17. Enumerar las características básicas del bus AGP.
18. Calcular el ancho de banda requerido para grabar sonido en función de la frecuencia de muestreo, nivel de cuantificación y número de canales.
19. Distinguir entre síntesis de sonido FM y tabla de ondas.
20. Enumerar las características más importantes de la especificación AC´97.
21. Enumerar brevemente la secuencia de programación de una SoundBlaster16 para transferencias DMA.
22. Describir el interfaz físico MIDI.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5 Microcomputadores

1. Introducción

Los sistemas basados en microprocesadores de propósito general, que aquí denominamos microcomputadores, han sufrido una tremenda evolución desde su aparición en los años 70. Inicialmente eran equipos de pocas pretensiones, sencillos, de relativo bajo coste y sin apenas conectividad. Posteriormente alcanzaron unas capacidades de cómputo y almacenamiento tan grandes que los hicieron competir con equipos especializados. Hoy en día, su portabilidad y conectividad los ha convertido en auténticos equipos de consumo donde profanos en el mundo de la informática los consideran casi productos de primera necesidad.

En este capítulo se estudia en detalle el IBM PC que marcó un hito en el mundo de los ordenadores personales y que, aún hoy en día, conserva su importancia, ya que los equipos actuales mantienen compatibilidad software con el mismo. Seguidamente se muestra su evolución hacia plataformas basadas en Pentium donde aparecen nuevas características relacionadas con la conectividad, la multimedia y el consumo. A continuación, se describe el procesador PXA27x de Intel, todo un sistema de propósito general integrado en un chip que incorpora muchas de las interfaces y prestaciones de los ordenadores personales.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2. El IBM PC

A finales de los 70, la división Data Entry Systems de IBM decidió ampliar su gama de productos. En lugar de terminales y minicomputadores o mainframes, los clientes demandaban sencillos ordenadores que fueran baratos, autónomos y mono-usuario. No se necesitaba mucha potencia de cálculo ni conectividad, pero se requería que cumplieran con los estándares del alfabeto ASCII y el lenguaje de programación BASIC. El producto más parecido a esto dentro de la línea de IBM era el 5110, una máquina con BASIC en ROM y con un pequeñísimo display de caracteres incorporado. El 5110 no era competitivo y IBM empezó a perder clientes importantes. Así que Data Entry encargó a una empresa el diseño de un nueva máquina hasta que IBM pudiera reemplazarla por un diseño propio. Algunos ejecutivos de IBM no confiaban en este proyecto al que consideraban un capricho que duraría poco. Fue el denominado IBM PC.

El diseño del IBM PC (ver figura 5.1) se basó en un computador monoplaca descrito en una nota de aplicación de Intel de 1978 para el microprocesador 8088-4,77MHz. La placa base del IBM-PC disponía de 4 canales de DMA, 3 contadores/timer, 8 interrupciones hardware, interfaz a casete, altavoz y teclado, RAM, ROM, lógica de control y un bus de expansión. El teclado, conducido por eventos, fue un diseño original, pero el concepto era de las estaciones de trabajo Star y Xerox Alto. El diseño del interfaz de casete era propio, pero similar en concepto a uno de TRS-80 de Radio Shack.

El bus de expansión era muy parecido a un bus de 8 bits, llamado Multibus 1, que Intel introdujo a mediados de los 70 dentro de su sistema de desarrollo de microprocesadores. Disponía de 5 ranuras de expansión. La mayor parte de la UES se implementaba en tarjetas de expansión: Ampliación de memoria, interfaz a joystick, puerto paralelo para impresora, adaptadoras gráficas para monitores y TV, puerto serie RS-232 y disqueteras. La tarjeta adaptadora de gráficos se basó en notas de aplicación del chip controlador de video 6845 de Motorola.

El sistema operativo usado fue el DOS (Disk Operating System), a menudo especificado como IBM-DOS o MS-DOS. La MS es de Microsoft que fue la empresa que lo desarrolló[259].

[259]Data Entry Systems se dirigió a Digital Research para adquirir su popular sistema operativo CP/M-86 pero no lo consiguió. Así que llamaron a la puerta de un vendedor de BASIC en ROM, una empresa denominada Microsoft, que no tenía productos de sistemas operativos en ese momento pero que consiguió comprar un sistema operativo de disco (el 86-DOS de Seattle Computer Products) para ofrecérselo a IBM. Su intérprete de comandos era una im-

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

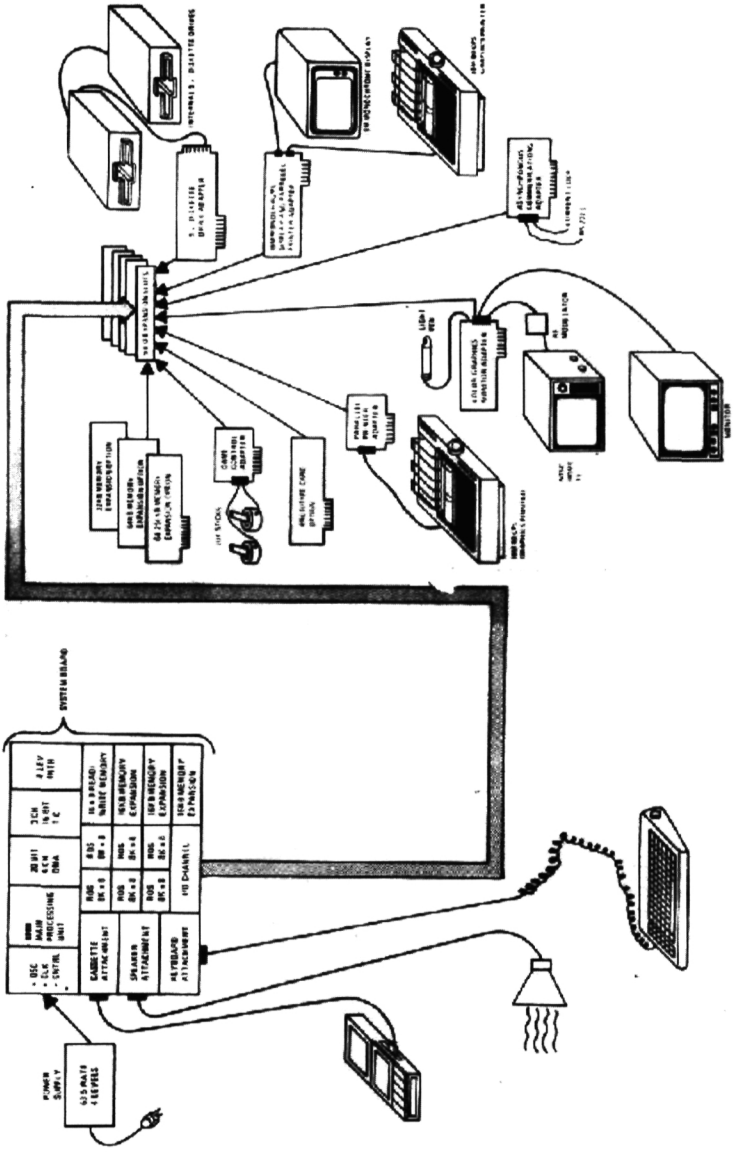


Figura 5.1: Elementos de un IBM PC. Cortesía de IBM

itación de Bourne Shell de Unix, con los caracteres especiales cambiados para evitar infringir los derechos de AT&T.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

En la figura 5.2 se muestran los niveles de abstracción del IBM PC. Las rutinas BIOS (Basic Input/Output System) son un firmware (software residente en ROM) usado por el sistema operativo para acceder a todos los dispositivos de E/S del sistema.

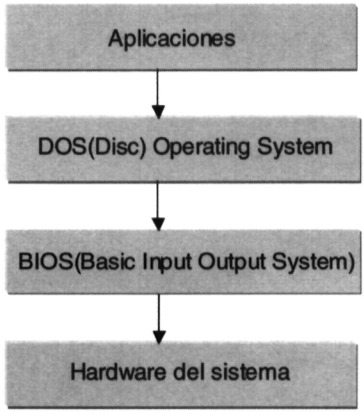


Figura 5.2: Sistema operativo del IBM PC

Data Entry Systems empezó a ofrecer el IBM PC a organismos oficiales sin ningún plan de ofertarlo al gran público.

2.1. Diagrama de bloques

La figura 5.3 muestra el diagrama de bloques original del IBM-PC. El microprocesador 8088 se apoya en un conjunto de chips para implementar la circuitería de la placa base: Generador de reloj 8284, coprocesador matemático 8087[260], controlador de interrupciones 8259, controlador de bus 8288, controlador de DMA 8237, puerto paralelo programable 8255 y generador de timers 8253.

2.1.1. Generador de reloj

El generador de reloj 8284 tiene varias funciones: Genera la señal de inicialización (`RESET`), de reloj (`CLK`) y de fin de ciclo de bus (`READY`). Una lógica de estados de espera proporciona al 8284 información para crear los tiempos de espera (señal `READY`) de memoria y dispositivos de E/S.

[260] La placa base dispone de un zócalo (auxiliary processor socket) para conectar el coprocesador matemático 8087.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

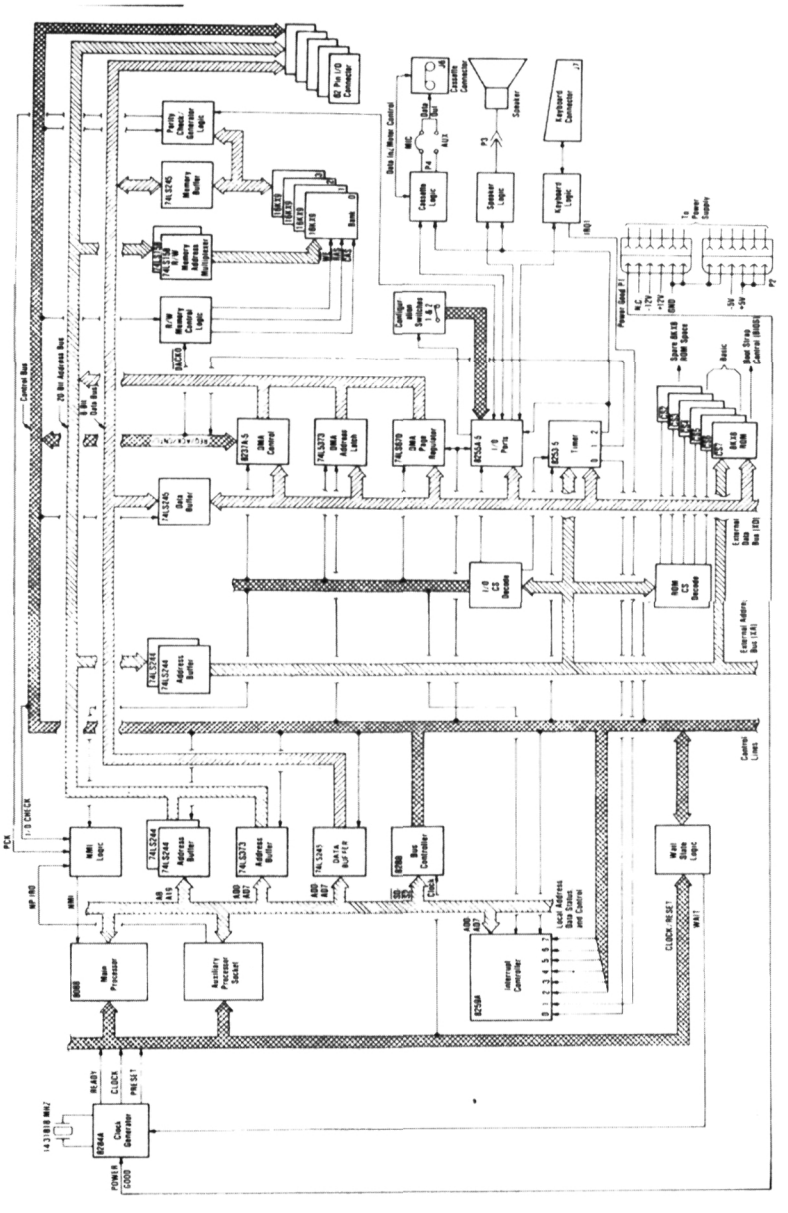


Figura 5.3: Diagrama de bloques de la placa base del IBM PC. Cortesía de IBM

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.1.2. Buses

El bus del microprocesador está formado por 20 líneas de direcciones (`A[19:0]`), 8 de datos (`D[7:0]`) y 3 de control (`S[2:0]#`). Los 8 bits menos significativos del bus de direcciones están multiplexados con los 8 bits del bus de datos (`AD[7:0]`).

Los buffers 74LS244, 74LS245 y el latch 74LS373 aíslan el bus del microprocesador del bus de expansión y proporcionan la corriente necesaria. El buffer 74LS244 (unidireccional) se usa para las direcciones altas (`A[19:8]`) y el buffer 74LS245 (bidireccional) para el bus de datos (`AD[7:0]`).

El latch 74LS373 tiene dos funciones: Desmultiplexar direcciones y datos y actuar como buffer para el byte de direcciones bajo `A[7:0]`. El latch usa una señal de control del microprocesador (`ALE`) para capturar las direcciones.

El controlador de bus 8288 generar las señales de control del bus de expansión.

Hay una extensión del bus de expansión que está aislada mediante buffers: `XA`(External Address) y `XD` (External Data). Esta extensión se usa para conectar la mayoría de los dispositivos de la placa base (controlador de DMA, puertos paralelo, timers y memoria ROM). También hay una lógica de decodificación de direcciones para todos estos dispositivos (`I/O CS DECODE` y `ROM CS DECODE`).

2.1.3. Interrupciones

El controlador de interrupciones 8259 gestiona 8 solicitudes de interrupciones hardware (`IRQ[7:0]`). `IRQ0` procede de la salida del timer 0 del generador de timers 8253 y se utiliza para implementar un reloj en tiempo real. `IRQ1` procede de la lógica de teclado y se activa cada vez que se pulsa/libera una tecla. El resto de solicitudes de interrupción son utilizadas por las tarjetas de expansión. El controlador de interrupciones solicita interrupción al microprocesador activando la entrada de interrupción enmascarable de 8088 (no figura en el diagrama).

El 8088 tiene una entrada de interrupción no enmascarable (`NMI`). La lógica NMI gestiona tres solicitudes de interrupción no enmascarables: 1) `PCK`: Error de paridad de memoria, generada por la lógica de chequeo de paridad, 2) `I/O CHECK`: error de paridad de dispositivos de E/S o memoria sita en tarjetas de expansión y 3) `NP IRO`: error del coprocesador matemático.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.1.4. Controlador de DMA

El controlador de DMA 8237 proporciona 4 canales de DMA al sistema. El canal 0 (el más prioritario) está reservado para llevar a cabo el refresco de la memoria DRAM[261]. Los tres restantes están disponibles para las tarjetas de expansión.

El 8237 requiere cierta lógica adicional para poder realizar las transferencias DMA. Su bus de datos está multiplexado con el byte alto de direcciones por lo que se hace necesario un latch del tipo 74LS373 para almacenar el byte alto de direcciones. Por otro lado, las 16 líneas de direcciones conseguidas entre el 8237 y el latch sólo alcanzan a direccionar 64KB y no 1MB. Por ello, se incluye un 74LS670, denominado registro de páginas (cada página o segmento es de 64KB), que genera las líneas de direcciones restantes `A[19:16]`. Por tanto, el controlador de DMA ve el espacio de direcciones de 1MB como 16 páginas de 64KB cada una[262].

2.1.5. Puertos paralelo

El chip 8255, también llamado PPI (Programmable Peripheral Interface), implementa 3 puertos paralelo. Se usa para control software de diversos dispositivos y para la lectura de la configuración (establecida en unos switches) de la placa base.

2.1.6. Temporizadores

El generador de timers 8253, también llamado PIT (Programmable Interval Timer) integra 3 timers/contadores. El timer 0 se utiliza para implementar un reloj en tiempo real, el timer 1 marca el ciclo de refresco de la memoria DRAM y el timer 2 proporciona una onda cuadrada de frecuencia audible que se utiliza para excitar un altavoz. Los tres timers/ contadores utilizan la misma entrada de reloj: 1.19318MHz.

El timer 0 genera a su salida un pulso cada 55 ms que se envía a la entrada `IRQ0` del controlador de interrupciones[263].

[261] El canal 0 se programa en modo single, con incremento de direcciones, autoinicialización y transferencia de lectura.

[262] Cuando se programa una transferencia de DMA es conveniente no cruzar un límite de página, ya que esto obliga a programar el registro de página con la consiguiente caída de prestaciones del DMA.

[263] El timer se programa para contar 65536 ciclos de reloj (es un registro de 16 bits) entre pulsos por lo que la frecuencia de interrupción es de 1.193.180/65.536 = 18,2 veces por segundo = 1/55ms.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

La salida del timer 1 está conectada a la entrada `DRQ0` (solicitud de DMA del canal 0) del 8237. Para la memoria de 1MB se requiere un refresco cada menos de 4ms.

Cada timer dispone de una entrada `GATE` que sirve para habilitar el contador. Los timers 0 y 1 están siempre habilitados (su entrada `GATE` está a 1). Sin embargo, la entrada `GATE` del timer 2 se puede programar a través del PPI (bit 0 del puerto `61H` de E/S).

2.1.7. Memoria

La memoria ROM se implementa con 6 chips de 8K×8. Su función es almacenar el código de arranque del sistema: inicialización del hardware, tests del sistema, programa de carga del sistema operativo (bootstrap), rutinas BIOS y BASIC.

La memoria DRAM se distribuye en 4 bancos de 16K×9 ó 64K×9. El interfaz de memoria DRAM incluye: 1) Lógica de control de memoria que genera las señales de lectura, escritura, `RAS#` y `CAS#` y gestiona el refresco de DRAM, 2) Multiplexor de direcciones de memoria mediante un par de 74LS158, 3) Un buffer 74LS245 para aislar el bus de la memoria del sistema y evitar así colisiones de datos y 4) Lógica de generación y chequeo de paridad.

La función de la memoria DRAM es almacenar el sistema operativo y los programas de aplicación y datos a procesar. Los programas se cargan desde la disquetera o el casete.

2.1.8. Interfaz de teclado

El interfaz de teclado usa un conversor serie/paralelo (74LS322), uno de los puertos del PPI para almacenar el código de muestreo de teclado y lógica adicional para generar la solicitud de interrupción de teclado `IRQ1`.

Como ejemplo, en la figura 5.4 se describe la secuencia típica que se produce en un IBM PC desde que se pulsa una tecla hasta que su valor se almacena en el buffer de teclado.

1. Se pulsa/libera una tecla
2. El controlador de teclado detecta la pulsación/liberación y genera un código de muestreo (scan code) de tecla pulsada (make code) o liberada (break code) que trasmite a través del interfaz de teclado.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

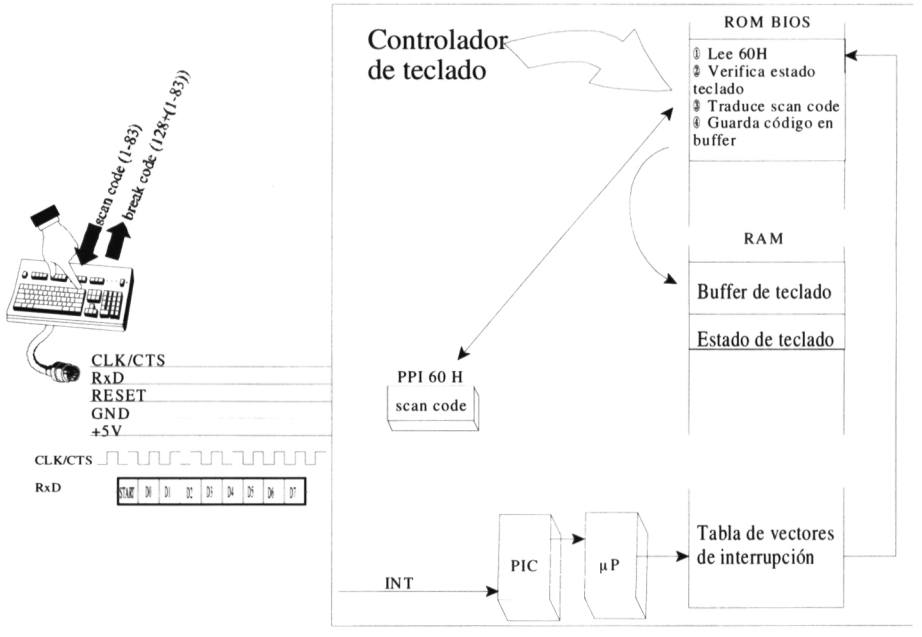


Figura 5.4: Secuencia de teclado XT

3. El código se almacena en un puerto de E/S[264]. El PC fuerza la señal de reloj del interfaz de teclado `CLK` a 0 (es una señal de tipo open collector) indicándole al teclado que no envíe más códigos hasta que no procese el que acaba de recibir.

4. La recepción del código de teclado dispara una solicitud de interrupción al controlador de interrupciones del sistema[265] que la gestiona haciendo llegar la petición al microprocesador.

5. El microprocesador accede a la dirección de memoria correspondiente[266] de la tabla de vectores de interrupción donde está la dirección de comienzo de la rutina de servicio de interrupción del teclado.

6. Se ejecuta la rutina del controlador de teclado: a) Lee el código de muestreo, b) Verifica el estado del teclado (Mayúsculas, Ctrl, Alt, Alt-Gr, etc.), c) Traduce el código de muestreo y lo convierte en un código

[264] En la dirección `60H` de E/S del PC.

[265] La interrupción de teclado se corresponde con la interrupción hardware `IRQ1` y el vector de interrupción 9 de la tabla de vectores de interrupción.

[266] 9×4=36H. Suponemos modo real.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

de carácter y d) Guarda los códigos de carácter y el estado del teclado en el buffer de teclado[267].

7. Finalmente, el PC libera la señal CLK del interfaz de teclado.

2.2. Espacio de direcciones

El IBM PC dispone de dos espacios de direcciones: Memoria y E/S. El espacio de direcciones de memoria es de 1MB (ver figura 5.5).

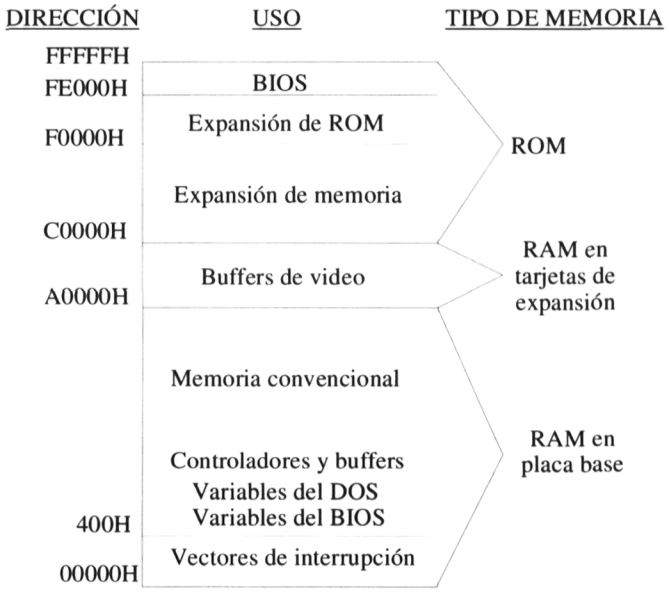


Figura 5.5: Espacio de memoria del IBM PC

El espacio se divide en dos partes: Los primeros 640KB (0-9FFFFH) se reservan para la DRAM en placa base y el resto (360KB) para ROM y memoria sobre tarjetas de expansión.

El espacio de direcciones de E/S es de 1KB. Este espacio se divide en dos partes iguales (ver figura 5.6). Las 512 direcciones más bajas (0-1FFH) son usadas exclusivamente por la placa base. Las 512 direcciones más altas (200-3FFH) son decodificadas por tarjetas conectadas en las ranuras de expansión. En la tabla 5.1 se muestra la asignación de direcciones de E/S a los diferentes dispositivos.

[267] El buffer de teclado está ubicado en la dirección de memoria 0000:041EH. Allí es donde accederá el programa de aplicaciones para recoger el dato.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

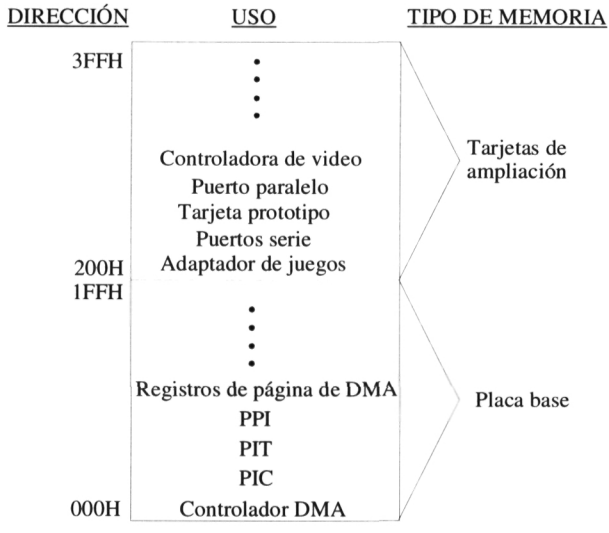


Figura 5.6: Espacio de E/S del IBM PC

Dirección	Asignación
000-00F	Controlador DMA 8253
020-021	Controlador de interrupciones 8259
040-03	Timer programable 8253
060-063	PPI 8255
080-083	Registros de página DMA
0AX	Registro de máscara NMI
0CX	Reservado
0EX	Reservado
200-20F	Adaptador de juegos (Joystick)
278-27F	Reservado
2F8-2FF	Tarjeta RS232 ALT
300-31F	Tarjeta prototipo
378-37F	Puerto de impresora paralelo
380-38F	Adaptador de impresora paralelo y display monocromo de IBM
3D0-3DF	Adaptador gráfico en color
3F0-3F7	Adaptador de disquetera 5,25"
3F8-3FF	Tarjeta RS232 TP

Tabla 5.1: Asignación de direcciones de E/S del IBM PC

2.3. Bus de expansión

El bus de expansión permite la ampliación del sistema, proporcionando a las tarjetas de expansión el acceso a casi todas las señales del bus del E/S[268]. La figura 5.7 y la tabla 5.2 describen el conector y cada una de las señales del

[268] Algunas señales del bus del sistema (`IRQ0`, `IRQ1` y `DRQ0`) no están en el bus de expansión ya que están dedicadas en exclusiva a dispositivos integrados en la placa base.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

bus de expansión.

B1	GND	I/O CHCK	A1
B2	RESET DRV	D7	A2
B3	+5v	D6	A3
B4	IRQ2	D5	A4
B5	-5v	D4	A5
B6	DRQ2	D3	A6
B7	-12v	D2	A7
B8	RESERVADO	D1	A8
B9	+12v	D0	A9
B10	GND	I/O CHRDY	A10
B11	MEMW#	AEN	A11
B12	MEMR#	A19	A12
B13	IOW#	A18	A13
B14	IOR#	A17	A14
B15	DACK3#	A16	A15
B16	DRQ3	A15	A16
B17	DACK1#	A14	A17
B18	DRQ1	A13	A18
B19	DACK0#	A12	A19
B20	CLK	A11	A20
B21	IRQ7	A10	A21
B22	IRQ6	A9	A22
B23	IRQ5	A8	A23
B24	IRQ4	A7	A24
B25	IRQ3	A6	A25
B26	DACK2#	A5	A26
B27	T/C	A4	A27
B28	ALE	A3	A28
B29	+5v	A2	A29
B30	OSC	A1	A30
B31	GND	A0	A31

Figura 5.7: Conector del bus de expansión del IBM PC

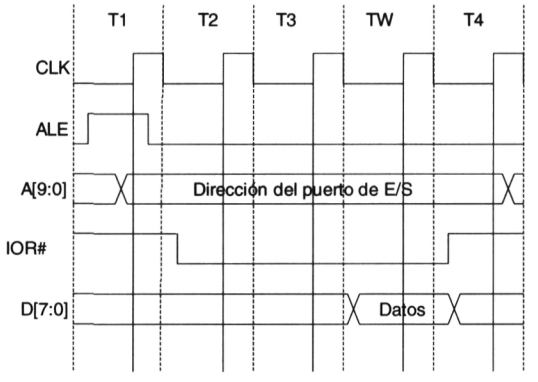
Señal	Dirección	Descripción
GND		Masa
I/O CH CK#	entrada	I/O Channel Check. Indica que ha habido un error en el periférico. Cuando se activa dispara una interrupción no enmascarable en el microprocesador
RESET DRV	salida	Reset activo a nivel alto
D[0:7]	bidireccional	Bus de datos. Los datos son válidos en el flanco de subida de las señales de lectura y escritura
+5 V	salida	Alimentación de +5V (7A)
-5 VDC	salida	Alimentación de -5V (300mA)
IRQ[7:2]	entrada	Solicitudes de interrupción del controlador de interrupciones 8259. Se activan con el flanco de subida. Se usa prioridad fija siendo IRQ2 la más prioritaria
DRQ[3:1]	entrada	Direct Memory Access Request. Solicitudes de DMA activas a nivel alto. Prioridad fija siendo DRQ1 la más prioritaria
-12 VDC	salida	Alimentación de -12 V (250 mA)
+12 VDC	salida	Alimentación de 12V (2A)
IO CHRDY	entrada	I/O Check Ready. Genera tiempos de espera para dispositivos lentos
AEN	salida	Address Enable. Indica que se está llevando a cabo un ciclo de DMA
MEMW#	salida	Memory Write. Señal de control de escritura de memoria
MEMR#	salida	Memory Read. Señal de control de lectura de memoria
IOW#	salida	I/O Write. Señal de control de escritura de E/S
IOR#	salida	I/O Read. Señal de control de lectura de E/S
A[19:0]	salida	Bus de direcciones
DACK[3:1]#	salida	DMA Acknowledge. Indica que la solicitud de DMA ha sido concedida
CLK	salida	Reloj de 4,77MHz y ciclo de trabajo de 30 %
T/C	salida	Terminal Count. Indica que ha terminado un ciclo de DMA. Es una señal activa con nivel lógico alto
ALE	salida	Address Latch Enable. Se activa al principio de un ciclo de bus e indica que el bus de direcciones tiene información válida. También se utiliza para capturar la dirección en un latch
OSC	salida	Oscilador de 14.31818 MHz

Tabla 5.2: Descripción de señales del bus de expansión del IBM-PC

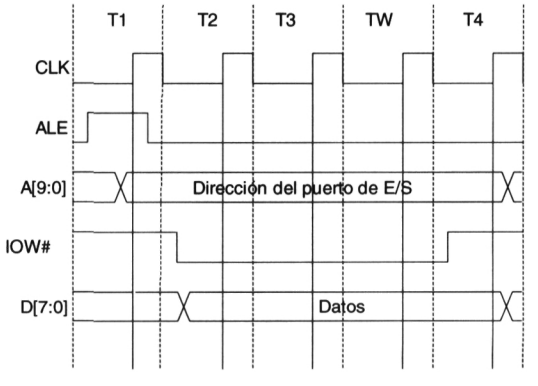
Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.3.1. Ciclos de bus

La figura 5.8 muestra los ciclos de lectura y escritura de acceso a E/S. Se requieren 5 ciclos de reloj (de 210 ns cada uno) incluyendo el estado de espera (`TW`) inyectado por lógica ubicada en la placa madre.



(a) lectura



(b) escritura

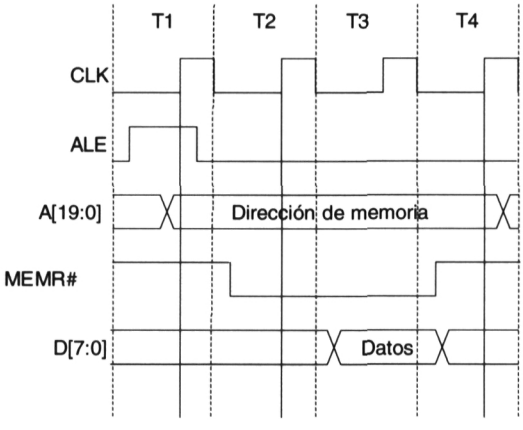
Figura 5.8: Acceso a E/S del bus de expansión del IBM PC

La señal `ALE` se activa (a nivel alto) al comienzo del ciclo e indica que las direcciones son válidas sobre el bus. La señal `AEN` (no aparece en el cronograma) debe permanecer a nivel bajo durante todo el ciclo de transferencia. `IOR#`(`IOW#`) se activa a nivel bajo para indicar un ciclo de lectura (escritura).

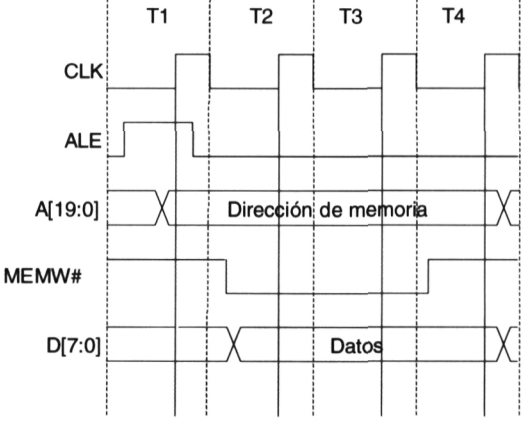
Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Los datos son válidos en el flanco de subida de la señal lectura (escritura). Sólo es necesario decodificar las 10 primeras líneas de direcciones `A[9:0]`. Se pueden añadir estados de espera si la señal `IO CHRDY#` se activa en la primera mitad de cada ciclo de espera.

La figura 5.9 muestra los ciclos de lectura y escritura de acceso a memoria. Se requieren 4 ciclos de reloj (de 210 ns cada uno).



(a) lectura



(b) escritura

Figura 5.9: Acceso a memoria del bus de expansión del IBM PC

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

La señal **ALE** se activa al comienzo del ciclo e indica que las direcciones son válidas sobre el bus. La señal **AEN** (no aparece en el cronograma) debe permanecer a nivel bajo durante todo el ciclo de transferencia. **MEMR#(MEMW#)** se activa a nivel bajo para indicar un ciclo de lectura (escritura). Los datos son válidos en el flanco de subida de la señal lectura(escritura). En este caso hay 20 líneas de direcciones: **A[19:0]**.

La figura 5.10 muestra un ciclo de DMA. Es un ciclo de transferencia de DMA que lee datos de un puerto de E/S de un periférico que solicitó la transferencia para el canal **X** vía la señal **DRQX** (no aparece en la figura).

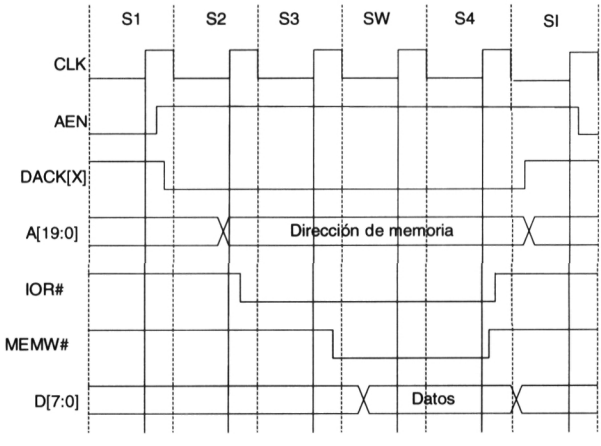


Figura 5.10: Acceso DMA del bus de expansión del IBM PC

La señal **AEN** se activa para indicar que se está llevando a cabo un ciclo de DMA; esto es necesario para evitar que las direcciones de memoria de la transferencia DMA sean confundidas con direcciones de E/S, ya que se activan las señales de control de E/S (en el caso de la figura **IORD#**). La señal de reconocimiento de DMA para el canal **X** (**DACK X**) se activa en el primer ciclo de reloj. En el segundo ciclo de reloj, el controlador de DMA pone las direcciones de memoria sobre el bus y activa la señal de lectura **IORD#**. Dos ciclos después aparece el dato leído sobre el bus y se mantiene hasta el ciclo **S4**. En el tercer ciclo, activa la señal de escritura de memoria **MEMWR#** y en el ciclo S4 los datos que hay en el bus se escriben en memoria con el flanco de subida de **MEMWR#**.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3. El PC-XT

El PC-XT (Personal Computer- Extended Technology) es una versión mejorada del IBM PC. El microprocesador 8088 trabaja a 8-10MHz, incluye una unidad de disco duro de 10MB[269] y tiene 8 ranuras de expansión[270].

El PC-XT fue ofrecido al gran público y superó todas las espectativas. IBM publicó los esquemas completos y los listados de la ROM, facilitando la aparición de clónicos. Los clónicos o PCs compatibles eran versiones compatibles de otros fabricantes. La compatibilidad con IBM venía determinada por la BIOS (Basic Input Output System). El que muchos fabricantes pudieran desarrollar PCs desencadenó una gran competencia en prestaciones y precio de los equipos.

4. El PC-AT

El PC-AT (PC-Advanced Technology) es una expansión de la arquitectura XT. Su arquitectura y funcionalidad es similar al PC-XT, pero un rendimiento muy superior (unas 3 veces más rápido). Las diferencias más significativas con el PC-XT son:

- Microprocesador 80286-8MHz de 16 bits.
- Reloj con batería backup CMOS RAM: Esta pequeña memoria guarda el reloj y los datos de la configuración del sistema.
- 15 interrupciones hardware. Hace uso de dos controladores de interrupciones 8259 conectados en cascada.
- 7 canales de DMA implementados con dos controladores de DMA 8237 conectados en cascada.
- Disco duro de 20MB y disquetera de 5,25" de 1,2MB.
- 256KB(ampliable a 512KB) de DRAM y 64KB de ROM[271] (ampliable a 128KB).

[269] Se eliminó el interfaz de casete. El controlador de disco fue diseñado con el chip 1010 de Western Digital y, de nuevo, su diseño fue directamente copiado de una nota de aplicación del chip.

[270] Dos de ellas ocupadas con una tarjeta controladora de disco y otra de video.

[271] 8 meses después de su aparición, la empresa Phoenix había "copiado" la ROM-BIOS del PC-AT, abriendo este mercado al resto de los fabricantes.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4.1. Espacio de direcciones

El espacio de direcciones de memoria es una expansión del que tiene el PC-XT (ver figura 5.11).

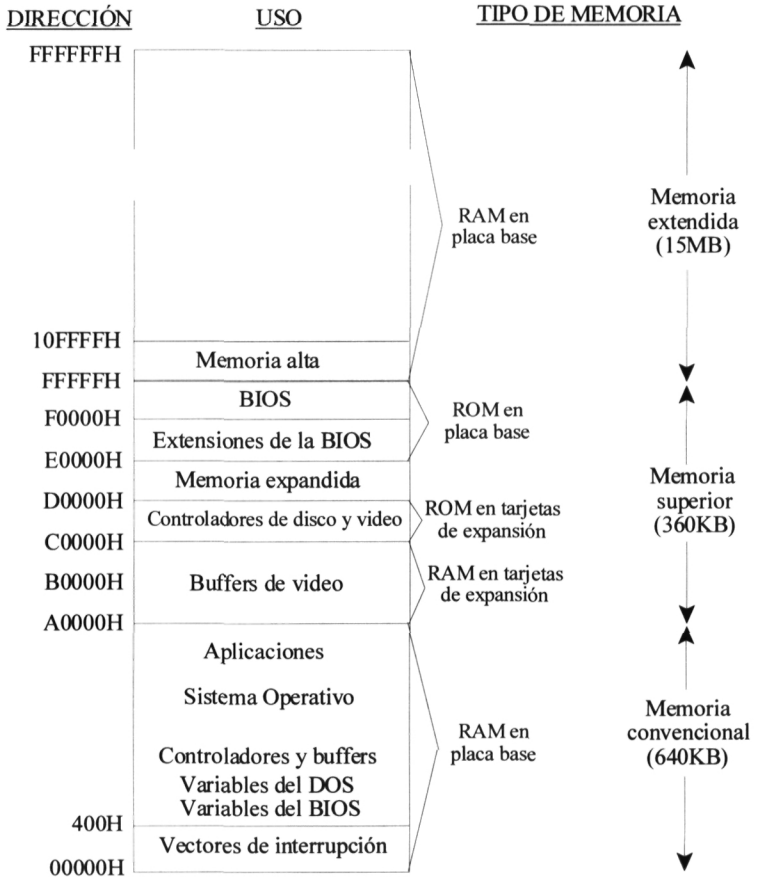


Figura 5.11: Espacio de memoria del PC-AT

Se distinguen tres zonas: a) memoria convencional, comprendida entre 0KB y 640KB; es memoria DRAM y se utiliza para cargar el sistema operativo y los programas de usuario, b) memoria superior, comprendida entre 640KB y 1MB; puede ser memoria ROM o RAM de tarjetas de expansión (ver tabla 5.3) y c) memoria extendida, comprendida entre 1MB y 16MB; es memoria DRAM sólo accesible desde el modo protegido del microprocesador (no es accesible desde DOS).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Segmento	Tipo	Direcciones	Características
A y B	Memoria de video	640KB-768KB	Memoria RAM ubicada en la tarjeta gráfica. La posición y comienzo de la memoria de video depende de la tarjeta gráfica y del modo de video activo.
C	Rutinas de controladores de disco y video	768KB-832KB	Memoria ROM de tarjetas de expansión.
D	Marco de página de la memoria expandida	832KB-896KB	También utilizado para interfaces de tarjeta de red.
E	Extensiones de la BIOS	896KB-960 KB	Suele estar libre.
F	Rutinas BIOS	960KB-1024KB	

Tabla 5.3: Segmentos de la memoria superior del PC-AT

En la figura 5.12 se muestra el espacio de direcciones de E/S del PC-AT.

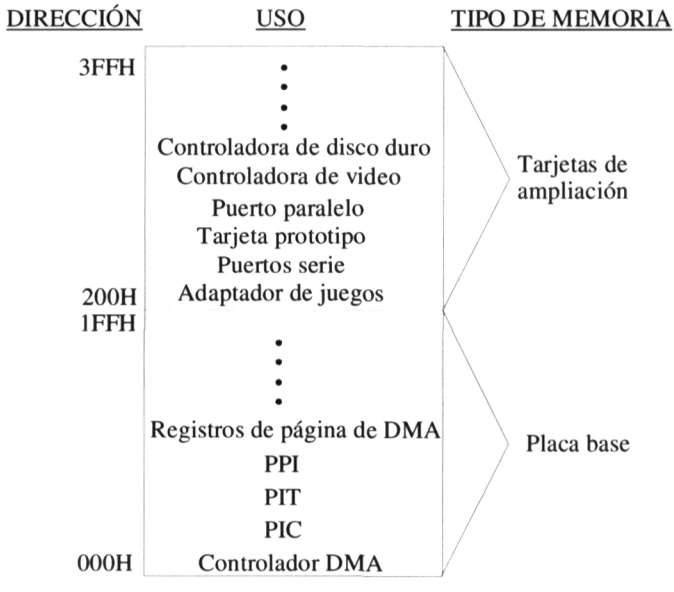


Figura 5.12: Espacio de E/S del PC-AT

4.2. Timers, interrupciones y DMA

Al igual que ocurre en el IBM PC, el PC-AT usa un 8254 de Intel (versión mejorada del 8253) para implementar el reloj en tiempo real (timer 0) y la onda cuadrada (timer 2) que excita al altavoz. El timer 1 está libre ya que dejó de usarse como solicitud de DMA para el refresco de la memoria principal.

Para ampliar el número de interrupciones, el PC-AT usa dos controladores

Publicaciones y Divulgación Científica UNIVERSIDAD DE MÁLAGA

de interrupciones 8259 conectados en cascada(ver figura 5.13). Se puede observar como la entrada **IRQ2** del controlador de interrupciones maestro se conecta a la salida **INT** del controlador esclavo[272].

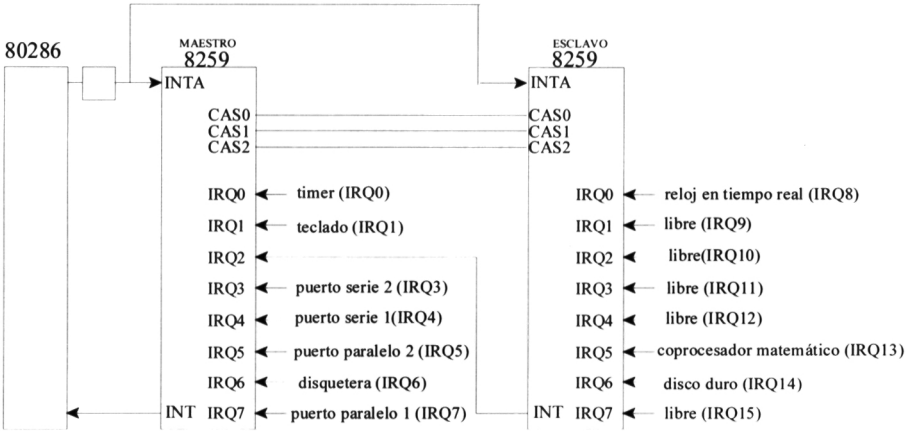


Figura 5.13: Controladores de interrupciones del PC-AT

La asignación de líneas **IRQx** para los diversos periféricos del ordenador es la que se muestra en la tabla 5.4 (por orden de prioridad).

Interrupciones	Función	Vector de interrupción	
		Maestro	Esclavo
IRQ0	Temporizador	INT 08h	
IRQ1	Teclado	INT 09h	
IRQ2	En los PC/XT: canal E/S	INT 0Ah	
IRQ8	Reloj de tiempo real		INT 70h
IRQ9	Simulación de IRQ2		INT 71h
IRQ10	Reservado		INT 72h
IRQ11	Reservado		INT 73h
IRQ12	Reservado		INT 74h
IRQ13	Coprocesador matemático		INT 75h
IRQ14	Controlador de disco duro		INT 76H
IRQ15	Reservado		INT 77h
IRQ3	COM2/COM4	INT 0Bh	
IRQ4	COM1/COM3	INT 0Ch	
IRQ5	LPT2 en el AT (Disco duro en PC/XT)	INT 0Dh	
IRQ6	Controlador de disquetes	INT 0Eh	
IRQ7	LPT1	INT 0Fh	

Tabla 5.4: Interrupciones del PC-AT

Un esquema similar (ver figura 5.14) se sigue para aumentar el número de canales de DMA: Hay dos 8237 conectados en cascada: un esclavo (canales 0,

[272]La **IRQ9** simula la **IRQ2** original que con está configuración se ha perdido. Cuando un periférico solicita una **IRQ2**, activa una **IRQ9**. El PC AT ejecuta una RSI en INT 71H que salta a INT 0AH después de enviar un comando EOI al controlador de interrupciones esclavo.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

1, 2 y 3) y un maestro (canales 5, 6 y 7). El canal 4 se pierde ya que se usa para interconectar los dos controladores. El 8237 esclavo es el encargado de las transferencias de 8 bits mientras que el maestro se encarga de las de 16 bits. El PC-AT no usa el canal 0 para refresco de memoria ni el 3 para el disco duro.

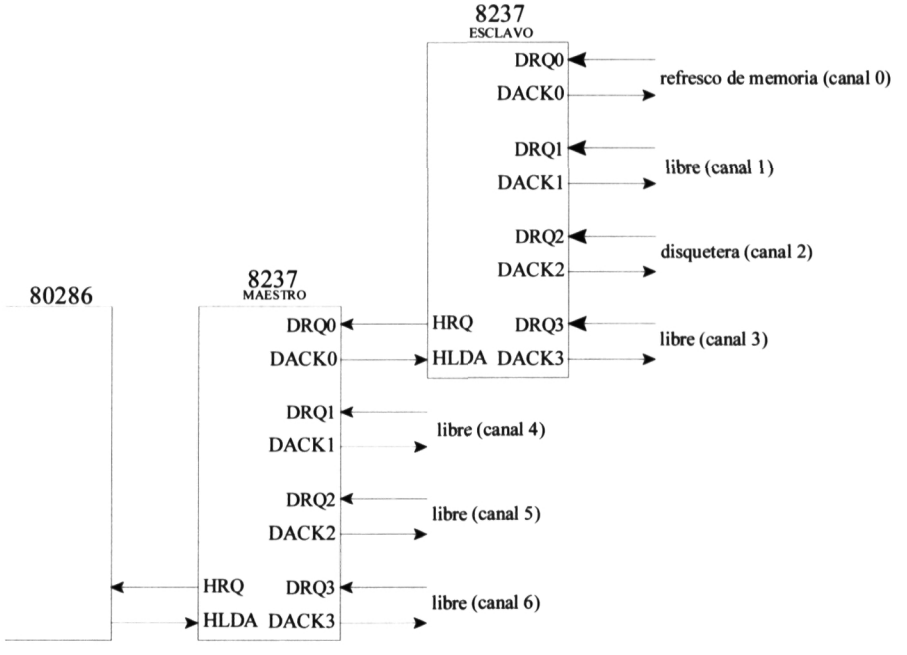


Figura 5.14: Controladores de DMA del PC-AT

El registro de páginas dispone de más bits (8 bits en total) que el utilizado en el IBM-PC y XT ya que el espacio de direcciones es mayor (ha pasado de 1MB a 16MB).

El 8237 esclavo ve el espacio de direcciones de 16MB como 256 páginas de 64 KB cada una mientras que el maestro, que trabaja con 16 bits, contempla 128 páginas de 128 KB cada una. Téngase en cuenta que en las transferencias de 16 bits sólo se usan las direcciones pares (`A0=0`), por lo que las páginas son de 128 KB.

4.3. Bus AT

El bus de expansión del PC-AT o bus ISA (Industry Standard Architecture) es compatible con el bus de expansión del IBM PC (ver figura 5.15).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Para añadir las señales que amplían el bus se incluye un segundo conector de 36 pines denominado bus de extensión que contiene: Nuevas líneas de datos (para ampliar a 16 bits) y direcciones (para ampliar a 16MB), señales `IRQx` y DMA adicionales, señales de control específicas para transferencias de 16 bits y multiproceso (ver tabla 5.5).

B1	GND	I/O CHCK	A1	D1	MEM CS16#	SBHE	C1
B2	RESET DRV	D7	A2	D2	I/O CS16#	LA23	C2
B3	+5v	D6	A3	D3	IRQ10	LA22	C3
B4	IRQ9	D5	A4	D4	IRQ11	LA21	C4
B5	-5v	D4	A5	D5	IRQ12	LA20	C5
B6	DRQ2	D3	A6	D6	IRQ13	LA19	C6
B7	-12v	D2	A7	D7	IRQ14	LA18	C7
B8	OWS	D1	A8	D8	DACK0#	LA17	C8
B9	+12v	D0	A9	D9	DRQ0	MEMR#	C9
B10	GND	I/O CHRDY	A10	D10	DACK5#	MEMW#	C10
B11	SMEMW#	AEN	A11	D11	DRQ5	SD08	C11
B12	SMEMR#	A19	A12	D12	DACK6#	SD09	C12
B13	IOW#	A18	A13	D13	DRQ6	SD10	C13
B14	IOR#	A17	A14	D14	DACK7#	SD11	C14
B15	DACK3#	A16	A15	D15	DRQ7	SD12	C15
B16	DRQ3	A15	A16	D16	+5v	SD13	C16
B17	DACK1#	A14	A17	D17	MASTER	SD14	C17
B18	DRQ1	A13	A18	D18	GND	SD15	C18
B19	REFRESH#	A12	A19				
B20	CLK	A11	A20				
B21	IRQ7	A10	A21				
B22	IRQ6	A9	A22				
B23	IRQ5	A8	A23				
B24	IRQ4	A7	A24				
B25	IRQ3	A6	A25				
B26	DACK2#	A5	A26				
B27	T/C	A4	A27				
B28	ALE	A3	A28				
B29	+5v	A2	A29				
B30	OSC	A1	A30				
B31	GND	A0	A31				

Figura 5.15: Bus de expansion del PC-AT

Señal	Dirección	Descripción
MEM CS16#	entrada	Memory Chip Select. Indica que el interfaz soporta un tiempo de espera, para la transferencia de 16 bits en el ciclo de acceso a memoria en curso
SBHE	salida	System Bus High Enable. Se usa para decodificar el tipo de operación en conjunción con SA0
I/O CS 16#	entrada	I/O Chip Select. Indica que el interfaz soporta un tiempo de espera, para la transferencia de 16 bits en el ciclo de acceso a E/S en curso
LA[23:17]	salida	Unlatched Address. Direcciones de memoria A[23:17]
IRQ[15:10]	entrada	Interrupt Request. Solicitudes de interrupción 15, 14, 13, 12 11 y 10
DACK[7:5]#	salida	Reconocimiento de DMA para los canales 7, 6 y 5
DRQ[7:5]	entrada	Solicitudes de DMA para los canales 7, 6 y 5
MEMR#	salida	Señal de control de lectura de memoria
MEMW#	salida	Señal de control de escritura de memoria
SD[15:8]	bidireccional	System Data Bus. Byte alto del bus de datos
MASTER#	entrada	Se usa en solicitudes de DMA, una vez activada la señal DACK correspondiente. Su activación causa la cancelación automática de la operación de DMA solicitada liberando el controlador DMA el bus del sistema

Tabla 5.5: Descripción de señales del bus de extension del PC-AT

El pin B8, que estaba reservado en el bus XT, es la señal `OWS` (Zero Wait

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

State) que permite evitar el estado de espera de los accesos a E/S. La señal `IRQ2` (pin B4) se convierte en `IRQ9`. Los pines `B12` y `B11` se convierten en `SMEMR#` y `SMEMW#` (específicos para accesos por debajo de 1MB) y las señales `MEMR#` y `MEMW#` se trasladan al bus de extensión. Por último, la señal `CLK` (pin `B20`) es un reloj de 8MHz.

La placa base tiene 8 ranuras de expansión, 6 con conector ISA y 2 con conector XT.

La potencia de la fuente de alimentación es superior a la del XT (unos 200W , 5V×20A+12V×7A).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5. PC-386

Compaq Deskpro 386 le ganó la batalla a IBM adelantándose en sacar al mercado un PC-AT compatible basado en el microprocesador 80386 de Intel[273]. Estos equipos hacían uso de circuitos integrados de alta integración (chipsets) que simplificaban el diseño de los PCs. El objetivo del chipset era reducir la complejidad y coste de la circuitería de la placa madre, incrementando la funcionalidad. Uno de estos chipsets fue el 82340 (figura 5.16).

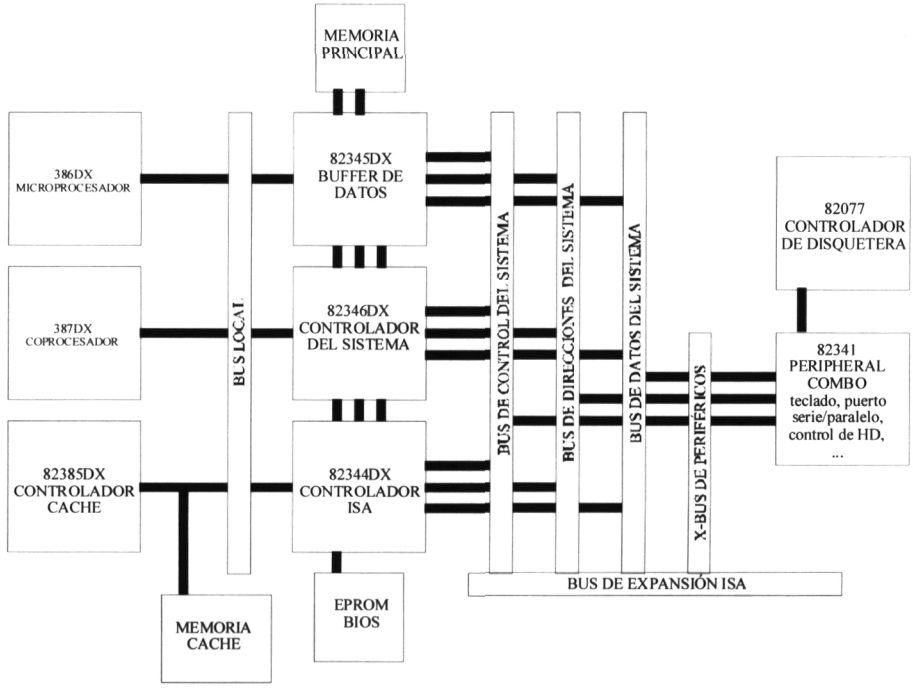


Figura 5.16: Plataforma 386DX basada en el chipset 82340 de Intel

5.1. Chipset 82340

El chipset 82340 ofrece una solución de alta integración en 4 chips para implementar una placa madre de un PC basado en el 80386DX. Además incluye como funciones adicionales: 2 canales de comunicaciones serie, un puerto de

[273]IBM apostó por su serie PS/2 que incluía un sistema de bus mejorado (más rápido y sencillo) pero incompatible con el bus AT: el MCA (Micro Channel Architecture).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

impresora paralelo y un interfaz a controlador de disco duro. No obstante, se necesitan algunas PALs, latches, transceptores, buffers y puertas lógicas para completar el sistema.

El chipset está compuesto de los siguientes chips:

- 82341. Periférico COMBO. Contiene 2 puertos de comunicaciones serie asíncrono, 1 puerto impresora paralelo (CENTRONICS), reloj en tiempo real, scratchpad RAM, controlador de ratón y teclado, interfaz a disco duro ATA y chip selects programables.
- 82345. Data Buffer. Realiza funciones de aislamiento y conmutación de buses. Es el responsable de la conmutación automática de datos entre el bus de datos de la UCP y el bus del BIOSROM/DRAM, bus I/O y bus del sistema. También implementa algunas otras funciones como lógica de generación y paridad.
- 82346. Controlador del sistema. Realiza muchas funciones del sistema: generación de las señales de reloj, señal de **RESET**, controlador de DRAM, interfaz al coprocesador matemático, etc..
- 82344. Controlador ISA. Implementa la mayoría de las señales del bus de expansión ISA, conduce el altavoz y selecciona el bus de datos de 8 ó 16 bits para la BIOS EPROM.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

6. PC-Pentium

La arquitectura PC-AT se mantuvo vigente con el 486 e incluso con el Pentium, apoyada por chipsets de cada vez mayor potencia e integración. Sin embargo, nuevas necesidades relacionadas con la demanda de mejores interfaces gráficos de usuario, multimedia, conectividad y gestión de consumo inteligente han ido transformando el diseño de esta exitosa arquitectura, aunque sin perjudicar su compatibilidad con el primigenio IBM PC. Algunos de los cambios más notorios ha sido la transición del bus ISA al PCI y la aparición de múltiples interfaces de E/S especializados (ATA, USB, AC'97, etc.).

Un chipset actual tiene dos chips principales (5.17): 1) el controlador del sistema[274] que proporciona el interfaz al microprocesador, memoria y buses locales de alta velocidad y se comunica con el controlador de E/S y 2) el controlador de E/S que implementa los buses de E/S del equipo: bus PCI, interfaz LPC, USB, etc.

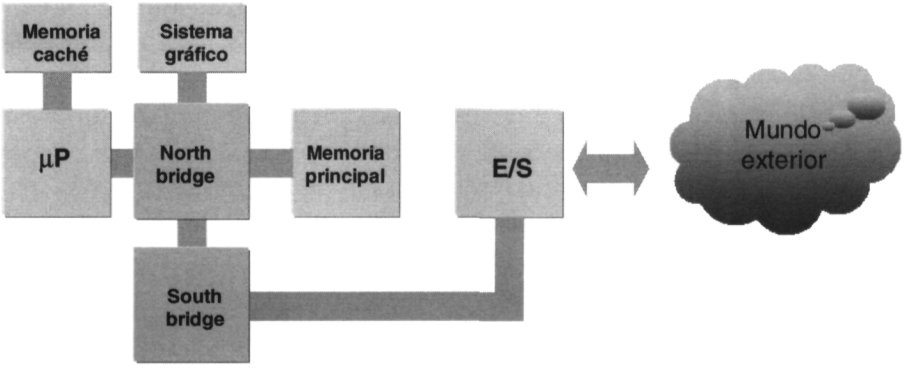


Figura 5.17: Chipset para Pentium

Hoy en día, el diseño digital de una placa base se reduce a interconectar el chipset con el microprocesador, memoria y los diferentes interfaces de E/S. La complicación se ha trasladado al diseño de la placa de circuito impreso y a la resolución de problemas relativos a disipación de potencia, interferencias, adaptación de impedancias en pistas de alta velocidad, etc.

La tabla 5.6 muestra una comparativa de algunos chipsets significativos de Intel para sus microprocesadores Pentium.

[274] El controlador del sistema ha ido cambiando de nombre con el tiempo: North Brigde, System Hub, Host Hub, Memory Controller, System Controller, etc. Igual le ha pasado al controlador de E/S: South Bridge, I/O Hub, I/O Controller, etc.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Característica	430FX	440	820	875
Microprocesador	Pentium	Pentium II	Pentium III	Pentium 4
Bus del sistema	66 MHz	100 MHz	133MHz	400-800MHz
Interfaz a memoria	EDO DRAM y PBSRAM	SDRAM	RDRAM	DDR DRAM
Bus de E/S	PCI(100MBps)	PCI 2.1	PCI 2.2	PCI2.3
Interfaz a disco duro	IDE	ATA33	ATA66	ATA100
Gestión de potencia	-	ACPI	SI	SI
AGP	-	AGP	AGPx4	AGPx8
USB	-	USB	USB	USB.2.0 (8)
LPC	-	-	SI	SI
Alert On LAN	-	-	SI	SI
AC'97	-	-	-	SI
Serial ATA	-	-	-	SI

Tabla 5.6: Tabla comparativa de chipsets de Intel para la familia Pentium

6.1. ACPI

La especificación ACPI (Advanced Configuration and Power Interface) permite un control inteligente de la energía consumida por cada dispositivo, recuperación rápida de estado de bajo consumo y encendido remoto.

Durante la inicialización del sistema operativo, el sistema de gestión de energía y configuración del sistema (OSPM, Operating System-directed configuration and Power Management) toma el control y se hace responsable del manejo de eventos de configuración de la placa base, de la gestión de energía, de las prestaciones del sistema y del estado térmico del equipo según preferencias del usuario y requerimientos de las aplicaciones y objetivos de usabilidad y calidad de servicio que impone el sistema operativo.

Las areas funcionales cubiertas por la especificación ACPI son:

- Gestión de energía. ACPI define mecanismos para que el ordenador entre y salga de diferentes estados de sleeping. También dispone de tablas que describen los dispositivos, incluido el microprocesador, de la placa base y sus estados de energía permitiendo un control del consumo por dispositivo basado en las necesidades de la aplicación en curso.
- Gestión de prestaciones. Estando el sistema activo, OSPM lleva a los dispositivos a diferentes estados de prestaciones, definidos por ACPI, para conseguir soluciones óptimas de compromiso entre velocidad y consumo.
- Configuración Plug and Play. ACPI especifica información utilizada para enumerar y configurar dispositivos de la placa base.
- Eventos del sistema. ACPI provee un mecanismo de eventos general que puede ser usado por diferentes eventos del sistema: térmicos, gestión energética, conexión/desconexión de dispositivos, etc..
- Gestión de la batería. Una batería compatible con ACPI necesita un interfaz de subsistema de batería inteligente, el cual es controlado por el

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

sistema operativo a través del controlador embebido, o un interfaz CMB (Control Method Battery). Este último está completamente definido por los métodos de control AML (ACPI Machine Language), permitiendo que un fabricante OEM (Original Equipment Manufacturer) elija cualquier tipo de batería e interfaz de comunicación soportado por ACPI. El sistema operativo puede alterar el comportamiento de la batería.

- Gestión térmica. ACPI provee un sencillo y escalable modelo que permite a los fabricantes OEM definir zonas e indicadores térmicos y métodos para enfriar zonas térmicas.
- Controlador embebido. ACPI define un interfaz de comunicaciones hardware/software entre un enumerador de bus del S.O. y un controlador embebido.
- Controlador SMBus. ACPI define un interfaz de comunicaciones hardware/software estándar entre un driver de bus del sistema operativo y un controlador SMbus. SMBus provee un interfaz basado en el bus I2C para manejar periféricos tales como SPD (Serial Presence Detection) sobre RAM, sensores térmicos, tarjetas CNR, tarjetas PCI, etc.

6.2. Interfaz LPC

El interfaz LPC (Low Pin Count) es un sencillo interfaz de 4 bits que pretende sustituir al antiguo bus ISA. Permite la conexión del puerto serie, puerto paralelo, puerto de juegos, teclado y ratón PS/2, interfaz de infrarrojos, interfaz a disquetera, etc.. La figura 5.18 muestra un ejemplo de transferencia LPC. `LCLK` es un reloj de 33MHz. `LFRAME#` indica el comienzo de un nuevo ciclo y `LAD[3:0]` multiplexa comandos, direcciones y datos.

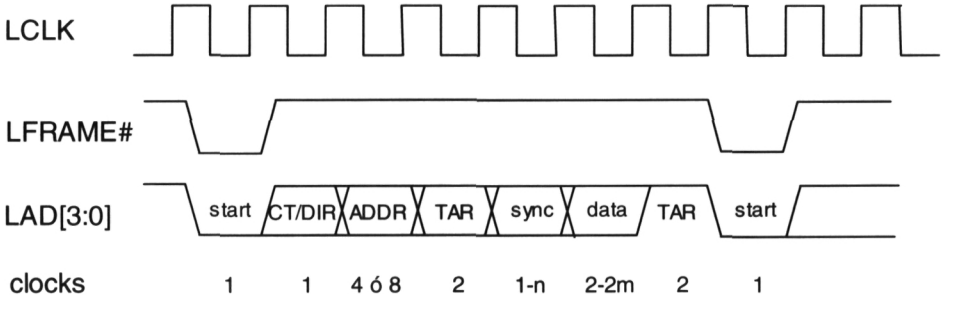


Figura 5.18: Interfaz LPC

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Un ciclo LPC empieza cuando el controlador activa la señal **LFRAME#** y pone la información apropiada sobre **LAD[3:0]**. Esta información puede ser tipo de ciclo (memoria, E/S o DMA), lectura o escritura, tamaño de la ráfaga, dirección, número de canal de DMA, o un comando de concesión del bus. Posteriormente, si es un ciclo de escritura también envía los datos. A continuación, el controlador invierte el bus mediante el comando TAR (Turn Around) y queda a la espera de que el periférico seleccionado le responda. Una vez culmina la operación, el ciclo termina con una nueva inversión del bus.

6.3. Chipset 875P

El chipset 875P de Intel está diseñado para sistemas basados en el procesador Pentium 4. Tiene dos componentes principales (ver figura 5.19): MCH y ICH5. MCH (Memory Controller Hub) es el controlador del sistema 82875P. ICH5 es el controlador de E/S y puede ser un 82801EB ICH5 o un 82801ER ICH5R.

El MCH arbitra el flujo de información entre los cinco interfaces del sistema:

- Bus del sistema (FSB, Front Side Bus). Puede alcanzar 400 MTps[275], 533 MTPs y 800 MTps usando la técnica Quad Pump[276].
- Sistema de memoria. El MCH integra un controlador DDR de memoria del sistema con dos "canales" de 64 bits cada uno.
- AGP(Accelerated Graphics Port). El MCH soporta un slot modo AGP 8X. Este slot cumple los requerimientos de la especificación AGP 3.0 incluyendo características eléctricas AGP 0.8V y 1.5V.
- HI (Hub Interface). MCH y ICH5 se interconectan a través de este interfaz propietario de Intel. HI corre a 266 MTps con un reloj base de 66 MHz y usa señalización 1.5V.
- CSA(Communications Streaming Architecture). El interfaz CSA conecta el MCH con el controlador 82541EI Gigabit Ethernet (GbE). El interfaz CSA corre a 266 MTps (con un reloj base de 66 MHz) y usa señalización 1.5V.

[275] MTps son Megatransferencias/segundo.

[276] La técnica Quad Pump permite llevar a cabo 4 transferencias en un ciclo de reloj.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

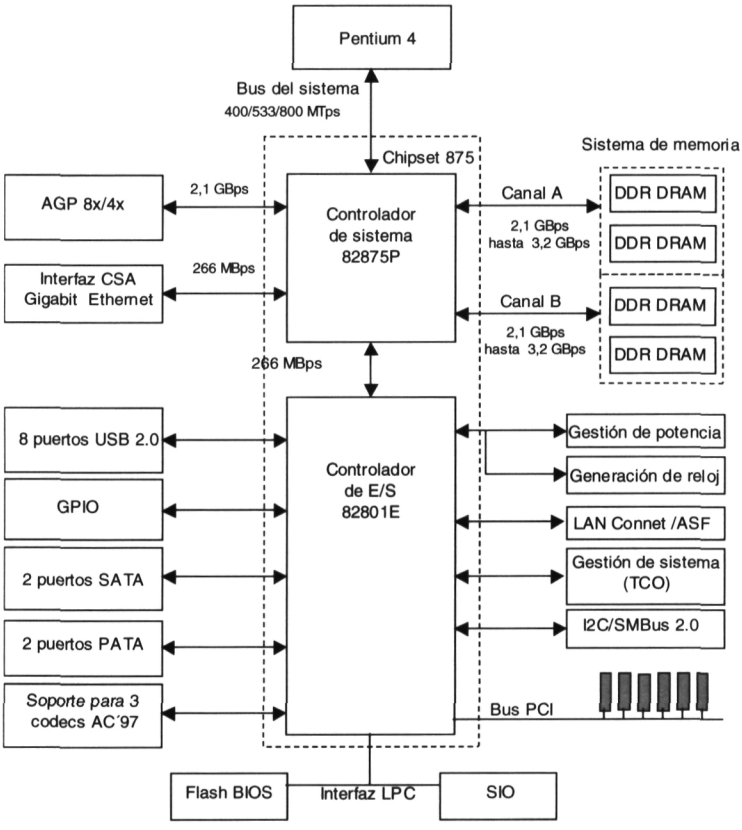


Figura 5.19: Plataforma Pentium 4 basada en el chipset 875P de Intel

El ICH5 o ICH5R provee el subsistema de E/S con acceso al resto del sistema. El ICH5/ ICH5R integra muchas funciones de E/S:

- Interfaz hub para acceso al MCH.
- Controlador de disco duro de 2 canales Ultra ATA/100.
- 2 controladores SerialATA.
- 8 puertos USB 2.0.
- APIC E/S. Las capacidades de interrupción de la plataforma ICH5 mantienen el soporte de hasta 8 pines de interrupciones PCI y interrupciones basadas en mensajes PCI 2.3.
- Controlador SMBus 2.0.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Controlador de gestión ASF (Alert Standard Format). El ICH5 integra un controlador de gestión ASF 1.03. El controlador ICH5 ASF usa el SMLink internamente como un bus dedicado para conectarse con el controlador ICH5 LAN.
- Interfaz LPC/Flash BIOS.
- Controlador de gestión del sistema integrado.
- Interfaz AC´97 2.3. Soporta hasta 3 codecs con funciones PCI independientes para audio y modem. El ICH5 expande la capacidad de audio con soporte para hasta 6 canales de salida de audio PCM (full AC3 decode) para sonido envolvente[277].
- Interfaz PCI 2.3.
- Controlador LAN integrado. El ICH5 soporta varios componentes: 82562ET/82562EZ de Intel para conexión básica Ethernet 10/100, 82562EM/82562EX de Intel que provee una conexión Ethernet 10/100 con capacidades de gestión añadidas, controlador 82540EM GbE de Intel y controlador 82551QM Fast Ethernet.
- RAID nivel 0 (disponible con el ICH5R 82801ER) que alivia cuellos de botella del disco aprovechando los dos controladores independientes SATA.

Para minimizar jitter, mejorar el routing, y reducir coste, los sistemas basados en el chipset 875P usan una solución de reloj único, el CK409. En esta configuración, el CK409 provee 3 pares de salida diferenciales seleccionables 100/133/200 MHz para todos los agentes del bus host, un par de salida diferencial de 100 MHz para serial ATA, 2 relojes de 48 MHz, 5 relojes de 66 MHz, 10 relojes de 33 MHz y 2 relojes de 14 MHz.

[277] Los canales son frontal izquierdo, frontal derecho, frontal central, trasero izquierdo, trasero derecho y subwoofer.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

7. El PXA27x

El procesador PXA27x es un sistema de propósito general integrado en un chip (SOC, System-On-a-Chip) basado en la microarquitectura XScale de Intel y especializado para el diseño de equipos portátiles y teléfonos celulares. Intel ha incorporado en este sistema potentes mecanismos de gestión de potencia y de seguridad de datos.

La figura 5.20 muestra un diagrama de bloques de su arquitectura. El núcleo es un microprocesador XScale de Intel rodeado de múltiples controladores: memoria, DMA y E/S, USB, LCD y videocámara. Dispone de 256 KB de memoria interna y de un potente sistema de gestión de potencia.

La microarquitectura XScale es un RISC superpipeline de alta velocidad y muy bajo consumo que trabaja desde 104 MHz hasta 624 MHz. Dispone de una gestión de voltaje dinámica que escala la frecuencia y el voltaje al vuelo (on-the-fly) para permitir a aplicaciones usar en cada momento la mejor combinación de prestaciones y consumo[278]. El núcleo cumple con el conjunto de instrucciones V5TE de la arquitectura ARM (excluyendo punto flotante) y sigue el modelo de programación ARM. El PXA27x también soporta la tecnología Wireless MMX de Intel.

La memoria interna es una SRAM distribuida en 4 bancos de 64 KB cada uno y puede ser usada tanto para código como para datos. La gestión de potencia es independiente para cada banco los cuales se pueden configurar para retener la información en modos de bajo consumo.

El PXA27x provee 121 señales de E/S de propósito general altamente multiplexadas (GPIO) para su uso en la generación y captura de señales de entradas/salida de aplicación específica.

También integra un interfaz de testeo JTAG que cumple el estándar IEEE 1149.1-1990 y IEEE 1149.1a-1993.

7.1. Controlador de memoria

El controlador de memoria externa se basa en una arquitectura unificada (UMA, Unified Memory Architecture), donde todos los dispositivos de memoria comparten un bus de datos y direcciones común. El procesador distingue tres espacios de direcciones:

- SDRAM con 4 particiones (bancos). Admite 100 MHz, bajo consumo y capacidades de hasta 1GB.

[278] El voltaje puede variar de 0.85 V a 1.55 V.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

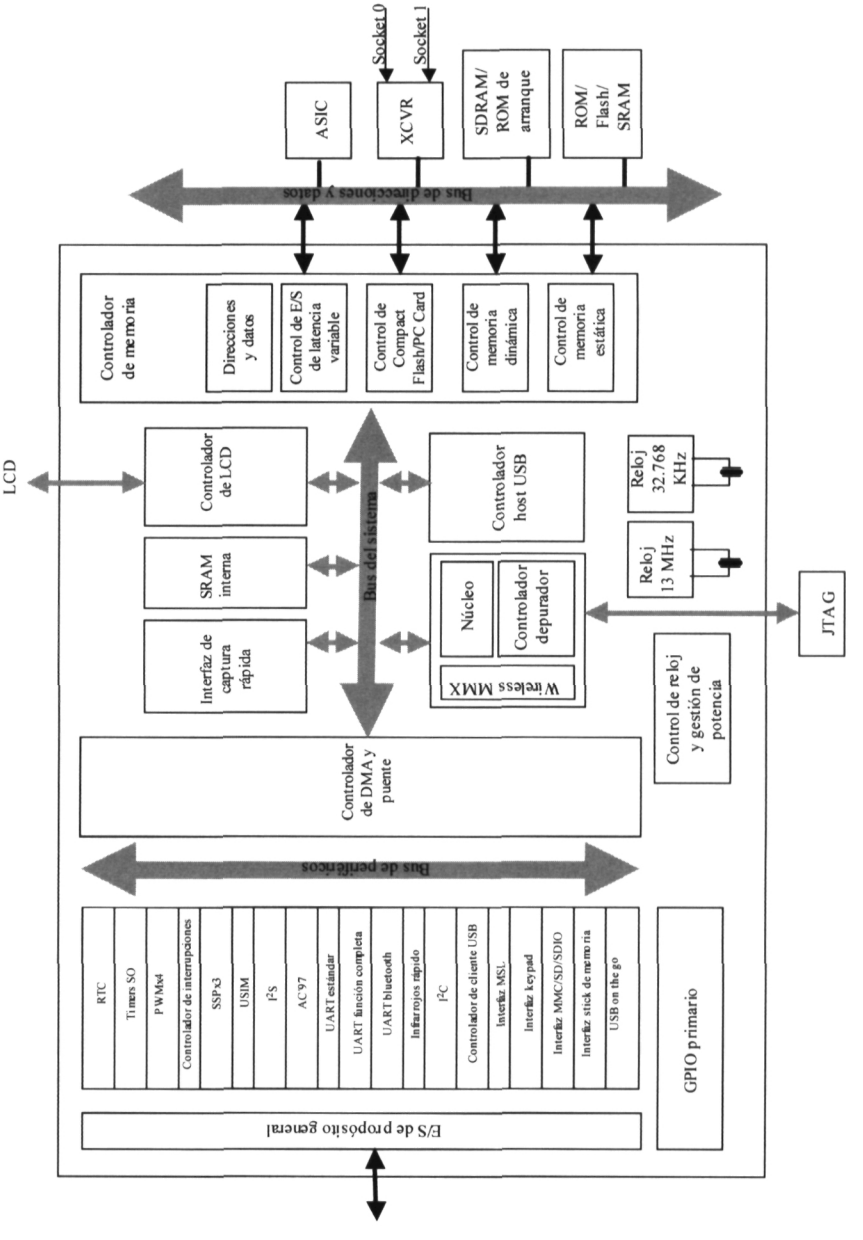


Figura 5.20: Diagrama de bloques de PXA27x de Intel

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Memoria estática con 6 particiones (bancos). Admite ROM, flash síncrona y asíncrona, SRAM y chips compañeros[279] de latencia variable (VLIO, Variable Latency I/O). El banco 0 es de arranque (bootable).
- PC card y memoria de expansión CompactFlash con 2 particiones (o sockets).

Todos estos tipos de memoria son programables a través de registros de configuración del interfaz de memoria. Las solicitudes de memoria son enviadas a una cola de tipo FIFO.

7.2. Controlador de DMA y E/S

El Controlador de DMA transfiere datos a/desde memoria en respuesta a solicitudes generadas por periféricos o chips compañeros. Soporta solicitudes de 63 dispositivos y 32 canales, con la capacidad de preprogramar cualquier solicitud de cualquier canal.

El controlador de E/S incluye:

- Un reloj de tiempo real (RTC, Real-Time Clock) de 32 bits de propósito general. Tiene 5 funciones básicas: Timer, wristwatch, stopwatch, periodic interrupt y trimmer. Usa un cristal de 32,76 KHz.
- Un bloque de timers del sistema operativo. Provee un conjunto de canales de timers que permiten al software generar interrupciones temporizadas (o eventos wake-up). Opera a 3,25 MHz.
- 4 canales de modulación de anchura de pulsos (PWM, Pulse Width Modulation) independientes. Entre sus utilidades está el control del brillo de la salida de un LED modulando el tiempo ON o el control del contraste del LCD.
- Un controlador de interrupciones. Provee una jerarquía de dos niveles de interrupciones. Puede recibir interrupciones desde periféricos (fuentes secundarias) o de dispositivos internos del PXA27x (fuentes primarias). Las fuentes secundarias múltiples son usualmente mapeadas en una fuente primaria[280].

[279] Un chip compañero (companion chip) se define como un dispositivo que tiene la habilidad de controlar el bus externo, mientras que un periférico externo tiene que ser controlado por el controlador de memoria.

[280] Por ejemplo, el controlador DMA es una fuente primaria con 32 posibles fuentes secundarias.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- 3 puertos serie síncronos (SSP, Synchronous Serial Ports). Se conectan a ADCs externos, codecs de telecomunicaciones y audio y otros dispositivos que usan protocolos serie para la transferencia de datos. Soportan los protocolos Microwire de National Semiconductor Microwire, SPP (Synchronous Serial Protocol) de Texas Instruments, SPI (Serial Peripheral Interface) de Motorola y PSP (Programmable Serial Protocol).

- Interfaz a USIM (Universal Subscriber Identity Module). Soporta comunicaciones con tarjetas inteligentes[281] (smart cards) de teléfonos móviles según estándar ISO 7816-3 y especificación 3G TS 31.101 del 3GPP (3rd Generation Partnership Project).

- Interfaz a I2S. I2S (Inter-IC sound) es un protocolo definido por Philips Semiconductor para transferir señales de audio digital de dos canales (de 16 bits y hasta 48KHz de frecuencia de muestreo) de un chip a otro.

- Interfaz AC´97. AC´97 proporciona canales independientes de 16 bits para entrada y salida de audio PCM estéreo, modem y entrada de micrófono mono.

- 3 puertos UART: Estándar, full-function y bluetooth. Cada puerto contiene una UART de velocidad máxima 921,600 bps y un interfaz de infrarrojos lento (hasta 115.2 kbps) según especificación IrDA. Soportan las funciones de los chips estándar 16550A and 16750. Las UARTs Bluetooth y tienen características extra para implementar las señales de control de los modems.

- Puerto de comunicaciones de infrarrojos rápido (FICP, Fast Infrared Communications Port). Opera en half-duplex a 4 Mbps, usa modulacion de pulsos de 4 posiciones (4PPM, 4 position pulse modulation) y un protocolo de paquete especializado. Provee una conexión directa a los transceivers LED que cumplen la norma IrDA.

- Interfaz a bus I2C. Soporta los modos estándar a 100 kbps y modo fast a 400 kbps. Permite al PXA27x comunicarse con dispositivos I2C y microcontroladores para funciones de gestión del sistema.

- Controlador de cliente USB. cumple con la versión 1.1. Soporta 24 endpoints, 12 Mbps y modo esclavo.

[281] Una tarjeta inteligente típica consiste en una UCP, memoria flash e interfaz de comunicaciones serie. Muchas incluyen aceleradores de encriptado ya que suelen estar orientadas a aplicaciones de seguridad.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Interfaz MSL (Mobile Scalable Link). Es un interfaz físico entre chips de alta velocidad, escalable y de bajo consumo especializado en plataformas móviles e inalámbricas[282]. Consiste en dos 2 enlaces unidireccionales de alta velocidad (192 Mbps at 48 MHz) independientes. Permite la transferencia de multiples canales lógicos y diferentes requerimientos de calidad de servicio (útil en señalización telefónica, voz o video en tiempo real, etc.).

- Interfaz a keypad. Soporta teclado de matriz de 8×8, hasta 8 teclas directas y hasta dos codificadores rotatorios usados en la implementación de scroll keys, jog-dials, and thumbwheels.

- Interfaz para tarjetas de memoria. Soporta los protocolos de MultiMediaCard (MMC) V3.2, Secure Digital (SD) V1.01 y SDIO (SD I/0) V1.0.

- Interfaz stick de memoria compatible con estándar Sony Memory Stick.

7.3. Controlador LCD

El controlador LCD genera el interfaz correspondiente y permite programar tipo de displays, resolución, buffer de video, profundidad de pixel, overlays, cursor hardware y formateo de salida de datos. Soporta pantallas duales de hasta 800×600 pixels y 24 bits de color. Está integrado con la tecnología Quick Capture para facilitar la previsualización de video.

7.4. Controlador de cámara

El interfaz Quick Capture Camera permite una conexión directa del procesador a una cámara[283]. El interfaz adquiere las señales de control y datos del sensor de imagen, formatea los datos y los encamina a memoria mediante DMA.

Soporta interfaces paralelo y serie y varios formatos para espacios de color RGB y YCbCr. Tiene tres modos de trabajo: modo Quick View mode (bajo consumo y previsualización en tiempo real), modo Quick Shot (imagen de alta resolución de hasta 4 Mpixels) y modo Quick Video (full-motion y captura de video de alta calidad).

[282] MSL cumple los requerimientos de la Personal Internet Client Architecture de Intel, la cual permite reducir los tiempos de desarrollo de los dispositivos móviles.

[283] Cámaras del tipo sensor de imagen CMOS o CCD.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

8. Ejemplos

8.1. Ejemplo de diseño de tarjeta de expansión

La figura 5.21 muestra el diagrama eléctrico de una tarjeta de expansión que maneja 3 contadores (integrados en el chip 8253). El diseño es muy general ya permite cambiar los márgenes de direcciones de E/S y generar hasta 16 chip selects de los que realmente sólo se aprovechan 4 (ver tabla 5.7): tres para acceder a los 4 registros del 8253 y un cuarto para implementar un puerto de escritura que permita controlar la entrada `GATE` de uno de los contadores del 8253.

DIRECCIONES E/S	CONTENIDO
304H	Contador 0 del 8253
305H	Contador 1 del 8253
306H	Contador 2 del 8253
307H	Registro de control del 8253
30CH	Gate del contador 0 del 8253

Tabla 5.7: Direcciones de E/S de la tarjeta de expansión

El buffer 74LS244 (tabla 5.8) aísla las señales del bus de expansión de la tarjeta y proporciona la corriente necesaria.

G#	IN	OUT
0	0	0
0	1	1
1	X	alta impedancia

Tabla 5.8: Tabla de verdad de buffer 74LS244

El buffer 74LS245 (tabla 5.9) es un 74LS244 bidireccional controlado por la señal `DIR`.

G#	DIR	OPERACIÓN
0	0	B → A
0	1	A → B
1	X	alta impedancia

Tabla 5.9: Tabla de verdad de transceptor 245

Los switches `S[7:0]` permiten programar el margen de direcciones de la tarjeta. Para ello, se hace uso de un comparador 74LS688 (tabla 5.10).

G#	IN	OUT
0	P[7:0]=Q[7:0]	0
0	P[7:0]≠Q[7:0]	1
1	X	1

Tabla 5.10: Tabla de verdad de comparador 688

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

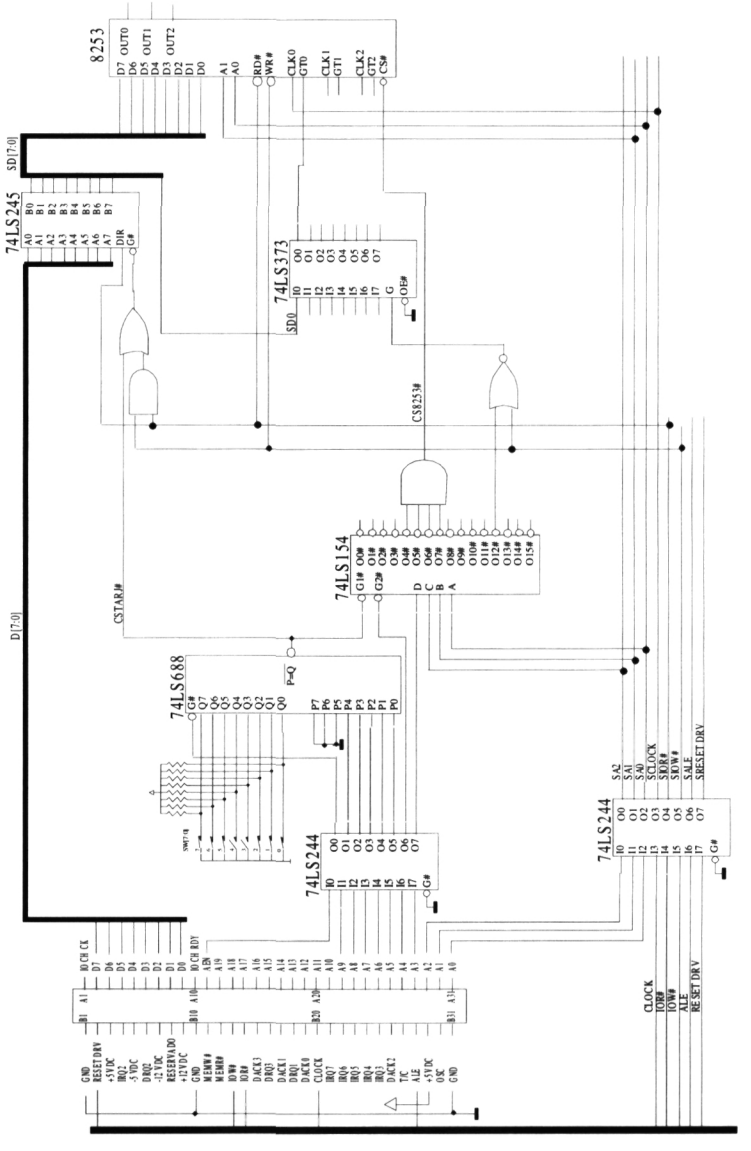


Figura 5.21: Ejemplo de tarjeta de expansión

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

El comparador sólo compara las líneas de direcciones `A[9:5]`, conectadas a sus entradas `P[4:0]` con los switches `S[4:0]` conectados a sus entradas `Q[4:0]`. Las entradas `P[7:5]` y `Q[7:5]` se ponen a 0. La señal `AEN` habilita el 74LS688 cuando el ciclo no es de DMA.

En el ejemplo los valores de los switches `S[4:0]` son `11000`, de manera que la salida `CSTARJ#` del comparador sólo se activa cuando la dirección de E/S está dentro del margen `300-31FH` (ver tabla 5.11).

A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	A[9:0]
↑	↑	↑	↑	↑						
P[4]	P[3]	P[2]	P[1]	P0]						Puerto P del comparador
Q[4]	Q[3]	Q[2]	Q[1]	Q[0]						Puerto Q de comparador
↓	↓	↓	↓	↓						
S[4]	S[3]	S[2]	S[1]	S[0]						Switches
1	1	0	0	0	X	X	X	X	X	300H-31FH

Tabla 5.11: Margen de direcciones seleccionado por el comparador

Para decodificar el resto de direcciones y generar los chip selects se usa un decodificador de 4 a 16 74LS154.

G1#	G2#	D	C	B	A	O[15:0]#
0	0	0	0	0	0	FFFEH
0	0	0	0	0	1	FFFDH
0	0	0	0	1	0	FFFBH
0	0	0	0	1	1	FFF7H
0	0	0	1	0	0	FFEFH
0	0	0	1	0	1	FFDFH
0	0	0	1	1	0	FFBFH
0	0	0	1	1	1	FF7FH
0	0	1	0	0	0	FEFFH
0	0	1	0	0	1	FDFFH
0	0	1	0	1	0	FBFFH
0	0	1	0	1	1	F7FFH
0	0	1	1	0	0	EFFFH
0	0	1	1	0	1	DFFFH
0	0	1	1	1	0	BFFFH
0	0	1	1	1	1	7FFFH
0	1	X	X	X	X	FFFFH
1	0	X	X	X	X	FFFFH
1	1	X	X	X	X	FFFFH

Tabla 5.12: Tabla de verdad de decodificador LS154

Para seleccionar el espacio 300H-30FH, `A4` tiene que estar a 0. Observe que la línea `A4` se conecta al decodificador 74LS154 y actúa junto con `CSTARJ#` como habilitadora del chip. Las líneas `A[3:0]` son utilizadas por este decodificador para generar un chip select distinto para cada dirección comprendida en el margen `300H-30FH`.

La señal `CS8253#` de chip select del 8253 (direcciones 304H-307H) se activa cuando se activa alguna de las señales `O[7:4]` del decodificador LS154:

```
CS8253#=O4#· O5#· O6#· O7#
```

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Para implementar el puerto de escritura (dirección 30CH) se usa un 74LS373 (ver tabla 5.13).

G	OE#	In	On
1	0	1	1
1	0	0	0
0	0	X	Qo
X	1	X	alta impedancia

Tabla 5.13: Tabla de verdad de registro 373

La entrada G se activa cuando se activa la salida O12 del decodificador LS154 y la señal de escritura SIOW#:

```
CS373= (SIOW#+ CS12#)#
```

La señal de habilitación G245# del buffer 245 se activa cuando se ha seleccionado la tarjeta (CSTARJ# activa) y se ha activado la señal de lectura (SIOR#) o escritura (SIOW#):

```
G245#=(SIOW#·SIOR#)+ CSTARJ#
```

La señal DIR de dirección se controla con la señal de lectura SIOR#:

```
DIR=SIOR#
```

Como ejemplo, para escribir en el puerto 30CH se utilizaría el siguiente código:

```
MOV DX, 30CH; Carga la dirección del puerto en el registro DX

MOV AL, 01H ; Carga el dato a escribir en el registro AL

OUT DX, AL  ; Escribe el dato en el puerto seleccionado
```

Observe que escribimos un byte pero sólo utilizamos el bit menos significativo SD0.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

8.2. Cálculo de anchos de banda de interfaces

La tabla 5.14 muestra un resumen del ancho de banda de los diferentes interfaces del controlador de sistema del chipset 875P.

Interfaz	Frecuencia de reloj (MHz)	Muestras por ciclo de reloj	Anchura de bus de datos (bytes)	Ancho de banda
Bus del sistema	200	4	8	6.4 GB/s
DDR-SDRAM	133/166/200	2	8	2.1 / 2.7 / 3.2 GB/s (por canal)
AGP	66	8	4	2.1 GB/s
CSA	66	4	1	266 MB/s

Tabla 5.14: Calculo de ancho de banda de interfaces del controlador de sistema del chipset 875P

La tabla 5.15 muestra un resumen del ancho de banda de los diferentes interfaces del ICH5 del chipset 875P.

Interfaz	Frecuencia de reloj	Muestras por ciclo de reloj	Tasa de datos (Megamuestras/s)	Anchura de bus de datos (bits)	Ancho de banda
HI	66MHz	4	266	8	266MB/s
PCI2.3	33MHz	1	33	32	133MB/s
IDE	$\leq$44.444MHz (escritura), $\leq$ 50MHz (lectura)	1	44.444 (escritura), 50 (lectura)	16	88.9MB/s (escritura) 100MB/s (lectura)
SATA	750MHz	2	1500	1	150MB/s
LCI	5MHz - 50MHz	1	5 - 50	3	1.875MB/s - 18.75MB/s
AC'97	12.288MHz	1	12.288	1	1.536MB/s
LPC	33MHz	1	33	4	16.5MB/s
USB 2.0	$\leq$ 240MHz	hasta 2	480	1	60MB/s
SMBus	10KHz - 16 kHz	1	10	1	1.25 KB/s

Tabla 5.15: Cálculo de ancho de banda de interfaces del controlador de E/S del chipset 875P

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

9. Bibliografía

Libros

Antonakos, J.L. and Adamson, T: Microcomputer Repair, Prentice Hall Career&Technology, 1998. ISBN: 0-13-893454-1.

Beeson, D.L.:Assembling and Repairing personal Computers, Prentice Hall Career&Technology, 1997. ISBN: 0-13-340225-8

Brenner, R.C.: Mantenimiento del IBM-PC, Paraninfo, 1989, ISBN 84-283-1673-2.

L.E.Frenzel: El libro del IBM PC,XT,AT Programación y aplicaciones, Anaya multimedia, 1988. ISBN 84-7614-036-3.

García Guerra, A: Los microprocesadores x86 y la arquitectura del PC. ETSIT de Madrid. Servicio de publicaciones, 1995. ISBN: 84-7402-244-4.

R.Jourdain: Solucionario del programador para IBM PC, XT, AT y compatibles, Anaya multimedia, 1986. ISBN: 84-7614-155-6.

Mazidi, M.A. y Mazidi, J.G.: The 80x86 IBM PC and compatible Computers, Vol. II: Design and Interfacing of the IBM PC and compatible Computers, Prentice Hall Career&Technology, 1998. ISBN: 0-13-758491-1.

P. Norton: Guía del programador para el IBM-PC, Anaya multimedia, 1987. ISBN: 84-7614-130-0.

Norton, P. y Socha, J.:Nueva guía del programador en ensamblador para IBM PC, XT, AT y compatibles, Anaya Multimedia, 1991. ISBN: 84-7614-287-0.

Oguic, P.: Control electrónico con el PC, Paraninfo, 1996.

Schuller, U., and Veddeler, H.: Ampliar y reparar su PC, 3ª Ed., Marcombo Importación, 1998. ISBN: 970-15-0084-9.

Seyer, M.D.: Conexiones en el IBM PC/XT/AT, teoría y práctica de periféricos, comunicaciones y configuraciones, Anaya multimedia, 1985, ISBN: 84-7614-119-X.

Stallings, W.: Organización y Arquitectura de Computadores. Diseño para optimizar prestaciones,4ª Ed., PrenticeHall, 1997. ISBN: 84-89660-24-7

Tanenbaum, A, and Goodman, J.R.: Structured Computer Organization, Prentice-Hall International, 1998. ISBN: 0-13-095990-1

Tisher, M. y Jennrich, B: PC interno 5, Marcombo Boixareu Editores, 1998. ISBN: 84-267-1081-6.. [Varios88] Varios autores:: Interfacing Sensors to the IBM PC. (Tompkings, W.J. and Webster, J., Eds.) Prentice Hall, 1988. ISBN: 0-13-469081-8.

Varios autores: Programación en Turbo C, McGraw Hill, 1990. ISBN: 84-7615-491-7.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Zoller, B.: Circuitos Electrónicos con el PC, Marcombo Importación, 1998. ISBN: 84-267-1125-1.

Manuales

IBM PC Technical Manual, IBM Press. 1981

Especificaciones

Especificaciones de APCI (Advanced Configuration and Power Interface), Hewlett-Packard, Intel, Microsoft, Phoenix y Toshiba, Rev. 3.0, 2004.

Revistas

Revistas divulgativas

Byte, McGraw Hill Companies Inc, mensual. ISSN: 0360-528.

Family PC, Ziff-Davis, mensual.

Home PC, CMP, mensual.

Mac User, Ziff-Davis, mensual. ISSN: 0884-0997.

Macworld, IDG Communications, mensual. ISSN: 0741-8647.

PC Actual, Business Publications España, S.A., mensual. ISSN: 1130-9954

PC Computing Ziff-Davis, mensual.

PCmanía, Hobby Press, S.A., mensual.

PC Magazine, America Iberica, mensual, ISSN: 1133-8113

PC World, IDG Communications, mensual.

Smart Computing, Sandhills, mensual.

Revistas profesionales

Computer Design, PennWell Publishing Company, ISSN-0010-4566.

Mundo Electrónico, Boixareu Ed. ISSN: 0300-3787

Revistas científicas

IEEE Annals of the History of Computing, IEEE Computer Society, trimestral. ISSN: 1058-6180.

Computers and Digitals Techniques, IEE Proceedings-E., IEE Publishing. ISSN: 0143-7062.

Computer, IEEE Computer Society, mensual, ISSN 0018-9162.

Computers Surveys, ACM

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Micro, IEEE Computer Society, bimensual. ISSN: 0272-1732.

Multimedia, IEEE Computer Society, trimestral.

Spectrum, IEEE Computer Society,

Transactions on Computers, mensual. IEEE Computer Society, ISSN 0018-9340.

Transactions on Computers System, ACM.

Páginas WWW

Guías

El Universo Digital del IBM PC, AT Y PS/2, 4ª Edición de Ciriaco García de Celis. http://www.gui.uva.es/udigital/

Pcguide. www.pcguide.com

PC Webopaedia. www.pcwebopedia.com

Tom hardware Guide. www2.tomhardware.com

PC Technique Guide. www.pctechguide.com

Manuales

Intel 875P Chipset: Intel 82875P Memory Controller Hub (MCH) Datasheet http://developer.intel.com/ design/chipsets/ datashts/252525.htm

Intel 875P Chipset Customer Reference Board Schematics http://developer.intel.com/ design/chipsets/ schematics/252812.htm

Intel Pentium 4 Processor with 512-KB L2 Cache on 0.13 Micron Process Datasheet http://developer.intel.com/ design/pentium4/ datashts/298643.htm

Intel 82801EB I/O Controller Hub 5 (ICH5) and Intel 82801ER I/O Controller Hub 5 R (ICH5R) Datasheet http://developer.intel.com/ design/chipsets/ datashts/252516

System Management Bus Specification http://www.smbus.org/

AC'97 Specification, Revision 2.3 http://www.intel.com/ ial/scalableplatforms/ audio/index.htm/

Communication and Network Riser Specification, Revision 1.2 http://developer.intel.com/ technology/cnr/ CNRspec_12.pdf

Alert Standard Format Specification, Revision 1.03 http://www.dmtf.org/standards/ standard_alert.php

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

10. Guía de estudio

1. Describir las características generales de un IBM PC.
2. Analizar el diagrama eléctrico a nivel de bloques de un IBM PC. Describir la función de cada elemento. Distinguir entre los diferentes buses de la placa base.
3. Explicar cómo son los espacios de direcciones de memoria y E/S de un IBM PC y cuál es su uso.
4. Entender el funcionamiento de cada señal del bus de expansión del PC.
5. Comparar un IBM PC-AT-286 con el IBM PC. Establecer diferencias con respecto a velocidad, prestaciones, espacio de direcciones, interrupciones y canales de DMA.
6. Indicar las diferencias entre el bus de expansión del IBM-PC y el bus ISA.
7. Definir el concepto de chip set.
8. Dibujar el diagrama de bloques de la placa base de un PC basado en Pentium (chipset, buses e interfaces).
9. Enumerar las características de la especificación ACPI.
10. Enumerar las características del interfaz LPC.
11. Describir la utilidad del interfaz SMbus.
12. Enumere las características del PX2A7x de Intel.
13. Calcular el ancho de banda de los diferentes interfaces de una placa base (ver tablas 5.14 y 5.15).
14. Analizar el hardware de una tarjeta de expansión ISA y describir la función de cada circuito.
15. Diseñar el decodificador de direcciones de una tarjeta de expansión ISA.
16. Diseñar un puerto de lectura/escritura sito en una tarjeta de expansión ISA.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

6 Microcontroladores

1. Introducción

En general, un microcontrolador se define como un pequeño sistema basado en microprocesador integrado en un chip. Su función básica es la de control de eventos y se utiliza profusamente en adquisición de datos y control de procesos industriales.

En su evolución, los microcontroladores y microprocesadores han seguido caminos muy distintos. Los microprocesadores, como UCP de propósito general, se han esforzado por conseguir más velocidad y más capacidad de proceso. Sin embargo, los microcontroladores, como sistemas de propósito específico, se han especializado en reducir coste, consumo, tamaño y tiempo de desarrollo.

En este capítulo se estudian los fundamentos de los microcontroladores y se particulariza para dos de ellos: El 8051 de Intel y el ADuC842 de Analog Devices. El clásico 8051 de Intel de los años 80 se utiliza todavía en la industria y es la base de muchos microcontroladores modernos. Tal es el caso del microconvertidor ADuC842 de Analog Devices que soporta el conjunto de instrucciones del 8051.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2. Fundamentos

2.1. Diagrama de bloques

En la figura 6.1 se puede ver un diagrama de bloques de un microcontrolador típico. El sistema incluye UCP, memoria (ROM y RAM), interfaces de E/S digitales (puertos paralelo, puertos serie) y analógicos (conversores A/D y D/A), timers, contadores de eventos, bus externo, etc.

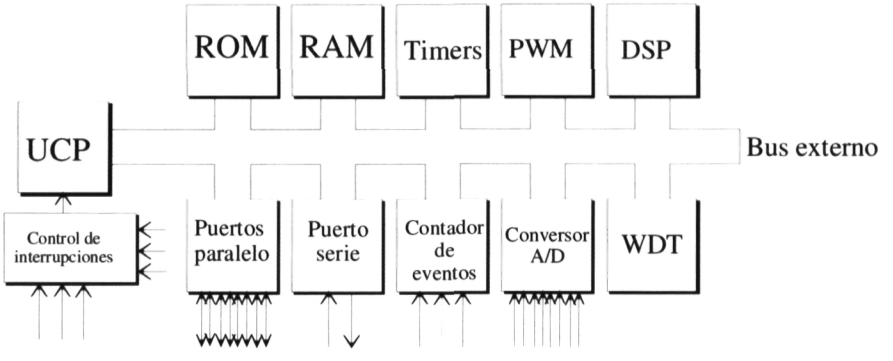


Figura 6.1: Diagrama de bloques de un microcontrolador

2.1.1. UCP

La arquitectura de la UCP puede ser muy variada (CISC, RISC, etc.), un espacio de direcciones o múltiples espacios de direcciones, basada en acumulador o en registros, etc.. Además el conjunto de instrucciones suele ser especializado para las aplicaciones en las que se recomienda el chip[284].

2.1.2. UCM

El código se suele ubicar en memoria de tipo ROM:

- ROM programada en fábrica. El cliente desarrolla su programa con un sistema de desarrollo con ROM externa o bien con un emulador. Cuando el programa está depurado se manda al fabricante y éste lo graba en la memoria ROM interna del microcontrolador.

[284] Hay que tener en cuenta que los microcontroladores son de propósito específico y no requieren la estricta compatibilidad de otros tipos de sistemas (como, por ejemplo, los PCs).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- PROM u OTP. El microcontrolador integra una PROM que puede ser grabada por el usuario (sólo una vez) una vez haya depurado su programa.
- EPROM . El microcontrolador integra una EPROM que le permite depurar el programa y salvar espacio. Es una solución de medio coste muy utilizada en prototipos y pequeños desarrollos.
- Flash. Es la memoria de lectura/escritura no volátil que se usa actualmente como memoria de código ya que permite reducir tiempo de diseño y de producción, actualizando las aplicaciones en segundos.
- sin ROM. Se hace uso de memoria externa, normalmente una EPROM. Es una solución económica para el desarrollo de un prototipo o para cuando no hay problemas de espacio.

La memoria RAM se suele usar para almacenar los variables del programa. Suele ser una pequeña memoria SRAM. Hoy en día también se hace uso de memorias EEPROM para aplicaciones de configuración, seguridad y diagnóstico.

2.1.3. UES

El interfaz digital incluye puertos paralelo y puertos serie. El puerto paralelo suele admitir direccionamiento de bit y de byte y facilita la conexión de múltiples periféricos (teclados, displays, CRT, impresoras, relés, motores, etc.). El puerto serie puede ser de muchos tipos: UART , USART, I2C, USB, etc.. Una sencilla UART es lo más común y permite implementar fácilmente una comunicación con otros equipos (por ejemplo, un PC).

Con el interfaz analógico se pueden adquirir señales del mundo real (tensiones, corrientes, fases, temperaturas, presiones, etc.) y se pueden generar señales analógicas. El microcontrolador suele incluir un multiplexador analógico, un circuito de muestreo y retención y el conversor A/D. También hay microcontroladores que disponen de conversores D/A aunque[285], en ocasiones, los sustituyen por moduladores de anchura de pulsos (PWM, Pulse Width Modulator). Los PWMs son generadores de ondas cuadradas con un ciclo de trabajo programable[286].

[285] El diseño de conversores D/A integrados es delicado ya que son muy sensibles al ruido digital y no admiten alta integración.

[286] Jugando con el ciclo de trabajo, podemos obtener una señal analógica filtrando (calculando su valor medio) la salida del PWM.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Muchos microcontroladores trabajan en ambientes industriales donde el nivel de ruido en la placa puede alterar el software. En estos casos, suelen integrar un watchdog timer (WDT). El WDT es un contador hardware que es reinicializado periódicamente por el programador. Si el programa se deteriora el WDT se desborda y reinicializa (reset) el sistema.

Los contadores también son un elemento importante en los microcontroladores. Se utilizan para contar eventos procedentes de circuitos externos y/o para contar tiempo.

Los microcontroladores disponen de un buen manejador de interrupciones tanto internas como externas que permite establecer máscaras, prioridades y diferentes configuraciones de activación(flanco o nivel).

2.2. Clasificación de microcontroladores

En las tablas 6.1 y 6.2 se puede ver una comparativa de diferentes familias de microcontroladores.

Familia	Memoria código (KB)	Memoria datos (bytes)	Puertos paralelo	Puertos serie	Timers	Interr.	Otros	Fabricante
COP800	1-8 (ROM)	64-256 (SRAM)	24-39	UART	1-3	3-14	PWM, A/D	National
MCS-51	4-16 (ROM)	128-256 (SRAM)	32-56	UART	2-3	6-19	DMA	Intel
ADuC812	8(Flash)	640 (Flash), 256 (SRAM)	32	UART, I2C, SPI,	3	9	ADC, DAC	Analog Devices
AVR	1-128 (Flash)	64-4096 (EEPROM), 128-4096 (SRAM)	8	SPI, UART, I2C	4	35	ADC	Atmel
68HC11	4-32 (OTP)	192-1024 (SRAM)	38	SCI,SPI	4	18	ADC, DCA	Motorola
PIC16C	0,768-14 (Flash)	64-256 (EEPROM),68-368 (SRAM)	33	USART, I2C,SPI, LIN, USB	3	4-12	ADC, DAC	Microchip
HPC16000	8-16 (ROM)	256-512 (SRAM)	52-56	UART	4	8	ADC, WDT	National
MCS-96	8 (ROM)	232(SRAM)	40	UART	4	8	ADC, WDT	Intel
80C166	32 (ROM)	1024 (SRAM)	76	USART	2	32	ADC	Siemens

Tabla 6.1: Comparativa de características funcionales de familias de microcontroladores

Los fabricantes han seguido varios caminos en su afán por reducir costes y aumentar la integración.

- Aumento de número de funciones. Algunos fabricantes aumentan el número de funciones integradas en el chip. Es el caso del 8051, el 8096 de Intel o del Z86C94 de Zilog que incorpora un DSP . En estos casos, suele haber

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Familia	Frecuencia CLK	Alimentación	Consumo	Bajo consumo	Empaquetado	Coste	Fabricante
ADuC812	16MHz	3/5	48mW(3V)	15 μW	52PQFP, 56 CSP	$6.85	Analog Devices
AVR	1-24MHz	1.8-5.5V	1-3mA	<1μA	8/20/28/40 DIP, 32/44/64 TQFP,...	0.5-$4	Atmel
68HC11	<5MHz	3/5	124mW (3MHz)	Wait,stop	DIP, CDIP,...	2-$15	Motorola
PIC16C	0-20MHz	2-6V	50mW	sleep, control individual	14-44DIP,..	1-$5	Microchip

Tabla 6.2: Comparativa de características eléctricas de familias de microcontroladores

muchas señales y funciones multiplexadas que no podrán funcionar de forma simultánea[287].

- Aumento del número de versiones de un microcontrolador, como por ejemplo la familia 68HC05 de Motorola que dispone de 70 variedades. El diseñador seleccionará el modelo que mejor se ajuste a sus requerimientos.

- Diseño ASIC (Application Specific Integrated Circuit). Es la solución más elegante, pero también la más cara: Diseñar nuestro microcontrolador a medida. El cliente diseña mediante herramientas CAD su propio microcontrolador basándose en un microcontrolador básico y añadiéndole lógica a su alrededor.

2.3. Criterios de selección de microcontroladores

El número de microcontroladores en el mercado es enorme, así que a la hora de elegir un modelo hay que seguir ciertos criterios para conseguir una buena elección. El objetivo general debe ser cumplir las especificaciones y ser barato, pero teniendo presente que se debe minimizar el coste del sistema. La solución ideal es encontrar un único chip que cumpla todas las especificaciones. Si no existe un chip que verifique todas estas exigencias habrá que buscar el que menos hardware extra necesite.

Los criterios a tener en cuenta pueden ser relativos a la funcionalidad, al desarrollo y/o a la fabricación.

Con respecto a la funcionalidad, habrá que tener en cuenta aspectos tales como: El número de puertos paralelos y serie que se requieren[288], el uso de

[287] A veces el fabricante, proporciona chips de soporte que permiten resolver el problema.

[288] Puertos paralelo para sensores, actuadores, interfaces a dispositivos, etc. y puertos serie para control remoto, monitorización, etc.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

conversores A/D y/o D/A para manejar señales analógicas, la cantidad de memoria necesaria[289], contadores de eventos, interrupciones, etc.

En cuanto al desarrollo hay muchas cuestiones importantes que afectan a su tiempo de ejecución[290]:

- La calidad y cantidad de herramientas de desarrollo disponibles: Ensambladores, compiladores, depuradores, emuladores, entorno de desarrollo integrado, placas de evaluación, etc..
- El soporte de la aplicación. Cuando el diseño no funciona o el chip no parece cumplir las especificaciones del fabricante, no es lo mismo contar con un vendedor sin conocimientos técnicos que con un grupo específico de apoyo.
- El lenguaje de programación. Hoy en día hay muchos microcontroladores que permite una programación eficiente en C con todas las ventajas que esto conlleva.

Si el producto se va a fabricar en grandes cantidades, hay que tener en cuenta aspectos relacionados con la fabricación:

- La solvencia y estabilidad del fabricante.
- La disponibilidad del chip. Hay ocasiones en las que, por usar una nueva tecnología, se diseña con muestras de nuevos microcontroladores que luego tardan en pasar a la fase de producción (a veces no llegan nunca); esto puede afectar al tiempo de puesta en el mercado y a la pérdida de la oportunidad de negocio.
- Las segundas fuentes. Si el sistema se va a fabricar en grandes cantidades durante mucho tiempo, es muy recomendable que el microcontrolador tenga más de un fabricante.

[289] El código puede ser simplemente un bucle que consulte un flag o puede incluir complejos algoritmos de procesado de señal.

[290] El tiempo de desarrollo es un factor crítico en muchos diseños, ya que no hay que olvidar que, en general, los productos resultantes se venden en mercados muy competitivos.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3. El 8051 de Intel

El 8051 de Intel es un microcontrolador de 8 bits optimizado para aplicaciones de control con un potente conjunto de instrucciones booleanas. Dispone de espacios separados de direcciones de código y datos, 32 líneas de E/S, 2 contadores/timer de 16 bits, una UART full duplex y 5 fuentes de interrupción.

3.1. Espacio de direcciones

Se pueden distinguir 3 espacios de direcciones de memoria (ver figura 6.2): Espacio de direcciones de memoria de programa de 64 Kbytes, espacio de direcciones de memoria externa de datos de 64 Kbytes y espacio de direcciones de memoria interna de datos de 128 bytes. Estos 3 espacios corresponden a 3 memorias físicamente distintas y se direccionan con diferentes instrucciones.

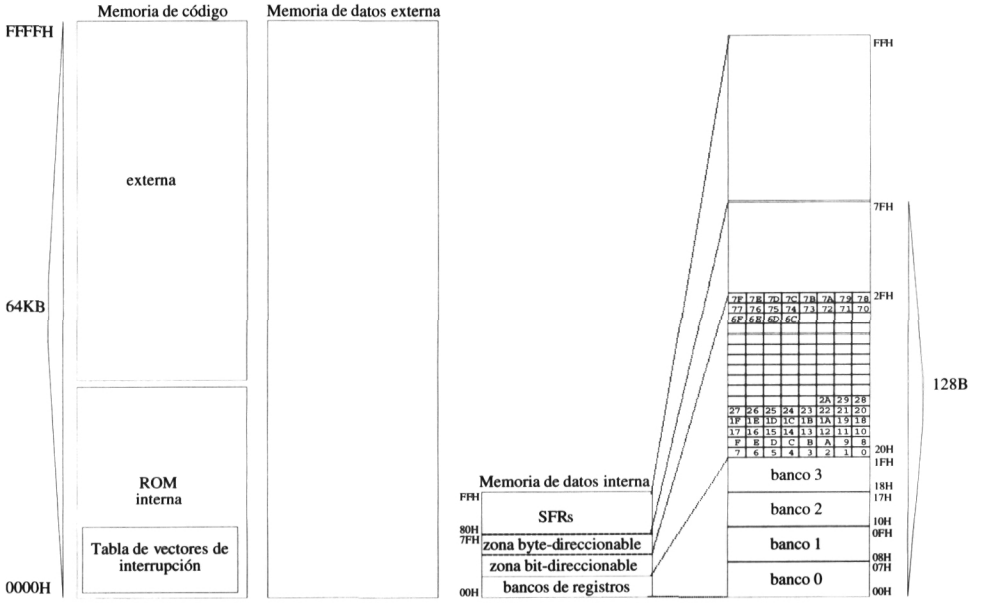


Figura 6.2: Espacios de direcciones del microcontrolador 8051

En el espacio de direcciones de memoria de código hay 2 opciones: a) Toda la memoria es externa[291] o b) los primeros Kbytes de memoria son internos (ROM interna) y el resto externos. El tamaño de la ROM interna varía según

[291] Para que toda la memoria sea externa el pin EA debe estar a 0.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

el modelo[292]. La tabla de vectores de interrupción se ubica en las direcciones bajas de la memoria de código (ver tabla 6.3).

Fuente de interrupción	bit	Dirección
Interrupción externa 0	IE0	0003H
Timer 0	TF0	000BH
Interrupción externa 1	IE1	0013H
Timer 1	TF1	001BH
Puerto serie	RI,TI	0023H

Tabla 6.3: Interrupciones del 8051

Es interesante destacar que en las direcciones de la tabla de vectores de interrupción no se ubican las direcciones de la rutina de interrupción sino instrucciones de salto a la rutina[293].

El espacio de direcciones de memoria de datos interna se divide en dos: a) Espacio disponible para el usuario de 128 bytes y b) Registros de función especial (SFRs, Special Function Registers). La pila reside en este espacio y crece hacia direcciones más altas. En el espacio de memoria de datos interna disponible para el usuario se definen tres zonas (ver figura 6.3):

- Bancos de registros: El 8051 tiene 4 bancos de 8 registros cada uno[294]. Los registros son de propósito general.
- Registros bit-direccionables. El 8051 tiene 16 registros bit-direccionables de propósito general[295]. Por tanto, se pueden direccionar 128 bits.
- Registros byte-direccionables.

Después de un reset, el registro de pila (SP, Stack Pointer) contiene 07H. Así que el primer byte llevado a la pila se almacena en la dirección 08H.

[292] 8031(sin ROM interna), 8051(4KB), 83C51FA(8KB), 83C51FB(16KB), etc..

[293] La razón de esto es que en algunas ocasiones la rutina de interrupción es tan pequeña que la instrucción de salto puede ser sustituida por la instrucción de la rutina misma.

[294] El banco i está formado por los registros Ri[7:0], $i = (0, 1, 2, 3)$. Sólo hay un banco activo; en el registro de estado (PSW) se selecciona cuál es el banco que se está usando.

[295] Sus direcciones van desde la 20H hasta la 2FH. Ésta es la dirección de byte. Cada uno de los 8 bits de cada byte se puede direccionar individualmente. Por ejemplo, el byte 20H contiene las direcciones de bit: 0H-7H, el byte 21H: 8H-FH, y así sucesivamente hasta el byte 2FH que contiene las direcciones de bit 78H-7FH.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

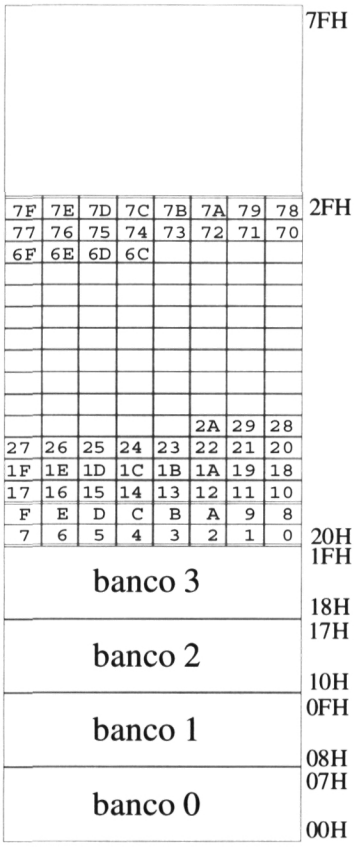


Figura 6.3: Primeros 128 bytes de la memoria interna de datos

3.2. Registros de función especial

Los registros de función especial permiten la programación del microcontrolador (tabla 6.4). La figura 6.4 muestra un diagrama de bloques ilustrativo de su arquitectura.

Para leer o escribir en los 4 puertos de E/S del 8051, se usan los registros de lectura/escritura P0, P1, P2 y P3. El 8051 dispone de acumulador (ACC), registro de estado (PSW), registro de puntero de pila (SP), registro auxiliar B[296] y registro de puntero de datos de 16 bits (DPTR). Este último consta de dos bytes (DPH y DPL) y se utiliza para direccionar memoria de datos externa.

[296]El registro B es una extensión del ACC para operaciones de multiplicación y división.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Tipo	Dirección	Descripción	símbolo
Puntero datos	83H	Byte alto del puntero de datos DPTR de 16 bits	DPH
Puntero datos	82H	Byte alto del puntero de datos DPTR de 16 bits	DPL
Puntero pila	81H	Puntero de pila	SP
Acumulador	E0H	Acumulador	ACC
Multiplicación	F0H	Se usa en operaciones de multiplicación y división	B
Estado	D0H	Registro de estado	PSW
Puerto 0 de E/S	80H	Puerto E/S bidireccional de 8 bits direccionables individualmente	P0[7:0]
Puerto 1 de E/S	90H	Puerto E/S bidireccional de 8 bits direccionables individualmente	P1[7:0]
Puerto 2 de E/S	A0H	Puerto E/S bidireccional de 8 bits direccionables individualmente	P2[7:0]
Puertos 3 de E/S	B0H	Puerto E/S bidireccional de 8 bits direccionables individualmente	P3[7:0]
Contador 0	8AH	byte bajo del timer 0	TL0
Contador 0	*8CH*	*byte alto del timer 0*	*TH0*
Contador 1	8BH	byte bajo del timer 1	TL1
Contador 1	8DH	byte alto del timer 1	TH1
Modo Contadores	89H	Registro de modo de contadores/timers	TMOD
Control Contadores	88H	Registro de control de contadores/timers	TCON
Control Puerto serie	98H	Registro de control de puerto serie	SCON
Puerto serie	99H	Buffer del puerto serie	SBUF
Gestión Interrupciones	B8H	Registro de prioridad de interrupciones	IP
Gestión Interrupciones	A8H	Registro de habilitación de interrupciones	IE
Alimentación	87H	Registro de control de alimentación	PCON

Tabla 6.4: Registros de función especial del 8051

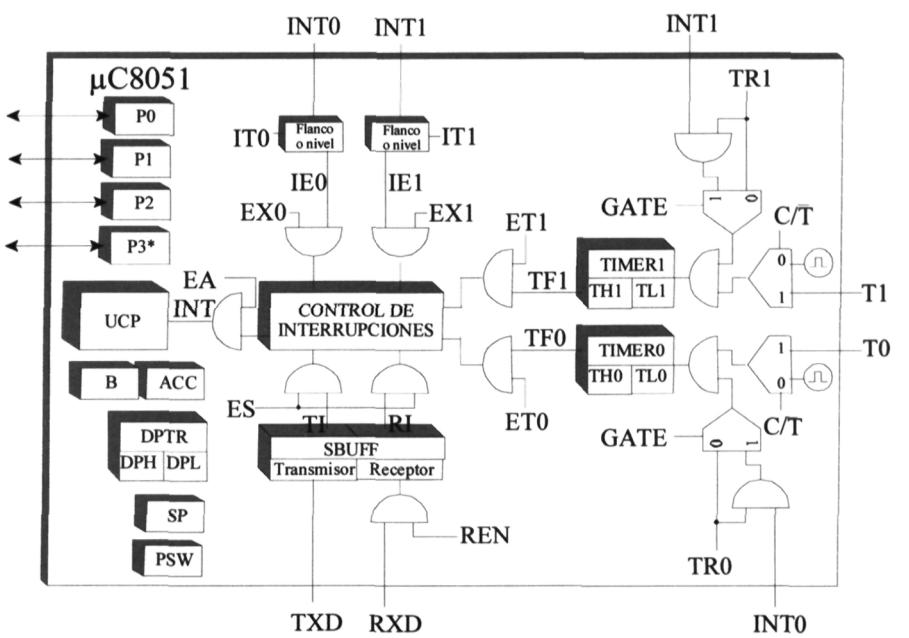


Figura 6.4: Registros de función especial del 8051

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Hay dos contadores, TIMER1 y TIMER0, de 16 bits. Cada contador tiene dos registros: THi y TLi. THi es el byte alto del contador/timer i y TLi el byte bajo. El contador i puede contar pulsos de tiempo (bit C/$\bar{T}$ = 0) o pulsos de eventos (bit C/$\bar{T}$ = 1) que entran por el pin Ti. Estos pulsos se habilitan con una puerta AND. La habilitación puede ser software (bit GATE = 0) o hardware (bit GATE = 1). Para habilitar por software, el bit TRi tiene que estar a 1. En la habilitación hardware, el pin INTi tiene que estar a 1 (TRi también tiene que valer 1). Cuando el contador se desborda (pasa del valor máximo a 0) activa el bit TFi. La activación del bit TFi puede provocar una interrupción si el bit ETi de habilitación de interrupción está a 1. Todos los bits de control mencionados se ubican en los registros TMOD y TCON que se describen en el apartado 3.2.1.

El puerto serie tiene como interfaz los pines `TXD` y `RXD` (transmisión y recepción, respectivamente). La recepción se puede deshabilitar desactivando el bit REN (REN=0). Para transmitir (recibir) un dato, éste se escribe en (lee de) el registro SBUF[297]. Cuando el dato ha sido transmitido(recibido) se activa el bit TI(RI). La activación de TI ó RI puede provocar una interrupción si el bit ES de habilitación de interrupción está activado (ES=1). Los bits de control mencionados se ubican en el registros SCON (apartado 3.2.2).

El 8051 dispone de 2 señales de interrupción externas: `INT0` e `INT1`. La interrupción se puede provocar por flanco (bit ITi = 1) o por nivel(bit ITi = 0) y provoca una interrupción si el bit EXi está a 1. Finalmente, para que la UCP del 8051 atienda cualquier interrupción el bit EA tiene que estar a 1. Los bits de control mencionados se ubican en los registros IE e TCON.

La tabla 6.5 muestra el registro de estado del 8051. Los bits RS[1:0] indican cuál es el banco de registros activo en cada momento[298]. El bit P marca paridad par de '1s' en el acumulador y se usa para implementar el bit de paridad para la transmisión asíncrona serie.

bit	Descripción	símbolo
B7	Flag de acarreo	CY
B6	Flag auxiliar de acarreo	AC
B5	Flag de estado de propósito general	F0
B4	Bit 1 de selección de banco de registros	RS1
B3	Bit 0 de selección de banco de registros	RS0
B2	Flag de overflow	OV
B1	Definible por el usuario	
B0	Paridad del acumulador	P

Tabla 6.5: Registro de estado del 8051

[297] En realidad, SBUF son dos registros: Un registro de sólo escritura (transmisión) y otro de sólo lectura (recepción).

[298] Si RS[1:0] = i, el banco seleccionado es el i ($i = 0, 1, 2, 3$).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.2.1. Registros de control de timers

Los dos contadores/timers de 16 bits del 8051 tienen 4 modos de trabajo (tabla 6.6). El modo 0 es un contador de 13 bits que facilita la compatibilidad con el microcontrolador 8048. El modo 1 es un contador de 16 bits. El modo 2 es un contador de 8 bits (el byte bajo TL) con autorrecarga; cuando el contador TL se desborda, se autorrecarga con el valor que previamente se ha almacenado en el byte alto TH. El modo 3 es un modo especial que mantiene dos contadores de 8 bits.

M1	M0	Descripción
0	0	Temporizador de 13 bits
0	1	Temporizador/contador de 16 bits (TL+TH)
1	0	Temporizador/Contador de 8 bits (TL)con autorrecarga (TH)
1	1	2 temporizadores/Contadores de 8-bits (TL0 y TH0) controlados por bits de control de timer0 y timer1.

Tabla 6.6: Modos de timers del 8051

Los contadores/timers se configuran y controlan mediante los registros TMOD (tabla 6.7) y TCON (tabla 6.8). La funcionalidad de cada bit ya ha sido descrita sobre la figura 6.4.

bit	Descripción	símbolo
B7	Control hardware(1)/software(0) del timer 1	GATE
B6	Selección de temporizador(0)/contador(1) del timer 1	T/C
B5	Modos de operación del timer 1	M1
B4	Modos de operación del timer 1	M0
B3	Control hardware(1)/software(0) del timer 0	GATE
B2	Selección de temporizador(0)/contador(1) del timer 0	T/C
B1	Modos de operación del timer 0	M1
B0	Modos de operación del timer 0	M0

Tabla 6.7: Registro TMOD del 8051

bit	Descripción	símbolo
B7	Flag de overflow del timer 1	TF1
B6	Habilitador de timer 1	TR1
B5	Flag de overflow del timer 0	TF0
B4	Habilitador de timer 0	TR0
B3	Flag de flanco de interrupción 1 detectado	IE1
B2	Control de tipo de interrupción externa 1: flanco(0) o nivel bajo (1)	IT1
B1	Flag de de flanco de interrupción 0 detectado	IE0
B0	Control de tipo interrupción externa 0: flanco(0) o nivel bajo (1)	IT0

Tabla 6.8: Registro TCON del 8051

Cuando los contadores se configuran como timers se incrementan cada 12 pulsos de reloj.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MALAGA

3.2.2. Registros del puerto serie

El 8051 dispone de una UART básica que usa el registro SBUF para transmitir y recibir los datos. El puerto serie tiene 4 modos de trabajo que se describen en la tabla 6.9.

SM0	SM1	Descripción
0	0	Registro de desplazamiento de frecuencia la *frecuencia de reloj)/12.*
0	1	UART de 8 bits de velocidad variable. El reloj de transmisión se implementa haciendo uso del TIMER1 en modo 2 (timer de 8 bits con autorrecarga).
1	0	UART de 9 bits de frecuencia la *frecuencia de reloj/64* ó la *frecuencia de reloj)/32.*
1	1	UART de 9 bits de velocidad variable. El noveno bit (TB8 y RB8) se suele utilizar para implementar sistemas multipunto cuando SM2=1

Tabla 6.9: Modos del puerto serie del 8051

En un sistema multipunto un 8051 actúa como maestro y n 8051s como esclavos. Cada esclavo tiene una dirección asignada. Todos los esclavos están en modo multipunto (SM2=1) a la escucha del maestro. Cuando el maestro se quiere comunicar con el esclavo i envía por el puerto serie un byte con la dirección i y el bit TB8=1. Sólo el esclavo i responde al maestro y se pone en modo datos. A continuación, el maestro envía bytes de información con el bit TB8 = 0 que son sólo recibidos por el esclavo i.

La ecuación que determina la velocidad de transmisión es la siguiente[299]:

$$Vbaudios = \frac{2^{SMOD}}{32} \times \frac{f_{CLK} \times 10^6}{(256 - TH1) \times 12} \tag{6.1}$$

SMOD es el bit 7 del registro de control de alimentación. La configuración de la UART se hace programando el registro SCON (tabla 6.10).

bit	Descripción	símbolo
B7	Modos de operación	SM0
B6	Modos de operación	SM1
B5	Modos de operación. Multipunto	SM2
B4	Habilitación de recepción	REN
B3	Noveno bit de transmisión para modos 2 y 3	TB8
B2	Noveno bit de recepción para modos 2 y 3	RB8
B1	Flag de interrupción de transmisión	TI
B0	Flag de interrupción de recepción	RI

Tabla 6.10: Registro SCON del 8051

[299]Para conseguir frecuencias baudibles (1200, 2400, 4800, etc.) se usa un reloj de 11.059MHz.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.2.3. Registros de gestión de interrupciones

El 8051 maneja 5 fuentes de interrupción: dos interrupciones externas (IE0 e IE1), dos interrupciones de timers (TF0 y TF1) y una interrupción del puerto serie (RI&TI). Para la gestión de interrupciones se hace uso de los registros IE (tabla 6.12) e IP (tabla 6.11).

bit	Descripción	símbolo
B7	Reservado	
B6	Reservado	
B5	Reservado	
B4	Prioridad alta(1)/baja(0) de la interrupción del puerto serie	PS
B3	Prioridad alta(1)/baja(0) de la interrupción del timer 1	PT1
B2	Prioridad alta(1)/baja(0) de la interrupción externa 1	PX1
B1	Prioridad alta(1)/baja(0) de la interrupción del timer 0	PT0
B0	Prioridad alta(1)/baja(0) de la interrupción externa 0	PX0

Tabla 6.11: Registro IP del 8051

bit	Descripción	símbolo
B7	Deshabilitador (0) de todas las interrupciones	EA
B6	Reservado	
B5	Reservado	
B4	Habilitador (1) de interrupción del puerto serie	ES
B3	Habilitador (1) de interrupción del timer 1	ET1
B2	Habilitador (1) de interrupción externa 1	EX1
B1	Habilitador (1) de interrupción del timer 0	ET0
B0	Habilitador (1) de interrupción externa 0	EX0

Tabla 6.12: Registro IE del 8051

Se establece el siguiente orden de prioridad (de más prioritario a menos prioritario) para el caso de que dos o más interrupciones simultáneas tengan en el registro IP el mismo nivel de prioridad:

$$IE0$$
$$TF0$$
$$IE1$$
$$TF1$$
$$RI\&TI$$

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.2.4. Registro de control de alimentación

El registro de control de alimentación (tabla 6.13) permite programar dos modos de bajo consumo del microcontrolador[300]: Power Down y Idle.

bit	Descripción	símbolo
B7	Duplica la tasa de bit	SMOD
B6	Reservado	
B5	Reservado	
B4	Reservado	
B3	Flag de propósito general	GF1
B2	Flag de propósito general	GF0
B1	PowerDown	PD
B0	Modo desocupado	IDL

Tabla 6.13: Registro PCON del 8051

En modo PowerDown (bit PD=1), el microcontrolador consume el 10 % de su consumo habitual. No funciona el reloj pero se mantienen los datos de memoria SRAM interna. Para salir de este estado hay que reinicializar el microcontrolador. El consumo típico es de 50 μA.

En modo Idle, el reloj funciona y suministra señal para interrupciones, puerto serie y timers, pero no a la UCP. El estado de la UCP se preserva. Se sale por interrupción. El consumo típico es de 5mA.

3.3. Modos de direccionamiento

El 8051 tiene 5 modos de direccionamiento : directo, indirecto, a registro, inmediato e indexado. En la tabla 6.14 se muestran algunos ejemplos de cada uno de ellos:

Direccionamiento	Instrucción	Descripción	Comentarios
Directo	ADD A,7FH	A←(7FH)+(A)	7FH es una dirección de la RAM interna
Indirecto	ADD A,@R0	A←((R0))+(A)	R0 es un registro de la RAM interna
Indirecto	MOVX A,@DPTR	A←((DPTR))	(DPTR) es una dirección de la RAM externa
Registro	ADD A,R7	A←(R7)+(A)	R7 es un registro de la RAM interna
Inmediato	MOV A,#100	A←100	
Indexado	MOVC A,@A+PC	PC←(PC)+1; A←((A)+(PC))	PC es el contador de programa. (A)+(PC) es una dirección de la ROM

Tabla 6.14: Ejemplos de modos de direccionamiento en el 8051

[300]El consumo habitual de un 8051 HCMOS es de unos 20mA.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.4. Conjunto de instrucciones

El 8051 tiene 111 instrucciones que se pueden clasificar en: transferencia de datos, aritméticas, lógicas, booleanas, saltos y misceláneas[301]. A continuación, se describen con más detalle usando la siguiente nomenclatura:

(X) = contenido de x; C= bit de acarreo; bit= dirección de bit; dir= dirección de byte; addr11: dirección de 11 bits; addr16: dirección de 16 bits. re= desplazamiento de 8 bits; data= dato.

3.4.1. Instrucciones de transferencia de datos

Hay instrucciones de transferencia para acceder a cualquiera de los tres espacios de direcciones: MOV para la RAM interna, MOVX para la RAM externa y MOVC para la memoria ROM (tabla 6.15).

Mnemónico	Descripción
MOV A,Rn	A←(Rn)
MOV A,@Ri	A←((Rn))
MOV A,dir	A←(dir)
MOV A,#data	A←data
MOV Rn,A	Rn← (A)
MOV Rn,dir	Rn←(dir)
MOV Rn,#data	Rn←data
MOV @Ri,A	(Rn)← (A)
MOV @Ri,#data	(Rn)←data
MOV @Ri,dir	(Ri)←(dir)
MOV dir,A	dir←(A)
MOV dir,Rn	dir←(Rn)
MOV dir,@Ri	dir←((Ri))
MOV dir,dir	dir←(dir)
MOV dir,#data	dir←data
MOV DPTR,#data	DPTR←data
MOVC A,@A+DPTR	A←((A)+(DPTR)); acceso a memoria de código
MOVC A,@A+PC	PC←PC+1; A←((A)+(PC)) acceso a memoria de código
MOVX A,@Ri	A←((Ri)); acceso a memoria externa de datos
MOVX A,@DPTR	A←((DPTR)); acceso a memoria externa de datos
MOVX @Ri,A	(Ri)←(A); acceso a memoria externa de datos
MOVX @DPTR,A	(DPTR)←(A); acceso a memoria externa de datos
PUSH dir	introduce (dir) en la pila
POP dir	Saca (dir) de la pila
XCH A,Rn	Intercambia (A) y (Rn)
XCH A,@Ri	Intercambia (A) y ((Ri))
XCH A,dir	Intercambia (A) y (dir)
XCHD A,@Ri	Intercambia (A) y nibble de ((Ri))

Tabla 6.15: Instrucciones de transferencia de datos del 8051 de Intel

3.4.2. Instrucciones aritméticas

Las instrucciones aritméticas incluyen operaciones de suma, resta, incremento, decremento, multiplicación, división y ajuste decimal (tabla 6.16).

[301] Son instrucciones que no encajan en ninguna de las categorías anteriores como NOP (no operación).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Mnemónico	Descripción
ADD A,Rn	A←(Rn)+(A)
ADD A,@Ri	A ←((Rn))+(A)
ADD A,dir	A←(dir)+(A)
ADD A,#data	A←data+(A)
ADDC A,Rn	A←(Rn)+(A) con acarreo
ADDC A,@Ri	A←((Rn))+(A) con acarreo
ADDC A,dir	A←(dir)+(A) con acarreo
ADDC A,#data	A←data+(A) con acarreo
SUBB A,Rn	A←(A)-(Rn) con borrow
SUBB A,@Ri	A←(A)-((Rn)con borrow
SUBB A,dir	A←(A)-(dir)con borrow
SUBB A,#data	A←(A)-data con borrow
INC A	A←(A)+1
INC Rn	Rn←(Rn)+1
INC @Ri	(Rn)←((Rn))+1
INC dir	dir←(dir)+1
INC DPTR	DPTR←(DPTR)+1
DEC A	A←(A)-1
DEC Rn	Rn←(Rn)-1
DEC @Ri	(Rn)←((Rn))-1
DEC dir	dir←(dir)-1
MUL AB	B,A←A*B
DIV AB	A/B, A←cociente, B←resto
DA A	Ajuste decimal

Tabla 6.16: Instrucciones aritméticas del 8051 de Intel

3.4.3. Instrucciones booleanas

Las instrucciones booleanas son específicas para operaciones con bits. Incluye instrucciones de transferencia, de bit, puesta a 1 (SET) y a 0 (CLR), AND, OR y NOT (tabla 6.17).

Mnemónico	Descripción
CLR C	Pone C a 0
CLR bit	Pone (bit) a 0
SETB C	Pone C a 1
SETB bit	Pone (bit) a 1
CPL C	Complementa C
CPL bit	Complementa (bit)
ANL C,bit	C← C AND (bit)
ANL C,bit#	complementa (bit); C← C AND (bit)
ORL C,bit	C← C OR (bit)
ORL C,bit#	complementa (bit); C← C OR (bit)
MOV C,bit	C←(bit)
MOV bit,C	bit←(C)

Tabla 6.17: Instrucciones booleanas del 8051 de Intel

3.4.4. Instrucciones lógicas

Las instrucciones lógicas incluyen operaciones AND, OR, XOR, NOT, rotación e intercambio de nibbles (tabla 6.18).

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Mnemónico	Descripción
ANL A,Rn	A←(A) AND (Rn)
ANL A,@Ri	A←(A) AND ((Rn))
ANL A,dir	A←(A) AND (dir)
ANL A,#data	A←(A) AND data
ANL dir,A	dir← (dir) AND (A)
ANL dir,#data	dir ← (A) AND data
ORL A,Rn	A←(A) OR (Rn)
ORL A,@Ri	A←(A) OR ((Rn))
ORL A,dir	A←(A) OR (dir)
ORL A,#data	A←(A) OR data
ORL dir,A	dir← (dir) OR (A)
ORL dir,#data	dir ← (A) OR data
XRL A,Rn	A←(A) XOR (Rn)
XRL A,@Ri	A←(A) XOR ((Rn))
XRL A,#data	A←(A) XOR data
XRL A,dir	A←(A) XOR (dir)
XRL dir,A	dir← (dir) XOR (A)
XRL dir,#data	dir ← (A) XOR data
CLR A	A = 0
CPL A	Complementa (A)
RL A	Rota A a la izquierda
RLC A	Rota A a la izquierda a través del bit de acarreo
RR A	Rota A a la derecha
RRC A	Rota a la derecha a través del bit de acarreo
SWAP A	Intercambia nibbles of A

Tabla 6.18: Instrucciones lógicas del 8051 de Intel

3.4.5. Instrucciones de salto

Las instrucciones de salto incluyen saltos condicionales e incondicionales. Hay instrucciones de salto booleanas (6.19).

Mnemónico	Descripción
LJMP addrr16	Salta a addr16
AJMP addr11	Salta a addr11
SJMP rel	Salta corto a rel
JMP @A+DPTR	Salta a (DPTR) + (A)
ACALL addr11	Salta a subrutina en addr11
LCALL addr16	Salta a subrutina en addr11
RET	Retorna de subrutina
RETI	Retorna de interrupción
JZ rel	Salta a rel si (A) es 0
JNZ rel	Salta a rel si (A) es distinto de 0
JC rel	Salta a rel si (C) es 1
JNC rel	Salta a rel si (C) es 0
JB bit,rel	Salta a rel si (bit) es 1
JNB bit,rel	Salta a rel si (bit) es 0
JBC bit,rel	Salta a rel si (bit) es 1; Pone (bit) a 0
CJNE A,dir,rel	Compara (A) con (dir) y si distinto salta a rel
CJNE A,#data,rel	Compara (A) y data y si distinto salta a rel
CJNE Rn,#data,rel	Compara (Rn) y data y si distinto salta a rel
CJNE @Ri,#data,rel	Compara ((Ri)) y data y si distinto salta a rel
DJNZ dir,rel	Decrementa (dir); Salta a el si (dir) es distinto de 0
DJNZ Rn,rel	Decrementa Rn; Salta a rel si Rn es 0

Tabla 6.19: Instrucciones de salto del 8051 de Intel

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.4.6. Ejemplos

A continuación, se explican algunas breves secuencias de código que muestran las prestaciones de algunas de las instrucciones más especializadas del 8051: Operaciones BCD, manipulación de bits y acceso a memoria externa.

3.4.6.1. Operaciones BCD El 8051 puede llevar a cabo operaciones BCD y de multiplicación y división. El código que se muestra a continuación convierte un número binario menor de 100 en BCD:

```
MOV A,#58 ;Carga el valor 58 en el acumulador

MOV B,#10 ;Carga el valor 10 en el registro auxiliar B

DIV A,B   ;Divide (A) entre (B) y deja el cociente
          ;en A y el resto en B, resultando (A)=5 y (B)=8

SWAP A    ;Intercambia los nibbles alto y bajo del acumulador:
          ; (A)=00000101 -> (A)=01010000

ADD A,B   ; Suma (A) y (B) y deja el resultado en A
          ; (A) = 01011000 que es 58 en BCD
```

3.4.6.2. Operaciones booleanas El 8051 dispone de instrucciones booleanas para operar con bits. En este caso, el acumulador booleano es el bit de acarreo C. Para implementar la función XOR se puede usar el siguiente código:

```
      MOV C,bit1     ;Carga el acumulador booleano C con el valor bi

      JNB bit2,SALTO ;Salta a SALTO si bit2=0

      CPL C          ;Complementa el contenido de C

SALTO: siguiente instruccion
```

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.4.6.3. Operaciones de transferencia de datos Los tres espacios de direcciones requieren 3 tipos de instrucciones de transferencia de datos distintas. Para acceder a la memoria de datos externa se usa `MOVX`. Si la dirección de acceso es de 8 bits se utiliza un registro del banco de registros para contenerla. Si la dirección es de 16 bits se usa el puntero de datos `DPTR`. Supongamos un 8051 con RAM externa donde la dirección `34H` de la RAM externa contiene el valor `56H`. La siguiente secuencia de código copiaría el valor `56H` en el acumulador y en la dirección `12H` de la memoria externa.

```
MOV R0,#12H;Carga 12H en el registro R0

MOV R1,#34H;Carga 34H en el registro R1

MOVX A,@R1 ;Carga el acumulador con el contenido de la dirección
           ;de memoria externa contenida en R1.

MOVX @R0,A ;Carga en la dirección de memoria externa contenida en
           ;R0 el valor del acumulador
```

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.5. Señales de interfaz

En la figura 6.5 y tabla 6.20 se muestran las señales del microcontrolador 8051.

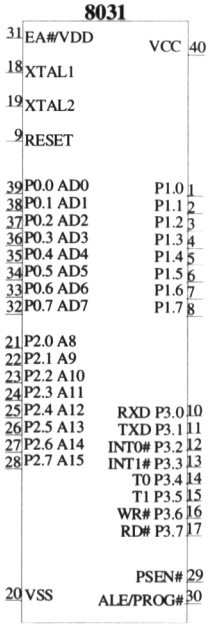


Figura 6.5: Señales del microcontrolador 8051

nombre	dirección	descripción
XTAL[1:2]	entrada	Oscilador
RESET	entrada	Reset del microcontrolador
ALE	salida	Habilitación de latch de direcciones. Sirve para "latchear" el byte bajo de las direcciones durante el acceso a memoria externa
PSEN	salida	Habilitación de lectura para programa en memoria externa
EA#/VDD	entrada	Habilitación de acceso a código externo
P0[7:0]	bidireccional	Puerto de E/S de propósito general. También es el bus de datos/direcciones bajas cuando se accede a memoria externa
P1[7:0]	bidireccional	Puerto de E/S de propósito general. También recibe los bytes de dirección baja durante la programación y verificación de la EPROM interna
P2[7:0]	bidireccional	Puerto de E/S de propósito general. También es el bus de direcciones altas para acceso a memoria externa y para la programación y verificación de la EPROM
P3[7:0]	bidireccional	Puerto de E/S de propósito general. Cada una de sus líneas tiene otra funcionalidad (ver tabla 6.21)

Tabla 6.20: Señales de microcontrolador 8051

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Pin	Puerto	Nombre	Descripción
10	P3[0]	RXD	Puerto de entrada serie
11	P3[1]	TXD	Puerto de salida serie
12	P3[2]	INT0#	Interrupción externa 0
13	P3[3]	INT1#	Interrupción externa 1
14	P3[4]	T0	Contador de eventos 0
15	P3[5]	T1	Contador de eventos 1
16	P3[6]	WR#	Señal de escritura de datos de memoria externa
17	P3[7]	RD#	Señal de lectura de datos de memoria externa

Tabla 6.21: Puerto P3[7:0] del 8051

3.6. Ciclos de bus

El 8051 dispone de interfaz de bus externo para acceder a la memoria de código y de datos. La frecuencia de reloj del 8051 básico puede variar desde 3.5 a 12 MHz.

La figura 6.6 muestra un ciclo de lectura de la memoria de código.

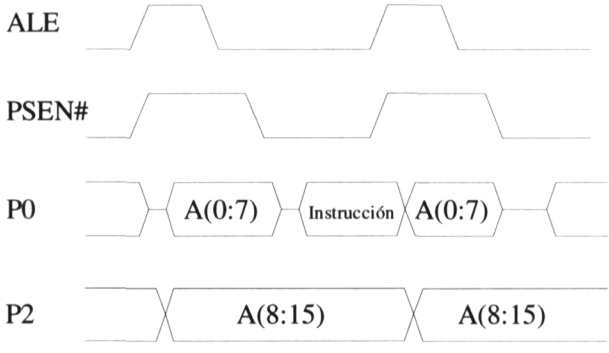


Figura 6.6: Ciclo de lectura de memoria de código externa de un 8051

El bus de datos y de direcciones bajas están multiplexados. Inicialmente se generan las direcciones: `A[0:7]` en el puerto `P0` y `A[8:15]` en el puerto `P2`. Con el flanco de bajada de la señal `ALE`, las direcciones bajas `A[0:7]` son capturadas por un registro (un típico 373) y el puerto `P0` se convierte en bus de datos (en este caso, de instrucciones).

Las figuras 6.7 y 6.8 muestran ciclos de lectura y escritura, respectivamente, de la memoria de datos. Para direcciones de 16 bits se usa el puntero de datos DPTR (el contenido de DPL sale por `P0` y el de DPH por `P2`). Para direcciones de 8 bits se puede usar un registro del banco de registros. En este ultimo caso, el puerto `P2` actúa como un puerto de E/S de propósito general.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

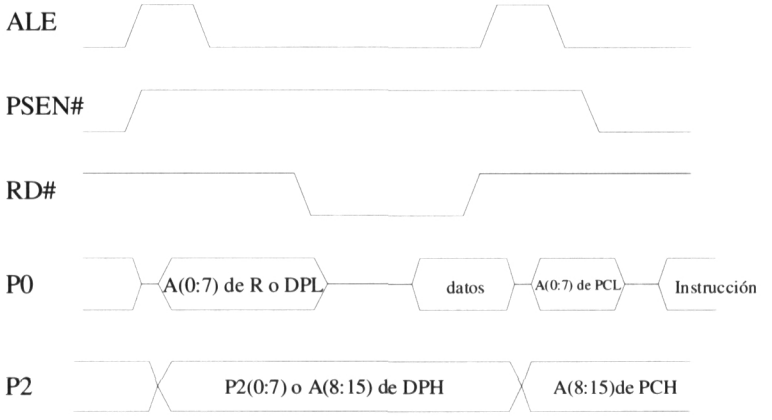


Figura 6.7: Ciclo de lectura de memoria de datos externa de un 8051

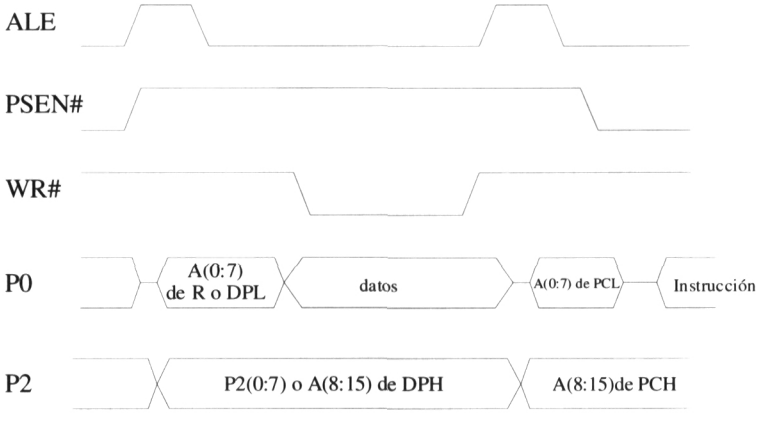


Figura 6.8: Ciclo de escritura de memoria de datos externa de un 8051

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.7. Ejemplos

Se muestran 4 sencillos ejemplos de programación del 8051 de Intel: Programación de un contador de eventos, control de un depósito de agua basado en una tabla de look-up e implementado tanto en ensamblador como en C, programación del puerto serie y manejo básico del entorno de desarrollo integrado μVision2 de Keil Software.

3.7.1. Programación de contador de eventos

Se pretende utilizar un 8051 como contador de pulsos externos. Se quiere que el flanco descendente de cada pulso incremente el contador y que cada 16 pulsos se genere una interrupción.

Para ello se programará el contador en modo 8 bits con autorrecarga (TLx es el contador y THx contiene el valor de autorrecarga) para que salte la interrupción cuando el contador pase de `FFH` a `00H` y se recargue de nuevo con un valor que le permita contar otra vez 16 pulsos. El valor de autorrecarga será `256-16` = `240` = `F0H`.

El 8051 proporciona dos contadores de eventos: TIMER0 y TIMER 1. Escogeremos el TIMER1 y lo programaremos configurando los registros TH1, TMOD y TCON. Para el caso de control software sus valores serán;

```
TH1 = F0H
TMOD = 0110|XXXXH = 60H
TCON = X1XX|XXXXH = 40H
IE   = 1XXX|1XXXH = 88H
```

y si el control es hardware (el pin `INT1` habilita el contador) el valor de TMOD será:

```
TMOD = 1110|XXXXH = E0H
```

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.7.2. Control de un depósito de agua

En la figura se muestra un sistema[302] de control de un depósito de agua basado en un microcontrolador 8051. El sistema consta del microcontrolador, tres sondas detectoras de niveles A, B y C y dos bombas (B1 y B2) (ver figura 6.9).

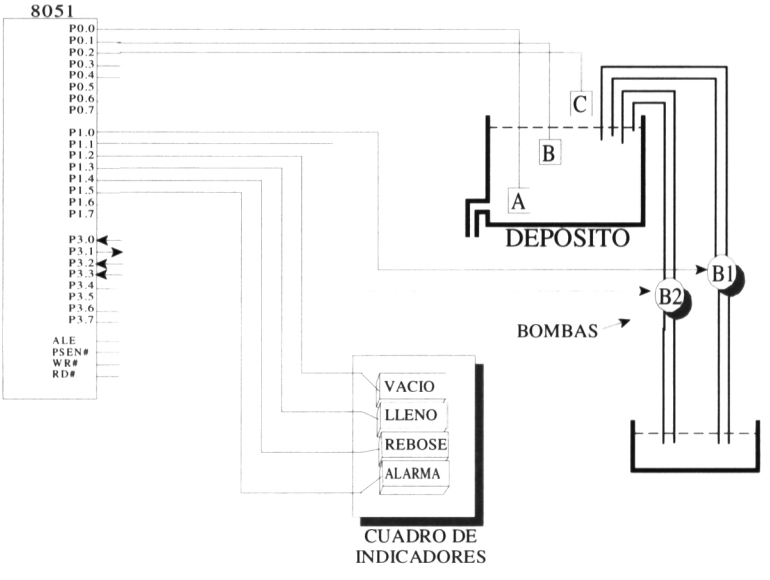


Figura 6.9: Ejemplo de diseño con el microcontrolador 8051

Las especificaciones del sistema son las siguientes:

1. La sonda A señala el mínimo nivel de agua. Por debajo de este nivel, el depósito se considera vacío.
2. La sonda B señala en nivel óptimo y cuando se alcanza se considera el depósito lleno.
3. La sonda C marca el nivel peligroso e indica que el agua está rebosando.
4. Cuando el depósito está vacío se activan las sondas B1 y B2.
5. Cuando el nivel de agua alcanza la sonda B se desactiva la bomba B2, quedando la B1 para mantenimiento del depósito.

[302]Este ejemplo se ha extraído de *Introducción a los microcontroladores* de González Vázquez, J.A., McGraw Hill, 1992.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

6. Si el nivel de agua alcanza la sonda C se desactiva la bomba B1.
7. Si se produce un malfuncionamiento en las sondas de entrada se activa una alarma.

Las sondas son leídas por el microcontrolador a través del puerto 0 y las bombas e indicadores se activan usando el puerto 1. En la figura 6.9 se muestran cuáles son las líneas de E/S utilizadas para cada puerto. En la tabla 6.22 se desarrolla la tabla de verdad correspondiente.

Entrada	C	B	A	Alarma	Rebose	Lleno	Vacío	B2	B1	**Salida**
P0[2:0]	P0.2	P0.1	P0.0	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	P1[5:0]
0H	0	0	0	0	0	0	1	1	1	07H
1H	0	0	1	0	0	0	0	1	1	03H
2H	0	1	0	1	0	0	0	0	0	20H
3H	0	1	1	0	0	1	0	0	1	09H
4H	1	0	0	1	0	0	0	0	0	20H
5H	1	0	1	1	0	0	0	0	0	20H
6H	1	1	0	1	0	0	0	0	0	20H
7H	1	1	1	0	1	1	0	0	0	18H

Tabla 6.22: Tabla de verdad del ejemplo de diseño con 8051

El sistema se implementa mediante una tabla de lookup. El código en ensamblador empleado se muestra a continuación:

```
        ORIGIN 300H
        .BYTE 07H,03H,20H,09H,20H,20H,20H,18H
        ORIGIN 200H
        MOV DPTR,#300H ;Carga 300H en DPTR
SALTO:  MOV A,P0       ;Carga el valor del puerto P0 en A
        ANL A,#07H     ;Multiplica el contenido de A por 7
        MOVC A, @A+DPTR;A = ((A)+(DPTR))
        MOV P1,A       ;Transmite el contenido de A al puerto P1
        JMP SALTO      ;Retorna a SALTO
        END
```

La directiva `ORIGIN` ubica el código que sigue a continuación (la tabla de lookup) en la dirección `300H`. Es decir, en la dirección `300H` de memoria de código se carga el valor `07H`, en la `301H` el valor `03H` y así sucesivamente.

El programa empieza en la dirección `200H`: Carga `300H` en DPTR, lee el puerto `P0` y le aplica una máscara para quedarse con los 3 bits menos significativos. A continuación, suma el contenido de A con el contenido de DPTR; el resultado es una dirección de la tabla de lookup cuyo contenido (una de las salidas de la tabla de lookup) se carga en el acumulador. Finalmente se escribe el valor del acumulador en el puerto `P1`.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.7.3. Programación del puerto serie

En este apartado se repite el ejemplo anterior utilizando lenguaje C y como salida de datos el puerto serie en lugar del puerto 1:

```
#include<reg52.h>
#include <stdio.h>

tabla[]={0x07,0x03,0x20,0x09,0x20,0x20,0x20,0x18};

unsigned char funcion(unsigned char valor){
                        return tabla[valor];
              }

void main(void){

    unsigned char entrada;
    unsigned char salida;

    SCON  = 0x50; // Modo 1, UART de 8 bits
    TMOD |= 0x20; // Modo 2, 8 bits con autorrecarga
    TH1   = 232;  // TH1: Valor de recarga para 1200 baudios
                  // a 11.059MHz
    TR1   = 1;    // TR1: Timer 1 habilitado
    TI    = 1;    // TI:  Flag TI activado para enviar caracter
                  // a la UART

    for(;;){
       entrada = P0 & 0x07;
       salida = funcion(entrada);
       printf("%bx\n", salida);
    }
}
```

El fichero reg52.h contiene definiciones de los registros del microcontrolador. La función *printf* de la librería stdio.h *escribe* en el puerto serie del microcontrolador, es decir, es una función de alto nivel para transmitir por el puerto serie.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.7.4. Entorno de desarrollo integrado de Keil

μVision2 de Keil Software es un entorno de desarrollo integrado (IDE, Integrated Development Environment) para microcontroladores que incluye gestor de proyectos, editor, compilador, ensamblador y simulador[303]. El simulador permite incluso programar funciones que simulan generadores de señal analógicos y digitales. A continuación, se describe con un ejemplo los pasos básicos a seguir compilar y ejecutar un programa.

En primer lugar, hay que crear un proyecto, seleccionar un microcontrolador y configurar el entorno de desarrollo. Para ello siga los siguientes pasos:

1. Cree una carpeta nueva llamada `miprograma` en el path:

   `c:\\keil\c51\examples`

2. Ejecute μVision2.

3. En primer lugar hay que crear un proyecto. Desde la ventana principal, elija el menu desplegable *Project* y seleccione *New project*. Aparecerá una nueva ventana. Seleccione la carpeta creada previamente (`miprograma`), escriba en el fondo de la ventana el nombre del nuevo proyecto, por ejemplo, `miprograma`, y pulse *Guardar*. Aparecerá una nueva ventana de configuración del sistema.

4. Seleccione el microcontrolador que quiere usar. Por ejemplo, acceda a la carpeta de los microcontroladores de Intel seleccione el 8051AH y pulse *Aceptar*.

5. Seleccione *Options for target* localizado en el menu desplegable *Project*. Aparecerá una nueva ventana. Seleccione la pestaña *Output* y habilite la opción para crear ficheros Hex marcando la check-box correspondiente y pulsando *Aceptar*.

Ahora ya se puede empezar a escribir el programa. Para ello siga los siguientes pasos:

1. En la ventana principal, seleccione *New* del menu desplegable *File*. Aparecerá una nueva ventana llamada *text1*.

2. Escriba el código fuente C del ejemplo anterior.

[303] En la página web de Keil (www.keil.com) se puede descargar gratuitamente un versión recortada pero operativa del programa.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3. Una vez escrito todo el código, guárdelo seleccionando *Save* del menu desplegable *File*. Aparecerá una nueva ventana de dialogo. Escriba en el fondo de la ventana el nombre del nuevo fichero, por ejemplo, `miprograma.c`, y pulse *Guardar*.

4. Hay que incluir el programa en el proyecto. Para ello, despliegue la carpeta *Target 1* de la ventana de la izquierda y haga clic con el botón derecho del ratón sobre *Source Group 1*. Seleccione *Add Files to Group Source Group 1*.

5. Seleccione el fichero `miprograma.c`, pulse *Add* y, a continuación *Close*.

Ahora ya podemos compilar el programa. Para ello, seleccione *Build Target* del menú desplegable *Project*[304].

Si la compilación se completa con éxito aparecerá en la pantalla un mensaje indicando que ha habido 0 errores[305]. Algunas veces pueden aparecer algunos avisos (warnings) pero que no impiden la ejecución del programa. En la carpeta `miprograma` se habrán creado varios ficheros. Entre ellos está el fichero `miprograma.hex` que es el que se volcará en el microcontrolador via puerto serie; su formato es un estándar de Intel.

Una vez obtenido el ejecutable, se puede simular. Para arrancar una sesión de simulación hay que configurar la simulación. Para ello, siga los siguientes pasos:

1. Seleccione *Start/Stop Debug Session* del menu desplegable *Debug*.

2. Para visualizar el puerto 0, seleccione *Port0* de *IO-Ports* del menu desplegable *Peripherals*.

3. Para visualizar el puerto serie, seleccione *Serial Windows #1* del menu desplegable *View*.

4. Para ejecutar el programa, seleccione *Go* del menu desplegable *Debug*.

5. En la ventana del puerto serie se verá el dato (18) que está siendo transmitido continuamente. En la ventana del puerto 0 se pueden cambiar los valores del mismo marcando las check-boxs. Puede observar como se modifica la salida del puerto serie.

[304] También puede iniciar la compilación pulsando la tecla de función especial F7 o haciendo clic sobre el icono *Build Target* en la barra de herramientas de *compile*.

[305] Cuando hay errores de compilación se puede hacer clic sobre el mensaje de error para que μVision2 resalte la línea de código que lo ha originado.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4. El ADuC842 de Analog Devices

El microconvertidor ADuC842 de Analog Devices es un microcontrolador moderno compatible con el conjunto de instrucciones del 8051 de Intel que tiene las siguientes características:

- Trabaja con un cristal de 32 KHz con un PLL integrado para generar un reloj programable de hasta 16,8MHz. Tiene unas prestaciones de 20 MIPS de pico.
- 8 canales de ADC de 12 bits autocalibrados.
- 2 canales DACs $\Sigma - \triangle$ de 12 bits.
- 62 KB de memoria de programa Flash/EE.
- 4KB de memoria de datos Flash de 4 KB.
- 256 bytes de memoria RAM interna más 2 KB de memoria RAM extendida on chip.
- 2 moduladores de anchura de pulsos (PWM) de 16 bits.
- 1 watchdog timer.
- 1 contador de intervalos de tiempo.
- 3 contadores/timers.
- 3 puertos serie: SPI, I2C y UART.
- Soporta carga del programa vía puerto serie UART (in-circuit serial download) y modos de depuración y emulación.

La figura 6.10 muestra un diagrama de bloques del mismo. El ADuC 842 tiene todos los registros SFR del 8051 y añade los que se describen en la tabla 6.10.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

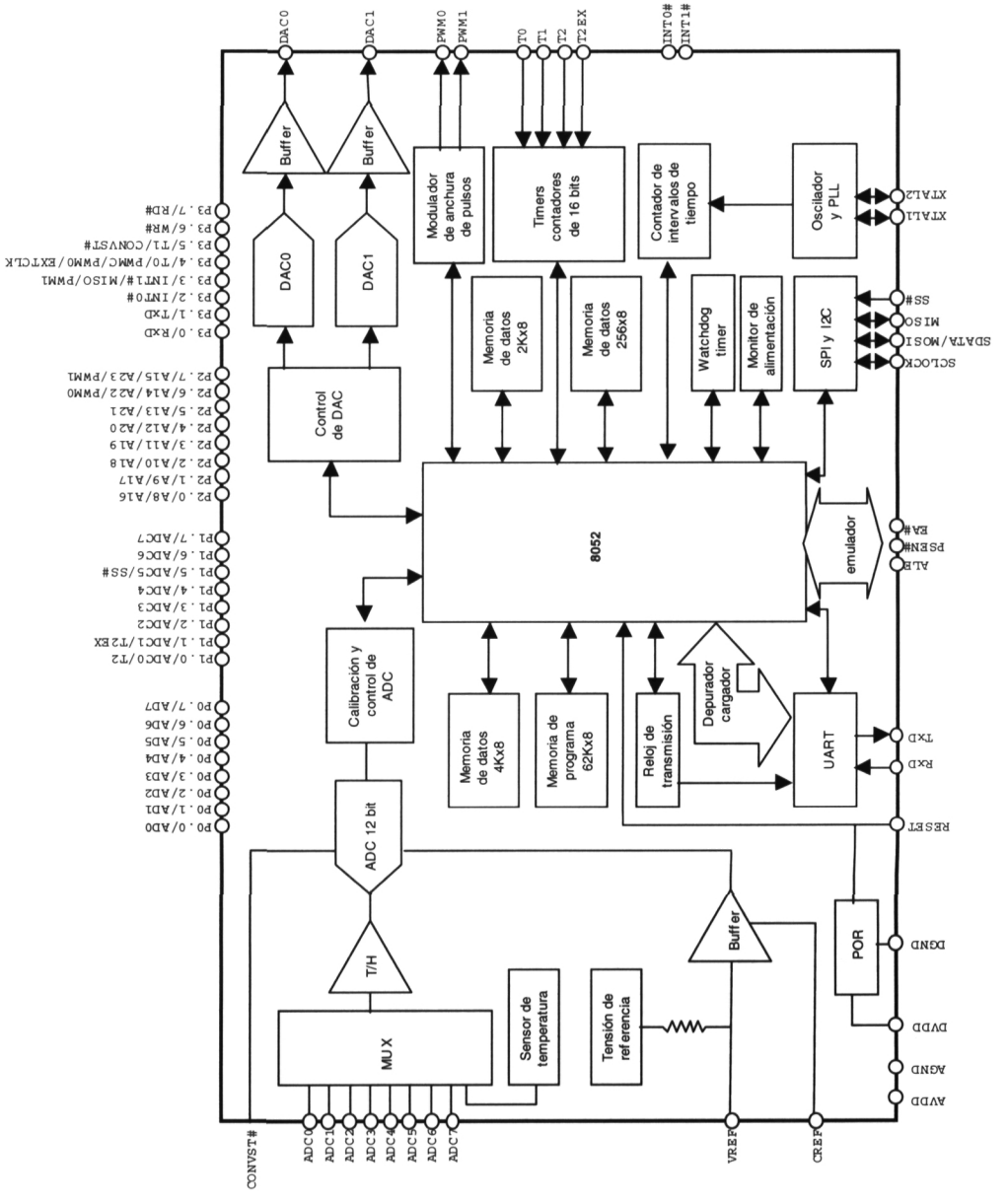


Figura 6.10: Diagrama de bloques del microconvertidor 842 de Analog Devices

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Tipo	Dirección	Descripción	símbolo
Puntero datos	84H	Puntero de datos de página	DPP
Puntero de datos	87H	Byte alto de puntero de pila	SP
Puntero de datos	D2H	Puntero de direcciones de DMA	DMAL
Puntero de datos	D3H	Puntero de direcciones de DMA	DMAH
Puntero de datos	D4H	Puntero de direcciones de DMA	DMAP
Control punteros de datos	A7H	Registro de control de puntero de datos	DPCON
Contador 2	CCH	Byte bajo del timer 2	TL2
Contador 2	CDH	Byte alto del timer 2	TH2
Contador de tiempo	A2H	Registro de 128-ésimas partes de segundos pasadas	HTHSEC
Contador de tiempo	A3H	Registro de segundos pasados	SEC
Contador de tiempo	A4H	Registro de minutos pasados	MIN
Contador de tiempo	A5H	Registro de horas pasadas	HOUR
Control Contadores	C8H	Registro de control de timer 2	T2CON
Control Contadores	CAH	Registro de recarga y captura de timer 2	RCAP2L
Control Contadores	CBH	Registro de recarga y captura de timer 2	RCAP2H
Control Contadores	9EH	Registro de control de timer 3	TCON
Control Contadores	A1H	Registro divisor fraccional del timer 3	TIMECON
Control Contadores	9DH	Registro de control de contador de intervalo de tiempo	TIMECON
Control Contadores	A6H	Registro de intervalo de tiempo	TIMECON
Control Contadores	C0H	Registro de control de watchdog timer	WDCON
PWM	81H	Byte bajo de registro PWM0	PWM1H
PWM	82H	Byte alto de registro PWM0	PWM1L
PWM	83H	Byte bajo de registro PWM1	PWM1H
PWM	84H	Byte alto de registro PWM1	PWM1H
Control PWM	AEH	Registro de control de modulación de anchura de pulsos	PWMCON
Interfaz I2C	9AH	Registro de datos I2C	I2CDAT
Interfaz I2C	9BH	Registro de direcciones de esclavo I2C	I2CADD
Interfaz I2C	91H	Registro de direcciones de esclavo secundario I2C	I2CADD1
Interfaz I2C	92H	Registro de direcciones de esclavo secundario I2C	I2CADD2
Interfaz I2C	93H	Registro de direcciones de esclavo secundario I2C	I2CADD3
Interfaz I2C	E8H	Registro de de control I2C en modo maestro	I2CCON
Interfaz SPI	F7H	Registro de datos SPI	SPIDAT
Interfaz SPI	F8H	Registro de control SPI	SPICON
Gestión Interrupciones	A9H	Registro 2 de prioridad y habilitación de interrupciones)	IEIP2
Alimentación	DFH	Registro de control de monitorización de alimentación	PSMCON
Configuración	87H	Registro de configuración de 842	CFG842
Identificación	C2H	Registro de identificación del chip	CHIPID
Reloj	D7H	Registro de control de PLL	PLLCON
Flash/EE	B9H	Registro de comando de Flash/EE de datos	ECON
Flash/EE	BCH	Registro de comando de Flash/EE de datos	EDATA1
Flash/EE	BDH	Registro de comando de Flash/EE de datos	EDATA2
Flash/EE	BEH	Registro de comando de Flash/EE de datos	EDATA3
Flash/EE	BFH	Registro de comando de Flash/EE de datos	EDATA4
Flash/EE	C6H	Byte bajo de Registro de direcciones de Flash/EE de datos	EADRL
Flash/EE	C7H	Byte alto de Registro de direcciones de Flash/EE de datos	EADRH
ADC	EFH	Registro 1 de control ADC	ADCCON1
ADC	D8H	Registro 2 de control ADC	ADCCON2
ADC	F5H	Registro 3 de control ADC	ADCCON3
ADC	D9H	Registro de datos de ADC	ADCDATAL
ADC	DAH	Registro de datos de ADC	ADCDATAH
ADC	F1H	Coeficientes de calibración de offset de ADC	ADCOFSL
ADC	F2H	Coeficientes de calibración de offset de ADC	ADCOFSH
ADC	F3H	Coeficientes de calibración de ganancia de ADC	ADCGAINL
ADC	F4H	Coeficientes de calibración de ganancia de ADC	ADCGAINH
DAC	F9H	Registro de datos de DAC0	DAC0L
DAC	FAH	Registro de datos de DAC0	DAC0H
DAC	FBH	Registro de datos de DAC1	DAC1L
DAC	FCH	Registro de datos de DAC1	DAC1H
DAC	FDH	Registro de control de DAC	DACCONH

Tabla 6.23: Registros de función especial del 842

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5. Bibliografía

La bibliografía incluye manuales del microcontrolador 8051, un buen número de libros sobre microcontroladores (la mayoría de ellos especializados en alguna familia y fabricante), algunas revistas relacionadas y páginas web. En particular, se hace un amplio listado de direcciones web de fabricantes de microcontroladores. Los fabricantes proporcionan información actualizada, notas de aplicación e incluso sistemas de desarrollo software gratuitos (simuladores, compiladores, etc.).

Manuales

8-bit Embedded Controllers Handbook", 1991. Intel.

16-/32-bit embedded processor handbook",1991. Intel.

Embedded controller aplications handbook",1991. Intel.

16-bit embedded controller handbook",1991. Intel.

Libros

Angulo, J.S.: Microcontroladores PiC. Diseño práctico de aplicaciones, MacGraw-Hill, 1997. ISBN: 84-4811-238-5.

Barnett, R.: The 8051 Family of Microcontrollers, Prentice Hall Career&Technology, 1997. ISBN: 0-02-306281-9.

Driscoll, F.F., Coughlin, R.F. and Villanucci, R.S.: Data acquisition and process control with the M68HC11 Microcontroller, Prentice Hall Career&Technology, 1994. ISBN: 0-02-330555-X.

González Vázquez, J.A.: Introducción a los Microcontroladores. McGraw Hill, 1992. ISBN: 84-7615-803-3.

González Vázquez, J.A.: Introducción a los Microcontroladores de 16 bits. Familia 96, McGraw Hill, 1994. ISBN: 84-4811-936-3.

Grover, D., and Deller,J.: Digital Signal processing and the Microcontroller, Prentice-Hall International, 1998. ISBN 0-13-081348-6.

Kheir, M.R.: The M68HC11 Microcontroller: Applications in Control, Instrumentation and Communication, Prentice-Hall International, 1997. ISBN: 0-13-205550-3.

MacKenzie, S.: The 8051 Microcontroller, Prentice Hall Career&Technology, 1998. ISBN: 0-13-780008-8.

Odant, B.: Microcontroladores 8051 y 8052, Paraninfo, 1995.

Schultz, T.: C and the 8051, Vol. 1: Hardware, Modular programming&Multitasking, Prentice Hall Career&Technology, 1998. ISBN: 0-13-754839-7.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Schultz, T.: C and the 8051, Vol. 2: Building Efficient Applications, Prentice Hall Career&Technology, 1998. ISBN: 0-13-521121-2.

Spasov, P.: Microcontroller Technology: The 68HC11, Prentice-Hall International, 1998. ISBN: 0-13-901240-0.

Stewart, J.W. and Miao, K.: The 8051 Microcontroller: Hardware, Software, and Interfacing, Prentice-Hall International, 1998. ISBN: 0-13-531948-X.

Tavernier, C.: Microcontroladores de 4 y 8 bits, Paraninfo, 1996. ISBN: 84-283-2406-9.

Tavernier, C.: Microcontroladores PiC, Paraninfo, 1997. ISBN: 84-283-2373-9.

Yeralan, S.: Programming and interfacing the 8051 microcontroller, Adisson-Wesley, 1995. ISBN: 0-201-63365-5.

Palacios, E., Remiro, F. y López, L.: Microcontrolador PIC16F84. Desarrollo de proyectos (2ª Edición), RA-MA, 2006. ISBN: 84-789-7691-4.

Revistas

Computer Design, PennWell Publishing Company, ISSN-0010-4566.

EDN, Cahners Publishing Company, ISSN 0012-7515.

Electronics Design, UNU Businesss Publications Inc., ISSN 0013-4872.

Mundo Electrónico, Boixareu Ed. ISSN: 0300-3787

Revista Española de Electrónica, Ediciones técnicas REDESA, ISSN: 0482-6396.

Páginas WWW

Revistas

IEEE Micro. www.computer.org/pubs/micro/micro.htm

MicroDesign Resources www.chipanalyst.com

EDN. www.ednmag.com

Fabricantes de microcontroladores

AMD. www.amd.com

Analog Devices. www.analog.com

Atmel. www.atmel.com

Cygnal Integrated Products. www.cygnal.com

Cybernetic Micro Systems. www.controlchips.com

Cypress Microsystems. www.cypressmicro.com

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Fujitsu. www.fmi.fujitsu.com

Hitachi. www.semiconductor.hitachi.com

Infineon. www.infineon.com/tricore

Intel. www.intel.com

Microchip Technology Inc. www.microchip.com

Motorola. www.motorola.com

NEC Electronics. www.necel.com

Philips Semiconductors. www.philips.com

Rabbit Semiconductor. www.rabbitsemiconductor.com

STMicroelectronics. www.st.com

Texas Instruments. www.ti.com

Toshiba. www.toshiba.com

Zilog Inc www.zilog.com

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

6. Guía de estudio

1. Explicar las diferencias entre un microcontrolador y un microprocesador.
2. Distinguir entre microcontrolador y controlador empotrado.
3. Dibujar el diagrama de bloques de un microcontrolador genérico. Explicar la función de cada bloque.
4. Describir los tipos de memoria que integran los microcontroladores. Explicar las ventajas e inconvenientes de usar diferentes tipos de memoria ROM (ROM, PROM, EPROM, EEPROM, Flash).
5. Explicar las diferentes tendencias de diseño de microcontroladores (chips de soporte, múltiples versiones, ASIC, etc.).
6. Elaborar una tabla comparativa de microcontroladores donde se comparen todos los parámetros relevantes (frecuencia de reloj, tamaño de RAM y de ROM, nº de timers, nº de puertos paralelos, puerto serie, etc.).
7. Discutir los criterios de selección de un microcontrolador incluyendo aspectos tales como los sistemas de desarrollo, el soporte técnico, las segundas fuentes, las versiones preliminares, etc.
8. Describir las características generales del 8051. Dibujar su diagrama de bloques.
9. Distinguir entre los diferentes espacios de memoria del 8051.
10. Entender la función de cada uno de los registros SFR del 8051.
11. Explicar el funcionamiento de la tabla de vectores de interrupción del 8051.
12. Analizar secuencias de código 8051 de un sistema sencillo de control y adquisición de datos.
13. Manejar las funciones básicas del entorno de desarrollo μVision2.
14. Explicar las diferencias entre un 8051 de Intel y un ADuC842 de Analog Devices.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

7 Guía de diseño de un sistema basado en microprocesador

1. Introducción

La metodología de diseño y desarrollo de un Sistema Basado en Microprocesador (SBM) no se restringe sólo al diseño técnico ya que hay muchos otros aspectos que, al final, determinan decisiones importantes en la implementación. La metodología empieza cuando una empresa se plantea una idea que desembocará después de todo el proceso en un nuevo producto en manos de los clientes.

En este capítulo se describen las recomendaciones y guías de diseño básicas en el diseño y desarrollo de estos sistemas[306].

[306] Muchas de las recomendaciones y guías que aquí se presentan no serán estrictamente necesarias para el diseño de un sencillo prototipo.

Publicaciones y Divulgación Científica
UMA UNIVERSIDAD DE MÁLAGA

2. Etapas de diseño de un SBM

La figura 7.1 muestra los pasos a seguir en el desarrollo de un proyecto que pretenda implementar un SBM. A continuación se describe cada uno de ellos.

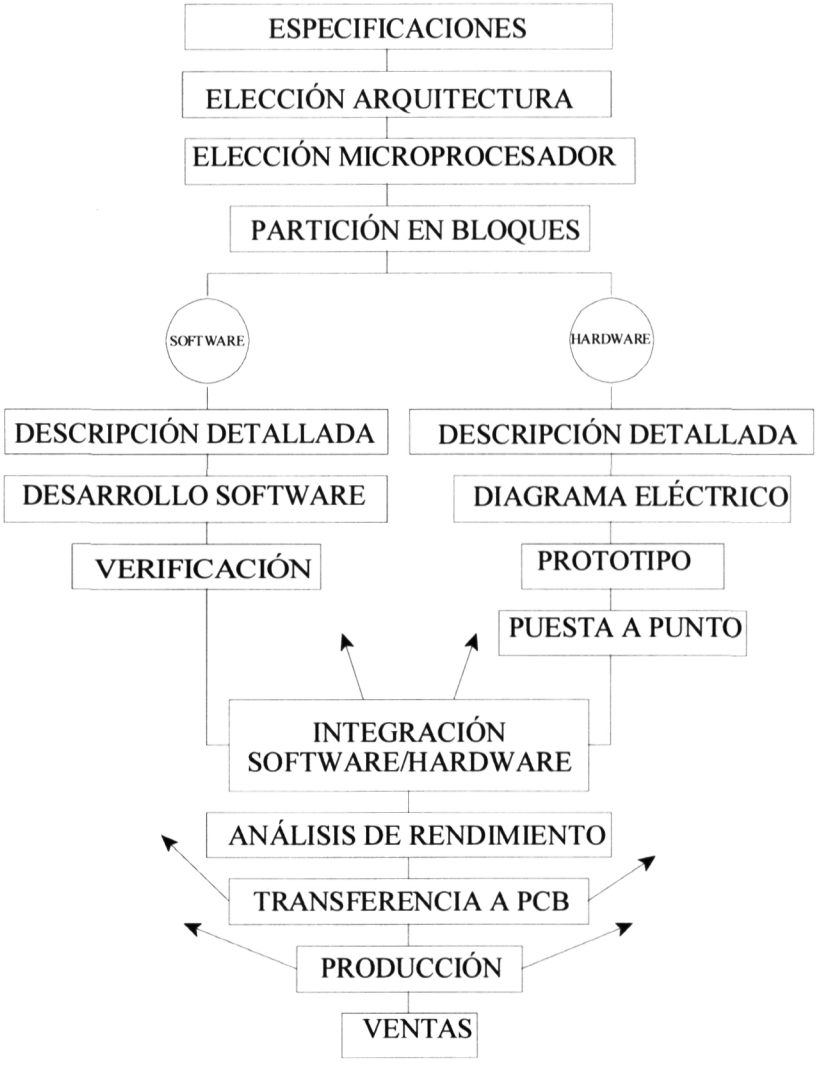


Figura 7.1: Etapas de diseño de un sistema basado en microprocesador

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.1. Especificaciones del producto

Cuando una empresa decide desarrollar un nuevo producto se debe estudiar su viabilidad desde muchos puntos de vista: Técnico, producción, ventas, calidad, etc. El resultado final suele ser un compromiso entre prestaciones y coste. No obstante, en muy pocas ocasiones, la mejor solución técnica será la solución elegida ya que intervienen muchos condicionantes: política de la empresa, oportunidad de mercado, consumo, espacio disponible, coste del producto final, stock[307], etc.

Una vez se decide la viabilidad del producto, hay que plasmar en un documento *qué* se quiere hacer (*especificación funcional y de requisitos*) y *en cuánto* tiempo (planificación temporal). Estos serán los documentos de partida de todo el proyecto.

2.2. Elección de la arquitectura

En la mayoría de los casos, la arquitectura elegida se apoya en arquitecturas ya conocidas o desarrolladas (con el objeto de reducir el tiempo de desarrollo) y utilizando tecnología y herramientas también conocidas.

Hay que establecer la arquitectura software y hardware. En la arquitectura hardware hay que tomar decisiones a muchos niveles: sistema monoprocesador o multiprocesador, tipo de arquitectura y de bus del sistema, uso de lógica custom y controladores, etc. En cuanto a la arquitectura software haya que decidir qué tipo de sistema operativo se requiere, la necesidad de gestión de memoria, el número de espacios de memoria, etc..

2.3. Bus del sistema

El bus del sistema merece una especial atención. En un sistema multiplaca hay que decidir qué tipo de bus del sistema se usa. Hay dos opciones básicas: Bus estándar (PCI, VME, Multibus, etc.) o bus específico para la aplicación.

El uso de un bus estándar tiene un conjunto de innegables ventajas:

- Existen muchas placas con el bus estándar disponibles en el mercado.

[307] Las empresas compran componentes electrónicos en grandes cantidades, consiguiendo así precios muy bajos. Este hecho hace que la empresa tenga, en muchas ocasiones, un stock (almacén) de componentes electrónicos de antiguos proyectos. Cuando se arranca un nuevo proyecto siempre se valora la posibilidad de usar los componentes del stock a fin de reducir costes y tiempo de desarrollo

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Los componentes de los buses estándar son fáciles de encontrar (conectores, backplanes, drivers y receivers, incluso controladores de bus estándar).
- Normalmente funciona. Esta es la gran ventaja. Siguiendo las especificaciones del bus al pie de la letra el bus funcionará.

Aunque también habría que destacar algunos inconvenientes:

- Pueden ser buses lentos. Como son buses de propósito general y pretenden cubrir muchas aplicaciones, pueden no ser óptimos (nº de bytes transferidos/s) para una aplicación en particular.
- Pueden resultar caros. Como son de propósito general soportan muchas características que no tienen porqué ser necesarias en nuestra aplicación.
- Pueden ser inconvenientes. Si la placa estándar es pequeña para nuestra aplicación, obliga a utilizar diversas técnicas como montaje superficial, placas hijas, etc.. El uso de PCBs más grandes simplificaría técnicas y componentes más asequibles que permitirían obtener una placa más barata y más fácil de fabricar.

En cuanto a un bus específico, normalmente no se suele partir de cero sino que se aprovechan elementos de los buses estándar y se adaptan a los requerimientos particulares. Por ejemplo, se puede usar backplane VME y protocolos del bus VME con una PCB no estándar, o usar una arquitectura de bus estándar de 16 bits y ponerle un conector de 32 bits.

2.4. Elección de microprocesador

Es una de las elecciones más importantes[308]. La elección técnica suele estar condicionada por las herramientas hardware (Analizadores lógicos programables, emuladores, placas de evaluación, etc.) y software (ensambladores, compiladores, etc.) disponibles o al menos conocidas por el diseñador.

Normalmente la compañías se especializan en una familia de microprocesadores y todos sus productos se basan en ellas (por ejemplo, Intel para PC y Motorola para Apple).

[308] En ocasiones, hay aspectos políticos y económicos en juego. Recuerde la elección del 8088 en lugar del 68000 en el diseño del IBM PC.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

2.5. Partición del sistema

La partición del sistema se hace atendiendo a criterios de funcionalidad y de abstracción. Las figuras 7.2 y 7.3 muestran un ejemplo de partición de un SBM complejo[309]: Una central de conmutación de paquetes basada en un sistema multiprocesador multiplaca.

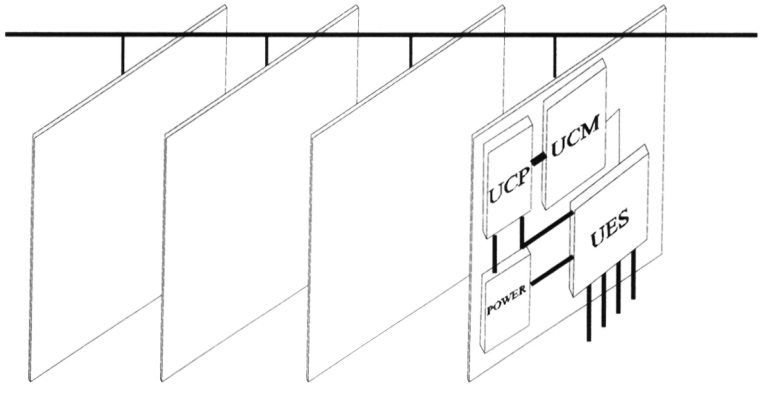


Figura 7.2: Ejemplo de SBM: Central de conmutación de paquetes

Cada placa incluye un SBM (UCP, ECM y UES). Las placas se comunican entre sí mediante un bus serie de alta velocidad. Hay varios tipos de placas: placas de comunicaciones (la mayoría de ellas), placas de conmutación, placas de gestión de alarmas y placas de control.

El sistema se divide en subsistemas (comunicaciones, conmutación, control y alarmas) y éstos en módulos (UCP, UCM y UES). Cada módulo tiene una función determinada y se comunica con el resto mediante interfaces de entrada y salida.

En la partición también se aplica el concepto de niveles de abstracción. La capa mecánica es la placa de circuito impreso (pistas, planos de masa y alimentación, conectores, zócalos) que da servicio a la capa hardware (componentes electrónicos). La capa hardware da servicio al software de bajo nivel (rutinas de control y gestión de los diferentes componentes electrónicos). La capa de software de bajo nivel da servicio a la capa de software de medio nivel (normalmente el sistema operativo). Finalmente la capa de software de medio nivel da servicio al software de aplicaciones.

[309] El ejemplo está basado en el diseño del conmutador de paquetes TesysB de Telefónica I+D de los años 90.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

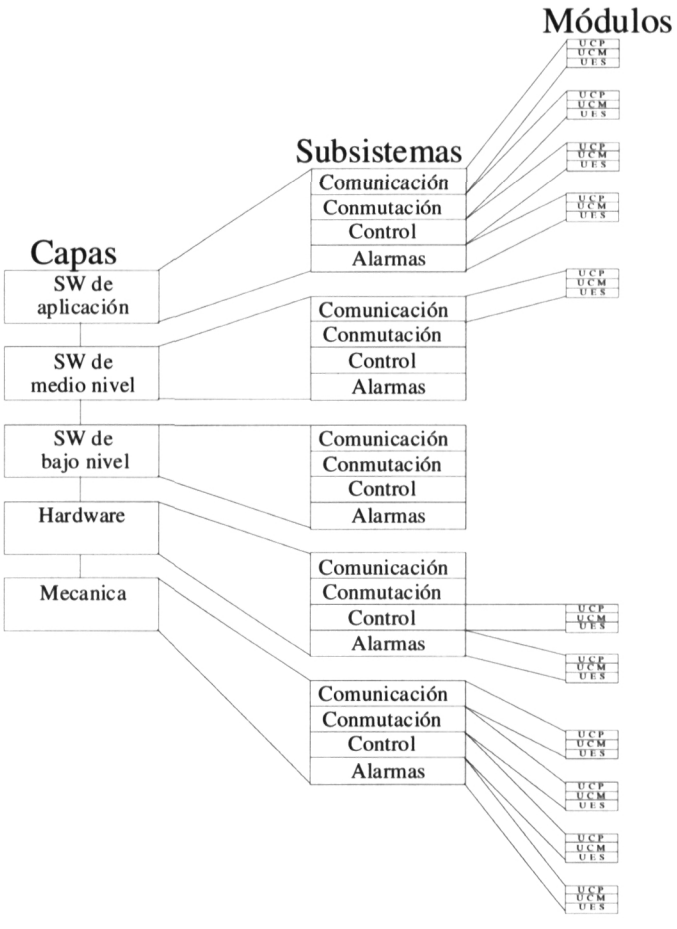


Figura 7.3: Ejemplo de partición del sistema de la figura 7.2

2.6. Descripción detallada del diseño

En la descripción detallada debe figurar el diseño y descripción de cada elemento. El elemento más pequeño es el módulo. Por cada módulo habrá que especificar: funcionalidad, señales de interfaz, componentes, consumo, etc..

La lista de decisiones detalladas no es sencilla y se puede llegar a hacer interminable. Habría que describir[310]:

- Familias lógica a usar: 74ALS, 74F, HCMOS, etc.

[310]En muchos casos, estas decisiones, vendrán ya impuestas en las especificaciones de requisitos.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- Tipo de empaquetado de chips: DIP, SMD, etc.
- Tamaño de los chips de memoria DRAM, posibilidad de ampliación, etc.
- Disponibilidad de componentes: muestras, en fabricación, dobles fuentes, etc.
- Espacio físico disponible en placa.
- Tipo de PCB (tecnología simple y barata de 2 capas o diseño más complejo de 4 ó 6 capas). Hay placas con substrato cerámico de más de 30 capas, pero son muy caras.
- Uso de planos de masa y alimentación.

2.6.1. Diagrama eléctrico del circuito

El diagrama eléctrico del circuito forma parte de la descripción detallada. Incluye tanto la circuitería convencional como el diseño y desarrollo de la lógica programable y custom (a medida).

Hoy en día los diagramas eléctricos no se diseñan a mano. Hay paquetes software de captura de esquemáticos que generan los diagramas (CAD, Computer Aided Design). Las ventajas que se obtiene son muchas: Diagramas impecables, actualización de esquemáticos casi inmediata, chequeo de conexiones[311] y simulaciones de todo tipo: lógicas, temporales, térmicas, etc..

Aunque la mayoría de los errores de diseño se *cazan* en simulación, algunos quedan ocultos, ya que normalmente los patrones de test no chequean el 100 % del funcionamiento del sistema.

2.7. Uso de lógica custom

Se usa lógica custom cuando el espacio es limitado, hay restricciones de potencia y/o el producto resultante se fabrica en grandes cantidades.

A no ser que se tenga mucha experiencia, no es mala idea terminar antes la lógica programable y los customs antes de hacer el prototipo ya que:

- La inclusión de muchos circuitos en un chip FPGA puede impedir el ruteo del software CAE (Computer Aided Engineering) y se hace necesario usar dos chips o un chip más grande.

[311] El chequeo de conexiones (net-list) detecta si hay cortocircuitos, fan out, colisión de señales, entradas no conectadas, etc.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

- El CAE asigna los pines para conseguir el menor retardo de señales. Los pines podrían estar en un sitio distinto al dibujado en el diagrama eléctrico.

2.8. Prototipo

Hay dos tipos de prototipos: grapinado y de circuito impreso (PCB, Printed Circuit Board).

El grapinado permite una implementación más rápida. Normalmente los chips se insertan en zócalos y son los zócalos los que se grapinan. Esto permite cambiar los chips fácilmente. Como inconvenientes caben citar que no funcionan bien a muy altas velocidades (60MHz como máximo) y que ciertos chips no tienen zócalos adecuados (aunque siempre se encuentran soluciones). Además, es común encontrarse con una maraña de cables donde no es fácil discernir cada una de las señales.

Los circuitos impresos se hacen cuando no hay tiempo de hacer un prototipo grapinado o cuando la frecuencia de trabajo es demasiado alta[312]. Se usan herramientas CAD que permiten reducir muchos errores de diseño. Los errores que no se hayan podido evitar se subsanan haciendo cortes en las pistas y puentes (cablecitos que establecen la conexión correcta).

2.9. Puesta a punto

La puesta a punto pretende chequear y mejorar el diseño del prototipo hardware. La instrumentación utilizada en la puesta a punto consiste[313] en: Multímetro, osciloscopio digital, analizador lógico, programador de EPROM y PAL, monitor y emulador (ICE, In-Circuit Emulator).

Las fases a seguir en la puesta a punto son:

1. Inspección visual para detectar errores de producción: Componentes mal montados o defectuosos, malas soldaduras, errores de lay-out, cortocircuitos, etc. También se hace uso del multímetro para chequear continuidades de las pistas y alimentaciones de los componentes.
2. Chequeo de las señales críticas (relojes y señales de alta velocidad). Se usa el osciloscopio digital y se registra la forma de estas señales para detectar posibles sobreoscilaciones o diafonías en pistas adyacentes.

[312] Para su fabricación se usan las tradicionales técnicas fotolitográficas o las novedosas herramientas de prototipado rápido que esculpen las pistas sobre una capa de cobre.

[313] Depende de la complejidad del sistema.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3. Análisis de tiempos donde se estudian las señales hardware mediante un muestreo a frecuencia fija. Permite detectar violaciones de los tiempos especificados por el fabricante para el acceso a un componente (por ejemplo, accesos a memoria y a registros de E/S). Se usa el analizador lógico que permite una representación visual de los niveles lógicos que ocurren el tiempo real en el hardware.

4. Análisis de estados que genera una traza del flujo del programa[314]. La traza sólo muestra cuándo hay un cambio de datos y permite detectar si los valores de direcciones y datos y/o el "handshake" de las señales de control se ejecuta correctamente[315]. Se usa el analizador lógico.

El emulador (figura 7.4) sustituye al microprocesador del prototipo y también a la memoria y ejecuta las instrucciones del sistema en el entorno del prototipo pero bajo el control del ICE.

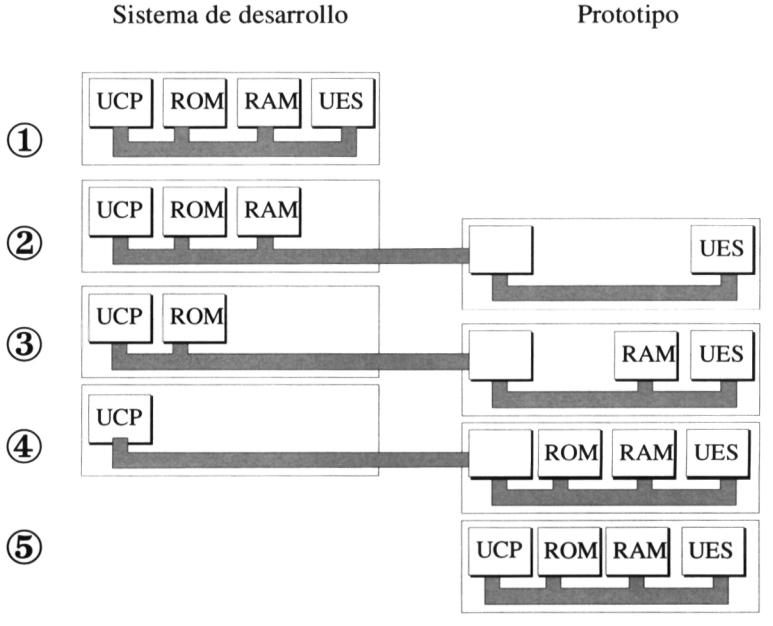


Figura 7.4: Etapas de desarrollo de un SBM usando un emulador

[314]Los programas que se ejecutan son muy simples y no tiene porque estar disponible el programa de aplicación, ni siquiera el sistema operativo.

[315]Normalmente el flujo de programa dirá qué funciona mal y dónde. Estudiando las señales hardware y las instrucciones se podrá determinar dónde está el error.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Cuando se usa un emulador para desarrollar un SBM, las etapas a seguir se parecen al desarrollo de un feto en el seno de su madre . El emulador es la madre y la placa prototipo el feto. Inicialmente sólo está la madre. El prototipo se desarrolla y crece de 1 a 4. Al final en 5, el prototipo puede andar por su propio pie.

2.10. Integración hardware/software

El sistema operativo y software de aplicación se carga en el prototipo. Lo más probable es que vuelvan a aparecer errores. Por ejemplo, se pierden interrupciones (o no se sirven correctamente) o las operaciones de E/S no funcionan correctamente. Estos errores pueden ser de hardware o de software y obligan a repetir en orden inverso los pasos anteriores de la puesta a punto.

2.11. Análisis de rendimiento

Es una información estadística sobre la ejecución del software que permite mejorarlo.

2.12. Transferencia a PCB

El diseño definitivo será una PCB. Para ello se elaboran las recomendaciones de lay-out[316]que describen: Disposición de componentes en la placa, trazado de pistas de alta velocidad (en muchos casos, se dibujan a mano), pistas de alimentación, conectores, conectores del backplane, pulsadores de reset, puntos de prueba, disposición del microprocesador para una correcta conexión del emulador, leds para el testeo de señales críticas, uso de zócalos (sockets), pulsadores, disposición del cristal (de pie o tumbado), etc.

La placa se "ruta", se compila y se obtienen unos negativos para cada capa de la placa. Con estos negativos se fabrica la placa.

[316]Trazado, composición.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3. Recomendaciones de diseño

3.1. Diseño de pistas de alta velocidad

En alta frecuencia, las pistas se comportan como líneas de transmisión apareciendo fenómenos de reflexión (rebotes) que provocan falsas transiciones de señal, errores en los datos y/o violaciones del nivel de voltaje de entrada[317] (ver figura 7.5). Son errores transitorios y muy difíciles de depurar. La solución es adaptar impedancias terminando las líneas.

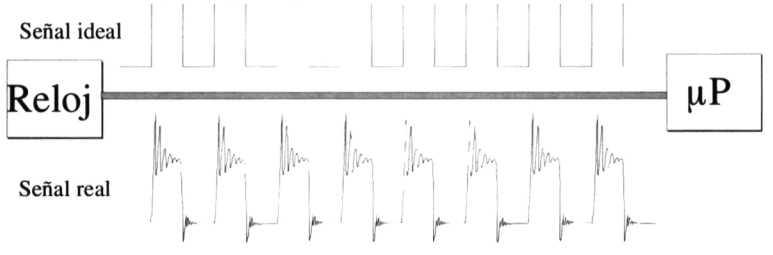


Figura 7.5: Diseño de pistas de alta velocidad

Esta terminación puede ser serie o paralelo. La terminación serie compensa el exceso de corriente, atenúa las sobreoscilaciones y consume poco, pero aumenta los tiempos de subida y bajada de la señal. La terminación paralelo ajusta la corriente al final de la línea y consume mucho pero no afecta a los tiempos de subida y bajada.

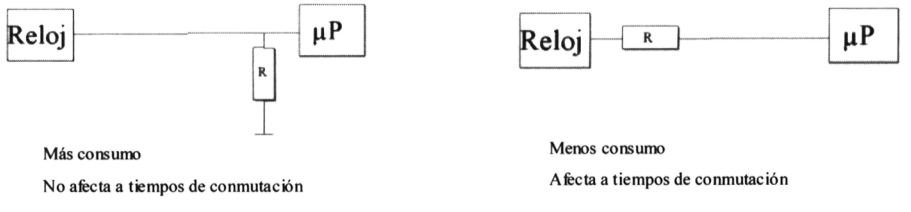


Figura 7.6: Terminaciones serie y paralelo

[317] Estas violaciones pueden dañar el circuito, o al menos, provocar un funcionamiento en zonas no especificadas por el fabricante.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.2. Planos de masa y alimentación

El uso de planos de masa y alimentación minimiza el ruido ya que la impedancia característica se reduce. En la figura 7.7 se hace equivalente un plano a un conjunto de pistas conectadas en paralelo.

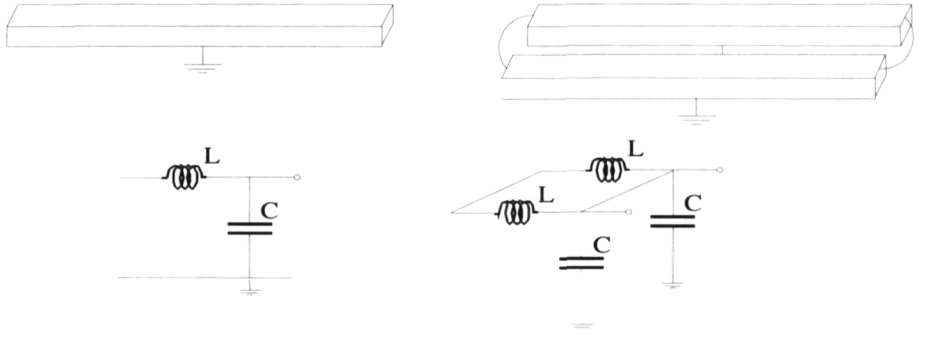


Figura 7.7: Planos de masa y alimentación. El circuito LC es el circuito equivalente de la pista

Si la impedancia característica de la pista es:

$$Z_0 = \sqrt{\frac{L_0}{C_0}} \tag{7.1}$$

la de dos pistas conectadas en paralelo será:

$$Z_0 = \sqrt{\frac{L_0/2}{C_0 \times 2}} = \frac{\sqrt{\frac{L_0}{C_0}}}{2} \tag{7.2}$$

es decir, la mitad.

Hay que separar los planos de masa de las señales analógicas de los de las señales digitales, ya que estos últimos introducen un ruido inaceptable en los primeros. Ambas masas se unirán en un único punto de la placa evitando bucles.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.3. Interferencias

La actividad eléctrica en un conductor hace que aparezcan voltajes transitorios en otro conductor (figura 7.8). La interferencia aumenta cuando aumenta la frecuencia ($f = 1/T \uparrow$) y/o cuando hay proximidad entre conductores ($d \downarrow$).

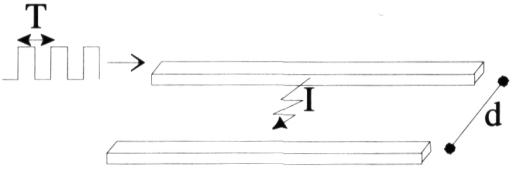


Figura 7.8: Diseño de pistas de alta velocidad

Hay dos tipos de interferencias: Electromagnética (EMI) y electroestática (ESI) (figura 7.9).

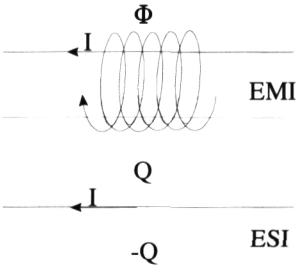


Figura 7.9: Interferencias electromagnéticas (EMI) y electroestáticas (ESI)

La EMI se produce por el campo magnético que existe alrededor de cualquier conductor que lleve corriente. El flujo magnético de un conductor induce corriente a otro conductor provocando un voltaje transitorio. Para reducir EMI se insertan pistas de masa entre pistas (figura 7.10) y se evitan los bucles cerrados que provocan ruido inductivo.

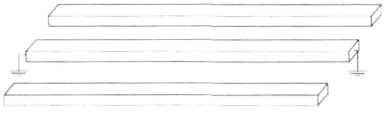


Figura 7.10: Ejemplo de pistas de masa para evitar interferencias

La ESI es el acoplamiento capacitivo de 2 conductores adyacentes. Las pistas actúan como placas de un condensador. Una carga en una pista genera

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

una carga opuesta en la pista adyacente. Para reducir ESI se separan las líneas y se ponen líneas de tierra para cancelar los campos.

3.4. Condensadores de desacoplo

La actividad de conmutación de un dispositivo se puede propagar a otros en forma de sobreimpulsos a través de las pistas de alimentación. Los condensadores de desacoplo atenúan estos sobreimpulsos y se deben colocar próximos al chip que se quiere proteger ya que la inductancia inherente de la pista anularía su efecto (ver figura 7.11).

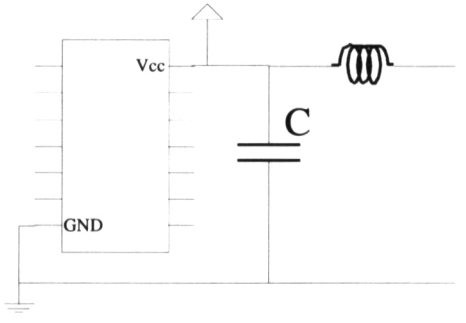


Figura 7.11: Condensadores de desacoplo

El valor del condensador de desacoplo depende del consumo del chip. En la tabla 7.1 se muestran algunos valores orientativos.

Consumo	Condensadores	Características
< 0.5W	C1	C1:100nF
0.5W-2W	C1‖C2	C1:100nF C2:2.2μF
>2W	C1‖C3	C1: 100nF C3: 16μF

Tabla 7.1: Valores orientativos de condensadores de desacoplo. C1 es cerámico multicapa y C2 y C3 electrolíticos.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

3.5. Ejemplos

3.5.1. Diseño de layout

En la figura 7.12 se muestra el layout de una placa con algunos errores de diseño.

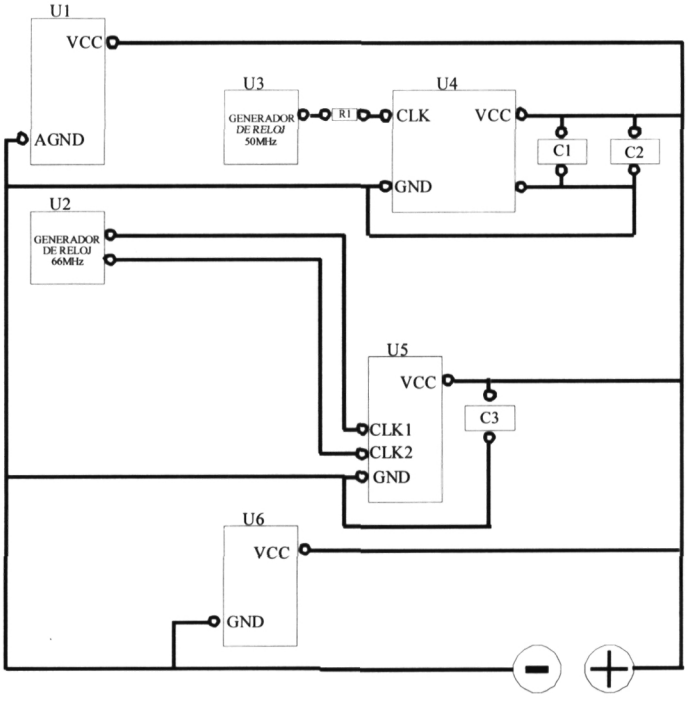


Figura 7.12: Ejemplo de fallos de diseño

Se puede observar que:

- Las pistas de reloj del componente U2 son demasiado largas y van en paralelo lo que podrían provocar diafonías. Habría que reducir su longitud y si no se puede insertar pistas de masa entre ellas y terminarlas mediante resistencias.
- El componente U1 requiere una masa analógica distinta de la digital.
- El componente U6 no tiene condensador de desacoplo.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

4. Bibliografía

Antonakos, J.L. and Adamson, T: Microcomputer Repair, Prentice Hall Career&Technology, 1998 ISBN: 0-13-893454-1

Barnett, R.: Microprocessor System Design Techniques, Sigma Press, 1991. ISBN 1-85058-152-5.

Beeson, D.L.: Assembling and Repairing Personal Computers, Prentice Hall Career&Technology, 1997 ISBN: 0-13-340225-8

Brenner, R.C.: Mantenimiento del IBM-PC, Paraninfo, 1989. ISBN 84-283-1673-2.

[Freer89] Freer, J.: Diseño de sistemas con microprocesadores avanzados, Anaya Multimedia, 1989. ISBN: 84-7614-219-6.

Roozenburg, N.F., and Eekels, J.: Product Design: Fundamentals and Methods. J. Wiley. Chichester. UK. 1996.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

5. Guía de estudio

1. Enumerar las etapas de diseño y desarrollo de un sistema basado en microprocesador y explicar las relaciones entre ellas.
2. Enumerar los criterios de selección de un microprocesador.
3. Comparar ventajas e inconvenientes de buses estándar y buses específicos.
4. Explicar en qué casos se usa lógica custom.
5. Explicar la puesta a punto de un prototipo. Relacionar cada etapa con una herramienta de desarrollo.
6. Describir el comportamiento de las pistas de alta velocidad y qué soluciones se suelen adoptar.
7. Explicar la utilidad de los planos de masa y alimentación.
8. Definir el concepto de radiointerferencia (EMI y ESI) y describir soluciones.
9. Explicar la necesidad de los condensadores de desacoplo.
10. Detectar malos diseños relacionados con la ubicación de componentes, los desacoplos, las radiointerferencias, las pistas de alta velocidad, etc.

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD DE MÁLAGA
Publicaciones y Divulgación Científica

Índice alfabético

3D, 43, 252
3DNow, 47
68HC05, 325
68HC11, 324, 325
875P, 304

4004, 35
5110, 278
6502, 35, 42
6800, 35
6809, 39
8008, 35
8051, 3, 30, 324
8080, 35
8085, 35
8086, 3, 39, 42, 54
8088, 42, 278, 292
8096, 324
8237, 189, 280, 283, 292, 295
8253, 280, 282, 283, 312
8254, 294
8255, 280, 283
8259, 186, 280, 282, 292, 295
8288, 280
16550, 199
21164, 40
21264, 40
68000, 39, 42
68020, 39
80286, 55, 292
80386, 39, 42, 55, 299
80486, 55
82340, 299

AC-link, 266
AC´97, 266, 306, 310
Accediendo a E/S, 194
Acces386, 42
ACPI, 302
actualización de caches, 11
ADC, 20
ADPCM, 259
ADuC842, 350
AGP, 210, 257, 304
almacenamiento masivo, 8, 168, 174
Altair 8800, 35
AMD, 42, 132
AmI, 48
Amiga, 47
ancho de banda, 34
APIC, 55
Apple I, 35
Apple II, 35
arbitración, 126, 129, 217, 236
ARM, 307
Arpanet, 47
arquitectura de caches, 12
ASCII, 278
ASIC, 325
ASRAM, 161
AT, 209
ATA, 223
ATAPI, 225
ATC, 56
Athlon, 132
ATM, 48
auto-refresco, 151
autoconfiguración, 219

bajo consumo, 335
bancos de memoria, 17, 146, 150, 155, 284, 307
bancos de registros, 328
BASIC, 278
BEDO DRAM, 142, 149
big endian, 7
BIOS, 280, 292
bit de arranque, 27
bit de parada, 27
bloque de cache, 10
bluetooth, 48, 310
BTB, 59, 64
buffer de video, 248
bus, 2, 209
bus AT, 209, 296
bus de E/S, 209
bus de expansión, 209, 219, 287
bus ISA, 296, 303
bus paralelo, 2
bus serie, 2

caché de trazas, 65
cache asociativa por conjuntos, 16
cache completamente asociativa, 12
cache de mapeo directo, 12
carácter, 27
carga especulativa, 66
Cayenne, 40
CCD, 311
CD RW, 177
CD-DA, 174, 180
CD-R, 177
CD-ROM, 47, 176, 178
Cell, 133
CENTRONICS, 204
CGA, 250
chequeo de errores, 28
chipset, 299, 301
ciclos de bus, 289, 342
cilindro, 169
cintas, 172
circuito virtual, 229
CISC, 39
coherencia de cache , 16
ColdFire, 40

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

Commodore64, 42
CompactFlash, 181
Compaq, 42
compatibilidad, 42, 53, 61, 125, 130, 277, 301, 322, 332
compilador, 5
conector PCI, 219
conmutación de tareas, 112
Control de flujo, 198
controlador de cache, 11
controlador de E/S, 186
controlador de interrupciones, 24, 309
controlador de teclado, 285
Conversion analógico digital, 20
COP800, 324
Core, 66
Core Duo, 57
CP-M, 42
CPI, 241
CRC, 225
CRT, 246

DAC, 20
DAT, 172
dBase, 36
DCE, 197
DDR SDRAM, 152, 153
DDR2, 153
debug, 79
depurador, 194
descriptor, 98
Deskpro386, 42
diafonías, 27
diagnóstico, 130
DIMM, 158, 165
direccionamiento, 335
DirectX, 252
disco duro, 168, 173
display, 244
DisplayWriter, 42
disquete, 171
DLT, 172
DMA, 26, 189, 278, 283, 294, 309, 352
DOS, 278
DRAM, 8, 144
DRDRAM, 155
DSP, 324
DTE, 197
DVD, 178
DVD-Video, 178
DVI, 251

E/S, 20, 95, 185, 194
ECC, 160
ECP, 206
EDO DRAM, 147
EEPROM, 143
EGA, 250
EIA, 197
EIDE, 225
EISA, 209
ejecución dinámica, 61
ejecución especulativa, 61
EM64T, 57
Empaquetado, 158
ENIAC, 35
ensamblador, 5
entradas de página, 108
entrelazado, 247
Enumeración USB, 234
EOI, 188, 196
EPP, 206
EPROM, 143, 323

error de cuantificación, 21
escritura continua de cache, 11
espacio de direcciones, 3, 21
espacio lógico, 18, 73
Espacios de E/S, 95
espacios de E/S, 21
ethernet, 47

F8, 35
FAT, 169
FIFO, 165
Firewire, 209
Flash, 144, 323
flowthrough, 164
FM, 260
fotolitografía, 37, 50
FPM DRAM, 147
frecuencia de muestreo, 20

GART, 257
gestión de memoria, 18, 98
GPIB, 209
GPRS, 48
GPU, 252
GSM, 48
GUI, 45

Hercules, 250
Hypercard, 36
hyperpipeline, 65

I2C, 209, 235, 310, 323
I2S, 310
i960, 40
IA-32, 53
IBM PC, 41, 42, 278
IEEE 1284, 204
intérprete, 4
interfaz de Pentium 4, 131
interfaz de Pentium II, 125
interfaz del Pentium, 117
Interfaz USB, 230
interleaving, 17, 146
Internet, 1, 47
interrupción, 24, 94, 123, 194, 216, 334
IrDA, 310
ISA, 43
Itanium, 40, 132
Itanium II, 132

Jaz, 170
jerarquía de memoria, 9
Joystick, 242
JTAG, 218, 307

K6, 40

línea de cache, 10
LAN, 47
latencia, 34, 56, 61, 66, 145, 154, 170, 172, 309
latencia de rotación, 170
LBA, 225
LCD, 309, 311
lenguaje C, 5
lenguaje de alto nivel, 5
lenguaje de bajo nivel, 5
lenguaje de medio nivel, 5
lenguaje máquina, 4
little endian, 7, 72
LPC, 303

Macintosh, 45

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

MacPaint, 45
macrofusion, 66
MacWrite, 45
MCA, 209, 299
MCGA, 250
MCS-51, 324
MDA, 250
mediaprocessor, 40
memoria, 7, 139
memoria cache, 10
memoria no volátil, 8
memoria virtual, 18
memoria volátil, 8
memorias ópticas, 174
memorias de video, 157
memorias magnéticas, 168
memorias no volátiles, 143
memorias síncronas, 150
memorias semiconductoras, 140
MESI, 16
MessagePad, 43
mickey, 241
MicroChannel, 42
microcomputador, 2
microcontrolador, 321
microelectronica, 36
microfusion, 66
microprocesador, 2
microsistema, 29
Microsoft, 45
MIDI, 259, 270
MIPS, 39, 40
Mips, 40
MMX, 47, 56, 307
modem nulo, 198
modo idle, 335
modo protegido, 71
modo real, 71
Modos de transferencia de E/S, 22
modos de transmisión, 27
Moore, 36
MPEG, 47
MRAM, 181
MS-DOS, 278
Multibus, 278
multiproceso, 29, 55, 96, 124, 297
multipunto, 333
multitarea, 6, 18, 112
multithread, 41

NetBurst, 65
Newton, 43
Nintendo, 40
nivel de abstracción, 4
nivel de cuantificación, 20
nivel de privilegio, 109, 110
NMI, 282
NoBL, 164
norma de diseño, 37
NS32032, 39
NVRAM, 144
Nyquist, 21

OpenGL, 252
organización de memoria, 7
OS/2, 42
OSPM, 302
OTP, 143, 323
OUM, 181

página, 18
página de memoria, 17
PA HP, 39

PA8000, 40
paginación, 18, 55, 107
paquete USB, 231
paquetes USB, 232
paridad, 27, 282
PBSRAM, 164
PC, 1
PC-386, 299
PC-AT, 42, 238, 292
PC-Pentium, 301
PC-XT, 42, 238, 292
PCI, 43, 209, 211, 306
PCI Express, 43, 210, 221
PDA, 43, 44
Pentium, 40, 55, 58
Pentium 4, 56, 131, 304
Pentium Extreme Edition, 57
Pentium II, 40, 56, 125
Pentium III, 56
Pentium M, 57
Pentium MMX, 56
Pentium Pro, 56
PET, 35, 42
PIC16C, 324, 325
PIO, 223
pipeline, 31, 55, 147
PlayStation, 40, 133
plug&play., 211, 225, 227
polling, 23
postescritura de cache, 11
Power3, 40
powerdown, 335
PowerPC, 40, 133
PowerPC 620, 40
programación, 4
programación de DMA, 196
programando una RSI, 194
PROM, 143
Protección, 109
protección, 18
PS/2, 42, 209, 238, 299
PSDRAM, 157
puerta, 104
puerto de entrada, 23
puerto de salida, 23
puerto paralelo, 204
puerto serie, 197
PWM, 309, 323
PXA27x, 307

QDR, 165
QIC, 172

R10000, 40
R12000, 40
R2000, 39
ráfaga, 122
RAID, 172, 226
Rambus, 155
RAMDAC, 250
Ratón, 239
RDRAM, 155
Reed-Solomon, 175
refresco, 247
registros bit-direccionables, 328
resolución, 245
RFID, 48
RISC, 39
ROM, 143
RS-232, 197, 241
RSI, 195

SATA, 225, 306

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

SBSRAM, 164
SCSI, 223
SDRAM, 150
sector, 169, 175, 176
segmentación, 18, 31, 98
segmento, 18
selector, 98
SFR, 328
SGRAM, 157
SIMD, 47, 56, 84
SIMM, 158
Sistema Operativo, 5
SMbus, 303, 305
SMS, 48
snooping, 16
SOC, 40, 44, 307
SoundBlaster16, 259
SPARC, 39
SPARCstation, 42
SPD, 158, 303
SPI, 310
SRAM, 8, 161
SSE, 56, 85
SSE2, 57, 85
SSE3, 57
SSP, 310
SSTL-2, 153
stack pointer, 328
StrongARM, 40
superescalar, 41, 59
SuperH, 40
superpipeline, 61
SVGA, 251

tablas de ondas, 260
tarea, 6
Tarjetas adaptadoras, 248
TCP/IP, 47
TDM, 266
teclado, 237
telealimentado, 227
TFT, 248
TI9900, 39
tiempo de acceso, 177
tiempo de búsqueda, 169
tiempo de precarga, 17
TLB, 59
TMDS, 252
TMS-1000, 35
traductor, 4
transacción, 125
transferencia bulk, 230
transferencia de interrupción, 230
transferencia directa, 22
transferencia isócrona, 230
transferencia USB, 230
transferencias PCI, 212
transferencias USB, 230
transmisión asíncrona, 27
transmisión paralelo, 27
transmisión síncrona, 28
transmisión serie, 27

UART, 310, 323, 333
UCM, 2
UCP, 2
UES, 2
UltraJava, 40
UltraSparc, 40
UltraSparcIII, 40
UMTS, 48
unidad aritmético lógica, 2
Universal Serial Bus, 227
USART, 323
USB, 44, 171, 227, 305, 323
USIM, 310

vector de interrupción, 194
velocidad lineal, 176
VESA, 247, 251
VGA, 250
VIC-20, 42
videojuegos, 41
vigilancia del bus, 16
VIS, 47
Visicalc, 36
Visual C++, 36
VL-bus, 209
VLIW, 132
VME, 209
VRAM, 157

WDT, 324
WiFi, 48
Windows, 45
WordStar, 36
WRAM, 157
write back, 11
write buffering, 11
write through, 11
WUSB, 227
WWW, 47

Xerox Alto, 45
Xerox PARC, 45
Xmodem, 198
Xon/Xoff, 198
XRAM, 157
XScale, 307

Z80, 35
Z8000, 39
Z80000, 39
Z86C94, 324
ZBT, 164
ZIP, 170
Zmodem, 198

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD DE MÁLAGA
Publicaciones y Divulgación Científica

UNIVERSIDAD DE MÁLAGA
Publicaciones y Divulgación Científica

UNIVERSIDAD DE MÁLAGA
Publicaciones y Divulgación Científica

Hoy en día estamos inundados de cientos de sistemas digitales que juegan un papel discreto pero agradecido en muchas de nuestras actividades diarias. No hace muchos años, no sabíamos lo que era un mando a distancia y, hoy en día, no sabemos vivir sin él. Muchos productos de consumo basados en sistemas digitales son ya artículos de casi primera necesidad.

La revolución digital, basada en la microelectrónica, la informática y las telecomunicaciones, comenzó en los años setenta con la aparición del microprocesador. En los 80, los microprocesadores facilitaron la llegada de los ordenadores personales (PC, Personal Computer) en un momento en el que nadie podía sospechar el éxito que alcanzarían. En los 90, triunfó el PC multimedia y su conexión a red junto con el auge de la telefonía móvil de segunda generación. La llegada del siglo XXI ha consolidado Internet y ha puesto las bases para el despliegue de la tercera generación de móviles. Los siguientes pasos apuntan entornos inteligentes donde terminales multimedia inalámbricos se comunicarán de forma espontánea con dispositivos próximos y podrán acceder a cualquier red de comunicaciones.

Este manual pretende repasar esta revolución estudiando, desde un punto de vista hardware, los sistemas digitales basados en microprocesadores de última generación. Consta de 7 capítulos que cubren conceptos básicos, microprocesadores (IA-32 de Intel), memorias, interfaces a periféricos (PCI, USB, ATA, teclados, ratones, sistemas gráficos, AC'97, etc.), microcomputadores (PCs y PDAs), microcontroladores (8051 de Intel y ADuC842 de Analog Devices) y recomendaciones de diseño. Todos los capítulos incluyen ejemplos que permiten asimilar conceptos y acercar la teoría a la práctica. Además se proporciona una amplia bibliografía y una guía de estudio para cada capítulo.

ISBN 978-84-9747-176-3

UNIVERSIDAD
DE MÁLAGA

SERVICIO de PUBLICACIONES
Universidad de Málaga