Ampliar, configurar y reparar su PC

Luis Durán Rodríguez

- La placa base
- Procesadores
- Todos los periféricos
- Ampliar y modificar
- Redes
- El BIOS y el Setup
- Sistemas operativos

Incluye CR-ROM

marcombo
ediciones técnicas

Ampliar, configurar y reparar su PC

Luis Duran Rodriguez

Ampliar, configurar y reparar su PC

Título: **Ampliar, configurar y reparar su PC**

Autor:
Luis Durán Rodríguez

© Reservados todos los derechos de publicación, reproducción, préstamo, alquiler o cualquier otra forma de cesión del uso de este ejemplar de la presente edición española por
MARCOMBO S. A. 2006
Gran Via de les Corts Catalanes, 594
08007 Barcelona

Queda prohibida, salvo excepción prevista en la ley, cualquier forma de reproducción, distribución, comunicación pública y transformación de esta obra sin contar con autorización de los titulares de la propiedad intelectual.
La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual (arts. 270 y siguientes, Código Penal). El Centro Español de Derechos Reprográficos (www.cedro.org) vela por el respeto de los citados derechos.

Consulte nuestro fondo editorial
http://www.marcombo.com

Pueden existir materiales adicionales a este libro atojados en la URL:
http://www.marcombo.com/descargas

El autor de la obra alojará en esta dirección materiales adicionales que considere de interés para sus lectores, y los irá actualizando periódicamente.

ISBN: 84-267-1413-7
EAN13: 978-84-267-1413-8
Depósito legal:
Impreso en Gráficas Díaz Tuduri, S.L.
Printed in Spain

Índice analítico

1. Introducción 1
1.1 El sistema binario 2
1.2 El funcionamiento 3
1.3 La arquitectura 5
1.4 Los componentes 6
1.5 La mejor configuración 9
1.5.1 La placa base 9
1.5.2 Memoria RAM 10
1.5.3 Procesador 10
1.5.4 Disco duro 11
1.5.5 Tarjeta gráfica 11
1.5.6 Tarjeta de sonido 11
1.6 Ampliaciones y actualizaciones 12
1.6.1 La relación precio-rendimiento 12
1.7 Las redes 13

2. Un poco de historia 17
2.1 La fabricación de los procesadores 19
2.1.1 Los transistores 21
2.1.2 La fabricación 22
2.2 El origen de los ordenadores personales 27
2.2.1 La década de los años 80 29

3. Procesadores 33
3.1 Características básicas 33
3.1.1 El coprocesador aritmético 34
3.1.2 La memoria caché 34
3.1.3 Procesadores RISC y procesadores CISC 36
3.1.4 Front End y Back End 37
3.2 Del 8086 al Pentium 38
3.2.1 El 8088 38

3.2.2 El 286 41
3.2.3 El 386 43
3.2.4 El 386SX 45
3.2.5 Los recursos del modo protegido 46
3.2.6 El 486 50
3.2.7 El Pentium 54
3.2.8 Otros procesadores 60
3.3 Del Pentium MMX al Pentium II 64
3.3.1 El Pentium MMX 65
3.3.2 El K6 de la casa AMD 67
3.3.3 La evolución de los procesadores 68
3.3.4 AMD K6-2 70
3.3.5 MMX vs 3D-Now! 72
3.3.6 AMD K6-3 72
3.3.7 Cyrix 6x86MX 74
3.3.8 Cyrix Media GX 75
3.3.9 IDT C6 75
3.3.10 Pentium Pro 76
3.3.11 El Pentium II 81
3.3.12 El Celeron 84
3.3.13 Pentium II Xeon 86
3.4 Hasta el primer Pentium 4 87
3.4.1 El problema de los encajes 88
3.4.2 Los siguientes procesadores 93
3.4.3 El Pentium III 94
3.4.4 Pentium III Xeon 99
3.4.5 Celeron Coppermine (Celeron II) 100
3.4.6 Cyrix/Via C3 101
3.4.7 AMD Athlon (K7) 102
3.4.8 Athlon Thunderbird 112
3.4.9 AMD Duron 113
3.4.10 Pentium 4 115
3.5 Los nuevos procesadores 121
3.5.1 Las nuevas evoluciones 121
3.5.2 Intel vs AMD 121
3.5.3 Spitfire vs Morgan 123
3.5.4 Thunderbird vs Palomino 124

3.5.5 Athlon XP 126
3.5.6 Celeron Tualatin 129
3.5.7 Intel Itanium 129
3.5.8 AMD Hammer. 132
3.5.9 La famosa ley de Moore. 135
3.6 El futuro 136
3.6.1 La era de los transistores 136
3.6.2 La promesa de los procesadores cuánticos 137
3.6.3 Nanotubos 138

4. La memoria RAM 141
4.1 El funcionamiento 143
4.2 El acceso a los datos 144
4.3 Formato 144
4.3.1 Los módulos DIP 145
4.3.2 Los módulos SIMM de 30 vías 145
4.3.3 Los módulos SIMM de 72 vías 146
4.3.4 Los módulos DIMM de 168 vías 147
4.4 Los bancos de memoria 147
4.5 Tecnologías 148
4.5.1 Las memorias regulares 149
4.5.2 Las memorias FPM 149
4.5.3 Las memorias EDO 150
4.5.4 Las memorias BEDO 150
4.5.5 Las memorias SDRAM 151
4.5.6 Las memorias PC-100 152
4.5.7 Las memorias PC-133 152
4.6 Las nuevas tecnologías 153
4.6.1 Las memorias DDR-SDRAM 153
4.6.2 Las memorias Rambus 154
4.6.3 Memorias DDR vs memorias Rambus 155
4.6.4 Cómo funcionan las memorias DDR 156
4.6.5 La necesidad de las memorias DDR 157
4.6.6 Como funcionan las memorias Rambus 158
4.7 ECC y paridad 159

5. La placa base 161
5.1 Los formatos de placas base 161
5.1.1 Variaciones del formato ATX 162
5.2 La fuente de alimentación 163
5.3 Los componentes de la placa base 164
5.3.1 Chipset 165
5.3.2 BIOS 166
5.3.3 Memoria caché L2 167
5.3.4 Encajes para los módulos de memoria 168
5.3.5 Interfaces de disco 169
5.3.6 Puertos paralelo y serie 170
5.3.7 Conector del teclado 171
5.3.8 Jumpers 172
5.3.9 Conectores para el panel de la torre 172
5.3.10 Seguimiento de funciones 173
5.4 Buses 174
5.4.1 ISA 174
5.4.2 ISA de 16 bits 175
5.4.3 MCA 175
5.4.4 EISA 176
5.4.5 VLB 176
5.4.6 PCI 177
5.4.7 AGP 180
5.4.8 AGP Pro 181
5.4.9 USB 181
5.4.10 PCMCIA 182
5.4.11 Slots AMR 184
5.4.12 ACR 185
5.4.13 IEEE 1394 (Fireware) 185
5.4.14 Serial ATA 187
5.5 El presente de los buses 187
5.6 La petición de interrupción (IRQ) 188
5.6.1 Conseguir más IRQs 193
5.6.2 PCI IRQ Activated By 194
5.7 DMA (Acceso directo a memoria) 195
5.8 Los componentes integrados 196
5.9 Entendiendo los nombres y códigos 196

5.9.1 El procesador 197
5.9.2 El chipset 198
5.9.3 Otros recursos adicionales 200
5.10 El suministro de energía 201
5.10.1 SAIs y estabilizadores 202
5.11 La torre 204

6. Los discos duros 205
6.1 El funcionamiento 206
6.1.1 La tarjeta controladora 209
6.1.2 El diámetro de los platters 209
6.1.3 Materiales utilizados 210
6.1.4 Pistas, sectores y cilindros 210
6.1.5 Zoned Bit Recording (ZBR) 211
6.1.6 Densidad 212
6.1.7 Lectura y escritura 213
6.1.8 Formateo 214
6.2 El sistema de archivos 215
6.2.1 FAT16 216
6.2.2 FAT32 218
6.2.3 Convertir unidades de FAT16 a FAT32 219
6.2.4 Estructuras lógicas en los sistemas FAT 219
6.2.5 VFAT 223
6.2.6 Problemas con la FAT 224
6.2.7 FAT12 225
6.2.8 NTFS 225
6.2.9 NTFS 5 227
6.2.10 Estructuras lógicas en el sistema NTFS 228
6.2.11 Sistema EXT2 231
6.2.12 VFS 231
6.2.13 Soporte al NTFS 232
6.2.14 Estructuras del EXT2 232
6.2.15 EXT3 233
6.2.16 HPFS 233
6.3 Sectores Defectuosos 234
6.4 Las particiones 235
6.5 La recuperación de datos 236

6.5.1 Los defectos más comunes 236
6.5.2 Y ahora, qué? 237
6.6 Eliminar datos con seguridad 240
6.7 Comprimir archivos 240
6.8 Interfaces 241
6.8.1 IDE 242
6.8.2 RAID en discos IDE 245
6.8.3 SCSI 247
6.8.4 RAID con discos duros SCSI 250
6.9 Evolución 254
6.9.1 Las barreras de los 504 MB y los 8 GB 255
6.9.2 Dynamic Drive Overlays (DDO) 257
6.9.3 Detección y corrección de errores 257
6.10 Rendimiento 258
6.10.1 Tiempo de búsqueda (Seek Time) 259
6.10.2 Tiempo de latencia (Latency Time) 259
6.10.3 Tiempo de acceso (Access Time) 260
6.10.4 Head Switch Time 260
6.10.5 Tasa de transferencia (Internal Transfer Rate) 260
6.10.6 Caché (Buffer) 261
6.10.7 Densidad 262
6.10.8 Velocidad de la interfaz 262
6.11 Pio 4 vs UDMA 33/66 vs UDMA 100/133 263
6.12 La unidad de disquetes 265

7. CD-ROM, CD-R y DVD 267
7.1 El medio físico 268
7.1.1 El proceso de grabación 268
7.1.2 El formato 269
7.1.3 Los distintos formatos 271
7.2 Rendimiento 272
7.2.1 Modo de lectura 272
7.2.2 Tiempo de acceso 273
7.2.3 Caché 273
7.2.4 Soporte físico 273
7.2.5 La importancia del rendimiento 274
7.2.6 Por encima del límite 274

7.3 La grabación de CDs 275
7.3.1 El soporte físico 275
7.3.2 Colores 276
7.3.3 Audio vs datos 277
7.3.4 Durabilidad 278
7.3.5 CD-RW 278
7.3.6 CDs de 80 minutos 280
7.3.7 Oversize 281
7.3.8 Buffer Underrun 282
7.4 DVD o Digital Versatile Disc 285
7.4.1 El principio de funcionamiento 285
7.4.2 Tipos 287
7.4.3 Vídeo y audio DVD 288
7.4.4 DVD-R, DVD+R, DVD-RW y DVD+RW 290

8. La tarjeta gráfica 293
8.1 2D vs 3D 294
8.1.1 La memoria 296
8.1.2 La frecuencia de actualización del monitor 297
8.1.3 Las ventajas 298
8.2 Los recursos de las tarjetas 3D 300
8.2.1 Efectos básicos 300
8.2.2 Recursos avanzados 303
8.3 Conceptos generales 307
8.3.1 La división de las tareas 307
8.3.2 Frame-rate y rendimiento 309
8.3.3 Los drivers 310
8.3.4 Las APIs 310
8.3.5 AGP: ser o no ser 312
8.3.6 Uso de la memoria 313
8.3.7 Chipsets 313
8.4 Rendimiento 313

9. El sonido 317
9.1 La generación del sonido digital 318
9.1.1 La compresión 318

9.1.2 La extracción de audio 320
9.1.3 Generando archivos menores 321
9.1.4 Síntesis de audio 322
9.1.5 Síntesis por FM 322
9.1.6 MIDI 323
9.1.7 Síntesis por Wave Table 323
9.1.8 Los conectores externos 324
9.2 La familia Sound Blaster 326
9.3 Las tarjetas de sonido PCI 329
9.4 Tarjetas de sonido 3D 330
9.4.1 Otros recursos 333
9.4.2 Monster Sound 334
9.4.3 Creative Sound Blaster Live 336
9.4.4 Turtle Beach Montego 337
9.4.5 Aureal SQ1500 y SQ2500 338
9.4.6 La sucesora de la SB Live 339

10. Los monitores 341
10.1 Una visión global 341
10.2 Los monitores LCD 344
10.2.1 Las ventajas 344
10.2.2 Las desventajas 345
10.2.3 Cómo funciona el LCD 346
10.2.4 Los monitores Touch Screen 347
10.3 Nuevas tecnologías 348
10.3.1 Los monitores de plasma 348
10.3.2 Los monitores FED 349
10.3.3 Los monitores OLED 351
10.4 El uso de dos monitores 352
10.4.1 Vídeo primario y secundario 354
10.4.2 Limitaciones 354
10.4.3 Interferencias 355

11. El teclado 357
11.1 Principio de funcionamiento 359
11.2 Configuración y solución de errores 361

12. El ratón 365
12.1 Principio de funcionamiento 366
12.2 Configuración y solución de errores 367
12.3 Los ratones ópticos 370

13. Las impresoras 375
13.1 Impresoras de margarita 375
13.2 Impresoras de agujas 376
13.2.1 Tipos y mantenimiento 377
13.3 Impresoras de inyección de tinta o chorro 380
13.3.1 Principio de funcionamiento: bubble jet 381
13.3.2 Funcionamiento impresoras piezoeléctricas 382
13.3.3 Tipos y cartuchos 382
13.3.4 Impresión y mantenimiento 383
13.4 Impresoras láser 385
13.4.1 La fotoimpresión 388
13.4.2 Resolution Enhancement Technology 388
13.4.3 Tipos y mantenimiento 389
13.5 Impresoras láser en color 391
13.6 Impresoras térmicas 394
13.7 Configuración 395

14. Módems 399
14.1 Los módems de 56K 400
14.1.1 Compresión de datos 401
14.1.2 Actualización del módem 401
14.1.3 La corrección de errores 402
14.1.4 Los protocolos 403
14.1.5 Softmodems vs hardmodems 404
14.1.6 Los componentes 404
14.1.7 Rendimiento 406
14.1.8 Instalación 407
14.1.9 A simple vista 407
14.1.10 Mantenimiento 407
14.1.11 Enviar un fax 408
14.2 ADSL 408

14.2.1 Las limitaciones de la ADSL 409
14.2.2 Compartir la conexión ADSL 410
14.3 RDSI (ISDN) 411
14.4 Otras opciones de acceso 411

15. Redes 413
15.1 Estructuras y estándares de red 414
15.1.1 La topología de estrella 414
15.1.2 La topología de anillo 415
15.1.3 La topología de bus 415
15.1.4 La topología de árbol 416
15.1.5 Token Passing 416
15.1.6 100BaseVG-Any-LAN 417
15.1.7 Asynchron Transfer Mode 418
15.1.8 Los estándares LAN más importantes 418
15.2 Instalación de redes 420
15.2.1 Conexiones de cable coaxial 421
15.2.2 Conexiones de par trenzado 422
15.2.3 Conexiones de fibra óptica 425
15.3 Ethernet 428
15.3.1 Principio de funcionamiento 428
15.3.2 Fast Ethernet y Gigabit Ethernet 432
15.4 Hubs y Switches 434
15.4.1 Hub o Switch? 436
15.4.2 Administración de red 438
15.4.3 Otros dispositivos de interconexión 440

16. Ampliar y modificar 445
16.1 Medidas de seguridad 445
16.2 El montaje de las tarjetas de expansión 445
16.2.1 El sistema de bus 446
16.2.2 Montaje de la tarjeta 446
16.2.3 Problemas mecánicos al montar una tarjeta 448
16.3 La ampliación de la memoria 450
16.3.1 La dotación correcta de memoria 451
16.3.2 El montaje de los módulos 451

16.3.3 Solucionar problemas de memoria 454
16.4 Instalación de unidades 455
16.4.1 Correctamente montado y conectado 457
16.4.2 Instalar discos duros y el sistema de ventilación 460
16.5 Instalación de la placa base 462
16.5.1 Desmontar la placa base 463
16.5.2 Sujeción de la placa base 464
16.5.3 Montaje de la placa base 466
16.6 El montaje del procesador 468
16.7 Los indicadores y elementos de control 473
16.8 La conexiones de los distintos puertos 474
16.9 Resumen de los distintos pasos a seguir 475

17. El BIOS y el Setup 479
17.1 CMOS 480
17.2 POST 480
17.3 Actualización del BIOS 481
17.4 Configurar el CMOS Setup 481
17.4.1 BIOS Features Setup 482
17.4.2 Chipset Features Setup 488
17.4.3 Power Management Setup 492
17.4.4 Plug & Play y PCI Setup 494
17.4.5 On Board Devices, Integrated Peripherals 502
17.4.6 Configuraciones del procesador o CPU 505
17.5 El proceso de actualización del BIOS 508
17.5.1 Chips de BIOS 509
17.5.2 La programación 510

18. Sistemas operativos 517
18.1 Nociones básicas de hardware 517
18.1.1 Los registros del procesador 517
18.1.2 La ejecución de las instrucciones 518
18.1.3 Las interrupciones 518
18.2 Las funciones de un sistema operativo 518
18.3 Conceptos básicos 519
18.3.1 Los procesos 519

18.3.2 Los archivos 521
18.3.3 Las llamadas 521
18.3.4 El núcleo (Kernel) 521
18.3.5 El intérprete de comandos (shell) 522
18.4 Tipos de sistemas operativos 522
18.4.1 Sistemas por la estructura 522
18.4.2 Sistemas por los servicios 524
18.4.3 Sistemas por la forma de ofrecer los servicios 526
18.5 La gestión de los procesos 528
18.5.1 Los ciclos de vida 528
18.5.2 Concurrencia 530
18.6 Comparativa de sistemas 533
18.6.1 Los sistemas operativos 534
18.7 Comparación a nivel administrativo 536
18.7.1 Licencia 536
18.7.2 Estabilidad y rendimiento 538
18.7.3 Facilidad de uso 540
18.7.4 Soporte 542
18.8 Comparación a nivel técnico 543
18.8.1 Compatibilidad con otras plataformas 544
18.8.2 Portabilidad 544
18.8.3 Requerimientos de hardware 544
18.9 El sistema operativo GNU/Linux 544
18.9.1 GNU/Linux 545
18.9.2 Distribuciones GNU/Linux 545
18.10 Windows XP 548
18.10.1 Introducción 548
18.10.2 Requisitos de hardware 550
18.10.3 La compatibilidad de hardware 551
18.10.4 Actualización del equipo 553
18.10.5 La instalación de Windows XP 555
18.10.6 El Panel de control 556
18.10.7 La actualización de hardware en Windows XP 586

Indice alfabético en el CD-ROM

1. Introducción

Un ordenador puede llegar a parecer una máquina misteriosa, una "caja negra" donde se guardan y procesan datos. Pero los ordenadores no tienen nada de misterioso. Todo funciona de una manera muy ordenada y hasta cierto punto simple de entender.

Hoy en día, casi todo el mundo tiene un PC en casa, aprende muy rápido a usar *Windows* o incluso *Linux*, navegar por *Internet*, etc. Pero, de entre todos estos usuarios, realmente pocos entienden cómo funciona el ordenador. Qué diferencias existen entre un procesador *Pentium 4* y un *Athlon XP*? Por qué un PC con poca memoria *RAM* es más lento? Cómo funciona un disco duro? En qué tareas son necesarias las tarjetas gráficas 3D? Cuál es la diferencia entre una tarjeta de sonido "genérica" y otra más cara? En definitiva, cómo funciona un PC?

Aspecto típico de un ordenador de sobremesa

En esta introducción daremos una visión general de los distintos componentes que conforman un ordenador actual y su funcionamiento general. También encontraremos varias explicaciones sobre los términos y convenciones más usados.

El objetivo es dar una visión inicial para poder estudiar los conceptos más avanzados en los siguientes capítulos.

1.1 El sistema binario

Existen dos formas distintas de representar una misma información: analógica o digital. Como ejemplo, la música es grabada en una cinta de forma analógica, codificada como una gran onda de señales magnéticas que puede asumir un número ilimitado de frecuencias. Un sonido grave sería representado por un punto más bajo de la onda, mientras que un punto más alto representaría un sonido agudo.

El problema con esta representación es que cualquier interferencia causa distorsiones en el sonido. Si los ordenadores trabajaran con datos analógicos serían demasiado sensibles a los errores, pues cualquier interferencia, por mínima que fuese, causaría graves alteraciones en los datos procesados y, consecuentemente, en los resultados.

Por su parte, el sistema digital permite almacenar cualquier información en una secuencia de valores positivos y negativos o sea, en forma de unos y ceros. El número 181 por ejemplo, es representado digitalmente como 10110101. Cualquier tipo de dato, ya sea un texto, una imagen, un vídeo, un programa, o cualquier otra cosa, es procesado y almacenado como una gran secuencia de unos y ceros.

Es justamente el uso del sistema binario lo que hace que los ordenadores sean realmente fiables, pues la posibilidad de que un valor 1 sea alterado hacia un valor 0, o al contrario, es realmente muy pequeña. Trabajando con sólo dos valores diferentes, la velocidad del procesamiento también es mucho mayor, debido a la mayor simplicidad de los cálculos.

Cada valor binario es llamado *bit*, contracción de la expresión *binary digit* (dígito binario). Un conjunto de 8 bits forman un *Byte*, y un conjunto de 1.024 Bytes forman un *KiloByte* (KByte o KB). El número 1.024 fue escogido pues es la potencia de 2 más próxima a 1000. Un conjunto de 1.024 KBytes forman un *MegaByte* (1.048.576 Bytes) y un conjunto de 1.024 MBytes forma un *GigaByte* (1.073.741.824 Bytes). Los próximos múltiplos son el *TeraByte* (1.024 GBytes) y el *PetaByte* (1.024 TBytes).

También usamos los términos *Kbit*, *Megabit* y *Gigabit*, para representar conjuntos y potencias de 1.024 bits. Dado que un Byte corresponde a 8 bits, un MByte se corresponde con 8 Mbits y así en los demás casos.

Datos	Descripción
1 bit	1 o 0
1 Byte	Conjunto de 8 bits
1 KByte	1.024 Bytes (8.192 bits)
1 MByte	1.024 KBytes, 1.048.576 Bytes (8.388.608 bits)
1 GByte	1.024 MBytes, 1.048.576 KBytes, 1.073.741.824 Bytes (8.589.934.592 bits)

Cuando vamos a abreviar, también existen diferencias para expresar las unidades. Cuando estamos hablando de *KiloBytes* o *MegaBytes*, abreviamos respectivamente como

KBytes (KB) y MBytes (MB), siempre con una "B" mayúscula. Cuando estamos hablando de *Kilobits* o *Megabits* abreviamos de la misma forma, pero usando la "b" minúscula, Kbits (Kb) y Mbits (Mb).

A simple vista puede parecer un dato irrelevante, pero llega a ser la fuente de muchas confusiones. Siempre que nos referimos a la velocidad de una red de ordenadores la medimos en bits por segundo: 10 Mbits/s, 100 Mbits/s, etc.

1.2 El funcionamiento

La arquitectura básica de cualquier ordenador completo está formada por sólo 5 componentes básicos: procesador, memoria *RAM*, disco duro, dispositivos de entrada y salida y software (sistema operativo y programas).

El procesador es el cerebro del sistema, es el encargado de procesar todos los datos e informaciones. Sin embargo, a pesar de toda su sofisticación, el procesador no puede hacer nada por sí solo. Para disponer de un ordenador funcional, necesitamos algunos componentes más: memoria *RAM*, unidades de disco, dispositivos de entrada y salida y finalmente, los programas.

La memoria principal, conocida como memoria *RAM*, es usada por el procesador para almacenar los datos que se están procesando, funcionando como una especie de mesa de trabajo. La cantidad de memoria *RAM* disponible determina las actividades que podrá ejecutar el procesador. Si la cantidad de memoria *RAM* disponible es insuficiente, el ordenador no será capaz de ejecutar aplicaciones muy complejas.

El PC IBM original, lanzado en el año 1981, por ejemplo, sólo poseía 64 KBytes de memoria *RAM* y, por eso, sólo era capaz de ejecutar programas muy simples, basados en texto. Un PC actual posee mucha más memoria *RAM*: 256 MB, 512 MB, o incluso más, y por eso es capaz de ejecutar programas realmente complejos.

La memoria *RAM* es capaz de responder a las solicitudes del procesador a una velocidad muy elevada. Sería perfecta si no existiesen dos problemas principales: el alto precio y el hecho de ser volátil, o sea, de perder todos los datos grabados cuando se corta el suministro de corriente.

Dado que la memoria *RAM* sólo sirve como borrador para el trabajo, usamos otro tipo de memoria para guardar los archivos y los programas: la *memoria de masa*. El principal dispositivo de memoria de masa es el disco duro, en el que quedan guardados los programas y los datos mientras no están en uso o cuando desconectamos el ordenador de la corriente eléctrica. Los disquetes y *CDs/DVDs* también son unos buenos representantes de esta categoría de memorias de masa.

Los sistemas operativos actuales le permiten al procesador usar parte del disco duro para grabar datos en el caso de que se agote la memoria *RAM*, recurso conocido como memoria virtual. Utilizando la memoria virtual, aunque la memoria *RAM* esté com-

pletamente ocupada, el programa se ejecutará igualmente. Sin embargo, la ejecución será más lenta, debido a la mayor lentitud del disco duro respecto a la memoria RAM.

Para permitir la comunicación entre el procesador y los demás componentes del ordenador, así como entre el ordenador y el usuario, disponemos de los dispositivos de *I/O* (*Input/Output*) o *E/S* (*Entrada/Salida*). Estos dispositivos son el sitio por donde el ordenador recibe y transmite la información. Existen dos categorías de dispositivos de entrada y salida:

- La primera categoría está formada por los dispositivos destinados a la comunicación entre el usuario y el ordenador. En esta categoría podemos encontrar el teclado, ratón, micrófono, etc. (para la entrada de datos) y el monitor, impresoras, altavoces, etc. (para la salida).
- La segunda categoría es la destinada a la intercomunicación entre el procesador y los demás componentes internos del ordenador, como la memoria *RAM* y el disco duro. Los dispositivos que forman parte de esta categoría están situados en la placa base.

Toda la parte física del ordenador (procesadores, memoria, discos duros, monitores, etc.) es conocida como *hardware*, mientras que los programas son conocidos como *software*.

Existen dos tipos de programas, llamados software de alto nivel y software de bajo nivel. Estas designaciones no indican el grado de sofisticación de los programas, pero sí su grado de implicación con el hardware.

El procesador sólo es capaz de entender el *lenguaje máquina*, instrucciones relativamente simples, que ordenan la ejecución de operaciones matemáticas, además de algunas otras tareas, como la lectura y escritura de datos, comparación y otras operaciones lógicas, etc. Al ser extremadamente difícil hacer que el procesador ejecute las órdenes escribiendo los programas en lenguaje máquina, se usan pequeños programas, como el *BIOS* y los controladores (*drivers*) de los dispositivos para ejecutar las tareas más básicas, funcionando cómo intermediarios, o intérpretes, entre los demás programas y el hardware. Estos programas son conocidos como *programas de bajo nivel*. Todas las demás aplicaciones, como los procesadores de texto, plantillas, juegos, etc. trabajan encima de estos programas residentes, sin tener la necesidad de interconectar directamente con el hardware. Es por eso que son conocidos como *programas de alto nivel*.

Debido a esta división, muchas veces un nuevo dispositivo, por ejemplo una tarjeta de sonido que acabamos de encajar en la placa base, no funciona hasta que no se instalan los *drivers* que vienen en el CD que acompaña a la tarjeta. El sistema operativo es capaz de percibir la presencia de la nueva tarjeta, pero para usarla necesita del *driver* adecuado para aprovechar todos sus recursos. El *driver* trabaja como una especie de intérprete, que transforma los comandos usados por el sistema operativo en los comandos entendidos por la tarjeta y viceversa.

El propio sistema operativo *Windows* posee una gran biblioteca de *drivers* que permiten instalar de forma automática muchos componentes de hardware, pero muchos dispositivos, principalmente los más recientes lanzados al mercado después de la versión de *Windows* que estemos usando, no funcionarán de la forma más adecuada hasta que no instalemos los *drivers* correspondientes. Sin estos *drivers* es imposible hacer funcionar cualquier dispositivo de hardware. Por suerte, hoy en día, es posible encontrar los *drivers* adecuados para cualquier tipo de dispositivo, por antiguo que sea, a través de las páginas *Web* de los distintos fabricantes.

Para instalar un nuevo dispositivo, el procedimiento básico es siempre el mismo. Después de realizar la instalación física del dispositivo y conectar el PC, el sistema operativo (*Windows*) nos mostrará un aviso de que ha encontrado un nuevo dispositivo de hardware, pidiéndonos, de forma inmediata, los *drivers* más adecuados. Escogemos la opción de buscar el *driver* para el dispositivo y le indicamos la localización de los archivos, ya sea una carpeta del *CD-ROM*, una carpeta del disco duro, un disquete, etc. Si hemos indicado los archivos correctos, el sistema operativo los instalará y el dispositivo funcionará correctamente. Debemos tener presente que existen *drivers* específicos para cada sistema operativo.

Continuando la explicación sobre el software no podemos pasar por alto al propio sistema operativo, que funciona como un intermediario entre el hardware y el usuario, automatizando el uso del ordenador y ofreciéndonos una base sólida a partir de la cual los programas pueden ser ejecutados. La interfaz de los distintos sistemas operativos también es diferente. En *MS-DOS*, por ejemplo, sólo teníamos un *prompt* de comando basado en texto, mientras que en las distintas versiones de *Windows* tenemos una interfaz gráfica basada en un sistema de ventanas.

1.3 La arquitectura

En los inicios de la informática, en las décadas de los años 50, 60 y 70, varios fabricantes se disputaban el mercado. Cada uno desarrollaba sus propios ordenadores, que eran incompatibles entre sí tanto a nivel de hardware como a nivel de software.

A pesar de que ejecutaban las mismas operaciones básicas, prácticamente todo era distinto: los componentes de uno no servían en el otro, los programas eran incompatibles e, incluso, los lenguajes de programación eran distintos. Sin embargo, con la popularización de los ordenadores personales era inevitable la existencia de unos patrones o formatos básicos. En los inicios de la década de los años 80, disponíamos sólo de dos arquitecturas, o familias, distintas de ordenadores personales diferentes: el PC, desarrollado por la casa IBM, y el Macintosh, desarrollado por la casa Apple.

Al ser más barato, el PC se hizo más popular, quedando restringido el uso de los Macintosh a la edición de imágenes o sonidos y a la edición electrónica, situaciones en las que era mucho más eficiente.

Durante mucho tiempo, la propia casa Apple pareció conformarse con esta posición, lanzando ordenadores orientados principalmente al sector profesional. Actualmente, existe una mayor aceptación de los Mac, principalmente debido al lanzamiento del *iMac*, pero los PC todavía representan la gran mayoría de los ordenadores personales.

Dado que los PC poseen una arquitectura abierta, o sea, la posibilidad de que varios fabricantes distintos desarrollen sus propios componentes basados en patrones ya definidos, disponemos de una lista enorme de componentes compatibles entre sí. Podemos escoger entre varias marcas y modelos para los componentes que mejor se adapten a nuestras necesidades y conseguir nuestra propia configuración. También es posible mejorar posteriormente el ordenador montado a través de actualizaciones y ampliaciones, cambiando y añadiendo algunos componentes para mejorar su rendimiento general.

Los ordenadores de marca, como los IBM, Compaq, Dell, etc. también son ordenadores montados, ya que casi todos sus componentes son comprados a otros fabricantes. La diferencia principal entre los ordenadores montados (clónicos) y los ordenadores de marca es que estos últimos son montados por grandes empresas y disponemos de todo el soporte técnico y la garantía. Sin embargo, adquiriendo un ordenador de marca, siempre pagamos más caro y, al mismo tiempo, no tenemos tanta libertad para configurar, posteriormente, el ordenador a nuestro gusto.

1.4 Los componentes

Ahora que ya hemos visto, de forma superficial, lo que pasa en el interior de un PC, pasaremos a ver la función de sus principales componentes. En este punto ya deberíamos estar un poco familiarizados con la función del procesador. En la actualidad, encontramos varios procesadores distintos en el mercado. En orden de evolución podemos citar el *486*, el *Pentium*, el *Pentium MMX*, el K6, el *K6-2*, el *Pentium II*, el *Celeron*, el *Pentium III*, el *Duron*, el *Athlon* y, finalmente, el *Pentium 4* entre otros.

Un procesador Pentium 4

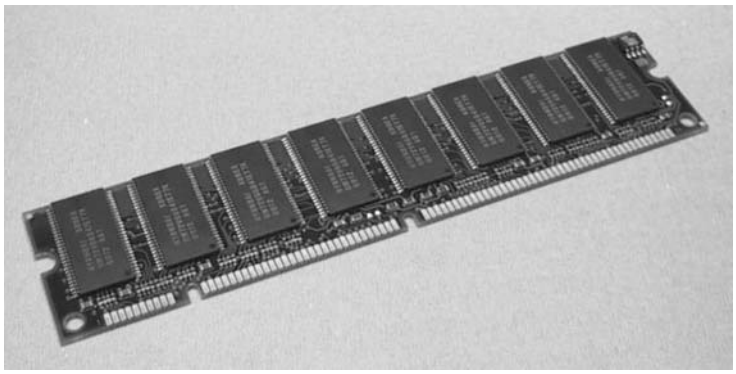
Definimos al procesador como el cerebro del ordenador. Podemos entender la placa base como el cuerpo del cerebro del ordenador. En ella encontramos todos los compo-

nentes que permiten al procesador comunicarse con los demás periféricos, como los discos duros, la tarjeta gráfica, etc. Otra función de la placa base es acomodar y suministrar electricidad al procesador.

Cada procesador necesita una placa base desarrollada especialmente para él, pues debido a las diferencias en la arquitectura de los procesadores, éstos poseen necesidades distintas. Cada procesador posee un número diferente de contactos, trabaja a un voltaje diferente y necesita de un conjunto de circuitos de apoyo desarrollados especialmente para él. El propio anclaje del procesador cambia dependiendo de la familia del mismo. Por ejemplo, nunca podremos instalar un *Athlon* en una placa base diseñada para una *Pentium 4*.

La placa base determina qué componentes se podrán usar en el ordenador (y consecuentemente las posibilidades de ampliación) y tiene una gran parte de influencia en el rendimiento general del equipo.

Para poder trabajar, el procesador también necesita a la memoria *RAM*, que es vendida en módulos de memoria y que son ensamblados en la placa base.



Un módulo de memoria RAM

Todos los programas y archivos son almacenados en el disco duro, también llamado *Hard Disk* (*HD*). La capacidad de almacenamiento del disco duro, en la actualidad medida ya en GBytes, determina la cantidad de archivos y programas que podremos almacenar. El disco duro también ejerce una gran influencia sobre el rendimiento global del equipo. El disco duro es acoplado a la caja del ordenador y conectado a la placa base a través de un cable de datos.

Otro componente esencial es la caja de metal que contiene y protege los frágiles componentes internos del ordenador. La caja también contiene la fuente de alimentación, responsable de convertir la corriente alterna de la toma (*AC*) en la corriente continua (*DC*) utilizada por los componentes electrónicos. La fuente de alimentación también sirve para atenuar pequeñas variaciones de tensión, protegiendo al equipo de posibles errores.

La fuente de alimentación

La placa base, el procesador, los módulos de memoria y el disco duro son los cuatro componentes básicos del ordenador. Sin embargo, hasta este momento sólo tenemos un equipo "autista", incapaz de recibir y transmitir información. Ahora necesitamos añadirle los "sentidos", en forma de más componentes. En este caso, los componentes más esenciales son la tarjeta gráfica, que permite que el ordenador pueda generar las imágenes que se mostrarán en el monitor, el teclado y el ratón (*mouse*), que le permiten al usuario interactuar con el ordenador.

Existen otros componentes que permiten ampliar los recursos del ordenador, pero pueden ser considerados opcionales, ya que el ordenador puede funcionar sin ellos. El *CD-ROM* permite que el ordenador lea *CDs* con juegos o programas. Si el ordenador también dispone de una tarjeta de sonido, podremos oír los *CDs* de música. También existen las unidades (*drives*) de *DVD*, que además de leer los *CDs* normales, leen los *DVDs* de programas o películas.

La tarjeta de sonido permite que el ordenador gestione los distintos sonidos, reproducidos a través de un par de altavoces. La tarjeta de sonido también dispone de una entrada para un micrófono y para un joystick.

El módem permite la comunicación entre dos ordenadores a través de la línea telefónica. El módem hace posible la recepción y transmisión de faxes y el acceso a *Internet*. Hoy en día, el módem es un componente prácticamente obligatorio: un ordenador que no puede ser conectado a *Internet* tiene una utilidad limitada hoy en día.

También tenemos la unidad de disquetes, que a pesar de ser un componente de baja tecnología, aún es necesario, pues los disquetes aún son muy usados para transportar datos de poco tamaño.

Además de estos componentes, disponemos de una gama enorme de otros accesorios: impresoras, escáneres (que permiten digitalizar imágenes), cámaras fotográficas digitales, cámaras de vídeo-conferencia, tarjetas de captura de vídeo y muchos otros.

1.5 La mejor configuración

Todos los componentes de un PC tienen una influencia directa en el rendimiento global del equipo. En este punto entra en escena, entonces, el factor coste-rendimiento: determinar qué configuración sería la mejor dentro de lo que podemos gastar. El objetivo de este apartado es justamente éste, ayudar a escoger la mejor configuración en términos de coste-rendimiento en cada caso. Para ello, veremos cómo cada componente influye en el rendimiento general del equipo y en qué aplicaciones es más importante cada uno.

Lo primero que debemos tener en cuenta es la aplicación a la que queremos destinar el ordenador, o sea, qué programas utilizaremos en él. Un PC utilizado en una oficina, donde se usan *Word, Excel* e *Internet* por ejemplo, no necesita de un procesador muy poderoso, pero es indispensable una cantidad razonable de memoria *RAM* y un disco duro razonablemente rápido. En cambio, en un ordenador destinado a los juegos, lo principal es un procesador rápido combinado con una buena y rápida tarjeta gráfica 3D.

1.5.1 La placa base

La placa base es el componente que debemos escoger con más cuidado. Una de baja calidad pondría en continuo riesgo el rendimiento del equipo. Al comprar una placa base debemos verificar qué procesadores soporta, si posee un *slot AGP* y si la cantidad de *slots ISA* y *PCI* es la suficiente para la cantidad de periféricos que queremos instalar en ella.

La cuestión más importante es la calidad de la placa. Además de los recursos, éste es el principal diferencial entre las distintas placas que encontraremos en el mercado. Las placas base de baja calidad, además de perjudicar al rendimiento, pueden hacer que el ordenador se vuelva inestable, causando bloqueos constantes en el sistema operativo. Bloqueos que frecuentemente son causados por fallos de hardware y no por *bugs* de los programas.

Deberíamos comprar una placa base de una buena marca, como Asus, Abit, Soyo y Supermicro. Las placas base de la casa Intel también son excelentes, pero debemos prestar atención a si la placa ha sido fabricada realmente por Intel: muchos vendedores venden placas con *chipsets* Intel como placas de la casas Intel. Muchos fabricantes usan *chipsets* Intel en sus placas, pero esto no es una garantía de calidad.

Deberíamos evitar al máximo comprar placas base que no traigan especificado el nombre del fabricante. A pesar de que sean más baratas y que traigan sonido, vídeo, módem y hasta la tarjeta de red *onboard*, estas placa,s invariablemente, son de baja calidad. Podemos preguntarnos por qué estas placas base son inferiores, ya que muchas veces usan el mismo *chipset* que las placas de buenas marcas. El diferencial es la calidad de los distintos circuitos de la placa.

Una placa base se confecciona usando una técnica llamada *MPCB* (*Multiple Layer Contact Board*), que consiste en varias placas apiladas como si fueran una de sola. Una placa de circuitos de este tipo tiene que ser proyectada y fabricada minuciosamente, pues cualquier mínimo error en la posición de las pistas, haría que surgiesen interferencias y convertirían la placa en inestable. Esto también perjudica al rendimiento, impidiendo que la comunicación entre los componentes sea realizada a la velocidad normal. La diferencia de rendimiento de un ordenador montado con una buena placa base con otro de configuración parecida pero usando una placa base de baja calidad puede llegar a un 20%.

1.5.2 Memoria RAM

Si el ordenador posee poca memoria *RAM*, el procesador tendrá que usar el disco duro para guardar los datos que deberían ser almacenados en la memoria, haciendo que el sistema se vuelva extremadamente lento. Por otro lado, instalar más memoria de la necesaria sería un desperdicio, pues no hará que el sistema sea más rápido. Notaremos que es necesario instalar más memoria *RAM* cuando el ordenador empiece a quedarse algo lento y a visitar intermitentemente el disco duro en los momentos de mayor actividad.

Si trabajamos sólo con aplicaciones sencillas, como *Word*, *Excel* o *Internet* y no solemos abrir más de una aplicación al mismo tiempo, 64 MB pueden ser suficientes, a pesar de que 128 MB sería lo ideal en este caso. Si, por otro lado, usamos programas de mayor tamaño o trabajamos con varios programas al mismo tiempo, entonces el mínimo sería 128 MB y lo ideal 256 MB.

Si destinamos el ordenador al procesamiento de imágenes, vídeo o a la edición, entonces deberíamos disponer de por lo menos 192 MB. Dependiendo del tamaño de los archivos que vamos a procesar, el ideal puede subir hasta los 256 MB o, incluso, hasta los 512 MB.

Hoy día, los módulos de memoria *SDRAM* (los usados en la mayoría de PCs actuales) son bastante baratos, por eso no vale la pena economizar en memoria *RAM*. Aunque el PC esté destinado a aplicaciones pequeñas, deberíamos instalar inmediatamente 256 o 512 MB. En este caso, es mejor pecar por exceso que por falta.

Con más memoria *RAM*, el PC se mantendrá rápido durante más tiempo. No debemos olvidar que los programas serán cada vez más grandes con el paso del tiempo. Lo que hoy puede ser un desperdicio de memoria, mañana será el requisito mínimo para muchos sistemas operativos.

1.5.3 Procesador

No siempre la instalación de un procesador más moderno hace que el ordenador sea más rápido. Muchas veces, aumentar la cantidad de memoria o cambiar el disco duro es más efectivo. Como siempre, depende de la aplicación que le demos al equipo.

Si el ordenador se destina principalmente a juegos, entonces vale la pena invertir en un procesador de última generación, como un *Athlon* o un *Pentium 4*. Si el ordenador lo destinamos al procesamiento de imágenes o a la edición, un procesador de última generación podría ayudar, pero sólo si el equipo dispone de la suficiente memoria *RAM*. Si se quiere ahorrar dinero, es preferible comprar un procesador de gama media, como un *Duron* e invertir más en memoria *RAM*.

Finalmente, si el ordenador es destinado a aplicaciones de poca carga de trabajo, lo ideal será adquirir un procesador más simple e invertir un poco más en la memoria, en un disco duro mejor o en una placa base de mejor calidad.

1.5.4 Disco duro

El rendimiento del disco duro determina la velocidad en la que se abrirán los programas y los archivos. Un disco duro rápido también ayuda en el caso de un ordenador que disponga de poca memoria *RAM*. En el caso de disponer de un procesador rapidísimo y mucha memoria *RAM*, todo puede volverse muy lento si el disco duro no acompaña en cuanto a rendimiento.

Es importante saber el tiempo de acceso, la velocidad de rotación y la densidad del disco duro. El tiempo de acceso del disco varía generalmente entre los 8 y 12 milisegundos, dependiendo del disco duro. El tiempo de acceso determina cuánto tiempo tarda el cabezal de lectura en encontrar el dato que debe leerse. Un valor más bajo de tiempo de acceso se corresponde con un mejor rendimiento.

La velocidad de rotación se mide en *rpm*, revoluciones (rotaciones) por minuto. Si un disco gira más rápido, el dato necesario se encuentra más rápidamente. La densidad, o la cantidad de datos que caben en cada disco duro, también determina su rendimiento, pues como los datos estarán más próximos entre sí, serán localizados de forma más rápida. Podemos saber la densidad del disco duro dividiendo la capacidad total del disco por la cantidad de cabezales de lectura que posee. Por ejemplo, un disco duro de 6 GB con 4 cabezales de lectura tiene una densidad de 1.5 GB por cada *platter*. Cuanto mayor es la densidad, mejor rendimiento obtendremos del disco duro.

1.5.5 Tarjeta gráfica

Existen tarjetas gráficas 2D y 3D. También existen tarjetas aceleradoras 3D que deben usarse conjuntamente con una tarjeta 2D común. También existen tarjetas gráficas "combo" (las más comunes), que realizan las funciones 3D y 2D (todas las tarjetas actuales). Al día de hoy, incluso las tarjetas gráficas *onboard* vienen con recursos 3D, aunque su rendimiento no puede compararse con el de una tarjeta gráfica de calidad.

1.5.6 Tarjeta de sonido

La tarjeta de sonido no tiene ninguna influencia en el rendimiento del equipo, sólo determina la calidad del audio. Para un uso normal, una tarjeta de sonido simple,

como una *Sound Blaster* de gama media, o una de las tarjetas "genéricas", generalmente equipadas con *chipsets* Yamaha, dan un buen resultado. Las tarjetas de sonido más caras marcan la diferencia si pretendemos trabajar en la edición musical, o queremos oír música *MIDI* con el máximo de calidad.

También existen tarjetas de sonido 3D, como la *Sound Blaster Live*, que generan sonidos que parecen venir de todas direcciones, aunque usemos unos altavoces comunes. Este efecto es muy interesante para los juegos, pues ofrece una mayor sensación de realidad.

1.6 Ampliaciones y actualizaciones

Realizar una ampliación significa cambiar algunos componentes de un equipo ya viejo a fin de mejorar su rendimiento. Sin embargo, muchas veces, el equipo está tan viejo que sería necesario cambiar casi todos los componentes para conseguir alcanzar un rendimiento aceptable. En este caso, sería más conveniente y adecuado comprar un ordenador nuevo.

El secreto para realizar una buena actualización es detectar los "puntos débiles" de la configuración, los componentes que tienen un rendimiento inferior al del resto del conjunto.

1.6.1 La relación precio-rendimiento

Ya deberíamos haber percibido que en el mundo de la informática las cosas evolucionan muy rápidamente. Cada semana salen nuevos componentes al mercado.

Pero, prestando algo de atención podemos darnos cuenta de dos cosas importantes:

1. En general, los fabricantes lanzan nuevos componentes con pequeños avances sobre los anteriores y, sin embargo, con un gran aumento del precio. Por ejemplo, en el sector de los procesadores, los nuevos modelos son siempre sólo 33 o 50 MHz más rápidos que los anteriores.

   Por ejemplo, en la familia de procesadores *Pentium III* se lanzaron al mercado en menos de un año las versiones de 450, 500, 533, 550, 600, 650, 667, 700, 733, 750, 800 y 1.000 MHz. Siempre que se lanza una nueva versión, las anteriores caen de precio y las más antiguas son retiradas del mercado.

   La diferencia de precio entre la versión más alta de la línea y la anterior, que es en general sólo un 5 o 6% más lenta, puede llegar a casi un 50%. Viendo esto, inmediatamente percibimos que no vale la pena comprar el procesador más rápido, pero sí pagar 3 o 4 veces menos por un procesador sólo un poco más lento.

   En otras áreas, como en el sector de las tarjetas gráficas 3D, la diferencia no es tan escandalosa, pero las tarjetas de gama alta valen, en general, 2 veces más que las versiones anteriores, siendo sólo un 25 o 30% más rápidas.

En el caso del disco duro, el factor más importante es la capacidad, pero el rendimiento general también es fundamental. Muchas veces, un disco duro de menor capacidad es mucho más rápido que uno de mayor capacidad.

2. En los últimos años, los equipos han evolucionado mucho más rápido que los requisitos de los programas. Al contrario de lo que ocurría algunos años atrás, un ordenador de hace dos años, completamente antiguo respecto a los patrones actuales, puede ejecutar con bastante facilidad casi todas las aplicaciones actuales. En estos momentos, sólo existen dos tipos de aplicaciones que utilizan todo el poder de procesamiento de un ordenador de última generación: las aplicaciones profesionales para renderizar imágenes y la edición de vídeo y los juegos más actuales del mercado.

   Aparte de saber escoger los componentes en relación a la calidad, también deberíamos tener presente si realmente vamos a necesitarlo todo cuando vayamos a comprar un ordenador nuevo. Es evidente que no vale la pena comprar un equipo muy anticuado, pero tampoco debe ser necesario comprar uno de última generación. El punto ideal está en un punto intermedio entre estos dos extremos.

1.7 Las redes

Las redes son, en los últimos tiempos, un campo cada vez más importante. Basta recordar que *Internet*, tan esencial hoy en día, no es más que una red de ordenadores. Existen varias topologías (arquitecturas) de red y cada día se crean de nuevas, pero por suerte, la tendencia es que sean pocos los patrones que sobrevivan en cada área.

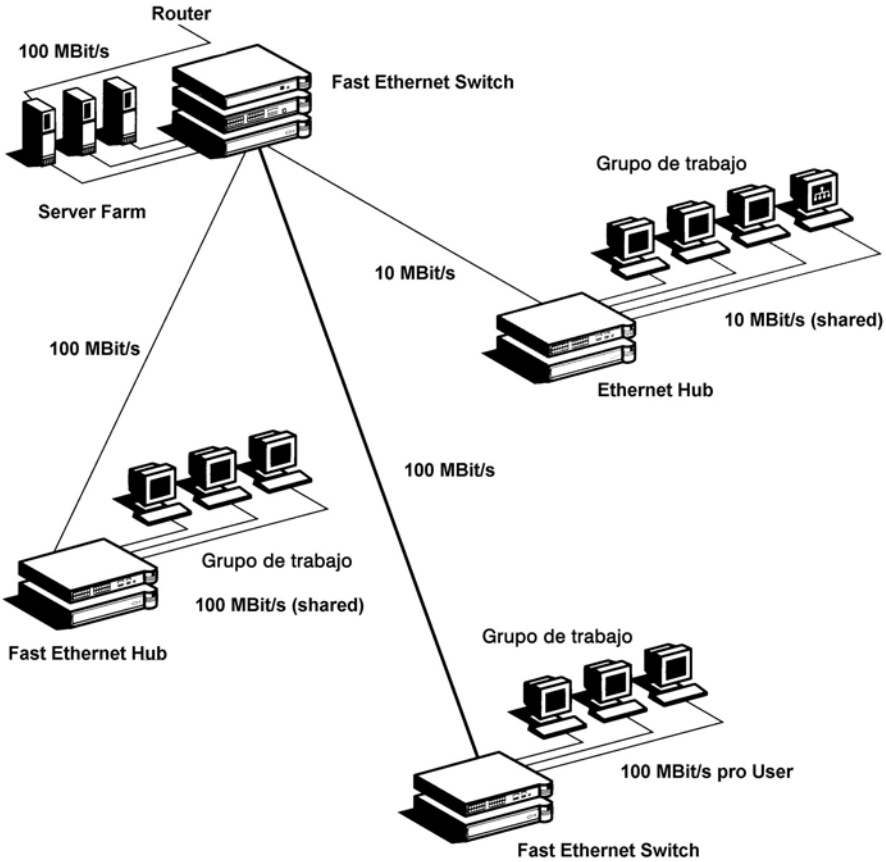
El patrón *Ethernet* se hizo casi omnipresente en las redes cableadas. Eso comportó algo muy positivo, que fue la facilidad en el montaje de las redes. Como todas las tarjetas de red son compatibles y los cables siguen un mismo patrón, es más fácil encontrar los componentes cuyo precio también disminuyó bastante.

Disponemos de tres patrones distintos de redes *Ethernet*: de 10 mbits/s, 100 mbits/s y 1 gbit/s. Las tarjetas de red son compatibles, pero al combinar tarjetas de velocidades diferentes, la red pasa a operar a la velocidad de la tarjeta de red más lenta, a menos que usemos un *switch*, que es capaz de aislar las transmisiones, permitiendo que cada tarjeta de red opere a su velocidad sin perjudicar a las demás.

Un *switch* vendría a ser un hermano más viejo del *hub*, teniendo los dos la misma función, o sea, servir como punto céntrico para una red. Todas las tarjetas de red son conectadas al *hub* o *switch*. También es posible, en el caso necesario, conectar varios *hubs* o *switchs* entre sí. La diferencia es que el *hub* sólo retransmite todo lo que recibe hacia todas las estaciones de trabajo. Esto significa que sólo una puede transmitir datos cada vez y que todas necesitan operar a la misma velocidad (siempre marcada por abajo). Esto funciona bien en pequeñas redes, pero conforme aumenta el número de PCs, el rendimiento disminuye rápidamente.

Entonces surgen los *switchs*, aparatos más inteligentes, que son capaces de establecer conexiones sólo entre el emisor y el destinatario de la transmisión. Eso permite que se realicen varias transmisiones al mismo tiempo (entre PCs distintos naturalmente) y que cada tarjeta de red pueda operar a su velocidad máxima. Con el uso de un *switch*, el rendimiento de la red se mantiene con un número mayor de estaciones de trabajo.

Finalmente, tenemos los *routers*, que son el último paso de la cadena evolutiva. Los *routers* son aún más inteligentes, pues son capaces de interconectar varias redes diferentes y siempre escoger la ruta más rápida para cada paquete de datos. Los *routers* forman la espina dorsal de *Internet*.



Distintas redes interconectadas mediate hubs, switches y routers

Finalmente, encontramos las redes inalámbricas que están en un rápido ascenso, lideradas por las tarjetas del estándar *802.11b*. En vez del *hub* disponemos del punto de acceso, que es capaz de centralizar las transmisiones de datos de algunas decenas de estaciones de trabajo.

La velocidad es similar a las de las redes *Ethernet* y el alcance varía entre los 15 y 100 metros, dependiendo de los obstáculos. En grandes distancias la señal se degrada y la velocidad de transmisión disminuye, hasta el punto de que la señal puede perderse completamente.

Además de usarse en los ordenadores portátiles, las interfaces *802.11b* pueden usarse en algunos ordenadores de mano y tiene muchos puntos a favor para convertirse en muy popular. Casi siempre, los puntos de acceso pueden conectarse a una red *Ethernet* ya existente, unificando las dos redes. Esto permite que en una oficina podamos conectar los ordenadores de sobremesa usando una red *Ethernet* convencional, que es más rápida y más barata, y usar la red inalámbrica sólo para los ordenadores portátiles. Además de que los componentes físicos de la red son casi siempre compatibles, los sistemas operativos también lo son en la actualidad.

Con algunos conocimientos básicos no deberíamos tener demasiados problemas para interconectar un PC bajo *Windows*, otro bajo *Linux* y un *Macintosh* en la misma red, intercambiar archivos y compartir la conexión a *Internet*. Algunos términos de red que necesitaríamos conocer son:

TCP/IP: es el principal protocolo de red, usado en *Internet* y en la gran mayoría de las redes locales. El protocolo es justamente el "idioma" universal que permite que varios sistemas operativos distintos se puedan entender.

ICS: es el programa, presente en *Windows 98 SE*, *Windows ME*, *Windows 2000* y *Windows XP* que permite compartir la conexión de *Internet*. Los clientes pueden ejecutar *Linux* u otros sistemas operativos pues toda la interconexión se realiza vía *TCP/IP*, que es universal.

Samba: es el software que permite compartir archivos entre ordenadores con *Linux* y *Windows*. Se ejecuta en los equipos que tienen *Linux*.

2. Un poco de historia

Hoy en día, cuando oímos hablar de procesadores de 3 GHz nos parecen de lo más común. Muy poca gente ha oído hablar del *8088*, que fue el procesador usado en el PC *XT*, hace más de 20 años, y mucho menos del Intel *4004*, el primer microprocesador lanzado al mercado en el año 1971.

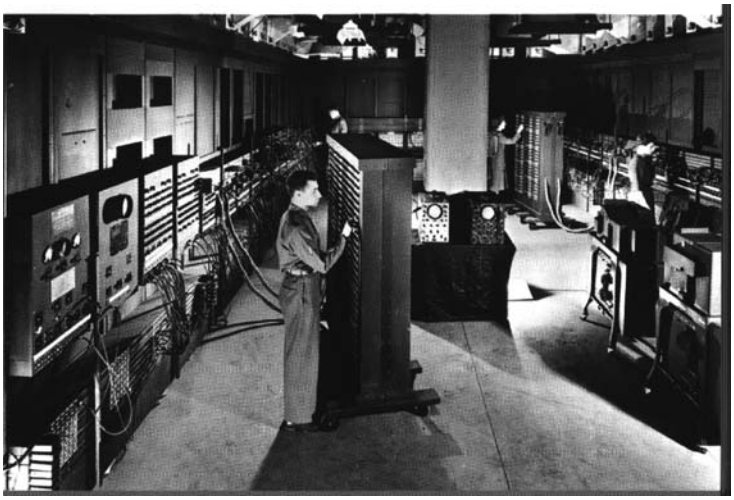
En este capítulo hablaremos de los procesadores y ordenadores que forman parte de la historia. En la época de nuestros bisabuelos ya existían ordenadores, a pesar de que eran muy rudimentarios. Eran los ordenadores mecánicos, que realizaban los cálculos a través de un sistema de engranajes, accionado por una manivela u otro sistema mecánico.

A finales del siglo XIX surgió el relé, un dispositivo electromagnético formado por una pieza magnética móvil, que se desplazaba uniendo dos contactos metálicos. El relé fue muy usado en los sistemas telefónicos y, de hecho, algunas centralitas analógicas todavía los utilizan. Los relés pueden ser considerados como antepasados de los transistores. Los relés eran demasiado grandes y, al mismo tiempo muy lentos: un relé tarda más de una milésima de segundo en cerrar un circuito.

También a finales del siglo XIX surgieron las primeras válvulas. Éstas se usaron para crear los primeros ordenadores electrónicos, en la década de los años 40. Las válvulas basan su funcionamiento en el flujo de electrones en el vacío. Todo empezó cuando Thomas Edison, inventor de la bombilla eléctrica, estaba jugando con su invento y percibió que al conectar la bombilla al polo positivo de una batería y una placa metálica al polo negativo, era posible medir una cierta corriente fluyendo del filamento de la bombilla a la placa metálica, aunque los dos estuvieran aislados. Había descubierto el efecto termoiónico, el principio de funcionamiento de las válvulas.

Las válvulas eran mucho más rápidas que los relés, y ya alcanzaban frecuencias de algunos MHz. El problema de las válvulas es que se calentaban en exceso, consumían mucha electricidad y se quemaban con facilidad. Era fácil usar válvulas en radios, ya que usaban pocas, pero construir un ordenador que usaba miles de ellas era extremadamente complicado y caro. A pesar de todo esto, los primeros ordenadores surgieron en la década de los años 40, naturalmente con propósitos militares. Su principal uso era la codificación y decodificación de mensajes.

Seguramente, el ordenador más famoso de aquella época fue el *ENIAC* (*Electronic Numerical Integrator Analyzer and Computer*), construido en 1945. El *ENIAC* estaba compuesto por nada menos que 17.468 válvulas. Sin embargo, a pesar del tamaño inmenso del mismo, el poder de procesamiento del *ENIAC* era ridículo comparado con los patrones de procesamiento actuales, aunque era capaz de procesar 5.000 sumas, 357 multiplicaciones y 38 divisiones por segundo, bastante menos que una calculadora de bolsillo actual.

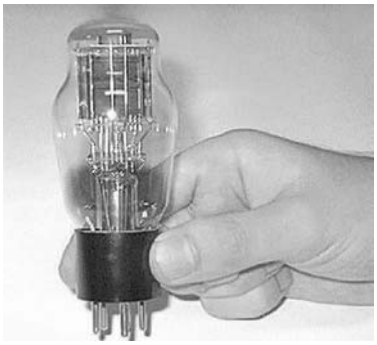


Una parte del ordenador ENIAC

La idea era construir un ordenador para realizar varios tipos de cálculos de artillería para ayudar a las tropas aliadas durante la segunda guerra mundial. Sin embargo, el *ENIAC* acabó siendo terminado 3 meses después del final de la guerra y siendo usado durante la guerra fría, contribuyendo por ejemplo, en el proyecto de la bomba de hidrógeno.

Si pensamos que programar en *C* o en *ensamblador* es complicado, podemos imaginar cómo era la vida de los programadores de aquella época. La programación del *ENIAC* se llevaba a cabo a través de 6.000 llaves manuales. Con cada cálculo nuevo, era necesario reprogramar varias de estas llaves. Eso sin hablar del resultado, que era dado de forma binaria a través de un conjunto de luces.

En la época, la mayor parte de la industria continuó trabajando en el perfeccionamiento de las válvulas, obteniendo modelos más fiables y de menor tamaño. Sin embargo, varios investigadores comenzaron a buscar alternativas menos problemáticas. Algunos de estos experimentos tenían como objetivo la investigación de nuevos materiales, tanto conductores como aislantes. Los investigadores empezaron a descubrir que algunos materiales no encajaban ni en un grupo ni en el otro, pues dependiendo de la circunstancia, podían actuar como aislantes o como conductores, formando una especie de grupo intermedio que fue denominado el grupo de los semiconductores.



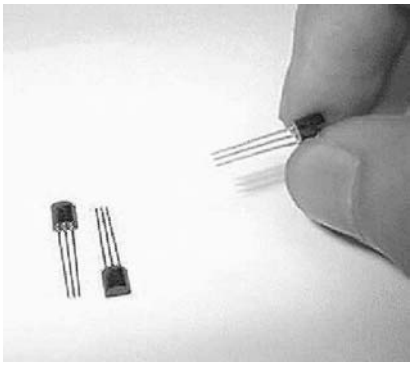
Una válvula electrónica

Habían encontrado la llave para desarrollar el transistor. El primer proyecto conocido de transistor surgió el 16 de diciembre del año 1947, donde se usó un pequeño bloque de germanio (que en la época era junto con el silicio el semiconductor más investigado) y tres filamentos de oro. Un filamento era el polo positivo, otro el polo negativo, mientras que el tercero tenía la función de control. Teniendo una sola carga eléctrica en el polo positivo no acontecía nada, el germanio actuaba como aislante, bloqueando la corriente. Sin embargo, cuando se aplicaba una cierta tensión eléctrica a través del filamento de control, la carga eléctrica fluía hacía el polo negativo. Habían creado un dispositivo que no tenía partes móviles y que sustituía a la válvula, gastando una fracción de la electricidad gastada por una válvula y, al mismo tiempo, mucho más rápido y de menor tamaño.

Este primer transistor era relativamente grande, pero no pasó mucho tiempo hasta que fue perfeccionado. Durante la década de los años 1950, el transistor fue gradualmente dominando la industria, sustituyendo a las problemáticas válvulas. Los modelos fueron disminuyendo de tamaño, bajando de precio y haciéndose más y más rápidos. Algunos transistores de la época podían trabajar a hasta 100 MHz. Claro que esta era la frecuencia que podía alcanzar un único transistor. En los ordenadores de la época, la frecuencia de trabajo era bastante menor, ya que en cada ciclo de procesamiento la señal necesitaba pasar por varios transistores.

Pero el gran salto fue la sustitución del germanio por el silicio. Este paso permitió miniaturizar aún más los transistores y bajar sus costes de producción. La idea del uso del silicio para la construcción de los transistores es que añadiendo ciertas sustancias, en pequeñas cantidades, es posible alterar las propiedades eléctricas del silicio. En las primeras investigaciones se usaron fósforo y boro, que transformaban el silicio en conductor para cargas negativas o conductor para cargas positivas, dependiendo del material que se usase.

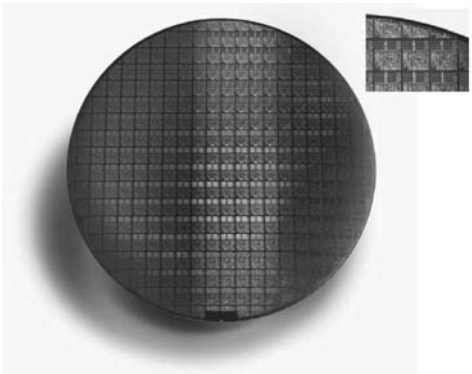
El funcionamiento de un transistor es bastante simple. Un transistor está compuesto de tres filamentos, llamados base, emisor y receptor. El emisor es el polo positivo, el receptor el polo negativo, mientras que el filamento base es el que controla el estado del transistor.



Unos transistores con sus tres filamentos

Cuando el transistor se encuentra desconectado no existe carga eléctrica en la base, y por eso, no existe corriente eléctrica entre el emisor y el receptor. Cuando se aplica una cierta tensión en la base, se cierra el circuito y se establece la corriente entre el emisor y el receptor.

También se logró otro gran avance cuando los fabricantes se dieron cuenta de que era posible construir varios transistores sobre el mismo *waffer* de silicio. Había surgido el circuito integrado, varios transistores dentro del mismo encapsulado. No pasó mucho tiempo hasta la aparición de los primeros *microchips*.



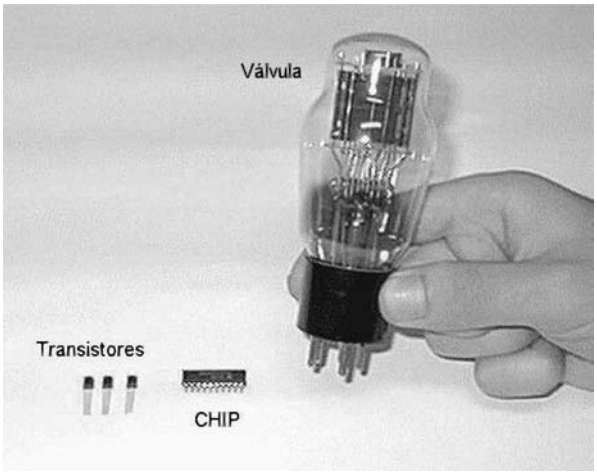
Un waffer de silicio

2.1 La fabricación de los procesadores

Hasta que llegaron las cosas al punto como las conocemos hoy, los *chips* pasaron por un largo proceso evolutivo. Todo empezó con las válvulas, que evolucionaron hasta la aparición de los transistores, considerados una de las mayores invenciones del siglo XX. La aparición de los primeros transistores, debido a su elevado coste, no fue ninguna revolución sobre las antiguas válvulas, tanto que éstas sobrevivieron durante varios años después de que los transistores comenzasen a ser producidos a escala

comercial. El gran salto se produjo al descubrirse que era posible construir varios transistores sobre el mismo *waffer* de silicio Eso permitió disminuir de forma descomunal el coste y el tamaño de los ordenadores. Entramos entonces en la era del *microchip*.

El primer *microchip* comercial fue lanzado al mercado por la casa Intel en el año 1971 y se llamaba *4004*. Como el nombre indica, se trataba de un procesador de sólo 4 bits que estaba compuesto por poco más de 2.000 transistores. Desde entonces se ha recorrido un largo camino. Se realizaron enormes inversiones y muchos de los mayores genios del planeta trabajaron en la búsqueda de soluciones para cuestiones cada vez más complejas. Ahora repasaremos como evolucionaron los ordenadores, de las válvulas a los *Athlon* y P*entium 4* actuales.



Una vista general de las diferencias en tamaño

2.1.1 Los transistores

Cada transistor funciona como una especie de interruptor, que puede estar conectado (encendido) o desconectado (apagado), como una válvula. La diferencia es que el transistor no tiene partes móviles y es mucho más pequeño en tamaño, más barato, más durable y mucho más rápido que una válvula.

El cambio de estado, en el transistor, se realiza a través de una corriente eléctrica. Por su parte, este cambio de estado puede comportar el cambio de estado de otros transistores conectados al primero, permitiendo el procesamiento lógico de los datos. En un transistor, este cambio de estado puede ser llevado a cabo billones de veces por segundo. Sin embargo, cada cambio de estado genera cierta cantidad de calor y consume cierta cantidad de electricidad. Por eso, mientras más rápidos se convierten los procesadores, más se calientan y más energía consumen.

Por ejemplo, un *386* consumía poco más de 1 vatio y podía funcionar sin ningún tipo de refrigeración. Un *486DX-4 100* consumía cerca de 5 vatios y ya necesitaba de un simple ventilador, mientras que un *Athlon* llega a consumir 80 vatios y necesita, como

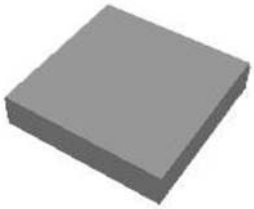
mínimo, de un buen ventilador para funcionar sin problemas. En compensación, el *386* operaba a 30 o 40 MHz mientras que el *Athlon* llega hasta los 3 GHz.

2.1.2 La fabricación

Llegamos al punto principal de este capítulo, que es mostrar cómo se fabrican los procesadores. El componente básico para cualquier chip es el *waffer* de silicio, que se obtiene a través de la fusión del silicio junto con algunos productos químicos. Inicialmente se producen cilindros de 20 a 30 centímetros de diámetro, que luego se cortan en partes bastante más finas. Estas partes más finas son pulidas, obteniendo los *waffers* de silicio.

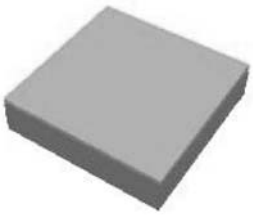
La calidad del *waffer* determinará el tipo de *chip* que podrá construirse. Para la construcción de un *chip* con media docena de transistores, puede utilizarse un *waffer* de baja calidad. Sin embargo, para fabricar un procesador moderno, es necesario usar un *waffer* de altísima calidad y que es extremadamente caro, dado que son muy pocas las compañías disponen de la tecnología adecuada para su producción.

Cada *waffer* es usado para la producción de varios procesadores, que son separados al final de la cadena de producción y encapsulados de forma individual. Sería imposible mostrar todos los procesos usados en la fabricación de un procesador, pero para poder tener una mínima idea de cómo se producen, vamos a intentar mostrar, paso a paso, la construcción de un único transistor. Después, imaginamos que el mismo proceso se repite algunos millones de veces formando un procesador funcional. Todo comienza con el *waffer* de silicio en su estado original.

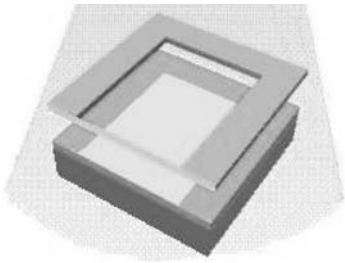


El waffer de silicio

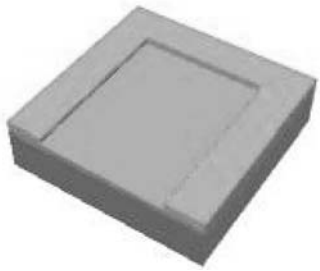
La primera etapa del proceso es oxidar la parte superior del *waffer*, transformándola en dióxido de silicio. Es decir, exponiendo el *waffer* a gases corrosivos y a altas temperaturas. La fina capa de dióxido de silicio que se forma es la que se usará como base para la construcción del transistor.



Luego se aplica una capa bastante fina de un material fotosensible sobre la capa de dióxido de silicio. Usando una máscara especial se aplica una luz ultravioleta sólo en algunas zonas de la superficie. Esta máscara tiene un patrón diferente para cada área del procesador, de acuerdo con el dibujo que se pretende obtener. La técnica usada se llama litografía óptica.



Originalmente, la capa fotosensible es sólida, pero al ser alcanzada por la luz ultravioleta se transforma en una sustancia gelatinosa, que se puede quitar fácilmente. Después de quitar las partes sobrantes de la capa fotosensible, algunas zonas del dióxido de silicio quedan al descubierto y otras continúan cubiertas por el resto de la capa.



Entonces, el *waffer* es bañado con un producto especial que elimina las partes del dióxido de silicio que no han quedadp protegidas por la capa fotosensible.

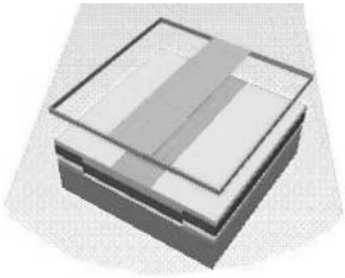
Finalmente, se quita la parte que quedó de la capa fotosensible. Debemos notar que al tener sustancias diferentes es posible quitar una capa cada vez, primero el dióxido de silicio, luego la propia capa fotosensible. Con lo que tenemos ahora es posible "diseñar" las estructuras necesarias para formar los transistores. En este punto tenemos lista la primera capa.

Cada transistor está formado por varias capas, dependiendo del procesador. En este ejemplo, tenemos un transistor simple, de sólo cuatro capas, pero los procesadores actuales utilizan un número mayor de capas, con más de veinte en algunos casos, dependiendo de la densidad que el fabricante pretenda alcanzar.

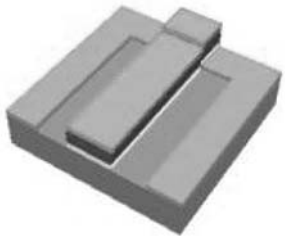
Entonces empieza la construcción de la segunda capa del transistor. Inicialmente, el *waffer* pasa de nuevo por el proceso de oxidación inicial, siendo cubierto por una

nueva capa (esta vez muy fina) de dióxido de silicio. Enseguida se aplica una capa de cristal de silicio sobre la estructura. Sobre esta capa se aplica una nueva capa de material fotosensible.

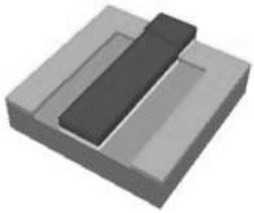
Nuevamente, el *waffer* pasa por el proceso de litografía, esta vez utilizando una máscara diferente.



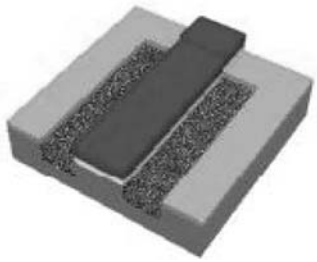
También ahora, la parte de la capa fotosensible que queda expuesta a la luz es eliminada, dejando expuestas las partes de las capas de cristal de silicio y de dióxido de silicio, que también son eliminadas a continuación.



Como en la etapa anterior, lo que queda de la capa fotosensible es eliminado. En este punto, terminamos la construcción de la segunda capa del transistor.

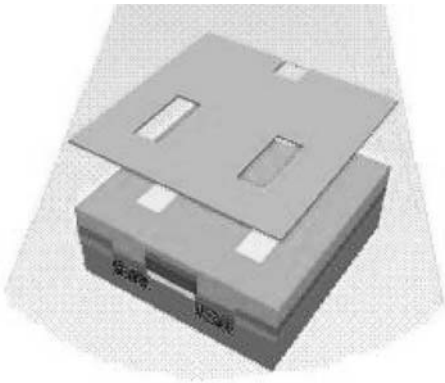


Llegamos a una de las principales etapas del proceso de fabricación, que es la aplicación de las impurezas, que transformarán partes del *waffer* de silicio en un material conductor. Estas impurezas también son llamadas *de iones*. Debemos notar que los iones sólo se adhieren a la capa de silicio que fue expuesta en el proceso anterior y no en las capas de dióxido de silicio o en la capa de cristal de silicio.



En este punto se le añade una tercera capa compuesta de un tipo diferente de cristal de silicio y, nuevamente, se aplica la capa fotosensible encima de todo.

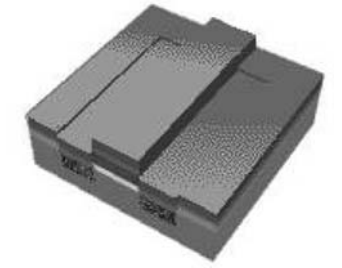
El *waffer* pasa otra vez por el proceso de litografía, usando una máscara diferente una vez más.



Las partes del material fotosensible expuestas a la luz son eliminadas, exponiendo partes de las capas inferiores, que también son eliminadas a continuación.

Ahora tenemos terminada la tercera capa del transistor. Podemos observar que la estructura del transistor ya está casi lista, a la espera de los tres filamentos conductores.

Se aplica una finísima capa de metal sobre la estructura anterior. En los procesadores que son producidos por medio de una técnica de producción de 0.13 micras, por ejemplo, esta capa metálica tiene el equivalente a sólo 6 átomos de espesor.



Se repite el proceso de aplicación de la capa fotosensible, de litografía y de eliminación de las capas una vez más, con el objetivo de eliminar las partes indeseables de la capa de metal. Finalmente, tenemos listo el transistor.

Cada procesador está constituido por varios millones de transistores. Un *Pentium II* posee poco más de 9 millones de transistores. Un *Pentium III Coppermine* ya posee 22 millones de transistores. Un *Athlon Thunderbird* posee 35 millones de transistores, mientras que un *Pentium 4* posee unos 42 millones de transistores.

Gracias al nivel de miniaturización que existe en la actualidad, estas cantidades de transistores ocupan un espacio muy pequeño. Por ejemplo, un *Athlon Thunderbird* sólo mide 112 milímetros cuadrados. Con esto, un único *waffer* de silicio es suficiente para la producción de varios procesadores, que son separados al final del proceso de fabricación

Distintos procesadores de un mismo waffer de silicio

Finalmente, los procesadores son encapsulados en una estructura de silicio, que los protege y facilita el manejo y la instalación. El formato del encapsulado varía según el modelo de procesador como podremos ver en el capítulo de los procesadores.

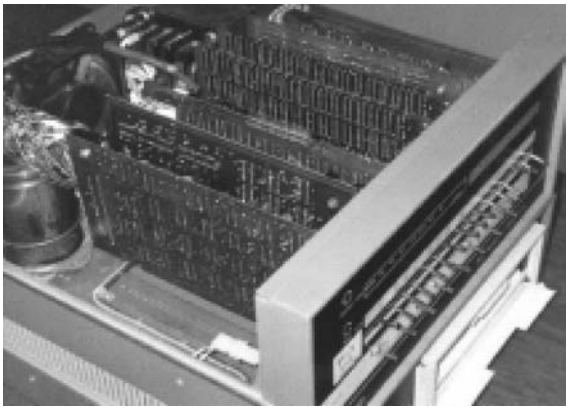
2.2 El origen de los ordenadores personales

Hasta este momento, hemos hablado de las supercomputadoras y de la creación de los procesadores, que evolucionaron de las válvulas al transistor y, después, hacia el circuito integrado. En lo que resta de este capítulo hablaremos sobre los primeros ordenadores personales, que comenzaron a escribir su historia a partir de la década de los años 1970.

El primer *microchip*, el *4004*, fue lanzado por la casa Intel en el año 1971. Era un proyecto bastante rudimentario que sólo procesaba 4 bits en cada ciclo y operaba a sólo 1 MHz. En realidad, el *4004* era tan lento que tardaba 10 ciclos para procesar cada instrucción, o sea, sólo procesaba 100.000 instrucciones por segundo. Hoy en día, estos números perecen ridículos, pero para la época era lo último en tecnología. El *4004* fue usado en varios modelos de calculadoras.

Poco tiempo después, la casa Intel lanzó un nuevo procesador que duró muchos años, el *8080*. Éste ya era un procesador de 8 bits y operaba a 2 MHz. El *Altair 8800*, que es considerado por muchos el primer ordenador personal de la historia, estaba basado en el *8080* de la casa Intel y venía con sólo 256 Bytes de memoria *RAM*, realmente muy poca cantidad incluso para la época. También se encontraba una tarjeta de expansión de 4 KB.

Poco tiempo después de su aparición, empezaron a surgir varios accesorios para el *Altair*: un teclado que sustituía el conjunto de llaves que servían para programar el aparato, una terminal de vídeo (mejor que ver los resultados en forma de luces), una unidad de disquetes (en aquella época aún se usaban disquetes de 8 pulgadas), tarjetas de expansión de memoria y hasta una impresora.



El Altair 8800

El *Altair* era un sistema muy simple que no tenía mucha aplicación práctica, pero sirvió para demostrar la gran pasión que podía despertar la informática en la gente y que, al contrario de lo que decían muchos analistas de la época, sí existía un gran mercado para los ordenadores personales.

La casa Apple fue fundada en 1976, después de que las casas Atari y HP rechazasen el proyecto del *Apple I*. El *Apple I* no tuvo un gran éxito en las ventas, ya que vendió poco más de 200 unidades, pero abrió el camino para el lanzamiento de versiones más poderosas.

El *Apple I* usaba un procesador de la casa Motorola, el *6502*, que operaba a sólo 1 MHz. En términos de poder de procesamiento perdía respecto al *8080*, pero tenía algunas ventajas a nivel de flexibilidad. El *Apple I* venía con 4 KB de memoria *RAM* y disponía de salidas para el teclado, la terminal de vídeo y para una unidad de cinta. También existía un conector reservado para futuras expansiones.

En aquella época, las cintas *K7* eran el medio más usado para guardar datos y programas. Los disquetes ya existían, pero eran muy caros. El gran problema de las cintas *K7* era su lentitud, tanto para leer como para grabar. Este fue el motivo de que las cintas se desgasten con el tiempo.

Este primer modelo fue inmediatamente perfeccionado, surgiendo el *Apple II*. Este sí que tuvo un cierto éxito. El *Apple II* venía con 4 KB de memoria *RAM*, como el modelo anterior. La novedad era una memoria *ROM* de 12 KB, que almacenaba una versión del *Basic*. La memoria *RAM* podía ampliarse hasta los 52 KB, pues el procesador Motorola *6502* sólo era capaz de gestionar 64 KB de memoria, y 12 KB ya correspon-

dían a la *ROM*. Una de las ventajas para la época era una tarjeta de expansión, fabricada por Microsoft, que permitía deshabilitar la *ROM* y usar los 64 KB completos de la memoria *RAM*.

El *Apple II* ya era muy parecido a un ordenador actual, pues venía con teclado y usaba una televisión como monitor. El equipo también venía con una unidad de cinta *K7*, pero era posible adquirir una unidad de disquetes por separado. Una variación del *Apple II*, el *Apple IIc*, lanzado en el año 1979, es considerado por muchos el primer ordenador portátil de la historia, pues disponía de hasta un monitor LCD opcional.

En 1979 surgió otro modelo interesante, esta vez de la casa Sinclair, el *ZX-80*. Este no era tan potente como el *Apple II* pero tenía la ventaja de ser mucho más barato. Fue el primer ordenador popular de la historia. El procesador era una *Z80*, de la casa Zilog, que operaba a sólo 1 MHz. La memoria *RAM* también era algo minúscula, de sólo 1 KB, combinados con 4 KB de memoria *ROM* que almacenaban *Basic*, usado por el equipo.

Como en cualquier sistema popular de la época, los programas estaban almacenados en cintas *K7*. A pesar de venir con una salida de vídeo la resolución gráfica era de sólo 64 x 48, aún en modo monocromático, ya que el adaptador de vídeo sólo tenía 386 Bytes de memoria. También existía un modo texto, de 32 x 24 caracteres.

Otro que no puede dejar de citarse es el *Atari 800*. A pesar de ser vendido más como videojuego, el *Atari 800* también podía usarse como un ordenador relativamente potente y llegó a ser usado en algunas universidades. Fue el antecesor del *Atari 2600*, que venía de fábrica con 16 KB de memoria *RAM* ampliables hasta 48 KB, con 10 KB más de memoria *ROM*. El sistema operativo era el *Atari-OS*, una versión del *Basic*. Originalmente, el sistema sólo venía con la entrada para los cartuchos con el sistema operativo o juegos, pero era posible adquirir una unidad de disquetes y un teclado por separado que lo transformaban en un ordenador completo.

2.2.1 La década de los años 80

Una década es una verdadera eternidad dentro del mercado de la informática, tiempo suficiente para que pasen auténticas revoluciones. Después de la década de los años 1970, los ordenadores personales comenzaron, finalmente, a alcanzar un nivel de desarrollo suficiente para permitir el uso de aplicaciones serias. Entonces, surgieron las primeras aplicaciones de procesamiento de texto, hojas de cálculo e, incluso, programas de edición y dibujo.

El primer PC fue lanzado al mercado por la casa IBM en 1981 y tenía una configuración bastante modesta, con sólo 64 KB de memoria *RAM*, dos unidades de disquetes de 5 1/4, un monitor *MDA* de solamente texto (existía la opción de comprar un monitor *CGA*) y sin disco duro. Esta configuración ya era suficiente para poder ejecutar el sistema operativo *DOS 1.0* y la mayoría de la programas de la época, que al ser muy pequeños cabían en apenas un disquete y ocupaban poca memoria *RAM*. Pero, una

ventaja que existe desde este primer PC es la arquitectura abierta, que permite que varios fabricantes saquen al mercado accesorios y tarjetas de expansión. Fue cuestión de meses que se empezasen a vender discos duros, tarjetas de expansión de memoria, tarjetas de vídeo, etc. de varios fabricantes.

La casa Apple había lanzado el *Apple III* pocos meses antes del PC. El *Apple III* venía con 128 o 256 KB de memoria *RAM*, dependiendo de la versión, un procesador *Synertek 6502A* de 2 MHz y una unidad de disquetes de 5 1/4. El gran pecado de la casa Apple fue el uso de un bus de expansión propio, lo que limitó las posibilidades de ampliación sólo a los accesorios ofrecidos por la propia casa Apple, una característica que fue la gran responsable de la supremacía del PC.

En 1983, la casa Apple apareció con una gran novedad, el *Lisa*. En su configuración original, el *Lisa* venía equipado con un procesador Motorola *68000* de 5 MHz, 1 MB de memoria *RAM*, dos unidades de disquete de 5 1/4 de 871 KB, un disco duro de 5 MB y un monitor de 12 pulgadas con una resolución de 720 x 360. Era una configuración mucho mejor que la de los PCs de la época, sin hablar de que el *Lisa* ya usaba una interfaz gráfica bastante elaborada y contaba ya con una *suite* de aplicaciones de oficina.

El Apple Lisa con su teclado y su pantalla

El problema era el precio, ya que el *Lisa* era muy caro, y por eso, nuevamente, no tuvo mucho éxito, pero el proyecto sirvió de base para el *Macintosh* lanzado en 1984. Este sí que tuvo un gran éxito, llegando a amenazar el imperio de los PCs. La configuración era compatible con los PCs de la época, con un procesador de 8 MHz, 128 KB de memoria *RAM* y un monitor de 9 pulgadas. La gran arma del *Macintosh* era el sistema operativo *MacOS 1.0*, un sistema innovador desde varios puntos de vista.

Al contrario del *MS-DOS*, el *MacOS 1.0* ya utilizaba una interfaz gráfica y un ratón, lo que lo hacía mucho más fácil para trabajar. El *MacOS* continuó evolucionando e incorporando nuevos recursos, pero siempre manteniendo la misma idea de interfaz. Como detalle, ya estamos en la décima versión del *MacOS*, el *MacOS X*.

En 1984 ya existía la primera versión de *Windows*, que era una opción para los usuarios de PCs interesados en poder ejecutar una interfaz gráfica. El sistema *Windows 1.0* se ejecutaba sobre el *MS-DOS* y podía ejecutar tanto aplicaciones de *Windows* como los programas para MS-DOS. El problema era la memoria. Los PCs de la época venían con cantidades muy pequeñas de memoria *RAM* y, en la época, aún no existía la posibilidad de usar memoria virtual (que sólo fue soportada a partir del *386*).

Para poder ejecutar *Windows*, primero era necesario cargar el sistema *MS-DOS*. Prácticamente, los dos juntos ya consumían toda la memoria de un PC básico de la época. Incluso en los PCs mejor equipados no era posible ejecutar muchas aplicaciones al mismo tiempo, nuevamente por la falta de memoria.

Como las aplicaciones para *Windows* eran muy raras para la época, pocos usuarios vieron la necesidad de utilizar Windows para ejecutar las mismas aplicaciones que ya se ejecutaban (con mucha más memoria disponible) en *MS-DOS*. Sin contar que la versión inicial de *Windows* era bastante lenta y tenía varios *bugs*.

Windows comenzó a tener un primer éxito con la versión *2.1*, cuando ya eran comunes los PCs *286* con 1 MB o más de memoria *RAM*. Con una configuración más poderosa, más memoria *RAM* y más aplicaciones, finalmente empezaba a tener sentido usar *Windows*. El sistema aún tenía varios *bugs* y se colgaba con frecuencia, pero algunos usuarios comenzaron a migrar hacia él.

La nueva versión de *Windows 3.1* fue un gran paso hacia delante y ya soportaba el uso de la memoria virtual, que permitía abrir varios programas de forma simultánea aunque se agotase la memoria *RAM*. Ya existían varias aplicaciones para *Windows* y los usuarios tenían la opción de cambiar a *MS-DOS* cuando lo deseaban. Fue en esta época en la que los PCs empezaron a recuperar el terreno perdido respecto a los *Macintoshs* de la casa Apple, aunque *Windows 3.1* también se colgaba con frecuencia, pero ya disponía de muchas aplicaciones y los PCs eran mucho más baratos.

Desde el inicio de la era de los PCs, las casas Microsoft e IBM trabajaban juntas en el desarrollo del *MS-DOS* y otros programas para la plataforma PC. Pero, en 1990, las casas IBM y Microsoft se separaron y cada una se quedó con una parte del trabajo hecho, con lo cual intentaron tomar el liderazgo del mercado de los sistemas por separado. Algunos cuentan que IBM se quedó con la parte que funcionaba y que Microsoft se quedó con el resto, pero la verdad es que a pesar del sistema operativo *OS/2* de la casa IBM era técnicamente muy superior al *Windows 95* de Microsoft, fue *Windows 95* el que se llevó la mejor parte, pues era más fácil de usar y contaba con la familiaridad de los usuarios de *Windows 3.1*. El sistema *OS/2* aún es utilizado por algunos entusiastas y existen movimientos para continuar el desarrollo del sistema, pero faltan programas y *drivers*.

Otro sistema operativo que comenzó a ser desarrollado en el inicio de la década de los años 1990 es el *Linux*, que todos conocemos. *Linux* tiene la ventaja de ser un sistema operativo abierto que, actualmente, cuenta con la colaboración de centenares de miles

de desarrolladores voluntarios esparcidos por todo el mundo, además del apoyo de empresas de peso, como IBM. Pero, en sus inicios, el sistema era mucho más complicado que las distribuciones actuales y todavía no contaba con las interfaces gráficas que tenemos hoy en día.

El desarrollo de *Linux* fue gradual, hasta que hubo la explosión del acceso a *Internet* en el año 1995, cuando el sistema comenzó a ser usado por un número cada vez mayor de servidores *Web*, pues era estable y gratuito. Hoy el *IIS* de Microsoft consigue plantarle cara (por lo menos en número de usuarios) pero, en sus primeros años, *Linux* era sinónimo de servidor *Web*.

Microsoft continuó mejorando su sistema. Después de *Windows 95*, lanzaron al mercado *Windows 98* y, finalmente, *Windows ME* con todos los problemas que conocemos, pero con la buena interfaz fácil de usar y un número suficiente de aplicaciones que garantizaron la popularización de estos sistemas. Paralelamente, Microsoft desarrollaba una familia de sistemas Windows destinados a servidores, *Windows NT*, que llegó hasta la versión 4, antes de ser transformado en *Windows 2000*.

Actualmente, las dos familias de *Windows* se han unido en *Windows XP*, un sistema destinado tanto de uso doméstico como para las estaciones de trabajo y servidores, y que puede ser considerado un sistema operativo estable (al contrario de *Windows 98* y *Windows ME*) pues está basado en el núcleo de *Windows 2000*.

Mientras tanto, *Linux* continúa avanzando. En la actualidad, el sistema sólo es usado en un 2% de los equipos de sobremesa (sin los usuarios ocasionales y los que mantienen un arranque dual entre *Windows* y *Linux*), pero tiene la oportunidad de crecer bastante en el futuro con la ayuda de las aplicaciones *Gimp* y *StarOffice*, que sustituyen a *Photoshop* y *Office*, respectivamente.

3. Procesadores

El procesador es el componente básico de cualquier PC. Con el avance cada vez más rápido de la tecnología y gracias a que varias empresas se están disputando el mercado de los procesadores, éstas se ven obligadas a desarrollar proyectos cada vez más osados, a fin de producir procesadores con un mayor rendimiento a un menor costo. Esto resulta muy bueno para nosotros, los usuarios, pero también tiene sus puntos débiles, ya que con proyectos tan diferentes no se acaba de fijar un único estándar.

En un primer punto, veremos las características básicas de los procesadores modernos para, luego, pasar a discutir algunas características básicas de los procesadores, conocer los procesadores pioneros de la década de los años 70 y avanzar por los años 80 y 90, hasta llegar a la actualidad y, así, poder entender mejor su funcionamiento.

3.1 Características básicas

En el mercado existen varios modelos de procesadores, que presentan precios y rendimientos muy distintos. Cuando vamos a comprar un procesador, la primera cosa que preguntamos es su frecuencia de trabajo, medida en *Megahercios* (MHz), o millones de ciclos por segundo, también conocida por frecuencia de reloj, que en la actualidad ya se mide en *Gigahercios* (GHz), billones de ciclos por segundo. Muchas veces, un procesador con una frecuencia de trabajo más alta no es más rápido que otro que trabaja a una frecuencia un poco menor. La frecuencia de trabajo de un procesador

sólo indica los ciclos de procesamiento que se realizan en cada segundo, lo que cada procesador es capaz de hacer en cada ciclo ya es otra cosa. Esto es debido a las diferencias en la arquitectura de los diversos procesadores.

3.1.1 El coprocesador aritmético

Todos los procesadores de la familia *x86*, usada en la mayoría de PCs, son básicamente procesadores de números enteros. Sin embargo, una gran mayoría de las aplicaciones necesitan utilizar valores numéricos de una mayor precisión, así como utilizar funciones matemáticas más complejas para llevar a cabo sus tareas. Este es el caso de los programas *CAD*, los juegos con gráficos tridimensionales, el procesamiento de imágenes, etc. La función del coprocesador aritmético es, justamente, ayudar al procesador principal en el cálculo de estas funciones complejas, cada vez más utilizadas, principalmente en los juegos.

Hasta el *386*, el coprocesador sólo era un accesorio que podía comprarse por separado para ser instalado en el encaje adecuado de la placa base, existiendo un modelo equivalente para cada modelo de procesador. El procesador *8088* utilizaba el coprocesador *8087*, el *286* utilizaba el *287*, los procesadores *386SX* y *386DX* utilizaban los coprocesadores *387SX* y *387DX* respectivamente, y el *486SX* utilizaba el *487SX*. El problema de esta forma de trabajar era que, al ser pocos los usuarios que equipaban sus equipos con los coprocesadores aritméticos, la producción de estos chips era baja y, consecuentemente, los precios eran altísimos, llegando al punto que en algunos casos el coprocesador era más caro que el procesador principal. Con el aumento del número de aplicaciones que necesitaban el uso del coprocesador, fue una evolución natural su incorporación en el procesador principal a partir del *486DX*. Con esta solución, se resolvió el problema del coste de producción de los coprocesadores aritméticos.

Actualmente, el rendimiento del coprocesador determina en gran parte el rendimiento del equipo en los juegos y aplicaciones gráficas, justamente las aplicaciones donde los procesadores actuales son más demandados. El rendimiento del coprocesador es una característica que varía mucho entre los diversos procesadores actuales.

3.1.2 La memoria caché

Mientras que los procesadores se hicieron casi 10 mil veces más rápidos desde el *8088* (el procesador usado en el formato *XT*), la memoria *RAM*, su principal herramienta de trabajo, evolucionó poco. Cuando se lanzaron al mercado los procesadores *386*, se notó que las memorias no eran capaces de acompañar al procesador en velocidad, haciendo que muchas veces el procesador tuviese que quedarse "esperando" a que los datos fuesen liberados por la memoria *RAM* para poder concluir sus tareas, perdiendo mucho tiempo de ejecución y rendimiento.

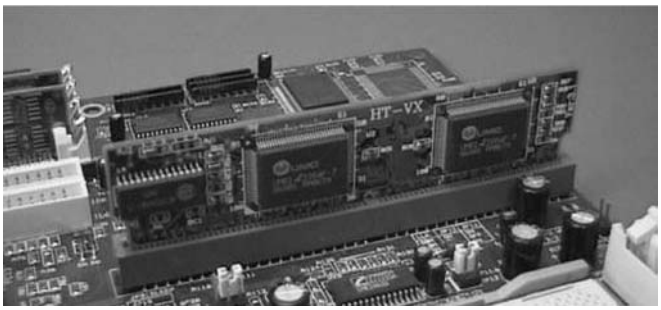
Si en la época del *386* la velocidad de las memorias ya era un factor que limitaba el rendimiento del equipo, podemos imaginarnos como entorpecería este problema el rendimiento de los procesadores que tenemos en la actualidad. Para solucionar este

problema se empezó a utilizar la memoria caché, un tipo de memoria ultrarápido que sirve para almacenar los datos usados con más frecuencia por el procesador, evitando que éste deba recurrir muchas veces a la memoria *RAM*, comparativamente mucho más lenta. Sin la memoria caché, el rendimiento del sistema quedaría limitado a la velocidad de la memoria *RAM*, pudiendo bajar hasta en un 95%. Se usan dos tipos de memoria caché, conocidas como caché primaria, o caché *L1* (*Level 1*), y caché secundaria, o caché *L2* (*Level 2*).

La memoria caché primaria está integrada en el propio procesador y es lo bastante rápida para acompañarlo en velocidad. Siempre que se desarrolla un nuevo procesador, también es preciso desarrollar un tipo más rápido de memoria caché para poder acompañarlo. Como este tipo de memoria es extremadamente caro, sólo se usa una pequeña cantidad de ella. El *486* sólo traía 8 KB, el *Pentium* traía 16 KB, el *Pentium II* y el *Pentium III* traían 32 KB, mientras que el *Athlon* y el *Duron,* de la casa AMD, traían 128 KB. Los Pentium 4 han llegado a traer hasta 256 KB.

Para complementar esta memoria caché *L1* también se usa un tipo un poco más lento de memoria caché, en forma de memoria caché secundaria que, por ser más barata, permite que se use en mayores cantidades. En los equipos *486*, el uso de 128 o 256 KB de caché *L2* era algo muy común, mientras que en los equipos más modernos el uso de 512 KB es lo mínimo. Dependiendo del procesador que se esté usando, la memoria caché *L2* puede venir integrada en el propio procesador (en todos los procesadores actuales) o formar parte de la placa base.

Siempre que el procesador necesita leer datos, primero los busca en la memoria caché *L1*. Si encuentra el dato, el procesador no perderá más tiempo ya que la memoria caché primaria funciona a la misma frecuencia. Si el dato no se encuentra en la memoria caché *L1*, entonces lo busca en la memoria caché *L2*. Si lo encuentra en la caché secundaria, el procesador perderá algún tiempo respecto a la caché *L1*, pero no tanto como el que perdería en el caso de tener que acceder directamente a la memoria *RAM*. Si los datos no están en ninguna de las dos memorias caché, al procesador no le queda más remedio que perder varios ciclos de procesamiento esperando que los datos sean entregados por la memoria *RAM*, mucho más lenta que las memorias caché.



Un módulo de memoria caché en la placa base

Antiguamente, era muy común que las placas base viniesen equipadas con soquetes apropiados para la memoria caché, que permitían añadir más memoria caché. Los módulos adicionales, llamados módulos *COAST* (*Cache On A STick*) eran relativamente accesibles, haciendo que mucha gente realizase la ampliación. Sin embargo, actualmente esta posibilidad ya no existe, pues la gran mayoría de los procesadores ya traen la memoria caché *L2* integrada, sin permitir ninguna modificación, ya que no es aconsejable abrir el procesador y soldar más memoria caché.

O sea que, actualmente, la cantidad de memoria caché que deseemos tener en el procesador, o raramente en la placa base, debe ser decidida antes de realizar la compra del equipo, eligiendo entre las distintas opciones disponibles. Una vez adquiridos el procesador y la placa base no nos será posible hacer ninguna modificación al respecto de forma simple y sencilla.

3.1.3 Procesadores RISC y procesadores CISC

Siempre ha existido una gran polémica en torno a cuál de estas dos plataformas es mejor. Tal vez podamos considerar inútil estar hablando sobre esto, pero es interesante comprender la diferencia entre estas dos plataformas para entender varios aspectos de los procesadores modernos.

Un procesador *CISC* (*Complex Instruction Set Computer*, u ordenador con un conjunto complejo de instrucciones), es capaz de ejecutar varios centenares de instrucciones complejas diferentes, siendo extremadamente versátil. Algunos ejemplos de procesadores *CISC* son el *386* y el *486*.

A inicios de la década de los años 80, la tendencia era construir chips con conjuntos de instrucciones cada vez más complejos. Sin embargo, algunos fabricantes consideraron seguir el camino contrario, creando el formato *RISC* (*Reduced Instruction Set Computer*, u ordenador con un conjunto reducido de instrucciones). Al contrario de los complejos procesadores *CISC*, los procesadores *RISC* sólo son capaces de ejecutar algunas instrucciones simples. Justamente por esto, los chips basados en esta arquitectura son más simples y más baratos. Otra ventaja de los procesadores *RISC*, es que al tener un número menor de circuitos internos, pueden trabajar a frecuencias más elevadas. Un ejemplo son los procesadores *Alpha,* que en el año 97 ya podían trabajar a unos nada despreciables 600 MHz para la época.

Puede parecer extraño que un chip que es capaz de ejecutar pocas instrucciones pueda ser considerado, por muchos, más rápido que otro que ejecuta centenares de instrucciones. Pero un procesador *RISC* es capaz de ejecutar tales instrucciones de forma mucho más rápida. La idea principal es que a pesar de que un procesador *CISC* sea capaz de ejecutar centenares de instrucciones diferentes, sólo algunas son usadas de forma frecuente.

Entonces, podríamos crear un procesador optimizado para ejecutar sólo las instrucciones simples que se utilizan con una mayor frecuencia. Como de cualquier forma, poca

gente programa directamente en lenguaje *ensamblador*, bastaría alterar los compiladores para que los programas fuesen compatibles con los nuevos procesadores.

Es indiscutible, sin embargo, que los procesadores *CISC* son mejores en la mayoría de tareas, principalmente por su gran número de recursos. Por eso, en vez de la consolidación de una de las dos tecnologías, actualmente vemos procesadores híbridos, que son esencialmente procesadores *CISC* pero que incorporan muchos recursos encontrados en los procesadores *RISC* (o viceversa).

Examinando desde un punto de vista un poco más práctico, la ventaja de una arquitectura *CISC* es que ya tenemos muchas de las instrucciones guardadas en el propio procesador, lo que facilita el trabajo de los programadores, que ya disponen de prácticamente todas las instrucciones que serán usadas en sus programas. En el caso de un chip estrictamente *RISC*, el programador tendría un poco más de trabajo, pues sólo dispondría de instrucciones simples y tendría que combinar varias instrucciones siempre que necesitase ejecutar alguna tarea más compleja.

En los chips actuales, que son una mezcla de las dos arquitecturas, unimos las dos capacidades. Internamente, el procesador lleva a cabo sólo instrucciones simples. Estas instrucciones internas, variando en función del tipo de procesador, se adaptan al proyecto del chip. Por ejemplo, las instrucciones internas de una *K6* son diferentes a las de un *Pentium*. Sobre estas instrucciones internas disponemos de un circuito decodificador, que convierte las instrucciones complejas utilizadas por los programas, en varias instrucciones simples que pueden ser entendidas por el procesador.

El conjunto básico de instrucciones usadas en los equipos PC es conocido por el conjunto *x86*. Este conjunto está compuesto por un total de 187 instrucciones, que son las más utilizadas por todos los programas. Además de este conjunto principal, algunos procesadores también traen añadidas instrucciones alternativas, que permiten a los programas ejecutar algunas tareas de forma más rápida de lo que sería posible sólo con el uso de las instrucciones *x86*. Algunos ejemplos de conjuntos alternativos de instrucciones son el *MMX* (usado a partir del *Pentium MMX*), el *3D-Now!* (usado por los procesadores de la casa AMD a partir del *K6-2*) y el *SSE* (soportado a partir del *Pentium III*).

3.1.4 Front End y Back End

Cualquier procesador actual puede ser dividido en dos bloques básicos, *Front End* y *Back End*. El *Front End* corresponde a los circuitos que decodifican las instrucciones, es el caso del *Hardware decoder* y el *Microcode decoder* junto con algunos componentes más, como los circuitos de *Branch Prediction*, que ordenan las instrucciones de forma que el procesador pueda procesar el mayor número posible de instrucciones por ciclo, y la memoria caché *L1*. Estos componentes son la "puerta de entrada" del procesador, teniendo la función de preparar las instrucciones para que sean realizadas a continuación por el procesador.

El *Back End* es la parte del procesador que procesa las instrucciones, estando compuesto, básicamente, por las diferentes unidades de ejecución. En las unidades de ejecución es donde los procesadores ofrecen más diferencias en la forma de procesar las instrucciones ya decodificadas.

Por ejemplo, el *Athlon* posee un total de 9 unidades de ejecución. De éstas, 3 son unidades de ejecución de lectura/escritura de los datos en la memoria. Otras 3 unidades de ejecución son de punto flotante (que forman el coprocesador aritmético). A efectos de comparación, el *Pentium III* sólo posee dos. Las otras 3 unidades de ejecución son unidades de procesamiento de números enteros.

En términos de instrucciones 3D, el *Athlon* cuenta con el conjunto de instrucciones *3D-Now!*, incorporado en los procesadores AMD y que permite mejorar el rendimiento del procesador en los juegos y aplicaciones 3D. El *Athlon* también lleva el viejo juego de instrucciones *MMX*, que garantiza alguna ganancia en las aplicaciones multimedia, aunque no sea mucha.

3.2 Del 8086 al Pentium

El primer microprocesador fue lanzado por la casa Intel en 1971 y se llamaba *4004*. Se trataba de un procesador extremadamente simple, formado por cerca de 2300 transistores, pero que fue el precursor de los procesadores que tenemos en la actualidad.

La llamada ley de *Moore*, que lleva el nombre del fundador de la casa Intel, Gordon Moore, proclama que la potencia de los procesadores se dobla cada 18 meses. A pesar de que esta previsión fue hecha a finales de la década de los años 70, continúa siendo cierta hasta los días de hoy, con una precisión notable.

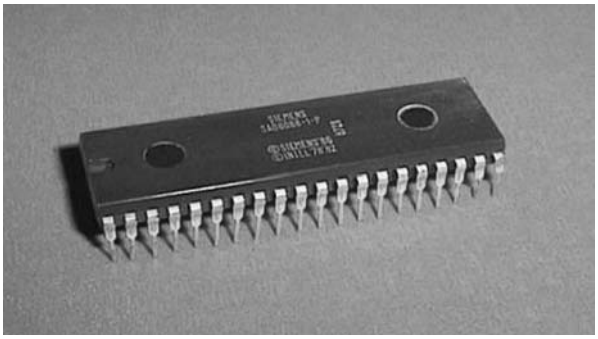
Desde entonces se ha recorrido un largo camino. Se realizaron enormes inversiones y muchos de los mayores genios del planeta trabajaron en la búsqueda de soluciones para cuestiones cada vez más complejas. A continuación vamos a examinar los avances hechos desde el *8088*, usado en el *XT*, hasta el *Pentium*, donde estudiaremos la introducción de recursos como el modo protegido y la multiplicación del reloj, y en lo que afectan al funcionamiento del procesador.

Entendiendo estos conceptos nos será más fácil entender las diferencias entre los distintos procesadores de la historia y los que aparecieron después, ya que continuaron utilizando los mismos conceptos básicos.

3.2.1 El 8088

El *8088* era una versión económica del procesador *8086*, que había sido lanzado por la casa Intel en el año 78. Cuando la casa IBM estaba desarrollando su ordenador personal, llegó a pensar en usar el *8086*, pero acabó escogiendo el *8088* debido a su menor coste de producción.

Tanto el *8086* como el *8088* eran procesadores de 16 bits y se consideraban avanzadísimos para la época, a pesar de que eran extremadamente simples comparados con los procesadores actuales. La diferencia entre ellos es que el *8088*, a pesar de trabajar internamente con palabras binarias de 16 bits, usaba un bus de sólo 8 bits, lo que permitió a la casa IBM utilizar los mismos componentes usados en los ordenadores de 8 bits de la época, que eran más baratos que los componentes de 16 bits.



El procesador 8088

Esta arquitectura permitió al primer PC competir en la misma franja de precio que los ordenadores de 8 bits más populares y, al mismo tiempo, poseer un rendimiento superior, debido a su procesador de 16 bits. El *8088* era capaz de soportar hasta 1 MB de memoria *RAM* y funcionaba a 4.77 MHz, recursos increíbles para la época, ya que estamos hablando de un procesador lanzado a finales del año 1979.

Hablando de recursos, sólo por curiosidad, el PC original de la casa IBM lanzado en agosto de 1981 sólo poseía 64 KB de memoria *RAM* (la versión más simple venía con sólo 16 KB), un monitor *MDA* monocolor de 12 pulgadas, usaba una unidad de disquetes de 5 pulgadas y 1/4 de sólo 160 KB y no llevaba ningún disco duro.

El sistema operativo usado era el *MS-DOS 1.0* (en la época aún llamado *PC-DOS*) que fue desarrollado por la casa Microsoft y basado en un sistema operativo más simple, llamado *QDOS*, comprado a la empresa Seattle Computers, una pequeña empresa desarrolladora de sistemas operativos. En realidad, la casa Microsoft fue la segunda opción de IBM, después de ser rechazada su propuesta de licencia por la casa Digital Research, que en la época desarrollaba versiones de su *CP/M* para varias arquitecturas diferentes.

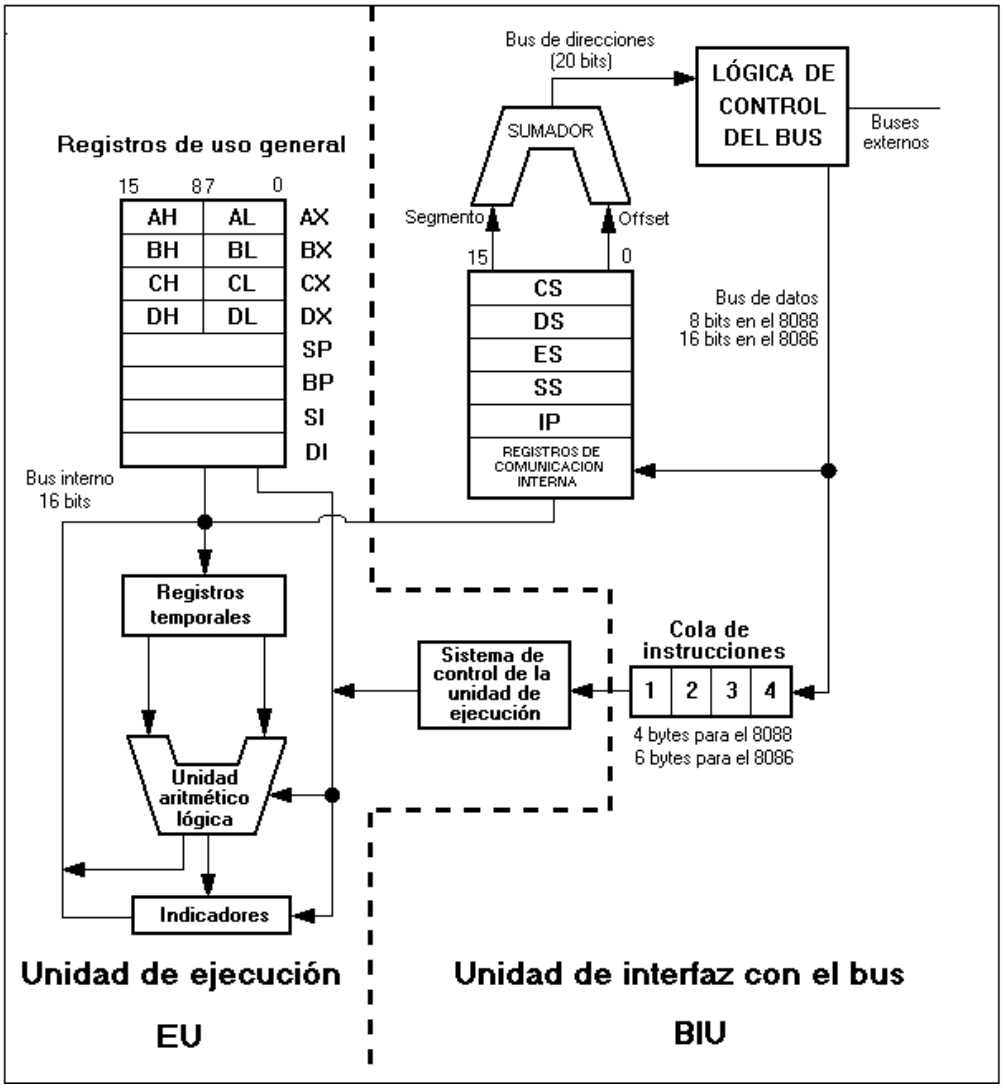
Dos años después, fue lanzado el PC *XT*, que a pesar de continuar usando el procesador *8088* de 4.77 MHz, venía incrementado con 256 KB de memoria *RAM*, un disco duro de 10 MB, un monitor *CGA* y el sistema operativo *MS-DOS 2.0*.

Aún con la aparición de los equipos *286*, el *XT* se continuó vendiendo ya que era más barato. Los fabricantes de ordenadores clónicos crearon proyectos de equipos *XT* más avanzados, equipados con procesadores *8088* de 8 MHz, discos duros de una mayor capacidad y hasta 640 KB de memoria *RAM*.

La segmentación de direcciones

Un recurso muy interesante usado en el *8088* es la segmentación de direcciones, que permitió aumentar la cantidad de memoria *RAM* soportada por el procesador.

Para que el procesador pueda acceder a la memoria *RAM* es necesario que la memoria se divida en direcciones. Cada Byte de la memoria recibe una dirección única. Como el *8088* sólo podía trabajar con palabras binarias de 16 bits, en un principio no le era posible acceder a más que 64 KBytes de memoria RAM, ya que 16 bits sólo permitían 65.536 combinaciones diferentes (2 elevado a 16).



Esquema del funcionamiento interno de los procesadores 8086 y 8088

Si el *8088* sólo podía acceder a 64 KB de memoria *RAM*, los equipos basados en él estaban limitados y sólo podían ejecutar programas muy simples. Para tener una idea, 64 KB no daban ni para cargar el sistema operativo *DOS 3.0*.

Para solucionar este problema fue adoptada una solución bastante ingeniosa: a pesar de que el procesador continuaba pudiendo acceder a sólo 64 KB de memoria *RAM* cada vez, se crearon 4 bits más de direccionamiento, que permitían el acceso a 16 bloques de memoria. Como cada bloque poseía 64 KB, llegamos a 1 MB de capacidad total en la memoria *RAM*. Básicamente, se crearon 16 áreas diferentes de memoria, cada una con 64 KB, que era el máximo que el *8088* podía gestionar.

El procesador podía acceder a una única área cada vez. Si por ejemplo, se estaba usando el bloque 1 y, de pronto, era preciso leer un dato grabado en el bloque 2, era preciso limpiar todas las direcciones relativas al bloque 1 y cargar las direcciones del bloque 2. En ese momento, el procesador perdía el acceso al bloque 1 y pasaba a acceder sólo al segundo bloque. Cuando era preciso lee o grabar datos de nuevo en el bloque 1 (o cualquier otro bloque), se cargaban las direcciones relativas a él y se perdía el acceso al bloque 2.

3.2.2 El 286

El procesador *286* fue lanzado al mercado a principios del año 1982, sólo 6 meses después de que la casa IBM lanzase su primer PC al mercado. Sin embargo, el *286* empezó a utilizarse a partir de 1984, cuando IBM lanzó su PC *AT*.

Un procesador 286

Esta demora es justificable, pues, para el lanzamiento de un ordenador que usase el nuevo procesador de la casa Intel, fue preciso desarrollar toda una nueva arquitectura. Desde la tarjeta gráfica a la torre, prácticamente se modificó todo, lo que sumado a la burocracia y a los largos periodos de pruebas antes del lanzamiento final, precisó de un cierto tiempo. Actualmente, el periodo de desarrollo de los periféricos es mucho más corto. Casi siempre, cuando se lanza un nuevo procesador ya existen placas base para él, pues el desarrollo se hace de forma simultánea.

El *286* conllevó varios avances respecto al *8088*. El *286* utilizaba palabras binarias de 16 bits, tanto interna como externamente, lo que permitía el uso de periféricos de 16

bits mucho más avanzados que los usados en el PC original y en el *XT*. El coste de estos periféricos no llegó, esta vez, a ser un gran obstáculo, pues mientras se estaba desarrollando el PC *AT* ya podían encontrarse a unos precios asequibles.

El principal avance del *286* fueron sus dos modos de trabajo, conocidos como *Modo Real* y *Modo Protegido*. En el modo real, el *286* se comportaba exactamente como un *8086* (a pesar de ser más rápido), ofreciendo una compatibilidad total con los programas ya existentes. En cambio, en el modo protegido, el *286* aprovechaba todo su potencial, incorporando funciones más avanzadas como la capacidad para acceder a hasta 16 MBytes de memoria RAM (usando los 24 bits de direccionamiento del *286*), la multitarea, la memoria virtual en el disco duro y la protección de memoria.

Al conectarse, el procesador operaba en modo real y, a través de una instrucción, pasaba a trabajar en modo protegido. El problema era que, trabajando en modo protegido, el *286* dejaba de ser compatible con los programas escritos para el modo real, inclusive con el propio *MS-DOS*.

Para empeorar la cosa, el *286* no poseía ninguna instrucción que hiciese posible que el procesador volviese al modo real, lo cual sólo era posible reiniciando el equipo. Esto significaba que un programa escrito para ser ejecutado en modo protegido no podía usar ninguna de las rutinas del *MS-DOS* de acceso a los dispositivos, haciendo inaccesibles el disco duro, la tarjeta gráfica, la unidad de disquetes, etc. a menos que se desarrollasen e incorporasen al programa todas las rutinas necesarias para un correcto acceso a los distintos dispositivos. Eso era completamente inviable para los desarrolladores, pues para proyectar un simple juego, era prácticamente necesario desarrollar todo un nuevo sistema operativo.

Además de esto, el programa desarrollado sólo funcionaba en los equipos con procesadores *286*, que aún eran una minoría en la época. De hecho, sólo algunas versiones del sistema operativo *UNIX* y una versión del *OS/2* fueron desarrolladas para utilizar el modo protegido del *286*.

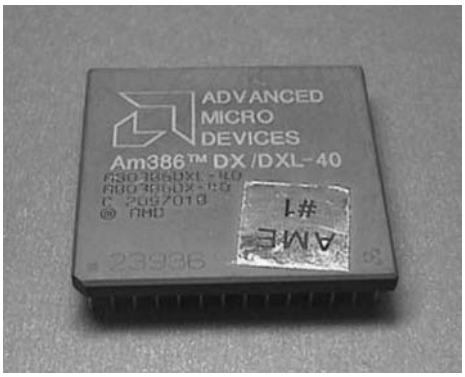
Básicamente, los equipos basados en el *286* eran usados para ejecutar aplicaciones en modo real, que también podían ser ejecutadas en un *XT*, sólo aprovechando la mayor velocidad de trabajo del *286*. Hablando en términos de velocidad, la primera versión del *286* funcionaba a sólo 6 MHz, siendo lanzada inmediatamente después una nueva versión a 8 MHz, que fue la usada en el PC *AT*. Posteriormente, fueron desarrolladas versiones de hasta 20 MHz. Debido a los varios y distintos cambios en la arquitectura, destacando el acceso más rápido a la memoria y las alteraciones en el conjunto de instrucciones, que permitían realizar muchas operaciones de manera más rápida y eficiente, un *286* conseguía ser casi 4 veces más rápido que un 8088.

3.2.3 El 386

El *386* fue lanzado a finales del año 85, tres años y medio después del *286*. Esta vez, la dirección de la casa IBM tardó mucho para llegar a un acuerdo y desarrollar un siste-

ma basado en el *386*, dando tiempo a la casa Compaq a salir al mercado. Este fue un verdadero cambio pues, de pronto, las compañías percibieron que ya no estaban obligadas a continuar con la casa IBM. Cualquiera que tuviera la tecnología suficiente podía salir al mercado, como hizo la casa Compaq.

A partir de ahí, la casa IBM comenzó, gradualmente, a perder el liderazgo en el mercado de los ordenadores, quedándose como uno más entre los distintos fabricantes de ordenadores personales.



Un procesador 386 de la casa AMD

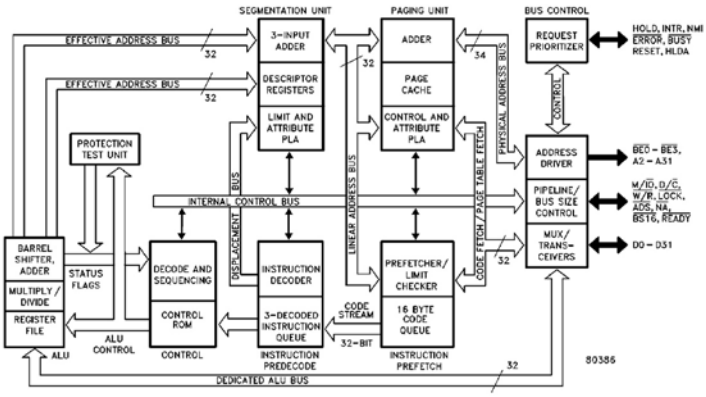
El *386* llevaba varios recursos nuevos. Para empezar, el 386 trabajaba, tanto interna como externamente, con palabras de 32 bits y era capaz de acceder a la memoria usando un bus de 32 bits, permitiendo una transferencia de datos dos veces mayor. Como el *386* podía trabajar con palabras binarias de 32 bits, era posible acceder a hasta a 4 GBytes de memoria *RAM* (2 elevado a 32), sin usar la segmentación de direcciones como en el *8088* y en el *286*.

Al igual que el *286*, el *386* continuaba soportando los dos modos de trabajo. La diferencia era que en el *386* ya era posible alternar entre el modo real y el protegido de forma totalmente libre. Un programa que funcionase sobre *DOS* podía cambiar el procesador al modo protegido para beneficiarse de sus ventajas y volver al modo real, siempre que necesitase usar alguna subrutina del *DOS* de manera transparente para el usuario. En este caso, era usado un programa DPMI (*DOS Protected Mode Interface* o interfaz DOS de modo protegido) para hacer el cambio entre los dos modos de trabajo.

Cuando el programa necesitaba usar alguna subrutina de *DOS*, éste pasaba el comando al intercambiador y se quedaba esperando. Por su parte, el intercambiador pasaba el procesador al modo real, ejecutaba el comando, cambiaba el procesador al modo protegido y entregaba el resultado a la aplicación, que continuaba trabajando como si nada.

Este esquema de cambio también se utilizaba en las versiones de *Windows 3.x*, que ya incluían todas las rutinas necesarias, dispensando a los programas *DPMI*. Los siste-

mas operativos *Windows 95/98* también podían cambiar al modo real en el caso de necesitar cargar algún *driver* de modo real. Sin embargo, debido al modo virtual *8086*, que veremos inmediatamente a continuación, no era necesario cambiar el procesador al modo real para ejecutar aplicaciones de *MS-DOS* en *Windows 95/98*.



El esquema de funcionamiento de un procesador 386

Disponer de un procesador *386* era el requisito mínimo para poder ejecutar cualquier sistema operativo o aplicación de modo protegido moderno. Con un *386*, un mínimo de memoria *RAM* y el espacio suficiente en el disco duro podíamos ejecutar *Windows 95* y la mayoría de las aplicaciones correspondientes, aunque muy lentamente debido a la poca potencia del procesador. Con un simple *286*, podíamos ejecutar, como máximo, el sistema operativo *DOS* y algunas aplicaciones más simples y que, además, sólo trabajasen en modo real. También era posible ejecutar *Windows 3.0* aunque en modo estándar, donde era posible acceder a todos los 16 MB de memoria *RAM* permitidos por el *286*, pero sin poder usar la memoria virtual ni la multitarea.

La introducción de la memoria caché

Los procesadores *386* con frecuencias de trabajo por encima de los 20 MHz eran muy rápidos para las memorias *RAM* de la época. Por eso, en cada acceso, el procesador tenía que quedarse "esperando" a que los datos fuesen liberados por la memoria *RAM* para poder concluir sus tareas, perdiendo mucho tiempo y rendimiento. Para solucionar este problema, pasaron a ser usadas pequeñas cantidades de memoria caché en la gran mayoría de las placas base para equipos *386* y superiores.

La memoria caché es un tipo de memoria ultrarápida que almacena los datos usados con más frecuencia por el procesador evitando, en la gran mayoría de los casos, que el procesador pierda tiempo recogiendo los datos directamente de la memoria *RAM*. Una pequeña cantidad de memoria caché, por poca que sea, es capaz de mejorar bastante la velocidad en el intercambio de datos entre el procesador y la memoria *RAM*.

A pesar de ser más rápido que la memoria *RAM*, el *386* aún no era un procesador muy rápido y, justamente por esto, aún no era tan dependiente del rendimiento de la

memoria caché como los procesadores actuales. Un 386 equipado con memoria caché era de un 20 a un 30% más rápido que un *386* de la misma frecuencia sin memoria caché, mientras que un procesador moderno puede ser hasta 20 veces más lento en el caso de deshabilitar tanto la memoria caché *L1* como la memoria caché *L2*.

3.2.4 El 386SX

Como el *386* era un procesador de 32 bits fue preciso desarrollar una nueva categoría de chipsets y circuitos para trabajar con él, lo que acabó encareciendo bastante los sistemas basados en el *386* y alejando a muchos compradores. Para intentar solucionar este problema, la casa Intel optó por lanzar una versión de bajo coste del *386*, bautizada como *386SX*, que a pesar de continuar funcionando internamente con palabras de 32 bits, se comunicaba con la memoria *RAM* y los demás periféricos usando palabras de 16 bits (como en el *286*). Sólo para diferenciar los dos procesadores, la casa Intel pasó a llamar *386DX* al *386* original.

Esta arquitectura permitió que se pudieran aprovechar los mismos periféricos usados en las placas base de los equipos *286*, haciendo mucho más asequibles los equipos basados en el *386SX*. Para hacernos una idea, un PC básico equipado con un *386SX* podía costar casi la mitad que un equipo con una configuración parecida basado en el *386DX*. A pesar de ser una buena opción en términos de relación coste-rendimiento, se quedaba corto en rendimiento comparado con un *386DX* de la misma frecuencia, ya que a pesar de que los procesadores eran internamente idénticos, el *386SX* usaba prácticamente los mismos componentes usados en los equipos *286*, accedía a la memoria *RAM* usando palabras de 16 bits y las placas base no tenían memoria caché.

Modo real y modo protegido

Trabajando en modo real, el procesador funcionaba exactamente como un *8086*, sólo variaba la mayor velocidad de proceso. No solamente el *386*, sino todos los procesadores posteriores pueden alternar entre el modo real y el modo protegido libremente, siempre que sea necesario. En el modo real, ejecutábamos *MS-DOS* y otras aplicaciones de modo real más antiguas, mientras que en el modo protegido ejecutábamos *Windows* y sus programas.

Seguramente, al intentar ejecutar un programa antiguo, ya nos hemos encontrado con un enigmático mensaje de falta de memoria a pesar de que los manuales del programa digan que sólo necesita 500 o 600 KB de memoria *RAM* y nosotros dispongamos de mucha más. Estos mensajes surgen porque estos programas se ejecutan con el procesador trabajando en modo real donde, como en el *8086*, sólo es capaz de reconocer el primer MB de la memoria *RAM*. Por su parte, este primer MB se encuentra subdividido en dos bloques, conocidos como memoria convencional y memoria extendida o superior.

La memoria convencional se corresponde con los primeros 640 KBytes y es el área de memoria usada por los programas que operan en modo real. Los restantes 384 KBytes

son llamados de memoria extendida y son reservados para almacenar una copia del *BIOS*, que pasa a poder ser ejecutada de forma más rápida, ya que la memoria *RAM* es más rápida que el chip de memoria *ROM*, o *Flash*, donde es almacenado el *BIOS* original. Esta copia del *BIOS* es llamada de *Shadow*, o sombra, y sirve para aumentar el rendimiento general del sistema. La memoria extendida también es usada para almacenar sombras de los BIOS de otros dispositivos, como tarjetas gráficas, también aumentando la velocidad de operación de estos periféricos.

A pesar de que existan 640 KBytes de memoria convencional para ser usada por cualquier programa que trabaje en modo real, no toda esta memoria queda disponible, ya que parte de ella es usada por el sistema operativo *MS-DOS* y los *drivers* de los dispositivos de modo real. Es posible liberar más memoria convencional editando los archivos de inicialización del *DOS*, consiguiendo poder ejecutar estos programas.

Cuando se inicia el ordenador, el procesador está operando en modo real. El encargado de pasarle el comando para que cambie al modo protegido es el sistema operativo. En el caso de *Windows*, este proceso pasa durante la carga del sistema. En el modo protegido, el procesador es capaz de reconocer toda la memoria *RAM* instalada en el sistema, además de incorporar recursos como la multitarea y la memoria virtual. En este modo podemos usar la interfaz gráfica de *Windows* y ejecutar sus aplicaciones.

3.2.5 Los recursos del modo protegido

Sólo en el modo protegido los procesadores incorporan los recursos más avanzados, que permiten la existencia de los programas que tenemos en la actualidad. A partir del *386* se incorporaron pocas funciones nuevas a los nuevos procesadores. Básicamente, evolucionaron sólo en términos de velocidad. El modo protegido lleva cuatro nuevos recursos: la memoria virtual, la multitarea, la memoria protegida y el modo virtual *8086*.

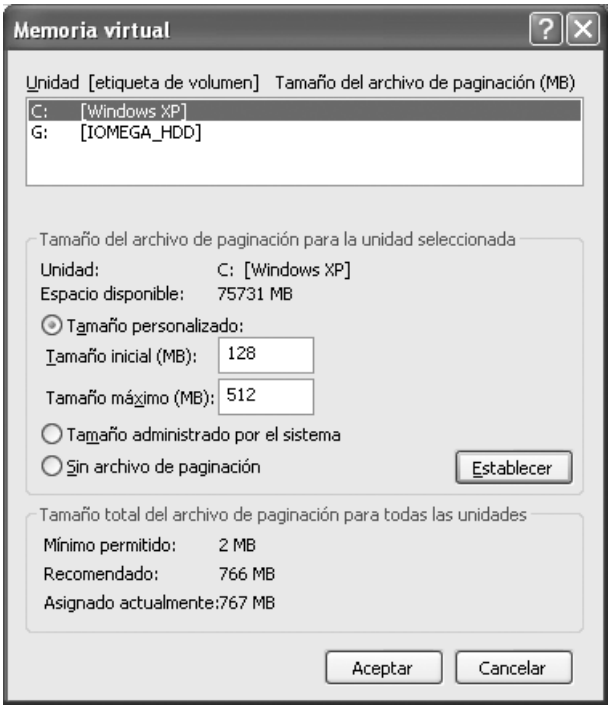
La memoria virtual

La capacidad del *386* de trabajar con varias aplicaciones al mismo tiempo (multitarea) era realmente muy útil, pero esta característica conllevaba un pequeño problema: al abrir varias aplicaciones de forma sucesiva, la memoria *RAM* del sistema se agotaba. Para corregir este problema, el modo protegido utilizó el recurso de la memoria virtual, que permitía crear un archivo temporal en el disco duro, llamado *Swap File*, o archivo de intercambio, que funcionaba como una extensión de la memoria *RAM*, permitiendo abrir cuántas aplicaciones fuesen necesarias, hasta que se agotaba el espacio del disco duro.

Por ejemplo, sólo W*indows 2000/XP Professional* junto con los servicios básicos ocupa cerca de 40 MB de memoria. Si deseamos abrir *Word*, serán necesarios 10 MB más. Si el equipo sólo poseyese 32 MB de memoria se crearía un archivo temporal de 18 MB en el disco duro, que almacenaría los datos que no se pudiesen almacenar en la memoria RAM.

El problema de usar memoria virtual es que el disco duro es centenares de veces más lento que la memoria *RAM*. Un disco duro razonablemente rápido posee un tiempo de acceso en torno a los 10 milisegundos (milésimas de segundo), mientras que un módulo de memoria *PC-100* posee un tiempo de acceso inferior a los 10 nanosegundos (billonésimas de segundo) o sea, un tiempo de acceso un millón de veces menor. En términos de tasas de transferencia, nuevamente tenemos un contraste destacado: 800 MB/s para el módulo de memoria y de 5 a 20 MB/s (dependiendo del modelo) para el disco duro, de forma aproximada. Debido a estas diferencias y, a pesar de que los programas funcionan normalmente usando memoria virtual, el sistema va volviéndose cada vez más lento.

En las versiones de *Windows 3.x* era necesario reservar una cantidad de espacio del disco duro para la memoria virtual, cantidad que podía configurarse libremente a través del *Panel de Control*. El problema es que este espacio no quedaba disponible para las otras tareas. Por ejemplo, si teníamos un disco de 800 MBytes, y reservábamos 200 MBytes para la memoria virtual, nos quedábamos con sólo 600 MB para instalar programas y guardar archivos. Si por otro lado, reservábamos poco espacio para la memoria virtual, nos quedábamos con poca memoria para abrir varios programas y trabajar con archivos de gran tamaño.



El cuadro de diálogo para configurar la cantidad de memoria virtual en Windows XP

A partir de *Windows 95* este problema fue resuelto con la adopción de un archivo de intercambio dinámico, que iba aumentando o disminuyendo de tamaño en consonan-

cia con la necesidad de memoria, evitando el desperdicio de espacio en disco que teníamos en las versiones anteriores del sistema *Windows*. A partir de la primera versión de *Windows 95*, también existió una administración más racional de los recursos del sistema, situando los archivos usados con más frecuencia en la memoria *RAM* (o memoria caché, dependiendo de la importancia del archivo), y sólo dejando los archivos menos usados medida en el archivo de intercambio. Esta medida disminuyó bastante la pérdida de rendimiento derivada del uso de la memoria virtual.

En *Windows 2000/XP* es posible determinar un valor inicial y un valor máximo para el archivo de intercambio. En el caso de *Linux*, a fin de mejorar el rendimiento, los desarrolladores optaron por crear un sistema de archivos propio para la memoria virtual.

La multitarea

Multitarea significa ejecutar más de una tarea a la vez. A pesar de que en la vida real no es muy fácil hacer dos cosas al mismo tiempo, desde el punto de vista de un ordenador este proceso es relativamente simple. Todas las aplicaciones son cargadas en la memoria *RAM* y el procesador ejecuta algunas instrucciones de cada aplicación a la vez. Como el procesador es capaz de ejecutar varios millones de instrucciones por segundo, este cambio es hecho de manera transparente, como si las aplicaciones se estuvieran realmente ejecutando al mismo tiempo. Mientras el procesador se dedica a una aplicación, las demás quedan paralizadas, esperando su turno.

La memoria protegida

Usando el recurso de la multitarea casi siempre tendremos varias aplicaciones cargadas en la memoria, ya sea en la memoria *RAM* o en el archivo de intercambio. Si no existiese ningún control por parte del procesador, una aplicación podría expandir su área de memoria, invadiendo áreas de otras aplicaciones y causando bloqueos en el equipo.

Un editor de imágenes, por ejemplo, necesita ocupar más memoria conforme las imágenes se van abriendo, creando o modificando. Sin ninguna orientación por parte del procesador, se ocuparían las áreas adyacentes, que podrían estar vacías u ocupadas por otra aplicación.

Para solucionar el problema fue desarrollado el recurso de la protección de memoria, que consiste en que el procesador aísla el área de memoria ocupada por cada aplicación, impidiendo que se ocupen las otras áreas sin ton ni son. Si, por casualidad, el programa necesita más memoria, el propio procesador irá a buscar un área vacía de memoria y ordenará a la aplicación que ocupe el área reservada.

Existen dos tipos de multitarea, denominadas multitarea preventiva y multitarea cooperativa, que difieren justamente por el uso o no de la memoria protegida. *Windows 3.x*, a pesar de ser considerado un sistema operativo multitarea, no era capaz de usar el recurso de la protección de memoria. En sus distintas versiones se usaba la multi-

tarea cooperativa, que consiste en que cada aplicación use los recursos del procesador en un determinado espacio de tiempo y esperar nuevamente, mientras el procesador se dedica a otra aplicación, a que llegue su turno para continuar ejecutando sus tareas. En este caso, la alternancia entre los distintos programas no es comandada por el sistema y sí por las propias aplicaciones. Con este escenario, una aplicación sin control podría monopolizar el sistema consumiendo todos los recursos del procesador por un largo espacio de tiempo o, peor todavía, ocupando áreas de memoria ya ocupadas por otras aplicaciones, causando uno de los famosos errores *GPF* (*General Protection Falt* o error de protección general) que tanto atormentaban a los usuarios de las versiones de *Windows 3.x*.

Las versiones de *Windows 95/98* usaban la multitarea preventiva, aislando las áreas de memoria ocupadas por las distintas aplicaciones en uso. Esto garantiza una mayor estabilidad que la que teníamos en *Windows 3.11*. Sin embargo, el modo de la multitarea preventiva implementada en *Windows 95*, así como en *Windows 98* y *Windows Millennium*, que están basados en el mismo *kernel* (núcleo) de *Windows 95*, aún tenía dos graves problemas. El primero es que, cuando se ejecutaba un programa de 16 bits, *Windows 95* pasaba a usar la multitarea cooperativa para poder ejecutar el programa dejando de proteger las áreas de memoria y volviéndose tan vulnerable como *Windows 3.11*.

Sin embargo, usando sólo aplicaciones de 32 bits los bloqueos también eran usuales, pues en *Windows 95*, los servicios del sistema no tenían prioridad sobre las aplicaciones. Esto significa que en el caso de que una aplicación entrase en un bucle, podía consumir todos los recursos del procesador y, en este caso, el sistema operativo se quedaba paralizado, sin saber cómo cerrar la aplicación y restaurar el sistema, obligando al usuario a reiniciar el equipo y perder el trabajo que no hubiese guardado. En realidad, se suele decir que *Windows 95/98* utilizaban la multitarea semipreventiva, pues no utilizaban todos los recursos de una verdadera multitarea.

La solución a este problema llegó con Windows NT. Desde sus primeras versiones, *Windows NT* ya era muy estable en este aspecto, pues implementaba la multitarea preventiva de forma completa. Las tareas ejecutadas por el sistema operativo eran prioritarias sobre las de cualquier otra aplicación. Esto significaba que en ninguna situación, una aplicación podía pasar por encima del sistema operativo y consumir todos los recursos del procesador como acontecía en Windows 95/98. En la práctica, significaba que el sistema también se podía bloquear debido a algún *bug*, pero si una aplicación se bloqueaba o intentaba invadir un área de memoria no correspondiente a ella, simplemente se cerraba, permitiendo que todas las demás aplicaciones continuasen trabajando sin problemas.

Windows 2000/XP están basados en el *kernel* de *Windows NT* y mantienen el mismo sistema de funcionamiento. Al haber derivado de *Unix*, *Linux* utiliza la multitarea preventiva desde sus primeras versiones y, por ello, *Linux* es considerado uno de los sistemas más estables, siendo usado en los servidores más importantes del planeta.

El modo virtual 8086

A pesar de que operando en modo real el procesador es totalmente compatible con cualquier programa antiguo, sería imposible ejecutar una aplicación en modo real sobre *Windows 95/98* o cualquiera otro sistema operativo que utilice el modo protegido. Sería necesario cerrar *Windows* y hacer que el procesador volviese al modo real para poder ejecutar la aplicación.

Pensando en esta posible limitación, los desarrolladores de la casa Intel idearon el modo virtual 8086, en el que el procesador, operando en modo protegido, era capaz de simular varias situaciones de modo real, cada una con 1 MB de memoria y total acceso al hardware del equipo, conocidas por máquinas virtuales. Es como si dentro del *386* se abriesen varios *XT* completos, uno para cada programa ejecutado en modo real. Es justamente el modo virtual *8086* lo que permite abrir ventanas de *DOS* en *Windows*. Como el procesador continúa en modo protegido, cada máquina virtual tiene su área aislada en la memoria y el programa se ejecuta sin perjudicar la estabilidad del sistema.

3.2.6 El 486

El *386* fue el primer procesador que llevaba el conjunto de instrucciones *x86*, que son soportadas por todos los procesadores modernos. A partir del *386*, surgieron varias mejoras, pero sólo en términos de rendimiento.

Un procesador 486DX2 de la casa AMD

A pesar de no traer nuevas instrucciones, el *486* conquistó un lugar en la historia de los procesadores por llevar varios recursos que todavía continúan siendo usados en los procesadores actuales. En primer lugar, el *486* fue el primer procesador que llevaba la memoria caché integrada. Sólo eran 8 KBytes, pero eran capaces de entregar datos en cada ciclo del procesador. Como los fabricantes continuaron incluyendo la memoria caché en la placa base, un poco más lenta pero en mayor cantidad, surgió la distinción entre la memoria caché *L1* y *L2*.

Otra evolución fue el coprocesador aritmético. En lugar del carísimo componente que debía adquirirse de forma separada, el coprocesador pasó a ser un componente de

serie del procesador. Éste fue el impulso que faltaba para la popularización de varios programas y el surgimiento de juegos muy elaborados. Con todas estas mejoras, un *486* era casi dos veces más rápido que un *386* de la misma frecuencia. En algunas aplicaciones que dependían del coprocesador aritmético en gran medida, un *486* llegaba a ser 10 veces más rápido que un *386*.

Como ya hizo anteriormente con el *386*, la casa Intel creó un *486* de bajo coste llamado *486SX*. Con la aparición del *486SX*, el *486* original pasó llamarse *486DX*. Los dos compartían la misma arquitectura, pero el 486SX venía sin el coprocesador aritmético, lo que lo hacía mucho más lento en las aplicaciones gráficas y científicas. Para los usuarios del *486SX* existía la opción de comprar el *487SX*, un coprocesador aritmético que se vendía por separado.

Salieron al mercado versiones del *486* que trabajaban a 25, 33 y 40 MHz. Sin embargo, se creó un tope pues no existían circuitos de apoyo capaces de trabajar a más de unos 40 MHz. Para solucionar ese problema fue creado el recurso de la multiplicación de reloj, a través del cual el procesador trabajaba internamente a una velocidad mayor que la que utilizaba la placa base. Entonces, se lanzaron al mercado los procesadores *486DX2* (que trabajaban al doble de la frecuencia de la placa base) e, inmediatamente después, los *486DX4* (que trabajaban al triple de la frecuencia de la placa base). Con esto, surgieron también las placas base ampliables, que permitían actualizar el procesador con sólo configurar algunos jumpers de la misma placa.

Un procesador 486DX4 que trabajaba a 100 MHz

Los procesadores *486*, a partir del *486DX* a 33 MHz fueron los primeros en utilizar un ventilador; hasta entonces se incorporaban disipadores con menos de un centímetro de altura. Conforme los procesadores pasaron a producir cada vez más calor, los ventiladores fueron creciendo en la misma proporción, hasta llegar a los que podemos ver en la actualidad.

La multiplicación de reloj

En cualquier ordenador, los datos son transmitidos y procesados en forma de señales eléctricas. El procesador es muy pequeño, ya que no mide más de 1 o 1,2 centímetros

cuadrados. La placa base, por su parte, es bastante mayor. Gracias a esta diferencia de proporciones, es más fácil desarrollar un procesador capaz de operar a 2 GHz que una placa base capaz de acompañarlo en velocidad. A pesar de que las señales eléctricas recorren los circuitos a una velocidad muy cercana a la velocidad de la luz, estamos hablando de billones de transmisiones por segundo.

El recurso de la multiplicación de reloj surgió para evitar que los procesadores se quedasen limitados a la frecuencia de la placa base. Por ejemplo, en un *Pentium III* a 800 MHz, la placa base sólo trabaja a 100 MHz. El multiplicador es de 8x.

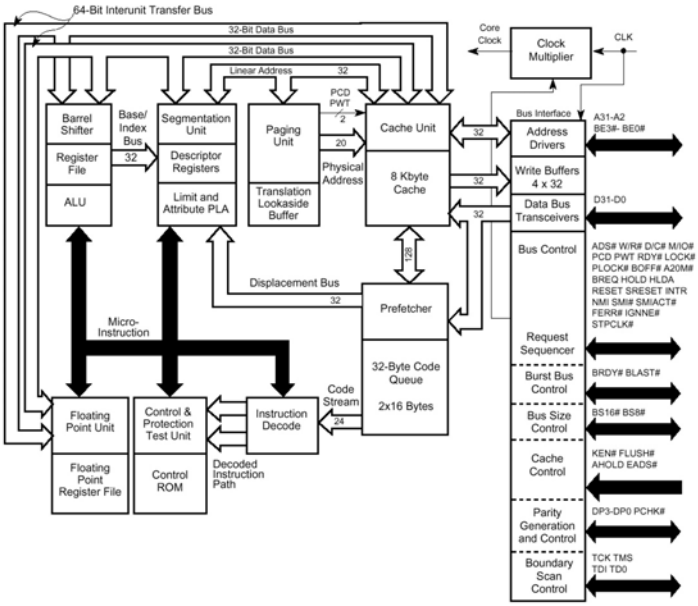


Diagrama de funcionamiento de un procesador 486

Hoy en día, los procesadores llevan memorias caché *L1* y *L2* integradas, trabajando a la misma frecuencia que el resto del procesador, lo que disminuye mucho la dependencia de la velocidad de la memoria *RAM*, que siempre opera a la misma frecuencia que la placa base. Aún así, cuanto mayor sea el multiplicador, mayor será la pérdida de rendimiento. Un buen ejemplo de esto es una comparación entre el *Celeron* 766 (que usa bus de 66 MHz) y el *Celeron* 800 (que ya usa un bus de 100 MHz). A pesar de que la frecuencia de operación es casi la misma, el *Celeron* 800 llega a ser un 20% más rápido, gracias al acceso más rápido a la memoria.

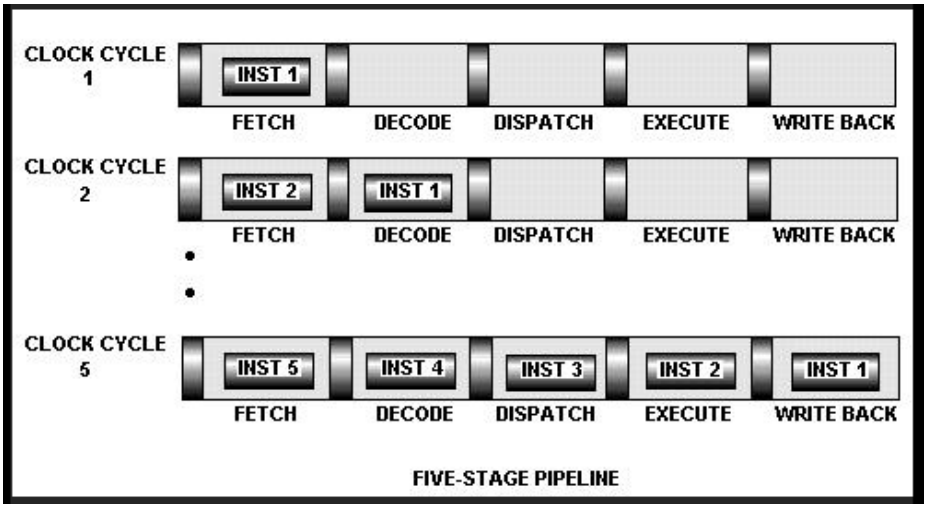
A pesar de sus limitaciones, el recurso de la multiplicación del reloj es indispensable en la actualidad, pues sin él sería imposible desarrollar procesadores más rápidos, ya que no es posible aumentar la frecuencia de las placas base y de los demás periféricos en la misma proporción que en los procesadores. Por ejemplo, si un *Pentium III* hubiese tenido que trabajar a la misma frecuencia de la placa base, no hubiésemos pasado de los 100 o 133 MHz.

En los *486*, *Pentium*, *Pentium MMX* y *K6* era necesario configurar el multiplicador manualmente, a través de algunos *jumpers* de la placa base. Pero, a partir del *Pentium II* la misma placa base es capaz de detectar el multiplicador de forma automática.

Pipeline

Hasta el *386*, los procesadores de la familia *x86* eran capaces de procesar una sola instrucción en cada ciclo de reloj. Una instrucción simple podía ser ejecutada en un sólo ciclo de reloj, mientras que las instrucciones más complejas tardaban varios ciclos de reloj en ser procesadas.

Para mejorar el rendimiento del *486*, la casa Intel decidió usar el *pipeline*, una técnica inicialmente usada en los procesadores *RISC* que consiste en dividir el procesador en varios estadios distintos. El *486* poseía un *pipeline* de 5 niveles, o sea, estaba dividido en 5 estadios. Cuando se cargaba una nueva instrucción, de entrada pasaba por el primer nivel, que trabajaba en ella durante un sólo ciclo de reloj, pasándola al segundo nivel. La instrucción continuaba siendo procesada sucesivamente por el segundo, tercer, cuarto y quinto nivel del procesador.



El funcionamiento del pipeline de 5 niveles

La ventaja de esta técnica era que el primer nivel no necesitaba quedarse esperando que la instrucción pasase por todos los demás niveles para cargar la próxima instrucción, y así podía cargar una nueva instrucción cuando se liberaba de la primera, o sea, después del primer ciclo del reloj.

Las instrucciones se movían dentro del procesador en el orden en que eran procesadas. Aunque la instrucción ya se hubiese procesado al pasar por el primer o segundo nivel, tenía que continuar su camino y pasar por todos los demás niveles. Si por casualidad la instrucción no hubiese sido completada después de pasar por los 5 niveles, volvía a pasar por el primero y era procesada de nuevo hasta que se concluía su pro-

cesamiento. De esta manera, conseguíamos que el procesador fuese capaz de procesar simultáneamente, en un único ciclo de reloj, varias instrucciones que normalmente tardaban varios ciclos en ser procesadas.

El uso de los 5 niveles de *pipeline* en el *486* no llegaba a multiplicar por cinco el rendimiento del procesador. En realidad, el rendimiento no llegaba ni a doblarse, pero la ganancia era significativa.

3.2.7 El Pentium

De la mima manera que el *486*, el *Pentium* era un procesador de 32 bits capaz de acceder a hasta 4 GB de memoria *RAM*. Pero el nuevo procesador trajo varias mejoras que lo hicieron mucho más rápido que los de la anterior generación. El *Pentium* era de un 65 a un 100% más rápido que un *486* de la misma frecuencia de reloj. Como el coprocesador aritmético también fue completamente remodelado, el *Pentium* también acabó siendo más rápido en las aplicaciones que demandaban un gran número de cálculos en punto flotante.

Un procesador Pentium

Los procesadores *Pentium* existieron en versiones de los 60 a los 200 MHz, utilizando siempre la multiplicación de reloj (sólo con la excepción de las versiones de 60 y 66 MHz). Como en la época de los equipos *486*, las placas base para los procesadores *Pentium* (con excepción de las placas más antiguas) soportaban varias frecuencias de bus y varios multiplicadores. En la mayoría de los casos era posible configurar la placa base para utilizar cualquier procesador de la familia *Pentium*.

Mejoras en la memoria caché L1

El primer cambio aportado por el *Pentium* fue el aumento de la cantidad de memoria caché *L1*, que pasó a ser de 16 KB, el doble que en el *486*. Además de este aumento de la capacidad también le fueron implementados tres nuevos recursos, que consiguieron que aumentase la eficiencia de la caché. La primera medida fue la división de la memoria caché en dos bloques de 8 KB, uno dedicado a almacenar los datos y el otro a almacenar las instrucciones. Esta división permitió que tanto las instrucciones que

iban a ser ejecutadas por el procesador (comparación, multiplicación, suma, decisión, etc.) como los datos que iban a ser procesados pudiesen accederse en la memoria caché de forma simultánea, aumentando su eficiencia.

Si, por ejemplo, un programa le ordenaba al procesador que leyese un número almacenado en la memoria y verificase si era mayor o menor que 10, nos encontrábamos con dos instrucciones (leer el número y compararlo con el número 10) y dos variables (el número 10 y el número almacenado). Con una memoria caché unificada, como en el *486*, primero se leían las instrucciones y después las variables. En la memoria caché dividida del *Pentium*, las instrucciones y los datos podían leerse en el mismo ciclo de reloj, ahorrando tiempo.

Otra modificación fue la ampliación del bus de datos entre el procesador y la memoria caché. Mientras que en el 486 podían leerse o grabarse hasta 128 bits de datos en cada ciclo de reloj, en el *Pentium* podían leerse o grabarse hasta 256 bits en la caché de instrucciones y 256 bits más en la caché de datos. Como ambas memorias caché podían accederse de forma simultánea, teníamos un bus total de 512 bits, cuatro veces mayor del teníamos en el *486*. Este bus más ancho permitía que mayores cantidades de datos se pudiesen leer a través de la caché en el mismo espacio de tiempo, permitiendo a la caché acompañar la mayor velocidad de procesamiento del *Pentium*.

La última mejora fue la adopción de una caché *Write Back*, que era capaz de comprobar las operaciones de lectura de datos en la memoria *RAM* y las operaciones de escritura. La memoria caché usada en el *486* sólo comprobaba las operaciones de lectura, lo que permitía ganar tiempo al procesador en la lectura de los datos, pero no ayudaba a la hora de grabar los datos, cuando se perdían varios ciclos de reloj hasta que la memoria *RAM* estaba disponible.

El desvío dinámico

Después de concluir una instrucción y para que el procesador no perdiese un ciclo de reloj esperando que la memoria caché o la memoria *RAM* enviasen la próxima instrucción a ser procesada, en el *Pentium* se incluyó un *búfer* de extracción previa. Este pequeño circuito almacenaba las próximas instrucciones que iban a procesarse, formando una especie de fila. En realidad, el *búfer* de extracción previa funcionaba como una especie de memoria caché *L0*, quedándose entre el procesador y la memoria caché *L1*. La mayor utilidad de este *búfer* era la previsión del resultado de las operaciones en la toma de decisiones. Si, por ejemplo, llegaba al procesador una instrucción como "Si X > Y entonces Z = K, sino Z = Q" el *búfer* cargaba tanto la instrucción siguiente para X < Y como para X > Y, haciendo que fuese el resultado que fuese de la operación anterior, la próxima instrucción ya estuviese cargada en el *búfer*.

El *búfer* de extracción previa también ayudaba a la memoria caché a cargar con antelación los datos que el procesador podía necesitar. En el caso del ejemplo anterior, se cargarían tanto el valor K como el valor Q en la memoria caché.

Coprocesador aritmético más rápido

El coprocesador aritmético del *Pentium* fue completamente remodelado. Se alteraron algunas de las instrucciones, que pasaron a ser ejecutadas de forma mucho más rápida y, como en el procesador principal, el coprocesador del *Pentium* también pasó a utilizar un *pipeline* para aumentar la velocidad de ejecución de las instrucciones.

Sumadas a las mejoras en la arquitectura, el coprocesador aritmético del Pentium pasó a ser cerca de 5 veces más rápido que el utilizado en el *486*, haciendo que el procesador fuese mucho más rápido en las aplicaciones que demandaban un gran número de operaciones.

La arquitectura superescalar

Otra mejora más del *Pentium* y uno de los principales motivos de su mayor rendimiento, fue la adopción de una arquitectura superescalar.

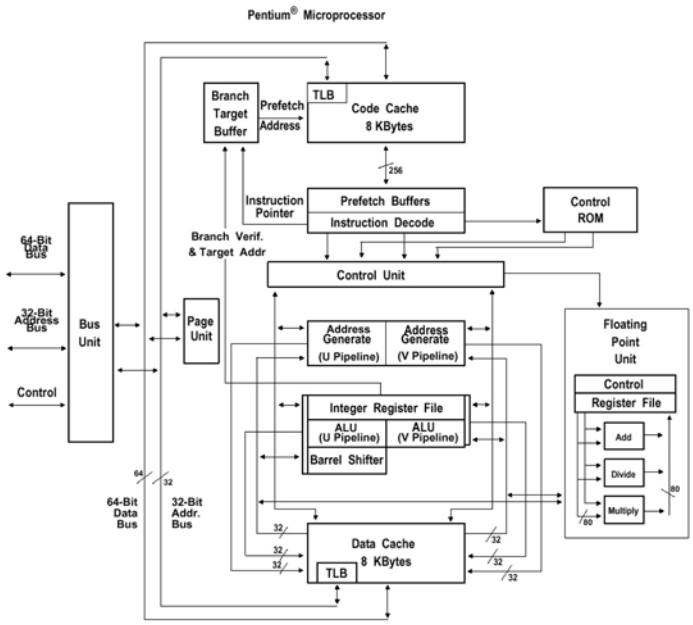


Diagrama de funcionamiento de un Pentium

Internamente, el *Pentium* trabajaba como dos procesadores de 32 bits distintos (conocidos como canaleta U y canaleta V), siendo capaz de procesar dos instrucciones en cada ciclo de reloj (una en cada canaleta). Cada canaleta poseía un acceso total a la memoria caché, a la memoria *RAM* y a los demás componentes del equipo. También fue incluida una unidad de control, con la función de supervisar el funcionamiento de las dos canaletas y dividir las tareas entre ellas. Teóricamente, el uso de dos canaletas distintas debía doblar el rendimiento del *Pentium*, ya que en vez de una, podrían eje-

cutarse dos instrucciones en cada ciclo de reloj, pero en la práctica, existían algunas limitaciones que no aumentaban tanto el rendimiento.

Si por ejemplo, un programa ordenaba que el procesador sumase 4 números, X + Y + Z + T, el procesador podía usar la canaleta U para sumar X e Y y la canaleta V para sumar Z y T en el primer ciclo de reloj, pero en el segundo ciclo sólo había un cálculo a realizar, el resultado de las dos sumas. En este caso, sólo una de las canaletas podía ser usada; la segunda se quedaba sin trabajo. A final de cuentas, sólo hubo una ganancia del 33%, ya que en vez de tardar 3 ciclos para ejecutar el cálculo, el procesador sólo tardó 2.

Si la continuación del cálculo dependiera de la conclusión del cálculo anterior, como en "(X + Y) x 3", la segunda canaleta no era usada de nuevo, pues el procesador tenía primero que sumar X e Y para después multiplicar el resultado por 3. En este caso, no había ninguna ganancia, pues el procesador tardaba los mismos dos ciclos de reloj que ya eran necesarios con una sola canalización.

En definitiva, la segunda canalización permitía un rendimiento de un 30 o un 40% superior al rendimiento que se obtenía con una sola canalización. Si el software era modificado y optimizado para ser ejecutado en un procesador con dos canalizaciones, ordenando las instrucciones para ocupar la segunda canaleta durante la mayor parte del tiempo, podíamos conseguir un 15% más de rendimiento, pero no era posible conseguir el doble de rendimiento como cabía esperar.

> **Nota:** por ejemplo, esto puede notarse al recompilar el *kernel* de *Linux*. Es posible optimizar el código para cualquier procesador, desde un *486* a un *Pentium 4*. Esta optimización permite que *Linux* utilice todos los recursos del procesador en cuestión, obteniendo un mayor rendimiento.

La optimización de un programa para ser ejecutado en un procesador *Pentium* no lo hacía incompatible con los procesadores de una sola canalización (el *486*), ni disminuía su rendimiento en estos procesadores, ya que se ejecutaban las mismas instrucciones, sólo que en un orden diferente.

Ejecución especulativa

Si el procesador tuviese que procesar una instrucción de toma de decisión como "Si X > 0 entonces Y = 30, sino Y = 20", mientras la primera canaleta del procesador verificaba si X era mayor o menor que 0, la segunda canaleta, en vez de quedarse ociosa podía ejecutar una de las dos opciones siguientes (atribuir el valor 30 o el valor 20 a la variable Y).

En el próximo ciclo de reloj, cuando la primera canaleta hubiese terminado de procesar su instrucción, tendríamos un 50% de oportunidades de que la segunda canaleta hubiese adivinado cual sería la instrucción siguiente y haberla ejecutado. El nombre de ejecución especulativa fue dado porqué la segunda canaleta escogía la instrucción a ejecutar de forma aleatoria, entre las posibles opciones.

Acceso más rápido a la memoria

El *Pentium* era capaz de acceder a la memoria usando palabras binarias de 64 bits, el doble que en el caso del *486*, que la accedía con palabras de 32 bits. Este nuevo recurso permitía que se leyesen 8 Bytes en cada ciclo, en vez de sólo 4, doblando la velocidad de acceso a la memoria. Esto disminuyó bastante el antiguo problema de la lentitud de las memorias, pero sólo de forma provisional, pues inmediatamente surgieron procesadores *Pentium* utilizando multiplicadores de reloj cada vez mayores.

Como la mayoría de las placas base para procesadores *Pentium* utilizaban módulos de memoria de 72 vías, que son módulos de 32 bits, era preciso usarlos a pares. El procesador accedía a cada pareja de módulos como si se tratase de un único módulo de memoria, llegando a los 64 bits necesarios. Aunque tenía la capacidad de acceder a la memoria con 64 bits y estaba compuesto internamente por dos procesadores de 32 bits, el *Pentium* continuaba siendo un procesador de 32 bits. Estos nuevos recursos sólo sirvieron para mejorar el rendimiento del procesador.

Multiprocesamiento

Echando un vistazo al mercado de *Workstations* (máquinas muy rápidas, destinadas a las aplicaciones pesadas como procesamiento de imágenes 3D o vídeo) y servidores de red, la casa Intel incluyó en el *Pentium* el recurso del multiprocesamiento simétrico, que permitía el uso de dos procesadores en la misma placa base. En este caso, era preciso adquirir una placa base especial, con los encajes para dos procesadores y un chipset que soportase el multiprocesamiento.

Dado que la división de las tareas entre los dos procesadores no se realiza de forma automática por el *chipset*, es preciso que el sistema operativo sea capaz de reconocer los dos procesadores y acceder a ellos de forma individual, dividiendo las tareas entre ellos de la mejor forma posible. Si el sistema operativo no ofrece soporte para el multiprocesamiento, como en *Windows 95/98*, sólo se usa uno de los dos procesadores. En este caso, será como si tuviéramos un único procesador instalado. La mayoría de los sistemas operativos, incluyendo *Windows NT*, *Windows 2000* y *Windows XP*, *Linux* y la mayoría de las versiones de *Unix* soportan el multiprocesamiento. Las excepciones más notables son *Windows 95/98* y *Windows ME*.

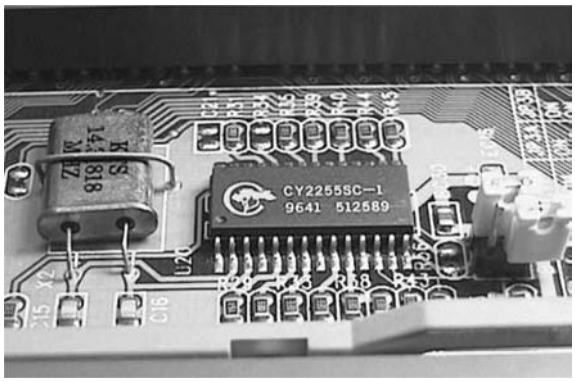
A pesar de que, por lógica, el rendimiento debería doblarse con dos procesadores trabajando en paralelo, la ganancia difícilmente pasa de un 40 o un 60% en la práctica, pues es difícil organizar la ejecución de las tareas de modo que ambos procesadores estén ocupados todo el tiempo, así como era difícil mantener las dos canaletas del *Pentium* ocupadas el 100% del tiempo.

Hasta hace poco tiempo, el recurso del multiprocesamiento fue exclusividad de los procesadores Intel. Tanto el *Pentium*, como el *Pentium MMX*, el *Pentium II* y el *Pentium III* soportaban el uso de hasta dos procesadores, mientras que el *Xeon* soporta el uso de hasta cuatro procesadores (8 con la adición de un circuito especial en la

placa base). Con excepción del *Athlon MP*, todos los demás procesadores de las casas *AMD* y *Cyrix* no soportaban el multiprocesamiento, lo que no llegó a ser una gran desventaja para un usuario doméstico, ya que fuera del ramo de los servidores de alto rendimiento este recurso era muy poco común.

Clock y overclock

Al contrario de lo que muchos piensan, la velocidad de trabajo de los procesadores no es fija, pero sí que está determinada por la placa base. En la placa base tenemos un pequeño cristal de cuarzo, llamado generador de reloj, que vibra algunos millones de veces por segundo, con una precisión casi absoluta. Las vibraciones de este cristal son usadas para sincronizar los ciclos de la placa base, que sabe que cada vibración del cristal debe generar un cierto número de ciclos de procesamiento.



El generador de reloj en la placa base

El procesador no posee un generador de reloj y por eso trabaja usando la señal recibida desde la placa base. Por ejemplo, en un *Pentium MMX* a 200 MHz, la placa base funciona a 66 MHz, y el multiplicador es de 3x, lo que significa que para cada ciclo de la placa base, el procesador genera 3 ciclos.

Justamente por estar limitada a la frecuencia indicada por la placa base, la frecuencia del procesador no es fija; puede ser mayor o menor que la especificada dependiendo de la configuración de la placa base. Hasta la actualidad, para mantener la compatibilidad con varios procesadores y dado que las placas base pueden trabajar a distintas frecuencias, es posible hacer que el procesador trabaje más rápido configurando la placa base para trabajar a una mayor frecuencia. Esta técnica se conoce por *overclock*.

Por ejemplo, un *Pentium* 120 usaba un bus 60 MHz y un multiplicador de 2x. Si configurábamos la placa base para trabajar a 66 MHz, manteniendo el multiplicador a 2x, el procesador pasaba a trabajar a 133 MHz. Si la frecuencia de la placa base se aumentaba a los 75 MHz, el procesador funcionaba a 150 MHz.

En muchos casos, el procesador también aceptaba un multiplicador mayor. Por ejemplo, un AMD *K6* a 266 MHz trabajaba con la placa base operando a 66 MHz y usando

un multiplicador de 4x. Si aumentábamos el multiplicador a 4.5x manteniendo la misma frecuencia de la placa base conseguíamos hacer funcionar al procesador a 300 MHz. El rendimiento de un procesador trabajando con overclock era idéntico al de un procesador "normal" funcionando a esa velocidad.

Generalmente, los fabricantes adoptan un cierto margen de seguridad, vendiendo el procesador con una frecuencia un poco inferior a la máxima frecuencia a la que puede trabajar. Es este margen de seguridad el que permite realizar el *overclock*; simplemente hacíamos que el procesador funcionase a su máxima frecuencia. Este margen varía en función del fabricante y del procesador. Por eso, algunos procesadores aceptan *overclocks* mayores que otros.

Obviamente, el *overclock* también comporta algunas desventajas. Haciendo que el procesador trabaje a una frecuencia mayor de la ideal podemos tener problemas de bloqueos, calentamiento y, además, disminuye vida útil del procesador.

3.2.8 Otros procesadores

Además de los procesadores "principales" que hemos visto hasta este punto, existieron algunos modelos lanzados como procesadores de bajo coste, que eran una buena alternativa de ampliación para los que tenían un procesador antiguo y no querían gastar mucho en un procesador.

486DLC y 486SLC

Estos dos procesadores fueron la tentativa de la casa Cyrix de entrar en el mercado de los procesadores de bajo coste, ofreciendo una opción barata de ampliación a los usuarios de equipos *386* y también una opción de procesador para equipos de bajo coste.

Estos procesadores eran básicamente procesadores *386* que incorporaban una pequeña memoria caché *L1* de sólo 1 KB. La caché no hacía milagros pero ya era suficiente para aumentar un poco el rendimiento del procesador, lo que sumado al bajo precio de venta fue suficiente para que varios usuarios invirtieran en la actualización, ya que los *486DLC* y *486SLC* eran totalmente compatibles con las placas base para equipos *386*. Debemos recordar que, como usaban el *386* de patrón, estos procesadores no poseían coprocesador aritmético, pudiendo ser acoplados a ellos el *387DLC* o el *387SLC*, que debían ser comprados por separado.

Bajo la licencia de la casa Cyrix, la casa Texas Instruments desarrolló versiones propias del *486DLC* y *486SLC*, preservando la misma arquitectura pero aumentando la cantidad de memoria caché *L1* hasta los 8 KB.

AMD 5x86

Al inicio, la casa AMD producía clones de los procesadores Intel, utilizando los proyectos desarrollados por la casa Intel y pagando *royalties* a cambio. Sin embargo, debido a varias divergencias la alianza acabó siendo disuelta y la casa AMD pasó a

crear sus propios proyectos de procesadores. A pesar de que, durante mucho tiempo, la casa AMD tenía que contentarse con un distante segundo lugar en el mercado, produciendo básicamente procesadores de bajo coste, actualmente compite directamente con la casa Intel también en el ramo de los procesadores de alto rendimiento, consiguiendo la mayoría de las veces mantener unos precios más bajos que la competencia.

Pero, volviendo a la cita con la historia, la casa Intel después de lanzar el *486DX4-100* abandonó el desarrollo de procesadores *486* para dedicarse al desarrollo del *Pentium*. Con la intención de presentar un procesador que poseyera un rendimiento semejante a un *Pentium* de bajo coste pero que, al mismo tiempo, tuviese un precio competitivo, la casa AMD continuó el desarrollo de su procesador *486*, lanzando una versión de 120 MHz (que trabajaba usando un bus de 40 MHz y un multiplicador de 3x), e inmediatamente después una versión de 133 MHz.

Por cuestiones de marketing, la casa AMD llamó *5x86* a este 486 de 133 MHz, lo que confundió a muchos usuarios, que pensaron que se trataba de un procesador semejante al *Pentium*. El AMD *5x86* utilizaba placas base para *486*, sólo necesitaba que la placa fuese capaz de situar el multiplicador a 4x. La frecuencia se quedaba en 33 MHz, totalizando sus 133 MHz.

El procesador 5x86 de la casa AMD

Como el AMD *5x86* no dejaba de ser un *486* funcionando a 133 MHz, su rendimiento fue sólo un 33% superior al de un *486DX4-100*, siendo más o menos equivalente a un *Pentium* de 75 MHz. De hecho, otra medida de marketing tomada por la casa AMD en la época fue crear el índice *Pr*, o *Pentium Rating*, comparando el rendimiento del *5x86* con el del *Pentium*. El *5x86* de 133 MHz recibió un índice *Pr* 75, indicando que poseía un rendimiento semejante al que presentaba un *Pentium* de 75 MHz.

Cyrix Cx5x86

Además de desarrollar proyectos de procesadores *486*, que fueron fabricados por la casa Texas Instruments, la casa Cyrix lanzó un procesador que mezclaba los recursos del *486* y del *Pentium*, ofreciendo un rendimiento bastante superior al de un *486*.

Este procesador fue bautizado como *Cx5x86* y traía una memoria caché *L1* de 16 KB, además de algunas otras mejoras que hacían que su rendimiento fuese cerca de un

35% superior al de un *486* de la misma frecuencia de reloj. La versión de 100 MHz del *Cx5x86* tenía un rendimiento equivalente al del AMD *5x86* de 133 MHz y al del *Pentium* 75, mientras que la versión de 120 MHz rivalizaba en términos de rendimiento con un *Pentium* 90.

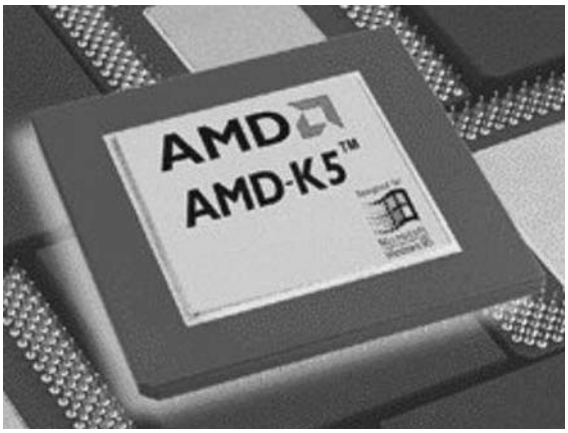
El procesador Cx5x86 de la casa Cyrix

Como en el caso del AMD 5x86, el *Cx5x86* era totalmente compatible con las placas base para los *486*, y sólo bastaba configurar la placa con un multiplicador de 3x y un bus de 33 MHz para instalar la versión de 100 MHz y un multiplicador de 3x y un bus de 40 MHz para utilizar la versión de 120 MHz.

AMD K5

Finalmente y después de muchos atrasos, la casa AMD consiguió lanzar un procesador que compitiese de forma directa con los *Pentium*. El *K5*, sin embargo, no llegó a convertirse en muy popular debido a su atrasado lanzamiento. Cuando sa-lieron las versiones con índices *Pr* 120 y *Pr* 133 del *K5*, la casa Intel ya había lanzado las versiones de 166 y 200 MHz del *Pentium*.

En vez de intentar copiar el proyecto de la casa Intel, la casa AMD optó por desarrollar un procesador completamente nuevo, técnicamente superior al *Pentium*.



Un K5 de la casa AMD

El *K5* también utilizaba una arquitectura superescalar, pero en vez de dos, poseía cuatro canalizaciones. La memoria caché *L1* también fue ampliada, pasando a ser de 24 KB dividida en dos bloques, uno de 16 KB para las instrucciones y otro de 8 KB para los datos.

Sin embargo, el coprocesador aritmético no fue muy mejorado, presentando un rendimiento casi un 50% inferior al presentado por el coprocesador aritmético del *Pentium*, debido principalmente a la ausencia de *pipeline*.

Pentium Overdrive

Como ya hizo con los antiguos *386SX*, Intel lanzó (o lo intentó, pues este procesador nunca llegó a ser muy vendido) también un *Pentium* de bajo coste. Este procesador, a pesar de tener un funcionamiento interno idéntico al de un *Pentium* utilizaba las placas base para procesadores *486* y, por eso, era llamando *Overdrive*.

La casa Intel lanzó el *Overdrive* en versiones de 63 MHz (25 MHz x 2.5) y 83 MHz (33 MHz x 2.5) pero, para que utilizasen las placas de los *486* que operaban a frecuencias mucho más bajas y accedían a la memoria con sólo 32 bits, estos procesadores perdían rendimiento al ser comparados con un *Pentium* normal. El *Overdrive* de 63 MHz presentaba un rendimiento idéntico al del *486DX4-100*.

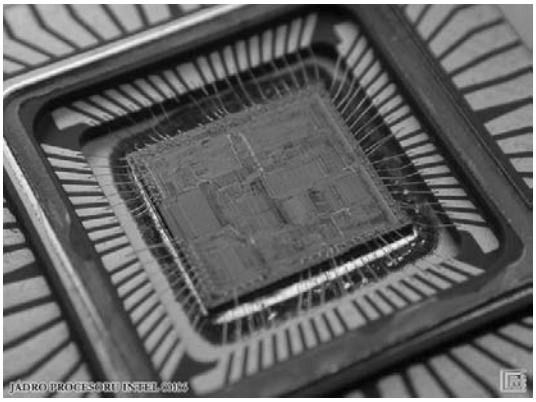
Además del bajo rendimiento, el *Overdrive* era extremadamente caro en contradicción con su origen de bajo coste y acabó siendo una mala opción de compra. Después de este primer fracaso, la casa Intel continuó intentando lanzar sucesivamente varios procesadores *Overdrive*, entre ellos una versión del *Pentium MMX* que funcionaba en placas base *soquete 7* antiguas y una versión del *Pentium II* que funcionaba en placas base para *Pentium Pro*.

A pesar de la propaganda hecha por algunos "especialistas", ninguno de estos modelos del *Overdrive* fue una opción de compra que mereciese ser tenida en cuenta, pues debido a su baja producción siempre fueron mucho más caros que los procesadores equivalentes.

La única ventaja de utilizar un procesador *Overdrive* era que se podía continuar utilizando la misma placa base, cambiando sólo el procesador. Sin embargo, en la mayoría de los casos, a causa de la baja producción costaba más el procesador *Overdrive* que unnuevo procesador normal más su correspondiente placa base.

Un tal 186

El *8086* fue el padre del *8088*, que equipó los primeros PCs. Después vinieron los procesadores *286*, que también hemos visto. Pero, entre las dos generaciones, existió un modelo intermedio, poco conocido pero igualmente importante. El *80186* fue una evolución del 8086 que trajo algunas instrucciones nuevas y un sistema de tolerancia a los errores.



Una imagen del inédito 80186

3.3 Del Pentium MMX al Pentium II

El *8086* fue lanzado en el año 79, doce años antes que el *Pentium MMX*, lanzado en el año 1997. Dieciocho años en la industria de los semiconductores se corresponden con 3 o 4 generaciones de procesadores, una eternidad.

Pero del año 97 hasta hoy las cosas han avanzado aún mucho más rápido. En parte a causa de la competencia entre las casas Intel, AMD, Cyrix y hasta empresas como IBM y Motorola, que también fabrican *chips*, a pesar de no ser directamente para el mercado del PC. Otro factor muy importante es el gran aumento en el número de procesadores vendidos.

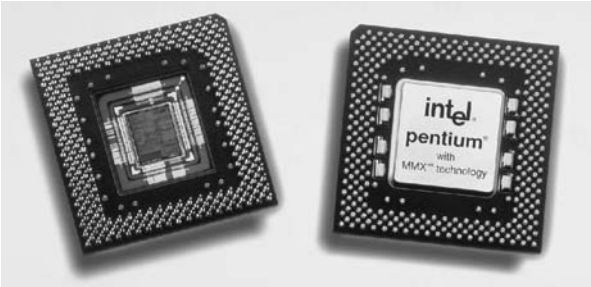
En la década de los años 70, los pocos ordenadores personales que podían encontrarse a la venta eran producidos casi artesanalmente y se vendían algunos centenares de unidades al mes. Hoy en día, las empresas como Dell venden millones de PCs todos los años. Con este crecimiento astronómico, fue un paso natural que las inversiones en el desarrollo de nuevos procesadores aumentasen en la misma proporción.

A pesar de todo, el precio de los procesadores continúa bajando. En el futuro, los procesadores serán más baratos e incorporarán más funciones. Sólo es necesario ver el caso del coprocesador aritmético y de la memoria caché. En la época del *386*, ambos se incorporaban a la placa base, mientras que hoy en día son componentes de fábrica, incluso en los procesadores más simples.

Componentes que hoy se añaden en forma de tarjetas de expansión o forman parte del *chipset* de la placa base, como el vídeo, sonido, red, etc. algún día también serán incorporados en los procesadores. Continuarán existiendo tarjetas de alto rendimiento para quién las necesite en el ámbito profesional, pero gracias a la integración será posible montar PCs básicos un poco más baratos que los actuales. Un procesador de bajo coste con todos los componentes integrados será más rápido que uno de actual.

3.3.1 El Pentium MMX

No existen muchas diferencias entre el *Pentium* y el *Pentium MMX*. Como el nombre ya sugiere, la principal modificación fueron las instrucciones *MMX*, que según la casa Intel tenían que ser suficientes para aumentar de forma considerable el rendimiento del procesador en las aplicaciones multimedia. El problema fue que las instrucciones *MMX* sólo ayudaban en las aplicaciones optimizadas para su uso. Era necesario que el desarrollador modificase el código del programa, sustituyendo las instrucciones *x86* por instrucciones *MMX*, recompilase y redistribuyese el programa, un proceso que exigía mucho trabajo y costaba su dinero. Incluso, algunos programas sacados al mercado después de la aparición del *Pentium MMX* no estaban optimizados para las nuevas instrucciones.



El Pentium MMX

Para no depender sólo de las nuevas instrucciones, la casa Intel aumentó la memoria caché *L1* del procesador de los 16 hasta los 32 KB. Con esto, el *Pentium MMX* pasó a ser un poco más rápido que un *Pentium* de la misma frecuencia, incluso en las aplicaciones que no estaban optimizadas para las instrucciones *MMX*. Debemos recordar que en aquella época la memoria caché *L2* del procesador aún formaba parte de la placa base y trabajaba a sólo 66 MHz.

La casa Intel también lanzó modelos de procesadores *MMX Overdrive*, que podían sustituir antiguos procesadores *Pentium* de 75, 100 o 120 MHz con un simple cambio del procesador y sin necesidad de cambiar la placa base. El problema es que estos procesadores eran más caros y difíciles de encontrar, así como los *Overdrive* anteriores.

Las instrucciones MMX

El *Pentium* era un procesador que trabajaba con palabras binarias de 32 bits. El problema consistía en que, muchas veces, era necesario realizar cálculos utilizando datos de 8 o 16 bits, que principalmente eran utilizados por los programas de edición de imágenes o sonido. Por lógica, debería ser posible procesar cuatro palabras de 8 bits o dos de 16 en cada ciclo pero, en la práctica, el procesador sólo era capaz de procesar un único valor en cada ciclo, independientemente de su número de bits. La suma de dos números de 8 bits tardaba lo mismo que la suma de dos números de 32 bits, simplemente no se utilizaban los bits adicionales del procesador.

Las instrucciones *MMX* permitían juntar varias palabras binarias de 8 o 16 bits y transformarlas en una única palabra de mayor tamaño que podía procesarse usando las instrucciones específicas del conjunto *MMX*.

Para hacer más fácil la tarea de adaptar los programas para utilizar las instrucciones *MMX*, la casa Intel optó por añadirlas al procesador en forma de una extensión del coprocesador aritmético. Esta arquitectura aportó la ventaja de no exigir prácticamente ninguna alteración en el proyecto del procesador, pues su funcionamiento continaba siendo idéntico. Un programa antiguo, simplemente ignoraba las instrucciones *MMX* y no presentaba ninguna ganancia de rendimiento.

Para sacarle provecho a las nuevas instrucciones era necesario que el programador modificase el código del programa, alterando sus rutinas para que se utilizasen las instrucciones *MMX* en lugar de las instrucciones *x86*. La ganancia de rendimiento dependía de la habilidad del programador en detectar, dónde y cómo podían usarse las instrucciones *MMX* para que la ejecución del programa fuese más rápida.

Teóricamente, las instrucciones *MMX* podían hacer que la ejecución de aplicaciones gráficas fuese hasta un 400% más rápida, ya que podían procesadarse hasta 4 palabras de 16 bits en cada ciclo. Otras aplicaciones también podían salir muy beneficiadas. Sin embargo, en la práctica, las ganancias conseguidas por las instrucciones *MMX* eran menores. Para empezar, sólo algunas de las rutinas utilizadas por los programas podían optimizadarse, ya que la única ventaja de ésta instrucciones era el procesamiento más rápido de las palabras de 8 o 16 bits. La mayoría de las instrucciones se continuaban ejecutando a través de las instrucciones *x86*. En la práctica, sólo un 30% o 40% de las instrucciones ejecutadas por los procesadores podían ejecutarse más rápidamente, resultando una ganancia de rendimiento en torno al 20% o 30%. Sin embargo, un pequeño problema con las instrucciones *MMX* hacía que la ganancia de rendimiento fuese aún menor.

Como el acceso a las instrucciones *MMX* se realizaba a través del mismo bus utilizado por el coprocesador aritmético, no era posible acceder a ambos dispositivos al mismo tiempo. O sea, el programa podía ejecutar una instrucción *MMX* o una instrucción de punto flotante, pero nunca podía hacer las dos cosas al mismo tiempo.

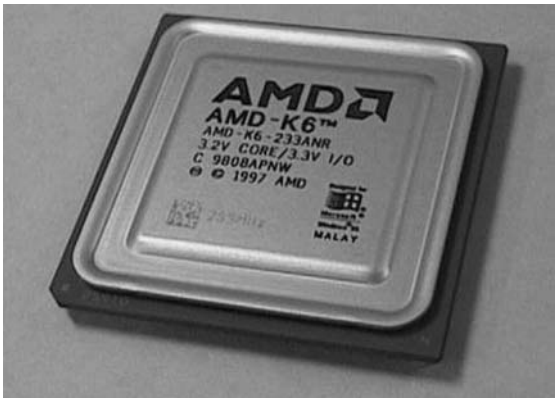
Para empeorar todavía un poco más la situación, se perdía un cierto tiempo para alternar entre el acceso a las instrucciones *MMX* y el uso del coprocesador aritmético. Como los programas gráficos dependen mucho del coprocesador aritmético, era necesario alternar constantemente entre las instrucciones *MMX* y las instrucciones *FPU* (del coprocesador aritmético), haciendo que la ganancia de rendimiento conseguida a través del uso de las instrucciones *MMX* sólo fuese de un 10% o 12% de media.

Viendo la pequeña ganancia real en el rendimiento que se podía conseguir a través de las nuevas instrucciones *MMX*, pocas empresas de software optimizaron sus programas para las nuevas instrucciones. El lanzamiento de un *Pentium MMX* fue más una estrategia de marketing que una mejora en los procesadores.

Aunque las ganancias de rendimiento eran escasas, las campañas de publicidad de Intel fueron tan fuertes que obligaron a sus competidores, AMD y Cyrix, a incorporar el conjunto de instrucciones *MMX* en sus procesadores, ya que de lo contrario, no conseguían venderlos.

3.3.2 El K6 de la casa AMD

Después del fiasco del *K5*, la casa AMD trabajó muy duro para actualizar su proyecto y lanzar el *K6* a tiempo de competir con el *Pentium MMX*.



El K6 de la casa AMD

En términos de recursos, el *K6* traía 64 KB de memoria caché *L1* integrada en el procesador y compatibilidad con las instrucciones *MMX*. Una gran ventaja del *K6* fue mantener su compatibilidad con las placas base *soquete 7* usadas por el *Pentium* y el *Pentium MMX*, facilitando así bastante la vida de los usuarios.

A causa de su arquitectura más avanzada, el *K6* superaba en rendimiento al *Pentium MMX*, llegando a estar cerca, incluso, del *Pentium II* en muchas aplicaciones. Sin embargo, el punto débil del *K6* era su coprocesador aritmético, que poseía una arquitectura más simple que la de los modelos utilizados por Intel en el *Pentium MMX* y en el *Pentium II*. A pesar de que este pequeño defecto no entorpecía el rendimiento del *K6* en las aplicaciones de oficina, su rendimiento en las aplicaciones gráficas y juegos con gráficos tridimensionales dejaba bastante que desear. En estas aplicaciones, el *K6* llegaba a ser un 20% más lento que un *Pentium MMX* de la misma frecuencia y un porcentaje aún mayor comparado con los procesadores *Pentium II* y *Pentium III*.

En la época de su lanzamiento, el *K6* no era precisamente un procesador de bajo coste y, por eso, el punto más destacado por AMD era su buen rendimiento en las operaciones con enteros y su arquitectura avanzada.

En realidad, el *K6* ya era un procesador de sexta generación, con una arquitectura muy semejante a la de los procesadores *Pentium II* y *Pentium III*. La principal diferencia entre la arquitectura de estos procesadores y la de los *Pentium* y *Pentium MMX*

antiguos era que esta nueva generación de procesadores ya incorporaba un núcleo *RISC*. Fue este cambio de filosofía lo que permitió que la frecuencia de los procesadores aumentase tanto en los últimos tiempos. Notemos que un *Pentium MMX* de 233 MHz utilizaba una arquitectura de 0.35 micrones. Cuanto menores son los transistores, más alta es la frecuencia de trabajo que es capaz de alcanzar el procesador.

La casa Intel ya lo ha demostrado en un *Pentium 4* de 0.13 micrones que trabaja a 3.5 GHz. O sea, no fue sólo la miniaturización de los transistores sino también la evolución en los proyectos de los procesadores lo que ha permitido que la industria de los semiconductores llegase al punto actual de desarrollo.

Frecuencia vs calentamiento

Mientras más elevada es la frecuencia de trabajo de un procesador, mayor es la cantidad de calor generado. Justamente por eso, los fabricantes buscaron desarrollar nuevas tecnologías de fabricación que permitiesen producir chips con transistores cada vez menores, a fin de disminuir el consumo de energía y, consecuentemente, la generación de calor.

Las primeras versiones del *K6* utilizaban una técnica de producción de 0.35 micrones y utilizaban una tensión interna de 2.9 o 3.2 voltios. Estas primeras versiones solían calentarse bastante y necesitaban un ventilador de buena calidad para no sobrecalentarse. A partir de la versión de 233 MHz, el *K6* pasó a producirse usando una técnica de producción de 0.25 micrones, lo que garantizaba una generación menor de calor. Estas versiones trabajaban con una tensión de sólo 2.2 voltios.

Debemos notar que a pesar de que el encaje era el mismo en estos nuevos *K6*, no todas las placas base *soquete 7* soportaban los 2.2 voltios. En realidad, la mayoría de las placas base antiguas sólo eran capaces de suministrar 3.3 o 3.5 voltios, las tensiones más utilizadas por el *Pentium* antiguo.

Todos los *K6* de 166 y 200 MHz fueron producidos con la técnica de producción de 0.35 micrones, mientras que todos los procesadores *K6* de 266 y 300 MHz lo fueron con la técnica de 0.25 micrones.

El problema eran los procesadores K6 de 233 MHz, pues fueron fabricados con ambas técnicas, dependiendo de la serie. Para diferenciar uno de otro, bastaba mirar el voltaje que se encontraba estampado en el procesador.

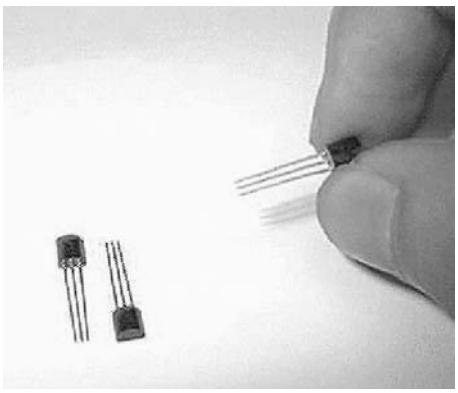
3.3.3 La evolución de los procesadores

Desde el *4004* lanzado en 1971 por la casa Intel, los procesadores evolucionaron vertiginosamente. No sólo fueron los componentes de los ordenadores que más evolucionaron, sino que también han sido los dispositivos que han evolucionado más rápido en toda la historia de la humanidad. Por ello, el transistor es considerado la invención del siglo XX.

El gran secreto para esta evolución vertiginosa puede reflejarse en una sola palabra: miniaturización. Fue justamente la miniaturización de los transistores lo que permitió la creación del circuito integrado, el *microchip* y los procesadores con más transistores y operando a frecuencias cada vez más altas. En la siguiente tabla podemos ver la evolución del número de transistores en los procesadores.

Procesador	**Transistores**
8088	29.000
286	134.000
386	275.000
486	1.200.000
Pentium	3.100.000
Pentium MMX	4.300.000
Pentium 2	9.500.000
Pentium III (Coppermine)	21.000.000
Athlon (Thunderbird)	35.000.000
Pentium 4	42.000.000

El primer transistor, creado en los inicios de la década de los años 50 fue hecho a mano y no era nada pequeño. Después de algún tiempo, se pasaron a construir transistores usando silicio y desarrollaron la litografía óptica, técnica utilizada hasta hoy en día, que usa luz, máscaras y varios productos químicos diferentes para crear las capas del transistor, permitiendo alcanzar un nivel de miniaturización impensable al principio.



Los transistores

En la siguiente tabla podemos observar el tamaño de los transistores usados en cada procesador. Un micrón equivale a 1 milésima de milímetro o a 1 millonésima parte de un metro.

Procesador	Tamaño de los transistores
Intel 4004	15 micrones
8088	3 micrones
486	1 micrones
Pentium 60 MHz	0.80 micrones
Pentium 100 MHz	0.60 micrones
Pentium 166 MHz	0.40 micrones
Pentium MMX	0.35 micrones
Pentium III 350 MHZ	0.25 micrones
Celeron 366	0.22 micrones
Pentium III Coppermine	0.18 micrones
Athlon Thunderbird	0.18 micrones
Pentium 4 Northwood	0.13 micrones
Athlon Thoroughbred	0.13 micrones
Año 2006 (según Intel)	0.07 micrones
Año 2010 (según Intel)	0.03 micrones
Alrededores 2015	0.02 micrones
Alrededores 2025	Procesadores cuánticos

3.3.4 AMD K6-2

Al desarrollar el proyecto del *K6*, los desarrolladores de la AMD estaban convencidos de que el punto más importante en un procesador era su rendimiento en las operaciones con números enteros. Pero hubo un pequeño error de cálculo en este razonamiento: las aplicaciones 3D no eran populares en la época por la falta de potencia en los procesadores para una ejecución decente. Cuando percibieron eso, el *K6* ya se estaba produciendo y era demasiado tarde para ponerle remedio.

El coprocesador aritmético del *K6* era un proyecto bastante simple. Sólo una unidad de ejecución realizaba todo el trabajo. Esto permitió eliminar todos los componentes de control que equipaban al coprocesador aritmético de los procesadores como el *Pentium III* y el *Athlon* que poseen, respectivamente, 2 y 3 unidades de ejecución en el coprocesador aritmético.

Con sólo una unidad de ejecución también era posible tener una latencia más baja, o sea, conseguir intervalos más cortos entre cada instrucción procesada. A pesar de todo, esto no fue suficiente para que el *K6* pudiese competir en igualdad de condiciones con el *Pentium II* en las operaciones de punto flotante.

Desarrollar un proyecto de coprocesador aritmético dura años. Por eso, la casa AMD optó por una solución elegante para intentar corregir este problema y el *K6-2* siguió

el ejemplo de la casa Intel e incorporó nuevas instrucciones a su procesador, el nuevo conjunto de instrucciones *3D-Now!*, formado por 27 nuevas instrucciones que tenían el objetivo de agilizar el procesamiento de las imágenes tridimensionales, trabajando conjuntamente con una tarjeta de aceleración 3D. Como acontecía con las instrucciones *MMX*, era necesario que el software usase el conjunto de instrucciones *3D-Now!*, ya que de lo contrario no existía ninguna ganancia.

El K6-2 de la casa AMD

Además de las nuevas instrucciones, los nuevos procesadores *K6-2* trajeron una novedad más para la época, que fue el uso de un bus que operaba a 100 MHz. Recordemos que el *K6-2* todavía utilizaba la memoria caché *L2* integrada en la placa base. Como la memoria caché trabajaba a la misma frecuencia que el resto de la placa base, utilizar placas de 100 MHz conllevó una gran ventaja para el *K6-2*, ya que la memoria caché *L2* se convirtió un 50% más rápida, sin hablar de la memoria *RAM*, que aumentó su rendimiento en la misma proporción.

A pesar de esto, la casa AMD continuó produciendo sus procesadores *K6-2* con el multiplicador desbloqueado. Esto permitía instalar procesadores *K6-2* en placas base antiguas que trabajasen a sólo 66 MHz, siempre que la placa soportase la tensión de 2.2 voltios (o los 2.4 voltios de las versiones posteriores) utilizada por el procesador. Un *K6-2* de 300 MHz podía utilizarse en una placa base configurada para operar a 100 MHz con un multiplicador de 3x, como en otra placa base configurada para operar a 66 MHz con un multiplicador de 4.5x.

Naturalmente, el rendimiento era mejor en la placa base de 100 MHz, por la diferencia en la frecuencia de trabajo de la memoria RAM y de la memoria caché *L2*. También se conservó la compatibilidad con las instrucciones *MMX*, con la posibilidad de ejecutar hasta 2 instrucciones *MMX* en cada ciclo de reloj. Todos los *K6-2* se fabricaron usando una técnica de producción de 0.25 micrones. Por eso, el *K6-2* sólo llegó a los 550 MHz antes de dar lugar al *Athlon* y al *Duron*, que aportaron las soluciones a la mayor parte de sus problemas.

3.3.5 MMX vs 3D-Now!

Por increíble que pueda parecer, al contrario del conjunto de instrucciones *MMX*, el conjunto *3D-Now!* sí que consiguió un gran apoyo por parte de los desarrolladores de juegos, consiguiendo que la mayoría de los juegos se optimizasen para estas nuevas instrucciones. Las aplicaciones 3D optimizadas para las instrucciones *3D-Now!* se ejecutaban de un 10 a un 20% más rápido en los procesadores *K6-2*, obteniendo un rendimiento similar al alcanzado por el *Pentium II*.

> **Nota:** También existen drivers de vídeo optimizados para el juego de instrucciones 3D-Now!, como los *drivers Detonator* para las tarjetas nVidia (TnT, TnT2, GeForce, etc.) que garantizan una ganancia de rendimiento considerable.

La diferencia básica entre el *MMX* y el *3D-Now!* es que mientras el *MMX* sólo permitía mejorar poco el funcionamiento del procesador en el cálculo con números enteros, donde el *Pentium*, como el *K6* y el *6x86* ya eran poderosos, el *3D-Now!* fue capaz de mejorar el rendimiento del *K6-2* donde más lo necesitaba, en los cálculos con números de punto flotante. Básicamente, tuvimos una extensión del coprocesador aritmético que permitió que sus funciones se ejecutasen más rápidamente, a través de estas instrucciones más eficientes.

El juego de instrucciones *3D-Now!* continúa presente en el *Athlon* y en el *Duron*, procesadores de la casa AMD que han llegado hasta hoy. A pesar de que ambos poseen un coprocesador muy poderoso, capaz de ejecutar 3 instrucciones en cada ciclo, las instrucciones *3D-Now!* continúan prestando unos buenos servicios.

3.3.6 AMD K6-3

A pesar de que desde el final de la era del *486* la casa AMD utilizaba una tecnología propia, siempre consiguió lanzar buenos procesadores que muchas veces introducían nuevas tecnologías que no sólo se utilizarían en las próximas generaciones de procesadores Intel, sino que muchas veces han superado a los Intel en rendimiento y, casi siempre, siendo más baratos.

El problema es que siempre que AMD conseguía lanzar un nuevo procesador, Intel ya estaba dos o tres pasos por delante, dejando a la casa AMD compitiendo con los procesadores Intel casi obsoletos en el mercado de los PCs de bajo coste. Así fue con el *K5*, *K6* y hasta cierto punto con el *K6-2*. Vender procesadores de bajo coste podía ser una buena manera de aumentar la participación en el mercado pero, definitivamente, no era la mejor manera de conseguir obtener los logros deseados.

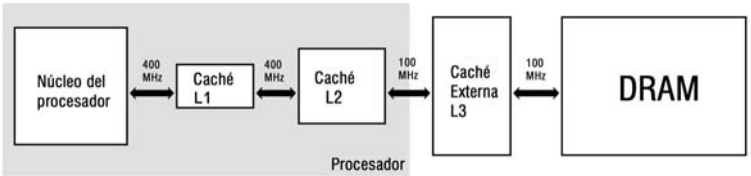
Lanzado en noviembre del año 1998, el *Sharptooth* o simplemente *K6-3* fue la primera tentativa de la casa AMD para competir directamente con los procesadores Intel, que en esa época eran el *Pentium II* de 500 y 550 MHz. Como el *K6-3* poseía una arquitectura muy distinta a la del Pentium II, no es fácil hacer una comparación directa, ya que cada procesador acababa mostrándose mejor en un cierto tipo de aplicaciones.

La memoria caché de tres niveles

La memoria caché surgió en la época de los procesadores *386* como una manera de acelerar el acceso a los datos almacenados en la memoria *RAM*. Inicialmente, sólo disponíamos de algunos KBytes de memoria caché incorporados en la placa base pero cuando la casa Intel desarrolló el *486* integró una pequeña cantidad de memoria caché dentro del procesador. Entonces teníamos dos tipos de memoria caché diferentes, una interna en el procesador y otra integrada en la placa base. Para diferenciar los dos tipos de memoria caché, la caché incorporada en el procesador pasó a llamarse caché de primer nivel o *L1*, mientras que la caché de la placa base pasó a llamarse de segundo nivel o *L2*.

La diferencia entre la caché *L1* y la caché *L2* residía en la velocidad. Mientras que la caché *L1* trabajaba a la misma frecuencia del procesador, poseyendo unos tiempos de acceso extremadamente bajos, la caché *L2* trabajaba a la misma frecuencia de la placa base, siendo más lenta que la caché *L1*. Justamente por ser más lenta, la caché *L2* era más barata, posibilitando el uso de mayores cantidades. Aún en los procesadores actuales, donde la caché *L2* ya forma parte del mismo procesador, la velocidad es menor que la de la caché *L1*.

Los datos más requeridos por el procesador son almacenados en la caché *L1*, siendo los datos que el procesador está procesando en ese momento, juntamente con otros datos importantes, transferidos por la caché *L2*. Así que el procesador termina de procesar una determinada rutina y pasa a la siguiente, se intercambian los datos de la caché *L1*, evitando al máximo que el procesador necesite acceder a datos contenidos en la caché *L2* (que siempre será más lenta) y las visitas directas a la memoria *RAM*, lo que conllevaría una gran pérdida de tiempo. En los procesadores actuales, la velocidad y la cantidad de memoria caché son más importantes que la propia velocidad de la memoria *RAM*.



La memoria caché en el K6-3

La idea del *K6-3* fue aumentar el rendimiento a través de una mayor eficiencia de la memoria caché. El *K6-3* traía los mismos 64 KB de caché *L1* que el *K6-2*, pero traía 256 KB de caché *L2* integrados en el encapsulado del procesador, caché que también trabajaba a la misma frecuencia que el procesador.

Para aumentar aún más el rendimiento del *K6-3*, la casa AMD decidió aprovechar la memoria caché *L2* encontrada en todas las placas base *soquete 7*, pasando a usarla como una caché *L3*. El funcionamiento de la caché de la placa base continuaba siendo

el mismo, sólo cambiamos su nombre. El uso de la caché *L3* conseguía aumentar entre un 3 y un 8% el rendimiento del *K6-3*, siempre dependiendo de la cantidad de memoria caché usada. No era ninguna revolución pero si una ganancia perceptible.

El mayor problema

Aparte de la inclusión de la memoria caché *L2*, trabajando a la misma frecuencia del procesador y del uso de la memoria caché *L3*, el K6-3 conllevó pocos cambios sobre el antiguo *K6-2*. Se realizaron algunas modificaciones en los registros del procesador, pero nada sustancial. El *K6-3* continuaba utilizando las 21 instrucciones *3D-Now!* y el mismo coprocesador aritmético del *K6-2*. Éste era el gran problema. El mayor defecto del *K6-2* no era la memoria caché sino el coprocesador aritmético con sólo una unidad de ejecución. Gracias a la rápida memoria caché, el *K6-3* conseguía, incluso, superar a un *Pentium III* de la misma frecuencia en las aplicaciones de oficina pero continuaba perdiendo en juegos y aplicaciones gráficas.

Compatibilidad

En la época del lanzamiento del *K6-3* existía una cantidad sustancial de usuarios de equipos basados en las placas base *soquete 7*, entre usuarios de procesadores AMD y de procesadores Cyrix que, a buen seguro, se sentían mucho más inclinados a cambiar su procesador por otro que continuase utilizando el mismo *soquete 7*, permitiendo una actualización sin la necesidad de cambiar la placa base.

Sabiendo esto, la casa AMD optó por hacer el *K6-3* totalmente compatible con las placas base *soquete 7*, sólo manteniendo la vieja exigencia de que la placa base fuese capaz de suministrar la tensión eléctrica correcta para el procesador y capaz de soportar el multiplicador correcto.

Cualquier placa base que soportaba el *K6-2* de 400 MHz también soportaba el *K6-3* de 400 y 450 MHz sin ningún problema, ofreciendo incluso actualizaciones del *BIOS*. El voltaje utilizado por el *K6-3* también era el mismo utilizado por el *K6-2*. Tampoco era necesario configurar nada relacionado con la memoria caché *L3*, pues con sólo encajar el *K6-3* en la placa base, la memoria caché de la placa pasaba automáticamente a ser reconocida como caché *L3*.

3.3.7 Cyrix 6x86MX

Retrocediendo un poco en el tiempo, el *6x86MX* fue el competidor de la casa Cyrix para el *Pentium MMX*. Este procesador llevaba 64 KB de memoria caché *L1*, las instrucciones *MMX* y también ofrecía compatibilidad con las placas base *soquete 7*. El rendimiento en las aplicaciones *Windows* era muy parecido al de un *K6* aunque, sin embargo, el coprocesador aritmético era más lento que el de los *K6*, resultando un rendimiento muy bajo en los juegos y aplicaciones gráficas. Mientras que los *K6* usaban un coprocesador aritmético simple, con sólo una unidad de ejecución pero con algunas optimizaciones para ser capaz de ejecutar una instrucción por ciclo, el copro-

cesador aritmético del *6x86MX* podía tardar varios ciclos en el procesamiento de cada instrucción. Sólo las instrucciones simples se ejecutaban en un único ciclo. Mientras el coprocesador aritmético de los *K6* era cerca de un 20% más lento que el del *Pentium MMX*, el coprocesador aritmético del *6x86MX* llegaba a ser un 50% más lento.

El 6x86MX de la casa Cyrix

Para las aplicaciones de oficina como Office, el procesador *6x86MX* fue una buena opción debido al bajo coste del procesador, que costaba la mitad que un *MMX*, pero no era nada adecuado para los PCs destinados a juegos o aplicaciones gráficas.

El *6x86MX* usaba una tensión de 2.9 voltios pero, según el fabricante, si la placa base no ofrecía esa tensión específica, podíamos usar sin problemas un voltaje de 2.8 voltios, como en el *Pentium MMX*.

3.3.8 Cyrix Media GX

El *Media GX* era un procesador *6x86MX* con más circuitos controladores de memoria *RAM* y memoria caché, así como controladores de vídeo y sonido, que se destinaba al mercado de los PCs de bajo coste y, principalmente, a los ordenadores portátiles. Usado en ordenadores portátiles, el *Media GX* conllevaba la ventaja de consumir poca electricidad, proporcionando una mayor autonomía de la batería.

Al exigir una placa base específica, el procesador *Media GX* sólo se destinaba a ordenadores con una arquitectura cerrada.

3.3.9 IDT C6

Lanzado en el año 1998, el *C6* fue la tentativa de la casa IDT de entrar en el mercado de los procesadores, compitiendo en la franja de procesadores de bajo coste con el *MMX* y el *6x86MX*.

Técnicamente, el *C6* no traía ninguna novedad. Básicamente teníamos una arquitectura no superescalar simple, que recordaba un poco a la de los equipos *486*, combinada

con una memoria caché *L1* de 64 KB, dividida en dos bloques de 32 KB. El coprocesador aritmético también era débil, quedándose por detrás de los procesadores de la casa Cyrix. El *C6* fue lanzado en versiones de 180, 200, 225 y 240 MHz y utilizaba placas base *soquete 7* y una tensión de 3.5 voltios.

Los procesadores de la casa IDT

A pesar del poco éxito alcanzado con el *C6*, la casa IDT lanzó después de algún tiempo el *C6+*, que incorporó instrucciones equivalentes al *3D-Now!* y pasó a usar un bus de 100 MHz pero que, como en el caso anterior, acabó siendo ignorado por el mercado. Fue el último procesador lanzado por esta compañía.

3.3.10 Pentium Pro

Hasta aquí hemos visto los procesadores en un orden más o menos cronológico, pero hemos realizado una excepción con el *Pentium Pro*. En realidad, este procesador fue lanzado antes que el *Pentium MMX*. Sin embargo, la arquitectura usada en el *Pentium Pro* fue usada como base para el *Pentium II* y el *Pentium III*, así como para el *Xeon* y el *Celeron*, procesadores que examinaremos más adelante y de forma más detallada.

El *Pentium Pro* fue desarrollado para competir en el mercado de los equipos de alto rendimiento, equipando *Workstations* y servidores. A pesar de usar un poco de la tecnología del *Pentium*, el *Pentium Pro* era un proyecto casi totalmente nuevo, que traía modificaciones en la arquitectura. Entre las innovaciones del *Pentium Pro* podemos destacar la arquitectura superescalar con tres canalizaciones, el soporte para el multiprocesamiento con hasta 4 procesadores trabajando en paralelo y la memoria caché *L2* integrada en el procesador.

El bus de direcciones del *Pentium Pro* también fue ampliado de 32 a 36 bits, permitiéndole el direccionamiento de hasta 64 GB de memoria *RAM* (2 elevado a 36). Puede parecer una exageración, pero muchos servidores de alto rendimiento usaban mucha más memoria *RAM* de la normal en un equipo de uso personal. Toda esta memoria *RAM* permitía ejecutar simulaciones complejas y cálculos inimaginables hasta hacía poco en un ordenador personal.

Tres unidades de ejecución

Mientras el *Pentium* podía procesar hasta dos instrucciones de forma simultánea, el *Pentium Pro* poseía tres unidades de ejecución, lo que le permitía procesar hasta tres instrucciones en cada ciclo de reloj.

Pero las cosas no eran tan simples como parecen, pues los programas continuaban esperando un procesador capaz de ejecutar una única instrucción por ciclo, como pasaba en un *486*. Sólo hasta hace pocos años las aplicaciones comerciales no empezaron a estar optimizadas para la arquitectura del *Pentium Pro* (y, consecuentemente, también para el *Pentium III*, *Celeron* y *Athlon* que tienen un funcionamiento semejante). *Linux* llevaba una cierta ventaja en este aspecto, pues era posible en cualquier momento recompilar el *kernel* del sistema para optimizarlo para cualquier procesador, garantizando un rendimiento bastante superior.

El interior de un Pentium Pro

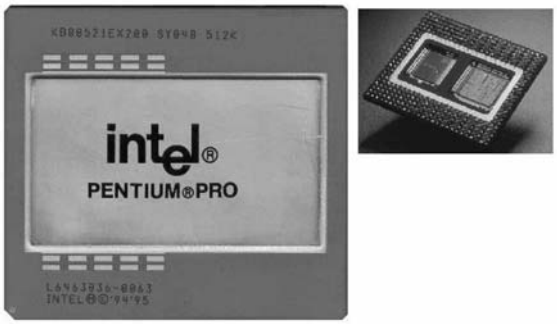
Para garantizar que las otras dos unidades de ejecución no acabaran estando ociosas la mayor parte del tiempo, fue incluido el recurso de ejecución de instrucciones fuera de orden. Si al procesador le llegaba una instrucción como "Si X >10, entonces Y = X + 2, sino Y = X + 5" donde las próximas instrucciones dependían de la resolución de la primera, la unidad de control escudriñaba el código del programa en busca de instrucciones que se pudieran ejecutar con antelación. A través de este recurso, mientras una de las canaletas se ocupaba de resolver el primer problema, las demás podían dedicarse a resolver las instrucciones siguientes, ganando tiempo.

Si no era posible localizar ninguna instrucción que pudiese "adelantarse", entraba en escena el recurso de la ejecución especulativa, escogiendo uno de los caminos posibles. Este último recurso no es tan ventajoso, pues existe una posibilidad de escoger el camino errado, y tener que recomenzar el trabajo al percibir el error, pero es mejor que no hacer nada mientras se está procesando la condición.

Memoria caché L2 integrada

El ingeniero de la casa Intel que inventó la multiplicación de reloj debe haber ganado varios premios, ya que este recurso fue uno de los factores que posibilitaron las asombrosas frecuencias de los procesadores actuales. Pero este recurso también tuvo sus efectos colaterales. Uno de ellos, que ya se notaba incluso en un *486*, era la lentitud de la memoria caché *L2*, pues al encontrarse localizada en la placa base tenía una frecuencia de trabajo limitada a la velocidad de la placa base, o sea, a sólo 33, 66 o 100 MHz, mucho más lenta que la velocidad de trabajo del procesador.

Teníamos entonces un cuello de botella pues, frecuentemente, los pocos datos grabados en la caché *L1* se agotaban haciendo que el procesador necesitase perder varios ciclos esperando que los nuevos datos fuesen entregados por la memoria caché *L2*. Para solucionar este problema, Intel resolvió integrar la memoria caché *L2* al *Pentium Pro*, creando también un nuevo bus, bautizado como *Backside Bus* únicamente dedicado a la comunicación entre el núcleo del procesador y la memoria caché *L2*. El uso de un bus separado permitió que la memoria caché *L2* trabajase a la misma frecuencia que el procesador, independientemente de la velocidad de la placa base, recurso que conllevaba la ventaja adicional de permitir el acceso simultáneo a la memoria *RAM*, ya que teníamos dos buses distintos.



Un Pentium Pro. Al lado podemos ver el núcleo y la memoria caché L2 integrada en el procesador

Aún con la memoria caché *L2* trabajando a la misma frecuencia que el procesador, la memoria caché *L1* continuaba siendo necesaria, pues a pesar de que las dos memorias caché trabajasen a la misma frecuencia, el bus de datos de la caché *L1* era mayor, lo que le permitía unas tasas de transferencia mucho más altas. El tiempo de respuesta de la memoria caché *L1* también era menor, pues además de usar memorias más rápidas, la distancia física recorrida por la señal era muy inferior, ya que la memoria caché *L1* formaba parte del mismo núcleo del procesador.

Justamente debido a la memoria caché *L2*, el *Pentium Pro* era muy difícil de producir, pues la complejidad de la caché *L2* conllevaba una alta tasa de defectos de fabricación. Como en el *Pentium Pro* la caché *L2* estaba embutida en el mismo encapsulado del procesador, un defecto en la memoria caché *L2* condenaba a todo el procesador. Los problemas de fabricación contribuían a hacer el *Pentium Pro* todavía más caro.

A pesar de compartir el mismo encapsulado del procesador, la memoria caché *L2* del *Pentium Pro* estaba formada por un chip separado. Eso permitió que la casa Intel lanzase varias versiones del *Pentium Pro* equipadas con distintas cantidades de caché *L2*. En realidad, existieron tres versiones con 256 KB, 512 KB y 1 MB de memoria caché *L2*. Evidentemente, el precio crecía exponencialmente junto con la caché y, por eso, los *Pentium Pro* de 256 KB fueron casi los únicos que fueron comercializados en PCs domésticos y *Workstations*.

En cuanto a la frecuencia de trabajo, existieron sólo dos versiones de 166 y 200 MHz. La limitación básica no fue la arquitectura del procesador, pero sí la memoria caché. En la época, la casa Intel tenía las condiciones para lanzar versiones más rápidas, pero no tenía cómo producir los *chips* de memoria caché capaces de trabajar a más de 200 MHz y sin los que era imposible trabajar.

Multiprocesamiento

El *Pentium Pro* permitía el uso de hasta 4 procesadores en la misma placa base. En este caso, las reglas eran las mismas que las del multiprocesamiento del *Pentium*, siendo necesaria una placa base especial y un sistema operativo con soporte para el multiprocesamiento. A pesar de que este recurso no era muy útil para un usuario doméstico, permitió el desarrollo de servidores con un rendimiento increíble a un coste relativamente bajo, usando un *Pentium Pro* en el lugar de los caros procesadores *RISC* de gama alta disponibles en la época.

Los *mainframes* (macrocomputadoras) con 2 o 4 procesadores *Pentium Pro* presentaban un rendimiento tan bueno, gracias a la rápidez de la nueva memoria caché, que muchas empresas continúan utilizándolos hoy en día, incluso después del lanzamiento de procesadores mucho más rápidos.

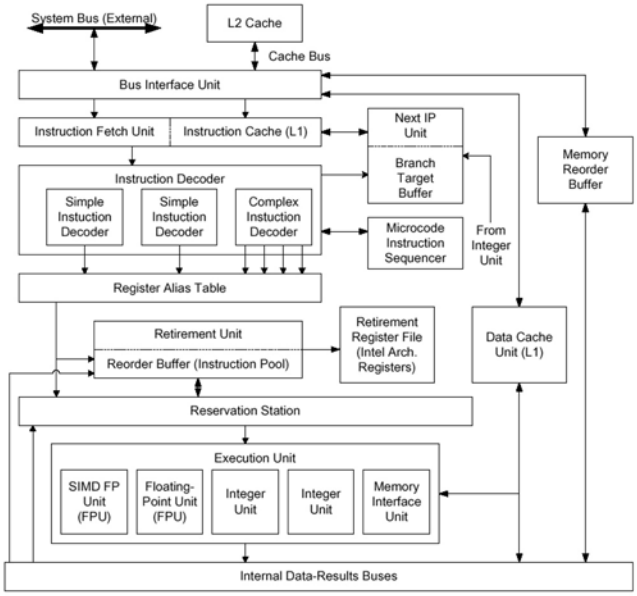
Soquete 8

Con el gran cambio en la arquitectura del procesador y el nuevo encapsulado, el Pentium Pro dejó de ser compatible con las viejas placas base *soquete 7* usadas por los procesadores anteriores. Entonces se creó un nuevo tipo de *soquete*, bautizado como *soquete 8*, que sólo fue utilizado en las placas base para procesadores *Pentium Pro*. Estas placas eran naturalmente incompatibles con los demás procesadores. Para permitir el uso de todos los nuevos recursos del *Pentium Pro*, también se creó el *chipset* i440FX que equipaba a la mayoría de las placas base para *Pentium Pro*.

Arquitectura optimizada para las instrucciones de 32 bits

El *Pentium Pro* podía ser considerado un procesador *Post-RISC*, pues básicamente el estaba compuesto por un núcleo formado por 3 procesadores *RISC* (las tres unidades de ejecución de las instrucciones) y tres decodificadores con la función de convertir las instrucciones *x86* usadas por los programas en instrucciones más simples que podían ser procesadas por el núcleo *RISC*. Las instrucciones simples, como las operaciones

aritméticas, podían convertirse en una única instrucción *RISC*, mientras que las instrucciones más complejas debían dividirse en varias instrucciones simples, necesitando varios ciclos para ser procesadas de forma correcta.



Esquema de funcionamiento de los procesadores de 6 generación

Este esquema funcionaba muy bien con instrucciones de modo protegido (32 bits), pero el decodificador del *Pentium Pr*o poseía una cierta dificultad para trabajar con instrucciones de modo real (16 bits), que acababan siendo decodificadas en muchas instrucciones *RISC*, tardando mucho más tiempo en ser procesadas.

Ejecutando sistemas operativos de 32 bits como *Windows NT*, *Windows 2000*, *OS/2*, *Linux* y *UNIX*, el *Pentium Pro* presentaba un rendimiento superior al que ofrecía el *Pentium* clásico. Sin embargo, ejecutando sistemas operativos de modo real, como *MS-DOS* o híbridos como *Windows 95/98*, incluso con una caché más rápida, el *Pentium Pro* acababa presentando un rendimiento equivalente, o incluso un poco menor, que el de un *Pentium* común.

En el caso de los servidores no llegaba a ser una limitación pues, invariablemente, estos equipos ejecutaban versiones de *Unix*, *Novell Netware* o *Windows NT*, siempre sistemas operativos de 32 bits. Sin embargo, los usuarios domésticos acabaron penalizando el débil desempeño del Pentium Pro en la ejecución de *Windows 95/98*, sistemas operativos que aún poseían mucho código de 16 bits.

Para un uso doméstico, no tenía mucho sentido el uso de un *Pentium Pro*. Sin embargo, en un servidor, la memoria caché *L2* funcionando a la misma velocidad que el procesador marcaba la diferencia, pues el procesamiento de los datos es muy repetitivo

en estas máquinas, tanto que, incluso con la aparición del *Pentium II* (donde la memoria caché *L2* funcionaba a la mitad de la velocidad del procesador) muchos prefirieron continuar usando el *Pentium Pro*, pues además de la memoria caché L2 ofrecía otros recursos interesantes para un servidor, como la posibilidad de usar hasta cuatro procesadores en paralelo (el *Pentium II* estaba limitado a dos procesadores), además de la mayor cantidad de memoria soportada, recursos que sólo fueron superados por el *Xeon* que trajo una mayor cantidad de memoria caché *L2*.

3.3.11 El Pentium II

La compañía Intel desarrolló el *Pentium II* usando el proyecto del Pentium Pro como base. Se realizaron algunas mejoras por un lado y se retiraron algunos recursos (como el soporte para 4 procesadores) por el otro, dejando un procesador más adecuado para el mercado doméstico.

El cambio más visible en el *Pentium II* fue el nuevo encapsulado *SEPP* (S*ingled Edge Processor Package*). En lugar de un pequeño encapsulado de cerámica, teníamos una tarjeta de circuito, que llevaba el procesador y la memoria caché *L2* integrada. Protegiendo esta tarjeta teníamos una capa plástica formando un cartucho muy parecido a un cartucho de videojuegos.

El procesador Pentium II en su encapsulado en forma de cartucho

Nuevamente fue modificado el anclaje usado por el procesador. El *Pentium II* no era compatible con las placas base *soquete 7* ni con las placas para *Pentium Pro*, exigiendo una placa base con un anclaje *slot 1*. A la mayoría de los usuarios no les gustó mucho la idea, ya que al utilizar un nuevo anclaje el *Pentium II* era incompatible con las placas base *soquete 7* disponibles hasta ese momento, lo que obligaba los usuarios a cambiar también la placa base en el caso de una actualización o ampliación. El uso del nuevo *slot 1* no dejaba de ser una política expoliadora de Intel, pues al haber sido creado y patentado por ella, los otros fabricantes no podían hacer uso de esa tecnología en sus procesadores. La utilización del *slot 1* fue el primer paso para la variedad de formatos y tecnologías que hemos tenido en el ramo de los procesadores.

Aparte del aspecto externo, el *Pentium II* traía una caché *L1* de 32 KB (dividida en dos bloques de 16 KB para datos e instrucciones), una caché *L2* integrada de 512 KB y compatibilidad con el conjunto de instrucciones *MMX*. Como los procesadores anteriores, el *Pentium II* también ofrecía soporte para hasta 4 GB de memoria *RAM*.

Como el *Pentium II* fue desarrollado para el mercado doméstico, donde *Windows 98* era el sistema operativo más utilizado en ese momento, Intel aportó un modo para solucionar el problema del *Pentium Pro* con las instrucciones de 16 bits, añadiendo al procesador un registro de segmento. Al contrario del *Pentium Pro*, el *Pentium II* podía procesar instrucciones de 16 bits a la misma velocidad con la que procesaba las instrucciones de 32 bits, ofreciendo un buen rendimiento ejecutando *DOS*, *Windows 3.x* o *Windows 95/98*.

Memoria caché L2 integrada

El *Pentium II* llevaba integrados en el procesador nada menos que 512 KB de memoria caché *L2*, el doble de la cantidad que podíamos encontrar en la versión más simple del *Pentium Pro*. Sin embargo, en el *Pentium II* la caché *L2* trabajaba a sólo la mitad de la frecuencia del procesador. Por ejemplo, en un *Pentium II* de 266 MHz la memoria caché *L2* trabajaba a 133 MHz, el doble de la frecuencia de la memoria caché que encontrábamos en las placas base *soquete 7*, pero menos que los 200 MHz de la memoria caché *L2* encontrada en el *Pentium Pro*. La casa Intel optó por usar esta caché *L2* más lenta para solucionar los tres problemas que entorpecieron el desarrollo y la popularización del *Pentium Pro*.

El primero era el alto grado de incidencia de defectos en la caché. La memoria caché de alta velocidad del *Pentium Pro* era muy difícil de producir con la tecnología existente en la época, lo que generaba un elevado número de defectos. Al no ser posible probar la caché separada del procesador, era necesario tener el procesador listo para probar todo el conjunto. Si la caché presentaba algún defecto, todo el procesador era condenado a la basura.

Este problema inutilizaba buena parte de los procesadores *Pentium Pro* que se producían, resultando en unos costes de producción más altos y, consecuentemente, en unos precios de venta más elevados que hacían que el *Pentium Pro* fuese aún más inaccesible para el consumidor final. La memoria caché más lenta utilizada en el *Pentium II* era más fácil de producir y, al mismo tiempo, el formato *SEPP* permitía sustituir los chips de caché durante el proceso de fabricación si estos presentaban algún defecto.

El segundo problema era la dificultad que la casa Intel encontró para producir memorias caché rápidas en la época del *Pentium Pro*; dificultad que acabó imposibilitando el lanzamiento de procesadores de esta familia que trabajasen a más de 200 MHz. En aquella época, la casa Intel aún no tenía la tecnología suficiente para producir chips de memoria caché *L2* capaces de trabajar a más de 200 MHz y con un coste de producción aceptable.

Por último, teníamos el factor del coste, pues utilizando memorias caché un poco más lentas en el *Pentium II*, los costes de producción se reducían haciendo un procesador más atractivo al mercado doméstico. Debido a la integración de la memoria caché *L2* en la placa de circuito del procesador, nunca encontrábamos a la venta una placa base para *Pentium II* con memoria caché.

Arquitectura

El *Pentium II* fue producido con dos arquitecturas diferentes. Las versiones de hasta 300 MHz utilizaban la arquitectura *Klamath*, que consistía en una técnica de fabricación de 0.35 micrones muy parecida a la utilizada en los procesadores *Pentium MMX*. En las versiones a partir de los 333 MHz ya se utilizaba la arquitectura *Deschutes* de 0.25 micrones que garantizaba una disipación de calor menor, lo que posibilitó el desarrollo de procesadores más rápidos. Vale la pena recordar que en el *Pentium II* no era preciso preocuparse de la configuración correcta de la tensión del procesador, pues esto era realizado automáticamente por la placa base. Sólo para matar nuestra curiosidad, los procesadores basados en la arquitectura *Klamath* utilizaban una tensión de 2.8 voltios, mientras que los que estaban basados en la arquitectura *Deschutes* utilizaban una tensión de 2.0 voltios.

Una última consideración acerca de los procesadores *Pentium II* es la frecuencia de bus utilizada por el procesador. Las versiones del *Pentium II* hasta los 333 MHz usaban un bus de 66 MHz, mientras que las versiones a partir de los 350 MHz usaban un bus de 100 MHz, lo que aceleraba el intercambio de datos entre el procesador y la memoria *RAM*.

Debemos recordar el recurso de *pipeline* introducido en el *486*. Mientras el *Pentium* clásico, así como el P*entium MMX*, mantenía la misma estructura básica del *486*, con un *pipeline* de 5 niveles, el *Pentium II*, al utilizar la misma arquitectura del *Pentium Pro*, poseía un *pipeline* de 10 niveles. Además de mejorar el rendimiento del procesador, el uso de un pipeline de más niveles permitía desarrollar procesadores capaces de operar a mayores frecuencias. La idea se basaba en que, al trabajar con un *pipeline* más largo, el procesador fuese capaz de procesar más instrucciones de forma simultánea. Sin embargo, al mismo tiempo, el *Pentium II* poseía un núcleo *RISC*, lo que significaba que internamente sólo procesaba instrucciones simples. La combinación de estos dos factores permitía simplificar la operación de cada nivel del *pipeline*, haciendo que cada nivel realizase menos procesamientos, pero en conjunto se conseguían ejecutar las mismas tareas. Ejecutando menos procesamientos por ciclo en cada nivel era posible que el procesador operase a frecuencias más altas sin sacrificar la estabilidad del sistema.

Multiprocesamiento

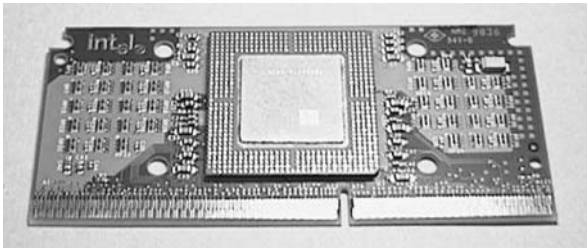
Al contrario que el *Pentium Pro*, el *Pentium II* sólo ofrecía soporte para el uso de dos procesadores, como el *Pentium* común. Esta era una adaptación hecha para disminuir un poco el precio de venta.

Otra razón de esta limitación era hacer más atractivo el *Pentium II Xeon*, un *Pentium II* equipado con una memoria caché más rápida, soporte para el uso de hasta 8 procesadores y soporte para el uso de más memoria *RAM*. Naturalmente, el *Xeon* era un procesador mucho más caro, dedicado al mercado de los servidores y *workstations*.

3.3.12 El Celeron

Con el lanzamiento del *Pentium II* la casa Intel abandonó la fabricación del *Pentium MMX* pasando sólo a vender los procesadores *Pentium II*, que eran más caros. El problema de esta estrategia fue que la casa Intel perdió terreno en el mercado de los PCs de bajo coste, en beneficio de los procesadores *K6* y *6x86*, que a pesar de presentar un rendimiento inferior, costaban menos de la mitad que un *Pentium II* de la misma frecuencia de trabajo.

Intentando reparar el error de estrategia, la casa Intel decidió lanzar una versión de bajo coste del *Pentium II*, bautizada como *Celeron*. El *Celeron* original era un *Pentium II* sin la memoria caché *L2* integrada y sin el envoltorio de plástico, responsables de buena parte de los costes de producción del *Pentium II*.



El Celeron de la casa Intel

Las primeras versiones del *Celeron*, que incluyen todos los de 266 MHz y algunos de los de 300 MHz, no traían memoria caché *L2* alguna y, por eso, presentaban un rendimiento muy débil en la mayoría de las aplicaciones, a pesar de que sí podían ofrecer un rendimiento razonable en juegos y en las aplicaciones que utilizaban el coprocesador aritmético de forma constante.

La memoria caché *L2* es un componente extremadamente importante en los procesadores actuales, pues a pesar de que la potencia de los procesadores ha aumentado muchísimo en las últimas dos décadas, la memoria *RAM* no ha evolucionado tanto en velocidad. Poco adelantamos con un procesador veloz si tiene que parar lo que está haciendo para esperar los datos provenientes de la memoria *RAM*.

Es justamente ahí donde entra en juego la memoria caché *L2*, reuniendo los datos más importantes de la memoria para que el procesador no tenga que quedarse esperando. Retirando la memoria caché *L2,* el rendimiento del equipo disminuye en casi un 40%. Justamente por esto, el *Celeron* sin caché perdía en comparación con, incluso, procesadores mucho más antiguos.

Debido a su bajo rendimiento, el *Celeron* sin memoria caché no tuvo una buena aceptación en el mercado, siendo muy criticado por la prensa especializada. En una nueva tentativa de reparar este error, la casa Intel decidió equipar las nuevas versiones del *Celeron* con 128 KB de memoria caché *L2,* que al contrario de la memoria caché del *Pentiun II*, funcionaba a la misma frecuencia que el procesador.

Mientras que en el *Pentium II* la caché *L2* estaba formada por chips separados y soldados en la tarjeta de circuito del procesador, en el *Celeron* la caché *L2* formaba parte del propio núcleo del procesador. Estos 128 KB de memoria caché *L2* conllevaban una diferencia increíble en el rendimiento del procesador. Mientras que un *Celeron* antiguo era casi un 40% más lento que un *Pentium II* de la misma frecuencia, el Celeron con caché *L2* era aproximadamente un 6% más lento. Esto acontecía a pesar de que el *Celeron* poseía una cantidad 4 veces menor de memoria caché *L2*, pero ésta funcionaba al doble de velocidad, compensando en gran medida la diferencia. Claro que esto dependía de la aplicación que estuviéramos ejecutando.

Otro punto a favor del *Celeron* era su coprocesador aritmético que, siendo idéntico al del *Pentium II,* era más rápido que el del *Pentium MMX* o de los *K6*, lo que le garantizaba un buen rendimiento en las aplicaciones gráficas.

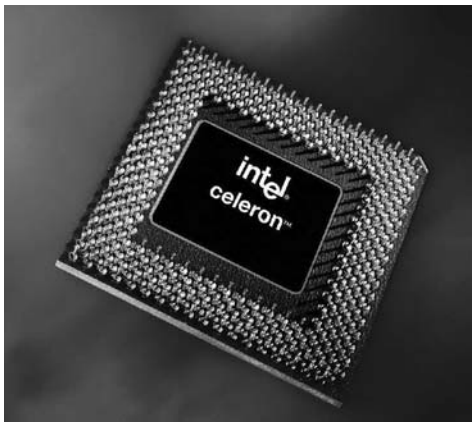
Todas las versiones del *Celeron* (con excepción de los *Celeron* de 800 MHz en adelante) utilizaban un bus de sólo 66 MHz. Esta era otra diferencia en relación al *Pentium II* y al *Pentium III*. A pesar de que en términos de procesamiento el *Celeron* llegaba a batir, a veces, a un *Pentium II* de la misma frecuencia, acababa siendo más lento al utilizar un multiplicador más elevado. Por ejemplo, un *Pentium II* de 400 MHz utilizaba un bus de 100 MHz y un multiplicador de 4x. Por su parte, un *Celeron* de 400 MHz utilizaba bus de 66 MHz y un multiplicador de 6x. A pesar de que en los dos casos el procesador trabajaba a la misma frecuencia, en el caso del *Celeron* la placa base y la memoria *RAM* funcionaban más lentamente, acabando por entorpecer el rendimiento del procesador. Por eso, muchas veces el *Celeron* acababa quedándose un 10%, o hasta un 15%, por detrás del *Pentium II* en los programas de medición del rendimiento.

Soquete 370 vs slot 1

Inicialmente, la casa Intel lanzó el *Celeron* en el mismo formato del *Pentium II*, o sea, en la forma de una tarjeta de circuito que utilizaba el *slot 1* a fin de mantener la compatibilidad con todas las placas base ya existentes y facilitar las ventas del nuevo procesador.

Sin embargo, inmediatamente después fue lanzado en un nuevo formato de encapsulado y con un nuevo anclaje para el *Celeron*, llamado *soquete 370*. El formato era muy parecido al de un Pentium MMX; la diferencia era que el *Celeron* poseía algunos contactos más. El *Celeron* para un *soquete 370* también era llamado *PPGA*, abreviatura de *Plastic Pin Grid Array*. Es preciso recordar que, a pesar de que los encajes eran parecidos, el *Celeron PPGA* no era compatible con las placas base *soquete 7* utilizadas

en conjunto para los procesadores *Pentium MMX* y *K6*. El *soquete 370* utilizaba el mismo sistema de contactos que el *slot 1* y las placas base utilizaban los mismos *chipsets* y demás componentes básicos. Inclusive, era posible ensamblar un *Celeron* para *soquete 370* en una placa base *slot 1* con la ayuda de un adaptador.



Un Celeron PPGA para un anclaje soquete 370

La explicación de la casa Intel para el cambio del formato del *Celeron* fue que, después de integrar la memoria caché *L2* en el procesador, la tarjeta de circuito usada en el *Pentium II* se volvió innecesaria, sirviendo únicamente para aumentar los costes de producción. Y retirándola del *Celeron* sería posible fabricar procesadores más baratos.

Durante mucho tiempo la casa Intel continuó fabricando el *Celeron* en ambos formatos, pero después canceló la producción de las versiones para *slot 1*, continuando fabricando sólo las versiones para *soquete 370*. Los *Celeron* para el *soquete 370* podían utilizarse tanto en placas base para *Pentium II* como para *Pentium III* con la ayuda del adaptador, como en placas base *soquete 370*. Como hemos visto anteriormente, los dos encapsulados eran eléctricamente compatibles, sólo cambiaba el formato.

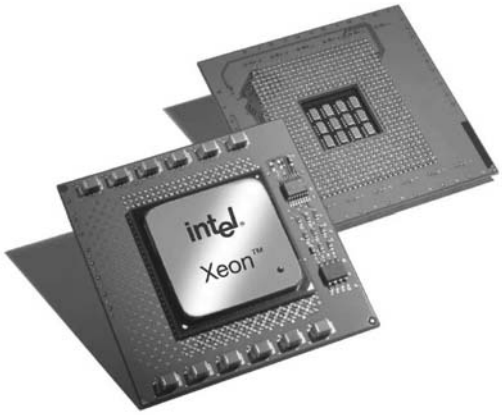
3.3.13 Pentium II Xeon

El *Pentium II* se acabó revelando como una solución mucho más adecuada que el *Pentium Pro* para el mercado doméstico. El Celeron también supuso un avance en este sentido, pues mantuvo un buen rendimiento a un coste bastante más bajo.

Pero no todo el mundo quedó satisfecho con la memoria caché *L2* operando a la mitad de la frecuencia del procesador usada en el *Pentium II*, ni con la poca caché *L2* de 128 KB del *Celeron*. Éstos fueron los propietarios de servidores de alto rendimiento que necesitaban equipos con varios procesadores y mucha memoria caché *L2*.

El *Pentium II Xeon* surgió como una alternativa para quiénes necesitaban un mayorrendimiento y podían pagarlo. De la misma manera que en el *Celeron*, la memoria caché *L2* trabajaba a la misma frecuencia del procesador, la única diferencia era la can-

tidad. Existieron versiones del *Pentium II Xeon* con 512 KB, 1 MB y 2 MB de caché y que trabajaban a 400, 450 y 500 MHz.



El Xeon de la casa Intel para equipos de alto rendimiento

Otra característica del *Xeon* era un mejor soporte para el multiprocesamiento. Además, ofrecía soporte para el uso de hasta 4 procesadores en la misma placa base, o de incluso 8 procesadores en *cluster*.

El *Xeon* fue concebido para equipar servidores, pues en estos sistemas el procesamiento es muy repetitivo y, por eso, la memoria caché *L2* más rápida y en una mayor cantidad marcaban una gran diferencia.

3.4 Hasta el primer Pentium 4

Finalmente, acabaremos nuestra clase de historia y vamos a hablar sobre los procesadores que han llegado a la actualidad. Notemos que las tecnologías que hemos discutido hasta ahora, como la multiplicación de reloj, el modo real y modo protegido, el coprocesador aritmético, RISC, *CISC* y las memorias caché, entre otras, se han continuado utilizando.

En los últimos 22 años, el rendimiento de los procesadores aumentó casi 50.000 veces, mucho más que cualquier otro componente. La memoria *RAM*, en la mejor de las hipótesis, se hizo 30 veces más rápida en este periodo. Comparadas con las placas base que teníamos en el *XT*, las placas base pasaron a trabajar a una frecuencia casi 30 veces mayor (4.7 contra 133 MHz). Sólo los discos duros consiguieron de cierta forma acompañar a los procesadores en este vertiginoso avance, pues la capacidad de almacenamiento aumentó entre 20 y 30 mil veces. Los módems pasaron de los 300 bps hasta los 56 Kbps, llegando al límite extremo del soportado por el sistema telefónico conmutado y la *ADSL* hizo realidad el viejo sueño del acceso rápido, y sin ocupar los canales de voz, con tasas de transferencia de hasta 6 Mbps a través de los hilos telefónicos comunes.

Todo esto debe verse como natural, ya que de nada serviría crear procesadores más rápidos si el resto del equipo no acompañase. A pesar de que los procesadores hayan evolucionado mucho más que el resto del conjunto, el uso de la memoria caché y de otros trucos ha servido para atenuar las diferencias.

Las páginas siguientes están dedicadas a los procesadores antiguos en uso hasta el día de hoy, juntamente con los actuales en el momento de redactar este capítulo y futuros lanzamientos.

Una diferencia notable entre los procesadores que hemos visto hasta aquí y los que tenemos en la actualidad es la gran variedad de encajes y formatos de placas base. Prácticamente todos los procesadores anteriores, incluyendo el *Pentium MMX*, *K5*, *K6*, *K6-2* , *6x86MX* e *IDT C6* podían usarse en la mayoría de las placas base *soquete 7*. Esto facilitaba mucho la elección, ya que no era preciso preocuparse tanto en saber si la placa base sería compatible o no con el procesador.

Pero, en la actualidad este problema se ha ido haciendo cada vez más evidente. Cada nueva familia de procesadores ha exigido nuevas placas base e, incluso muchas veces, ha existido incompatibilidad dentro de la misma familia de procesadores. Veamos el caso del *Athlon* por ejemplo: los modelos antiguos venían en el formato *slot A* mientras que los posteriores utilizaban el formato *soquete A*, sin que existiese ni siquiera un adaptador que permitiese aprovechar las placas base.

Al contrario de lo que ocurría dos o tres años atrás, casi siempre al hacer una actualización del procesador es preciso cambiar también la placa base y, en muchos casos, también la memoria *RAM*.

3.4.1 El problema de los encajes

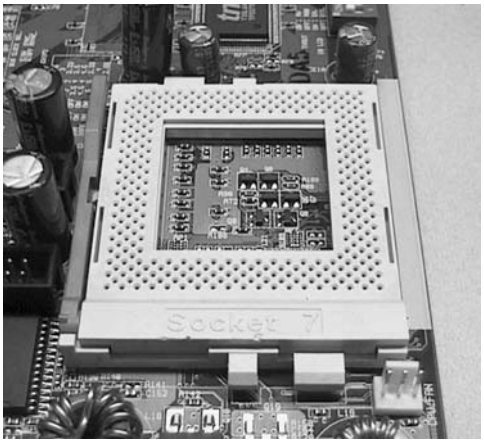
Ya que la enorme variedad de encajes y patrones es un problema crónico de los procesadores actuales, lo mejor es aprender a convivir con el problema. El encaje de la placa base da una buena pista de los procesadores que admite.

Soquete 7

Este era el encaje utilizado por la mayoría de los procesadores fabricados hasta el año 1998. Existían básicamente dos familias de placas base con este encaje. Las placas más antiguas, en general las fabricadas hasta 1997, que soportaban un bus de sólo 66 MHz y sólo eran compatibles con el *Pentium* y, en general, también con el P*entium MMX*, Cyrix *6x86* y los *K6* de hasta 233 MHz.

Las placas base *soquete 7* más modernas, fueron apodadas placas *super 7* pues soportaban un bus de 100 MHz y también eran compatibles con los procesadores de la casa AMD *K6-2* y *K6-3*. Algunos modelos ya traían hasta un *slot AGP*. Entre las dos generaciones existió una familia intermedia que estaba compuesta por las placas base que soportaban un bus de 66 MHz, pero que ya soportaban unas tensiones de 2.2 y 2.4 voltios utilizadas por los procesadores *K6-2*.

Para saber con certeza el tipo de placa base necesitábamos tener a mano el manual de la placa. Muchas de ellas venían con esquemas resumidos, con el esquema de los *jumpers* para cada voltaje, bus y multiplicador soportados por la placa.



El soquete 7 usado por los procesadores más antiguos

Si, por casualidad, no teníamos el manual de nuestra placa base, podíamos conseguirlo con cierta facilidad a través de Internet. Los fabricantes ya disponían los manuales en formato digital y de forma gratuita ya que, además, mantienen un archivo con los manuales de placas antiguas. En algunos casos es posible conseguir manuales de placas base para los procesadores *486*. Si sabemos la marca y el modelo de la placa base, basta ir directamente a la página Web del fabricante.

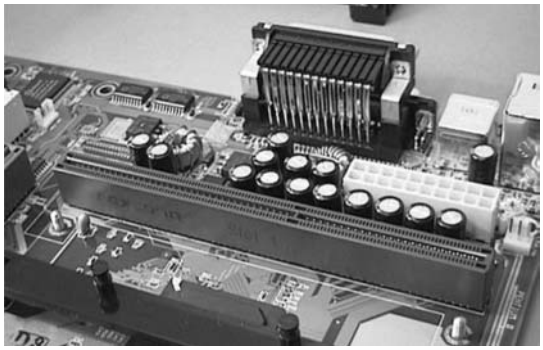
Slot 1

Mientras el *Pentium* y el *Pentium MMX* utilizaban el *soquete 7,* la siguiente familia de procesadores de la casa Intel pasó a utilizar un nuevo encaje, el famoso *slot 1*, que fue utilizado por todos los procesadores *Pentium II* y por algunas versiones del *Celeron* y del *Pentium III*.

Como en el caso de las placas *soquete 7*, también existió la división entre placas antiguas y placas más modernas y, como siempre, el manual de la placa base servía para ayudar a descubrir los procesadores que eran soportados por cada una. Pero podíamos tener una idea de qué procesadores soportaba en base al *chipset* utilizado en la placa base. Podemos descubrir el chipset de la placa base por el número de identificación del *BIOS* o con ayuda de la *Web* del fabricante.

Las placas con *chipsets* i440FX, i440LX e i440EX sólo soportaban el *Pentium II* de hasta 333 MHz y el *Celeron*. En el caso del *Celeron*, la placa soportaba los modelos de hasta 500 MHz y los modelos de 533 MHz que usasen un voltaje de 1.65 voltios. En general, no eran soportados los modelos de 533, 566 y 600 que utilizaban un voltaje de 1.5 voltios, pues estos procesadores poseían unos requisitos eléctricos diferentes, pues ya estaban basados en la arquitectura *Coppermine*.

En el caso de que el *Celeron* viniese en el formato *PPGA* era necesario un adaptador *soquete 370/slot 1* que costaba alrededor de las 5.000 pesetas y era fácil de encontrar.



El slot 1 para los procesadores en forma de tarjeta

Si al ser instalado, la placa base reconocía de forma incorrecta al procesador o simplemente no funcionaba en conjunto con él, era necesario hacer un actualización del *BIOS*. En resumen, era necesario bajar los archivos de actualización adecuados para nuestro modelo de placa base desde la página *Web* del fabricante, junto con el programa que realizaba la grabación, arrancar el sistema vía disquete y ejecutar el programa de instalación vía *MS-DOS*.

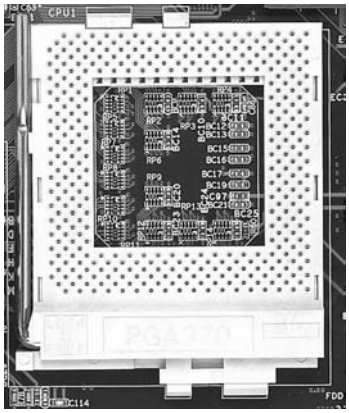
Las placas posteriores, basadas en los *chipsets* i440BX, i440ZX, i815 o Via *Apollo Pro* soportaban todas las versiones del *Pentium II*, todas las versiones del *Celeron* hasta los 500 MHz (o 533 MHz con una tensión de 1.65 voltios) y, en la gran mayoría de los casos, también el *Pentium III* y los *Celeron* de 566 y 600 MHz. Consultando la *Web* del fabricante podíamos obtener una idea de estas posibilidades. En algunos casos era necesario actualizar el *BIOS* para añadir el soporte adecuado para el procesador.

Soquete 370

El *soquete 370* utilizaba el mismo sistema de contactos del *slot 1*, la diferencia sólo estaba en el formato del encaje. Era el encaje utilizado por los procesadores *Celeron* y *Pentium III* que todavía podemos encontrar en la actualidad.

El *slot 1* surgió a causa del Pentium II, que al usar memoria caché externa era mucho mayor que los procesadores anteriores, demasiado grande para ser ensamblado en un *soquete*. Tanto el *Celeron* como las últimas versiones del *Pentium III* traían la memoria caché *L2* integrada en el núcleo del procesador, por lo cual utilizar un encaje mayor sólo servía para encarecer el precio de los procesadores y placas base. Para bajar costes, se volvió a utilizar un *soquete*.

Físicamente, el *soquete 370* era parecido al antiguo *soquete 7*, la diferencia consistía en que tenía algunos contactos más. A pesar de ser parecidos, los dos encajes eran incompatibles y no existía ningún adaptador para ensamblar un procesador *soquete 7* en una placa base *slot 1* y viceversa.

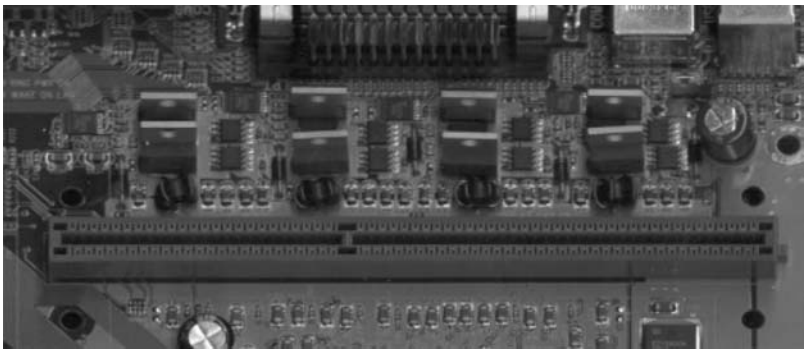


El soquete 370 de los procesadores Pentium III

Éste es el encaje usado por la gran mayoría de los procesadores Intel posteriores, lo que garantizó una compatibilidad con todos los procesadores *Pentium III* y *Celeron*. Sólo las placas base *soquete 370* más antiguas son incompatibles con los procesadores *Pentium III* debido a la falta de soporte eléctrico y, en algunos casos, por la falta de soporte al bus de 100 MHz. En algunos casos, esta limitación podía corregirse con una actualización del *BIOS*, como en las placas base *slot 1*.

Slot A y soquete A

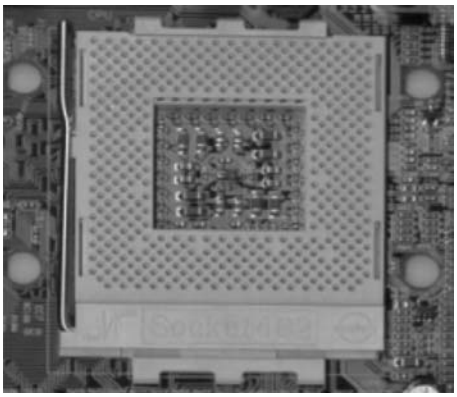
Tanto el *Athlon* como el *Duron* de la casa AMD poseen algunas semejanzas con el *Pentium II* y el *Pentium III* de la casa Intel. Así como el *Pentium II* y las primeras versiones del *Pentium III*, el *Athlon* original (el AMD *K7*) venían equipado con 512 KB de memoria caché *L2* externa, trabajando a la mitad de la frecuencia del procesador. Entonces, la casa AMD optó por utilizar un encaje semejante al *slot 1*, el *slot A*.



Un Athlon para el slot A

Veamos que el formato del *slot A* era prácticamente idéntico al del *slot 1*. La única diferencia visible era la posición del anclaje central. Esta diferencia servía para impedir que alguien encajara un *Athlon* en una placa base para *Pentium III* o viceversa, lo que podía dañar al procesador. Los *Athlon* para el *slot A* fueron vendidos durante

algunos meses hasta que fueron sustituidos por los *Athlon Thunderbird*, que traían la memoria caché *L2* integrada en el propio procesador. Tanto los *Athlon Thunderbird*, como el *Duron* sacados al mercado poco tiempo después utilizaron el encaje *soquete* A (o *soquete 462*).



El soquete A para muchos procesadores de la casa AMD

El *soquete A* es parecido al *soquete 370* utilizado por los procesadores de la casa Intel, pero posee más contactos (462 en total), organizados de forma que el procesador no pueda ensamblarse invertido. En el caso de los procesadores AMD no existe tanta confusión en cuanto a la compatibilidad.

Las placas base *slot A* son compatibles con todos los procesadores Athlon para el *slot A*, y las placas base *soquete A* son compatibles con todos los procesadores *Athlon* y *Duron* para *soquete A*.

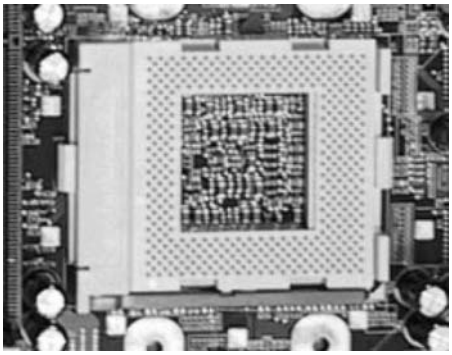
El único problema fue que las versiones posteriores del *Athlon* utilizaron un bus de 133 MHz, contra los 100 MHz utilizados en las placas base antiguas. Podemos instalar hasta un *Athlon* de 1.33 GHz (con un bus de 133 MHz) en una placa base que sólo soporte un bus de 100 MHz, pero el procesador trabajará a sólo 1.0 GHz.

Las nuevas versiones del *Athlon*, el Athlon MP y el *Athlon XP*, necesitan algunas modificaciones en el *BIOS* y en la señalización eléctrica de la placa base, lo que los hace incompatibles con muchas placas base antiguas, a pesar de que una actualización del *BIOS* resuelve el problema en algunos casos.

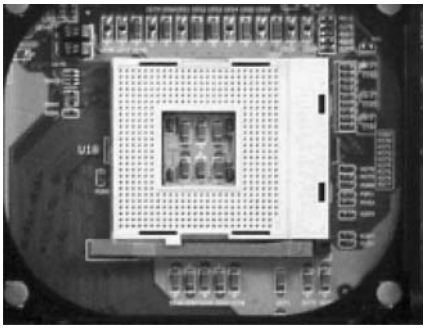
Soquete 428 y soquete 478

Estos son los dos encajes utilizados por el *Pentium 4*. El *soquete 428* es el encaje antiguo, utilizado por los procesadores *Pentium 4 Willamette*, mientras que el *soquete 478* es utilizado por los nuevos modelos, los *Pentium 4 Northwood*.

No es difícil diferenciar los dos encajes, pues a pesar de poseer más contactos, el *soquete 478* es menor, casi del tamaño de los encajes para los procesadores *386*. Por su parte, el *soquete 428* es un poco mayor que un *soquete 370*.



El soquete 428 de los procesadores Pentium 4 core Willamette



El soquete 478 de los procesadores Pentium 4 core Northwood

3.4.2 Los siguientes procesadores

Muchas cosas cambiaron después de los procesadores *Pentium II* y *K6-2*. Nunca antes, los nuevos procesadores y formatos se sacaron al mercado de forma tan rápida. Buena parte de la culpa la tuvo la casa AMD que viene manteniendo una fuerte competencia, obligando a Intel a bajar los precios y anticipar los lanzamientos para mantener su posición de liderazgo en el mercado.

El *Pentium II* y el *Celeron* fueron sustituidos por los procesadores *Pentium III* y el *Celeron core Coppermine*. Posteriormente se realizó un nuevo cambio y ambos pasaron a ser producidos usando el *core Tualatin*, de 0.13 micrones. *El Pentium 4* apareció y ya fue sustituido por una versión más moderna, el *Pentium 4 Northwood*, también fabricado en una arquitectura de 0.13 micrones.

Después del *K6-2*, la casa AMD atacó con el *Athlon*, que fue producido en varias arquitecturas. La versión original (*K7*) fue sustituida por una arquitectura de 0.18 micrones (*K75*) y enseguida por el *Athlon Thunderbird*, que dio origen al *Duron*. Inmediatamente, la familia *Athlon* se ramificó y surgieron los *Athlon 4*, *Athlon MP* y Athlon *XP*, todos ellos basados en el *core Palomino*. El *Duron* también evolucionó y los actuales utilizan el *core Morgan*, un poco más rápido que el anterior.

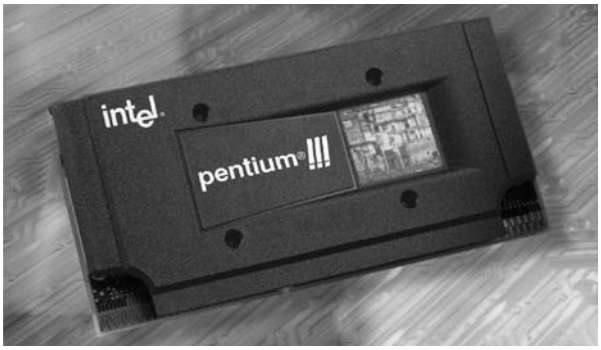
Como si no bastase, ya han surgido los procesadores de 64 bits que acabarán sustituyendo a los actuales, junto con varias tecnologías prometedoras para el futuro.

3.4.3 El Pentium III

El *Pentium III* fue el procesador principal de la casa Intel durante un buen tiempo, hasta que comenzó a ser definitivamente sustituido por el *Pentium 4*. En toda la historia de la informática, el *Pentium III* es probablemente el procesador con un mayor número de variaciones.

Existen versiones que utilizaban un bus de 100 MHz, versiones que utilizaban un bus de 133 MHz, versiones con 512 KB de caché *half-speed* (a la mitad de la frecuencia del procesador, como en el *Pentium II*), con 256 KB de caché *full-speed* (a la misma frecuencia que el procesador, como en el *Pentium Pro*), versiones que usaban el formato *SEPP*, versiones que utilizaban un nuevo formato, llamado *FC-PGA*, versiones que utilizaban el *core Katmai*, versiones que utilizaban el *core Coppermine* (más avanzado), que operaban a 2.0 voltios, a 1.65 voltios, a 1.6 voltios, etc.

Dependiendo de la versión del procesador, era necesario utilizar una placa base diferente y, en algunos casos, diferentes módulos de memoria *RAM*. Para entender todas estas variaciones, vamos a empezar a ver cada uno de los nuevos recursos introducidos en el *Pentium III*, además de la propia evolución de este procesador.



Un procesador Pentium III en su formato para el slot 1

Las instrucciones SSE

Mientras la casa Intel daba un paso en falso lanzando las instrucciones *MMX*, algo que el mercado no necesitaba y que hasta hoy no ha obtenido una aceptación significativa, la AMD apareció con su *K6-2* y sus instrucciones *3D-Now!* que realmente sí conseguían mejorar el rendimiento del procesador en un 20% en los juegos 3D con soporte para esta tecnología. A pesar de su débil coprocesador aritmético, el *K6-2* llegaba a un rendimiento muy cercano al de un *Pentium II* de la misma frecuencia en algunos juegos 3D, gracias a sus nuevas instrucciones 3D, que pasaron a ser soportadas por los principales juegos.

Como las instrucciones *3D-Now!* estaban patentadas, sólo la casa AMD (y eventualmente otra compañía que pagase por ello) podía utilizarlas en sus procesadores. Entonces, la casa Intel optó por crear su propio conjunto de nuevas instrucciones, bautizado como *SEE*, a fin de mejorar el rendimiento del *Pentium III*.

Mientras las instrucciones *MMX* mejoraban el rendimiento del sistema en el cálculo con números enteros (lo que explica su débil aceptación), las instrucciones *SSE* y las instrucciones *3D-Now!* sirvieron para agilizar los cálculos con números de punto flotante, usados principalmente en los juegos 3D y en las aplicaciones gráficas que usan las 3 dimensiones, justamente donde son más exprimidos los procesadores actuales. El conjunto de instrucciones *3D-Now!* estaba compuesto por un juego de 21 nuevas instrucciones. Por su parte, el conjunto de instrucciones *SEE* estaba compuesto por un juego de 70 nuevas instrucciones que eran capaces de mejorar el rendimiento del procesador, no sólo en juegos y aplicaciones gráficas, sino también en los programas de descompresión de vídeo, reconocimiento de voz y aplicaciones multimedia.

Como las nuevas instrucciones eran sólo software, era necesario que los programas estuviesen optimizados para hacer uso de ellas. En el caso de los juegos 3D también existía una ganancia de rendimiento si el juego utilizaba *DirectX 6.1* (o superior), optimizado para las nuevas instrucciones, o si los drivers de la tarjeta gráfica 3D estaban optimizados para las instrucciones *SSE*. Sin embargo, en este caso la ganancia era menor que en una aplicación realmente optimizada, y prácticamente imperceptible en muchos casos.

Antes de su lanzamiento hubo muchas especulaciones sobre cuál sería la ganancia real de rendimiento aportada por las nuevas instrucciones del *Pentium III*. Algunos especialistas llegaron a argumentar que la ganancia sería casi nula, pues al contrario del *K6-2*, el *Pentium III* ya poseía un excelente coprocesador aritmético. Felizmente, estas previsiones no fueron ciertas. Básicamente, las instrucciones *SSE* diferían de las instrucciones *3D-Now!* en la forma en que eran ejecutadas.

Las instrucciones *3D-Now!* permitían que algunas rutinas de los programas se ejecutasen 3 veces, o incluso 4, más rápidamente de lo que lo eran usando las instrucciones básicas del coprocesador aritmético. Las instrucciones *SSE* funcionaban de forma parecida, permitiendo que muchas rutinas utilizadas por los programas se pudiesen ejecutar de forma mucho más rápida. La ventaja era que el *Pentium III* era capaz de procesar, simultáneamente, instrucciones normales e instrucciones *SSE*, lo que conllevaba una ganancia todavía mayor de rendimiento.

Mientras que en el conjunto *3D-Now!* el programa tenía que escoger en cada momento entre utilizar una de las instrucciones básicas o una de las instrucciones *3D-Now!*, en el *Pentium III* era posible usar los dos tipos de instrucciones de forma simultánea, manteniendo las tres unidades de ejecución del coprocesador aritmético llenas durante más tiempo. La única desventaja de este proceso era que para tener un beneficio total de las instrucciones *SSE* era necesario un número mayor de modificaciones en el código de los programas, lo que en su momento desmotivó a muchos productores de

software para hacer compatibles sus programas con las nuevas instrucciones. Parece claro que las instrucciones *SSE*, debido a la manera como eran implementadas, eran más eficientes que las instrucciones *3D-Now!*, tal vez por el hecho de que los desarrolladores de la casa Intel tuvieron bastante más tiempo para crearlas e implementarlas.

El último órdago de Intel para estimular a los programadores a que incorporasen el soporte a las instrucciones *SSE* en sus programas fue el lanzamiento de un *compilador C* que añadía el soporte a las instrucciones de forma automática. Usando este *compilador*, que se distribuía gratuitamente, el programador podía crear su programa de forma normal, sin preocuparse por incluir cualquier soporte a las instrucciones *SSE*, y después de tener terminado el programa ejecutar el *compilador* para añadir el soporte de forma automática. Entonces, el compilador verificaba todas las líneas de código y añadía las instrucciones optimizadas siempre que era posible.

Debemos notar que el programa continuaba siendo compatible con los procesadores sin el conjunto de instrucciones *SSE*: si el programa era ejecutado en un *Pentium III* se utilizaban las líneas optimizadas y en caso de ser usado en otro procesador se utilizaban las instrucciones normales.

De cualquier forma, el uso de este *compilador* acabó haciendo que la mayoría de los títulos 3D y multimedia lanzados durante el año 2001 ya aportasen alguna optimización para el juego de instrucciones *SSE*. Las instrucciones *SSE* se encuentran en todas las versiones del *Pentium III* y *Pentium III Xeon*, además de las versiones del *Celeron core Coppermine*. El *Pentium 4* trae un juego de instrucciones mejorado, el *SSE 2* que veremos más adelante.

Katmai vs Coppermine y 100 vs 133 MHz

Las primeras versiones del *Pentium III*, de 450, 500, 550 y 600 MHz fueron construidas usando la misma técnica de fabricación que el *Pentium II*, o sea, utilizando el mismo encaje *slot 1*, la misma tensión de 2.0 voltios, los mismos 512 KB de caché *L2* a la mitad de la frecuencia que el procesador y la misma caché *L1* de 32 KB y un bus de 100 MHz. En esencia, no teníamos nada más que un *Pentium II* con las instrucciones *SSE*. Esto significa que en las aplicaciones que no fueron optimizados para las nuevas instrucciones, el rendimiento de los procesadores de esta versión era el mismo que presentaba un *Pentium II* de la misma frecuencia. La arquitectura (o *core*) utilizada en estos procesadores recibe el nombre de código de *Katmai*.

La excepción fue la versión de 600 MHz, que debido a la mayor frecuencia de operación utilizaba una tensión de 2.05 voltios. Este procesador poseía exactamente la misma arquitectura y se calentaba mucho más que las versiones más lentas, lo que podía ser debido a que se trataba de un *Pentium III* de 550 MHz con el *overclock* de fábrica.

Las siguientes versiones del *Pentium III* fueron las 533B y 600B. Así como las versiones anteriores, estas versiones utilizan el *core Katmai*, la diferencia era que mientras

las versiones anteriores utilizaban placas base con un bus de 100 MHz, las nuevas versiones utilizaban unas placas base con un bus de 133 MHz. La versión 533A trabajaba a 4x 133 MHz mientras que la versión 600A opera a 4.5x 133 MHz.

El bus de 133 MHz sólo servía para la memoria *RAM*; todos los demás componentes del equipo, como las tarjetas gráficas, etc. continúaban trabajando a la misma frecuencia de 66 o 100 MHz. Por ejemplo, el bus *PCI* que era el bus utilizado por los discos duros, tarjetas *SCSI* y algunas tarjetas gráficas, sonido y módems, trabajaba siempre a sus 33 MHz, independientemente de que la frecuencia de la placa base fuese de 66, 100 o 133 MHz. En realidad, sólo teníamos la frecuencia de la placa base dividida por 2, 3 o 4. El bus *AGP* que era el bus utilizado por las tarjetas gráficas *AGP* trabajaba a sus invariables 66 MHz.

Todas las versiones siguientes del *Pentium III*, lo que incluye las versiones de 650, 667, 700, 733, 750, 800, 850 y 900 MHz; 500Y, 550Y, 600Y, 533EB, 600EB, 800EB además, claro, de la versión de 1 GHz, utilizaron una arquitectura más avanzada, llamada *Coppermine*. Esta nueva arquitectura trajo varios avances sobre la arquitecura *Katmai* utilizada en los procesadores anteriores.

Para empezar, teníamos transistores menores, de sólo 0.18 micrones (contra los 0.25 del *Katmai*). Los transistores menores generaban menos calor, lo que permitió sacar procesadores más rápidos sin aumentar mucho la generación de calor. Mientras se utilizaba el *core Katmai* el límite fue el *Pentium III* de 600 MHz; utilizando el *core Coppermine* se lanzaron procesadores de hasta 1 GHz.

Los transistores menores también ocupan menos espacio, lo que permitió incluir más componentes en el núcleo del procesador; llegamos entonces al segundo avance. Mientras que en el *Pentium II* y en el *Pentium III core Katmai* la caché *L2* estaba soldada en la placa de circuito y compuesta por dos chips separados operando a la mitad de la frecuencia del procesador, en el *core Coppermine* fue movida hacia dentro del núcleo del procesador, como en el *Celeron*. Esto permitió que la memoria caché *L2* operase a la misma frecuencia que el procesador, en vez de la mitad, lo que mejoraba bastante el rendimiento del procesador. El único pero era que el *core Coppermine* sólo poseía 256 KB de memoria caché *L2*, la mitad de la que encontrábamos en las versiones anteriores del *Pentium III*.

En el *Pentium II*, el bus de comunicación entre el núcleo del procesador y la memoria caché *L2* sólo poseía 64 bits de ancho, lo que permitía transmitir sólo 8 Bytes de datos por ciclo, limitación que se extendió al *core Katmai* y también a las versiones antiguas del *Celeron*. No adelantaría mucho tener una memoria caché más rápida si no fuese posible transmitir los datos al procesador de forma más rápida.

Previendo esto, los desarrolladores de Intel ampliaron el bus de comunicación de la caché *L2* con el procesador hasta los 256 bits. Esto permitió transferir 32 Bytes de datos en cada ciclo, cuatro veces más que en los procesadores anteriores. La cuarta y última de las principales modificaciones fue la inclusión del recurso de *Advanced System*

Buffering que consistía en aumentar el tamaño de los *búfers* de datos del procesador, lo que mejoraba la capacidad de procesamiento y le permitía beneficiarse de toda la potencia de la memoria caché *L2*. Debido a todas estas modificaciones, el *Pentium III Coppermine* poseía aproximadamente 21 millones de transistores, la mayor parte consumidos por la caché *L2*, en contra de los pocos más de 9.5 millones de transistores del *Pentium II* y del *Pentium III Katmai*.

FC-PGA

En el curso del desarrollo del *Pentium III* se acabó siguiendo el mismo camino que en el *Celeron*, incorporando su memoria caché *L2* en el núcleo del procesador. A fin de abaratar costes, la casa Intel decidió lanzar versiones del *Pentium III Coppermine* en el mismo formato *PPGA* del *Celeron*. Por un lado fue bueno, pues permitió una disminución de precio en el coste final de cada procesador, ya que no se usaba la tarjeta de circuito, pero por otro lado fue malo, pues nos obligaba a comprar un adaptador para poder ensamblarlo en una placa base *slot 1*. Pero en el caso del nuevo *Pentium III Coppermine*, el nuevo encaje fue llamado *FC-PGA*.

El Pentium III en el formato FC-PGA

El mayor problema fue que, a pesar de que el encaje era el mismo que el utilizado por el *Celeron*, los requisitos del procesador a nivel de suministro eléctrico eran diferentes, ya que el *Celeron* usaba una tensión de 2.0 voltios mientras que el *Pentium III Coppermine* utilizaba una tensión de 1.6 o 1.65 voltios.

El *Pentium III Coppermine* fue producido tanto en versiones *FC-PGA* como *slot 1*. Existían adaptadores que permitían la instalación de un procesador *FC-PGA* en una placa base *slot 1*, pero de cualquier manera, era preciso que la placa base soportase a los procesadores *Pentium III Coppermine*.

Las variaciones de los distintos Pentium III

Como hemos visto hasta aquí, existieron varias variaciones del *Pentium III* en cuanto a la arquitectura, a la frecuencia del bus y al encaje. En la siguiente tabla aparecen marcados los recursos de cada versión del *Pentium III*.

Recursos	PIII 450, 500, 550, 600 MHz	PIII 533B y 600B	PIII 500E y 550E	PIII 650, 700,750, 800 y 600E	PIII 667 y 733, 533EB, 600EB, 800EB y 1 GHz
Arquitectura	Katmai	Katmai	Coppermine	Coppermine	Coppermine
Slot 1	Sí	Sí	No	Sí	Sí
FC-PGA	No	No	Sí	Sí	Sí
Bus 100 MHz	Sí	No	Sí	Sí	No
Bus 133 MHZ	No	Sí	No	No	Sí
Caché L2	512 KB half speed	512 KB half speed	256 KB full speed	256 KB full speed	256 KB full speed
Buffering	No	No	Sí	Sí	Sí

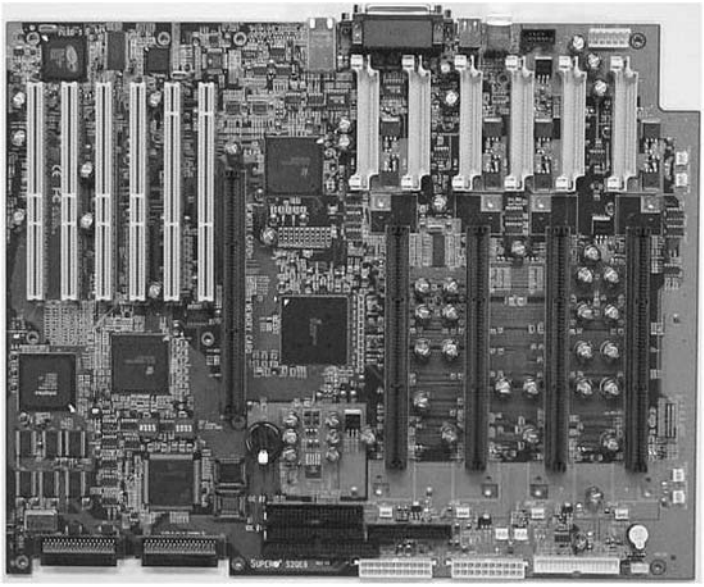
La letra **E** diferencia los procesadores *core Coppermine* de los *core Katmai* en el caso de versiones de la misma frecuencia, como en el caso de las versiones de 500, 550 y 600 MHz. En este caso, los procesadores con la letra E son los *core Coppermine*.

La letra **B** (de bus) indica a los procesadores con un bus de 133 MHz, mientras que la combinación EB indica a los procesadores que, al mismo tiempo, utilizan *core Coppermine* y un bus de 133 MHz, como en el caso de la versión EB de 800 MHz. Veamos que, en general, estas siglas son utilizadas para diferenciar a los procesadores de la misma frecuencia, no siendo usadas en el caso de los procesadores de 667 MHz y 733 MHz por ejemplo, ya que éstos sólo utilizan un bus de 133 MHz y la arquitectura *core Coppermine*.

3.4.4 Pentium III Xeon

El *Pentium III Xeon* fue la encarnación del *Xeon* basado en la arquitectura del *Pentium III*. Las características eran semejantes a las del *Pentium II Xeon*, con versiones equipadas con hasta 2 MB de memoria caché *L2* y soporte para el multiprocesamiento, pero le fue añadido el soporte a las instrucciones *SSE*. Ambos *Xeon* utilizaban un encaje diferente, el *slot 2*, y por eso requerían el uso de unas placas base específicas, destinadas a servidores y que eran bastante más caras.

Una curiosidad es que las versiones con 1 y 2 MB de memoria caché *L2* sólo existieron con frecuencias de hasta 700 MHz, por encima sólo existió la versión de 256 KB, que no era más que un *Pentium III Coppermine* común que usaba el encaje *slot 2* y permitía el uso de hasta 4 procesadores. Estos "supuestos" *Xeon* existieron en versiones de hasta 1 GHz, como el *Pentium III Coppermine*.



Una placa base para los Pentium III Xeon

3.4.5 Celeron Coppermine (Celeron II)

Así como el *Pentium III*, que después de la versión de 600 MHz pasó a producirse utilizando la arquitectura *Coppermine*, permitió frecuencias mayores debido al uso de unos transistores de menor tamaño, el *Celeron*, siguiendo su evolución natural, también pasó a ser producido con la nueva técnica de producción a partir de la versión de 533 MHz. Notemos que, a pesar de ser más raros, también existieron *Celeron* de 533 MHz producidos usando la arquitectura antigua. Para diferenciar las dos series, el *Celeron Coppermine* de 533 MHz fue llamado 533A.

Manteniendo la intención de producir un procesador de bajo coste que tuviese un buen rendimiento y que, al mismo tiempo, no compitiese directamente con los procesadores más caros, aparecieron cuatro desventajas del *Celeron Coppermine* sobre un *Pentium III Coppermine* de la misma frecuencia. La primera fue una frecuencia más baja del bus. El *Celeron Coppermine* continuó utilizando un bus de 66 MHz con el multiplicador bloqueado. Por ejemplo, un *Celeron* de 600 MHz utilizaba un multiplicador de 9x. Mientras más alto era el multiplicador, más bajo era el rendimiento. A partir del *Celeron* 800 ya se utilizó un bus de 100 MHz, atenuando (que no resolviendo) el problema.

La segunda desventaja fue el hecho de que el *Celeron* continuaba viniendo con sólo 128 KB de memoria caché *L2 full-speed* (trabajando a la misma frecuencia del procesador), mientras que el *Pentium III Coppermine* venía con el doble de cantidad: 256 KB de caché *L2 full-speed*. Las dos últimas desventajas también estaban relacionadas con la memoria caché *L2*. Además de ser de menor tamaño, la memoria caché *L2* del *Celeron Coppermine* estaba recortada de dos formas más. Primero, la caché del

Celeron poseía 4 líneas de asociación, en contra de las 8 líneas de la caché del *Pentium III*. Más líneas de asociación mejoraban la eficiencia de la memoria caché, principalmente cuando se usaba mucha memoria *RAM*. Según en que casos, la caché *L2* del *Celeron* tenía una latencia de dos ciclos, mientras que en el *Pentium III* no existía latencia alguna.

Sumando todos estos factores, el *Celeron* se quedaba detrás del *Pentium III Coppermine* en términos de rendimiento. En algunas aplicaciones, el *Celeron Coppermine* llegaba a ser hasta un 20% más lento que un *Pentium III Coppermine* de la misma frecuencia. Ya no era lo mismo como en los tiempos del *Pentium II*, donde el Celeron con memoria caché presentaba casi el mismo rendimiento que los procesadores *Pentium II* de la misma frecuencia.

En términos de la placa base, necesitábamos los mismos requisitos que en el *Pentium III Coppermine*, ya que el *Celeron* era, esencialmente, un *Pentium III Coppermine* con las modificaciones que limitaban su rendimiento.

Los modelos 533A, 566 y 600 MHz y posteriores del *Celeron* utilizaban la misma arquitectura del *Pentium III*, el famoso *core Coppermine*. En realidad, los procesadores no sólo compartían la misma arquitectura, sino que eran procesadores *Pentium III* que tenían la mitad de su memoria caché *L2* deshabilitada desde la fábrica.

Este fue otro caso más de la famosa línea de producción unificada de la casa Intel. Al llevar la memoria caché *L2* integrada en el procesador, existían un gran número de defectos de fabricación en la caché de los procesadores Pentium III. Como la memoria caché formaba parte del procesador, un único bit defectuoso de la caché condenaba todo el procesador a la basura, como acontecía en la época del *Pentium Pro*. Cuando ocurría un defecto en la caché, generalmente sólo pocos bits estaban defectuosos y, además, bits casi siempre próximos.

Antes de salir de fábrica, todos los procesadores son rigurosamente probados y los que presentan defectos en la memoria caché son apartados, ya que el defecto causaría bloqueos en el sistema. Entonces aparecía un problema. Qué hacer con esos procesadores *Pentium III* con defectos en la memoria caché? La solución adoptada por la casa Intel fue transformarlos en *Celeron*.

Como ya hemos dicho, cuando existía un defecto en la caché, casi siempre el error ocurría en algunos bits físicamente muy próximos. Dado que el *Celeron* sólo poseía 128 KB de memoria caché *L2*, bastaba deshabilitar la mitad de la caché donde estaba el defecto. Como sólo se usaba la mitad "buena" de la caché, el procesador funcionaba perfectamente.

3.4.6 Cyrix/Via C3

Después de lanzar los procesadores *6x86MII* que acabaron siendo un fracaso de ventas, la casa Cyrix empezó a trabajar en un nuevo proyecto de procesador que pudiera competir con los procesadores de las casa Intel y AMD.

La indecisión en el nombre del proyecto duró hasta que la casa Via compró la casa Cyrix y fue cuando el proyecto obtuvo su nombre definitivo, *Joshua*. A pesar de una historia complicada, *Joshua* no pasó de ser un mero proyecto, pues no llegó a ser lanzado al mercado. *Joshua* venía con 64 KB de caché *L1* y 256 KB de caché *L2*, ambas trabajando a la misma frecuencia que el procesador. El problema era que esta combinación derivó en un procesador caro de producir, que no podía competir con los *Celeron* y *Duron*, que además de ser más rápidos, también eran más baratos. *Samuel 1* fue la primera tentativa de la casa Cyrix de producir un procesador más barato, con 128 KB de caché *L1*, pero sin memoria caché *L2*, pero aún era más lento que el *Joshua*.

El próximo de la lista fue llamado *Samuel 2*, que llegó a ser lanzado al mercado con el nombre de *Cyrix III* a pesar de volver a tener poco éxito. Tenía 128 KB de caché *L1* y 64 KB de caché *L2* y era más caro que un *Duron* (a pesar de ser un poco más barato que un *Celeron*) y existió en una única versión de 700 MHz.

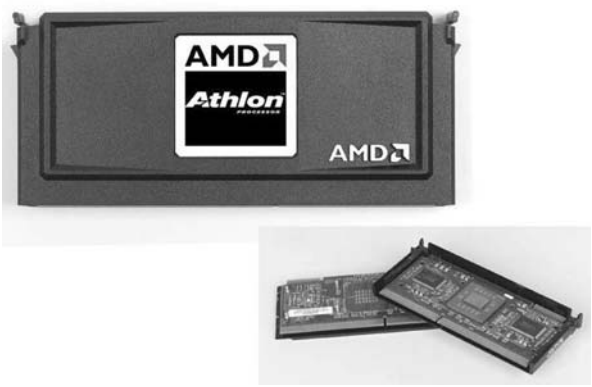
Después de toda la confusión, finalmente la casa Cyrix llegó a su proyecto definitivo, que fue lanzado bajo el nombre de *Via C3*. El *C3* era un procesador de bajo coste compatible con las placas base para *Pentium III* y *Celeron*. El rendimiento del *C3* no era muy espectacular, pues era levemente inferior al de un *Celeron* de la misma frecuencia en las aplicaciones de oficina pero perdía con gran diferencia en las aplicaciones que exigían un alto rendimiento en las operaciones de punto flotante. Se trataba de un procesador más adecuado para PCs básicos, que se usaban para *Office* e *Internet*.

El procesador VIA C3

3.4.7 AMD Athlon (K7)

Del *K6* al *K6-3* poca cosa cambió en la arquitectura de los procesadores AMD. El *K6-2* trajo las nuevas instrucciones *3D-Now!*, mientras que el *K6-3* ganó una memoria caché *L2* que trabajaba a la misma frecuencia del procesador y pasó a utilizar la caché que encontraba en la placa base en forma de caché *L3*. También se mejoraron las técnicas de producción, lo que permitió el lanzamiento de procesadores que trabajaban a frecuencias más altas. Sin embargo, el núcleo del procesador, así como el coprocesador aritmético, no fueron modificados en prácticamente nada. Sólo fueron añadidos algunos "accesorios" al *K6*, que lo hicieron más rápido.

Durante mucho tiempo, la casa AMD prometió un nuevo procesador en el que se solucionaría el viejo problema de rendimiento de los procesadores AMD en las aplicaciones gráficas y que, finalmente, sería capaz de presentar un rendimiento igual o superior al de un procesador Intel equivalente en todas las aplicaciones. Cuando finalmente fue lanzado, el *K7* se ganó el apodo de *Athlon*.



El procesador Athlon para slot A

El *Athlon* fue un proyecto de procesador completamente nuevo, al contrario del *Pentium III* que utilizó la misma arquitectura del *Pentium II*, sólo con la incorporación de las nuevas instrucciones *SSE* y una caché *L2* más rápida, en el caso de la versión *Coppermine*.

A fin de alcanzar todas las metas propuestas de rendimiento, la casa AMD optó por abandonar la idea de un procesador de bajo coste, como teníamos en los *K6* y optar por un proyecto realmente "generoso" en términos del número de transistores y recursos. A pesar de todo, la casa AMD consiguió mantener los precios de venta más bajos que los procesadores Intel.

Arquitectura

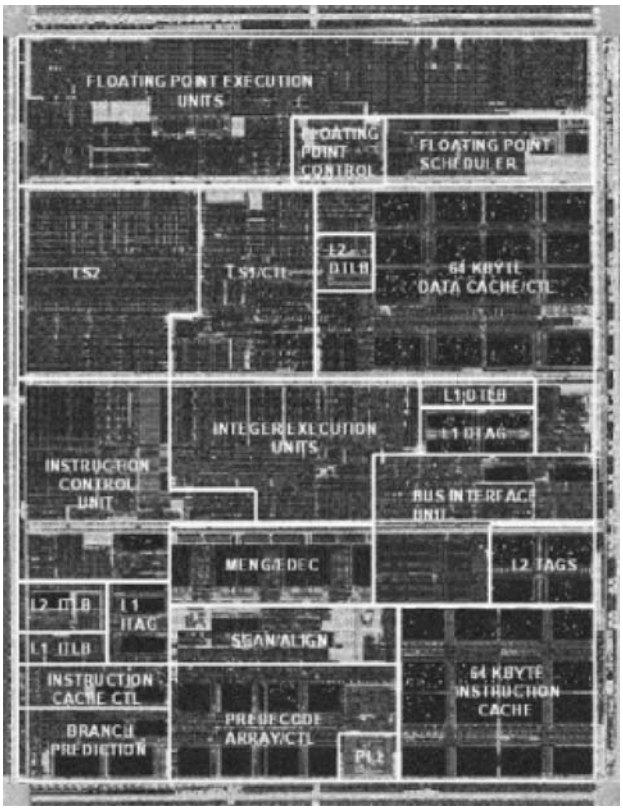
La idea fundamental de la casa AMD al desarrollar el *Athlon* parece que fue la de "más y más". El *Athlon* tenía más niveles de *pipeline*, más unidades de *FPU*, más unidades de ejecución y más registros, lo que dio lugar a un mejor rendimiento. El *Athlon* se encontraba dividido en las siguientes partes:

Floating Point Execution Units: son las unidades de ejecución de punto flotante. No es más que la parte fundamental del coprocesador aritmético, fundamental en varias las aplicaciones más actuales.

Floating Point Control: este componente controla el orden en que las instrucciones son ejecutadas por el coprocesador aritmético, permitiendo que las unidades de punto flotante queden ocupadas durante la mayor parte del tiempo.

Floating Point Scheduler: almacena las próximas instrucciones que van a ser procesadas por el coprocesador aritmético. Este componente es esencial para el trabajo del *Floating Point Control*.

64 KByte Data Cache: los 128 KB de memoria caché *L1* del *Athlon* estaban divididos en dos bloques, 64 KB para datos y 64 KB para instrucciones. Esta división era la utilizada en la mayoría de los procesadores y mejoraba la velocidad de acceso, ya que permite que los dos bloques puedan ser accedidos de forma simultánea. El *Data Cache* es la mitad que almacena los datos.



El interior de un procesador Athlon

Integer Execution Unit: estas son las unidades de ejecución de enteros. Este es el componente básico de cualquier procesador. Es aquí donde son procesadas las operaciones en las que participan los números enteros.

Instruction Control Unit: es el circuito que controla el envío de instrucciones a las unidades de ejecución de enteros. También ordena las instrucciones de forma que puedan ser procesadas de la forma más rápida.

Bus Interface Units: es por donde entran y salen los datos del procesador. Controla la comunicación del procesador con el *chipset* y con los demás componentes del equipo.

64 KByte Instruction Cache: es el segundo bloque de la memoria caché *L1*, el bloque que almacena las instrucciones.

Branch Prediction: debemos recordar el recurso de ejecución especulativa, usado a partir del *Pentium*, donde eran procesadas instrucciones aleatorias en los ciclos ociosos del procesador, ayudando a aumentar el rendimiento. Éste es el circuito que controla la ejecución de estas instrucciones.

Predecode Array: es la puerta de entrada del procesador. Estos circuitos convierten las instrucciones *x86* enviadas por los programas en las instrucciones simples que el procesador ejecuta internamente.

El funcionamiento

Para entender lo que hacía ser más rápido al *Athlon* que a los procesadores anteriores de la casa AMD, nada mejor que dar una vuelta por cómo procesaba las instrucciones. Hasta cierto punto, tanto el *Pentium III* como el *Athlon* y otros procesadores *x86* trabajaban de la misma manera. Internamente, el procesador sólo era capaz de ejecutar instrucciones simples y, más exactamente, sólo cuatro instrucciones: suma, asignación, lectura y escritura.

Si ya hemos estudiado algún lenguaje de programación, sabremos de la importancia de las variables, que son pequeños espacios de memoria reservados para guardar algún tipo de dato. Existen varios tipos de variables, de 8, 16, 32 bits, etc. que cambian de nombre dependiendo del lenguaje de programación usado. La instrucción de asignación es usada siempre que es necesario crear o alterar el valor de una variable.

Por ejemplo, imaginemos que un programa creó una variable de 8 bits que almacena el número 5. La próxima instrucción manda que el programa compare el valor de la variable con el número 6 y, si el número es menor, modifique el valor a 9. Como 5 es menor que 6, el programa decide hacer la modificación llevada a cabo utilizando la operación de asignación del procesador, que lee el valor 9 y lo escribe en el espacio de memoria de la variable, que pasa a tener almacenado el valor 9 en vez del 5. La instrucción de suma es la operación básica que permite hacer todo tipo de procesamientos, mientras que las instrucciones de lectura y escritura permiten mover los datos. Básicamente, esto es lo que sólo sabe hacer el procesador. Las operaciones más complejas, son ejecutadas a través de la combinación de varias instrucciones simples.

Para realizar una multiplicación, por ejemplo, el procesador utilizará de forma secuencial varias operaciones de suma. En realidad, dentro del procesador todas las operaciones, incluso las más complejas, son realizadas en base a varias operaciones de suma, realizadas entre los valores binarios procesados por el procesador. Una operación de resta es llevada a cabo a través de una asignación que transforma uno de los valores en su valor negativo, seguida por una operación de suma. Una operación de división es llevada a cabo ejecutando una secuencia de operaciones de resta y todos los demás cálculos, incluso los cálculos más complejos ejecutados por el coprocesador

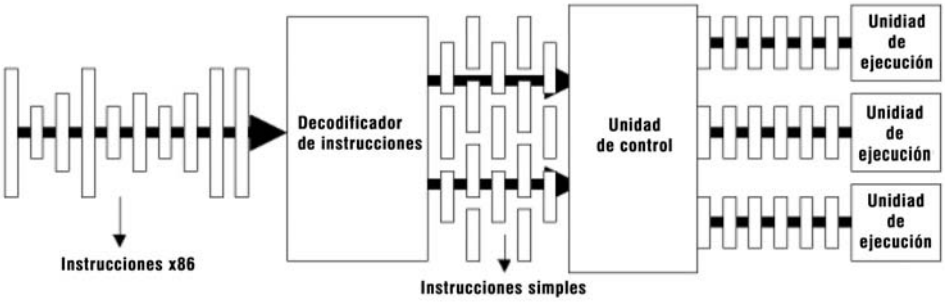
aritmético, son resueltos usando las cuatro operaciones, obtenidas a partir de la simple instrucción de suma.

Pues bien, el conjunto de instrucciones *x86* utilizadas por los programas y con las que cualquier procesador destinado a equipos PC debe ser compatible, está compuesto de instrucciones simples (suma, resta, lectura, escritura, comparación, etc.) como por instrucciones más complejas, que deben ser separadas en varias instrucciones simples para que puedan ser ejecutadas por el procesador.

Excluyendo los componentes de apoyo, como la memoria caché *L1*, y sólo dejando la parte "funcional" del procesador, podemos dividir el procesador en tres partes. La primera parte es el decodificador de instrucciones. Este componente tiene la función de convertir las instrucciones *x86* usadas por los programas en las instrucciones simples que pueden ser ejecutadas por el procesador. Las instrucciones simples van a una unidad de control, que organiza las instrucciones para que puedan ser ejecutadas de la forma más rápida. Entonces, las instrucciones forman una fila a fin de permitir que la unidad de control tenga tiempo de hacer este trabajo.

Debemos recordar que los procesadores actuales son superescalares, ejecutan varias instrucciones por ciclo de forma simultánea, lo que hace esencial la existencia de algún circuito que las coloque en el orden en el que la ejecución de una no dependa del resultado de la ejecución de otra.

Finalmente, tenemos las unidades de ejecución, donde las instrucciones preparadas y organizadas son finalmente procesadas. Veamos que todos los circuitos trabajan simultáneamente, supervisando que las unidades de ejecución tengan siempre algo para procesar.



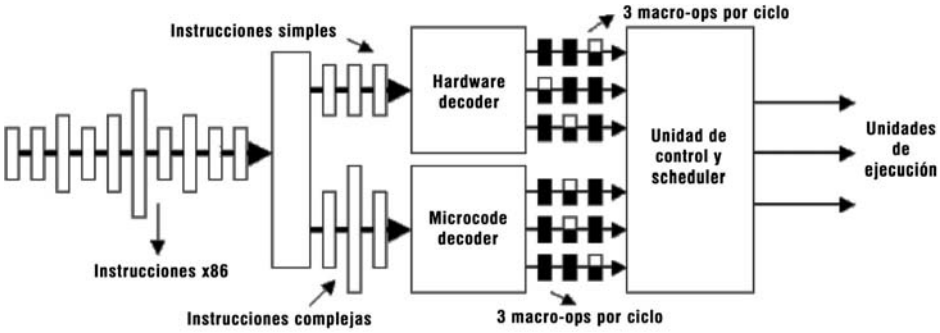
El paso de las instrucciones hasta las unidades de ejecución

Podríamos entonces dividir el procesador en dos partes. La parte frontal, o *front-end*, está compuesta por todos los circuitos que decodifican y ordenan las instrucciones de los programas. Esto incluye el decodificador de instrucciones, la unidad de control y también los registros y la memoria caché. La parte trasera, o *back-end*, está compuesta por las unidades de ejecución, donde realmente son procesadas las instrucciones. Cada una de las dos partes es igual de importante. Las unidades de ejecución deben

ser capaces de procesar el máximo de instrucciones, mientras que el *front-end* que está compuesto por los demás componentes, tiene la función de preparar las instrucciones lo más rápidamente posible, de modo que las unidades de ejecución permanezcan ocupadas la mayor parte del tiempo. La lógica es que mientras más unidades de ejecución estén trabajando al mismo tiempo, serán capaces de procesar un mayor número de instrucciones todas juntas y mientras más circuitos de decodificación y de control tengamos, más eficiente será la decodificación de las instrucciones, resultando un procesador más rápido. A continuación, vamos a ver una comparación entre un *Athlon* y un *Pentium III* para ver los distintos funcionamientos y entender mejor los distintos componentes del mismo.

El decodificador de instrucciones

La primera gran diferencia entre el *Athlon* y el *Pentium III* de la época residía en la manera como eran decodificadas y procesadas las instrucciones. Básicamente existen dos tipos de instrucciones simples que puede ejecutar el procesador: las operaciones aritméticas (suma o asignación) y las instrucciones de lectura o escritura. La mayoría de las veces, una instrucción aritmética viene acompañada de una operación de lectura o escritura, ya que después de ejecutar la operación es necesario guardar el resultado en algún lugar.



La decodificación de instrucciones en el Athlon

En el *Athlon*, las instrucciones eran decodificadas en forma de pares. Cada par, llamado "*macro-ops*" podía tener una instrucción aritmética y una instrucción de lectura o escritura. Si la opción de lectura o escritura no era necesaria, el espacio reservado para ella quedaba libre. El *Athlon* tenía dos decodificadores de instrucciones. El primer decodificador es conocido por *Hardware decoder* y se encargaba de convertir todas las instrucciones simples, mientras que el segundo decodificador, llamado Microcode decoder tenía la función de convertir las instrucciones más complejas. Cada uno de los dos decodificadores podía enviar 3 pares de instrucciones, o tres *macro-ops*, por cada ciclo a la unidad de control, donde las instrucciones eran ordenadas y enviadas a las unidades de ejecución. Como las unidades de ejecución del *Athlon* procesaban tres instrucciones por ciclo, este modo aseguraba que en cualquier caso, aún si sólo entraban instrucciones simples, o sólo instrucciones complejas, el procesador tenía por lo menos tres instrucciones para procesar en cada ciclo.

El *Pentium III* procesaba cada instrucción de forma separada y no en pares como en el *Athlon*. Cada instrucción era llamada "*uops*". Disponíamos de un total de tres decodificadores de instrucciones: dos decodificadores simples, que se encargaban de las instrucciones simples y decodificaban una instrucción (cada uno) por ciclo, y un decodificador más complejo que se encargaba de las instrucciones *x86* más complejas. Este segundo decodificador era capaz de decodificar hasta 4 instrucciones por ciclo.

En un principio parece que existía un empate técnico, ya que sumados los tres decodificadores tenemos la decodificación de 6 instrucciones por ciclo, como en el *Athlon*. En la práctica, sin embargo, cuando el decodificador de instrucciones complejas estaba ocupado, los dos decodificadores simples se paraban. Esto hacía que, en muchas situaciones, los decodificadores no fuesen capaces de producir todas las instrucciones que podían ser procesadas por las unidades de ejecución, haciendo que se quedasen ociosas durante buena parte del tiempo, lo que naturalmente significaba una pérdida de rendimiento.

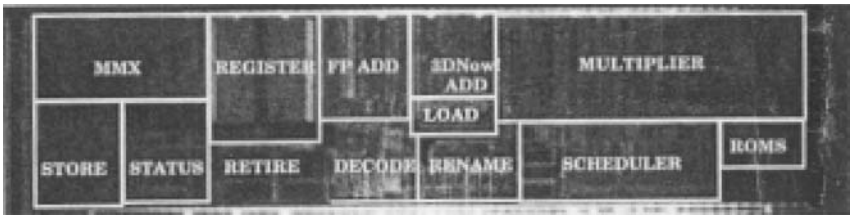
Coprocesador aritmético

El gran problema de los procesadores *K6* era el rendimiento del coprocesador aritmético, cosa que ya hemos comentado anteriormente. La gran ventaja del coprocesador aritmético utilizado en los procesadores Intel desde el *Pentium* era el recurso del *pipeline*, que combinaba varias unidades de ejecución en el mismo procesador, permitiendo que se procesasen varias instrucciones por ciclo.

Toda la familia de procesadores *K6* compartía el mismo proyecto de coprocesador aritmético, equipado con una única unidad de ejecución. El hecho de que sólo tuviesen una única unidad eliminaba la necesidad de adjuntar los circuitos que ordenaban las instrucciones, lo que simplificaba bastante el proyecto del procesador, pero que tenía un elevado coste en términos de rendimiento.

Para intentarlo compensar, la casa AMD proyectó esta unidad de ejecución solitaria con mucho cuidado, consiguiendo que tuviera una latencia más baja. Esto se tradujo en una ganancia considerable de rendimiento, que hizo que la unidad de ejecución fuese más rápida que cada una de las unidades de ejecución del *Pentium III*. El problema era que el *Pentium III* tenía tres unidades de ejecución de punto flotante, o sea, ganaba en cantidad. Felizmente, esto cambió en el *Athlon*, que pasó a adoptar un coprocesador aritmético con tres unidades de ejecución, superior al del *Pentium III* en varios aspectos.

Tanto en el *Pentium III* como en el *Athlon*, el coprocesador estaba compuesto por tres unidades de ejecución, llamadas *FADD*, *FMUL* y *FSTORE*. *ADD* es la abreviación de adición, *MUL* es la abreviación de multiplicación y *STORE* significa guardar. La *FADD* era la encargada de ejecutar las operaciones de suma, en las que intervenían números de punto flotante, la *FMUL* ejecutaba las operaciones de multiplicación, división, las instrucciones *MMX* e instrucciones *3D-Now!*, mientras que la *FSTORE* ejecutaba las operaciones de lectura/escritura y algunas operaciones más diversas.



La distribución de los componentes del coprocesador en el Athlon

Tanto en el *Pentium III* como en el *Athlon*, las tres unidades de ejecución de punto flotante poseían *pipeline* y en un mismo número de niveles, y por ello ejecutaban las mismas funciones. Entonces, en principio, eran equivalentes, pero sólo en principio. En el *Athlon* era posible ejecutar simultáneamente operaciones de suma, operaciones de multiplicación y operaciones de lectura/escritura, pues cada unidad era completamente independiente de las demás. Esto significa que en circunstancias ideales, el coprocesador aritmético del *Athlon* era capaz de procesar tres instrucciones por ciclo.

Para economizar transistores, los desarrolladores de la casa Intel optaron por compartir las mismas subunidades de ejecución entre las unidades de suma y multiplicación del *Pentium III*. Esto significa que sólo una de las dos unidades podía funcionar a la vez: o se realizaba una multiplicación y una operación de lectura/escritura, o entonces se realizaba una suma y una operación de lectura/escritura, nunca se ejecutaban las tres operaciones al mismo tiempo. Al contrario que en el *Athlon*, el coprocesador aritmético del *Pentium III* sólo era capaz de ejecutar dos instrucciones por ciclo en condiciones ideales. El motivo de esto es simple: las operaciones de multiplicación son secuencias de varias operaciones de suma. Como las dos unidades poseían el mismo núcleo básico (que realizaba las sumas), era posible crear un proyecto donde tanto la unidad de suma como la de multiplicación utilizasen el mismo núcleo. Con eso, era posible economizar un gran número de transistores, sacrificando también parte del rendimiento, ya que sólo una de las dos unidades podria usar el núcleo de suma.

En la práctica, el coprocesador del *Athlon* no llegaba a ser dos veces más rápido que el coprocesador del *Pentium III*, ya que sólo era posible utilizar una instrucción de lectura/escritura por ciclo y tampoco siempre era posible entregar una instrucción de suma y otra de multiplicación en un mismo ciclo, pues la demanda de cada tipo de instrucción dependía de la aplicación que se estaba ejecutando. De cualquier manera, el coprocesador aritmético del *Athlon* siempre era capaz de trabajar de una forma más eficiente que el coprocesador del *Pentium III*.

Es preciso recordar que a pesar de esta superioridad técnica, los primeros modelos del *Athlon* no tenían soporte para las instrucciones *SSE* y estaban equipados con una memoria caché *L2* que trabajaba a la mitad de la frecuencia del procesador, como en el *Pentium II*. Sumadas estas dos desventajas anulaban la superioridad del coprocesador aritmético, haciendo que los *Athlon K7* y *K75* continuasen siendo más lentos que los *Pentium III Coppermine* en varias aplicaciones.

La situación cambió con la aparición del *Athlon Thunderbird*, equipado con una memoria caché L2 que trabajaba a la misma frecuencia del procesador y, más recientemente, con los *Athlon* basados en el *core Palomino*, que además de conservar una memoria caché más rápida, trajeron un soporte parcial a las instrucciones *SSE* del *Pentium III*. Actualmente, el concurrente del *Athlon* ya no es el *Pentium III* sino el *Pentium 4*, que aún es inferior a un *Athlon* de la misma frecuencia en la mayoría de las aplicaciones, pero que en cambio es capaz de alcanzar unas frecuencias de trabajo mucho más altas.

El bus EV6

Todos los componentes del equipo, incluyendo el bus *PCI*, el bus *AGP*, la memoria *RAM*, etc. están conectados al *chipset*, que funciona como una especie de intermediario entre el procesador y los demás componentes del equipo. Si fuera preciso acceder a algún dato de la memoria *RAM* por ejemplo, el procesador entraría en contacto con el *chipset* y éste recogería el dato de la memoria y se lo entregaría al procesador.

En el *Pentium III*, el bus de comunicación entre el procesador y el *chipset* se llamaba *GTL+* y operaba a la frecuencia de la placa base, 100 o 133 MHz, dependiendo del modelo de procesador. Como el bus GLT+ era una arquitectura propiedad de la casa Intel, la casa AMD optó por utilizar el bus *EV6* desarrollado por la compañía Alpha Digital. El bus *EV6* también trabajaba a la misma frecuencia que la placa base (que también podía ser de 100 o 133 MHZ, dependiendo del modelo de *Athlon* utilizado) pero permitía dos transferencias de datos por ciclo, lo que en la práctica resulta en un rendimiento equivalente al de un bus que trabajaba a 200 o 266 MHz. Las dos transferencias de datos por ciclo del bus *EV6* permiten que los procesadores *Athlon* y *Duron* saquen provecho de las memorias *DDR*, que también realizan dos transferencias por ciclo.

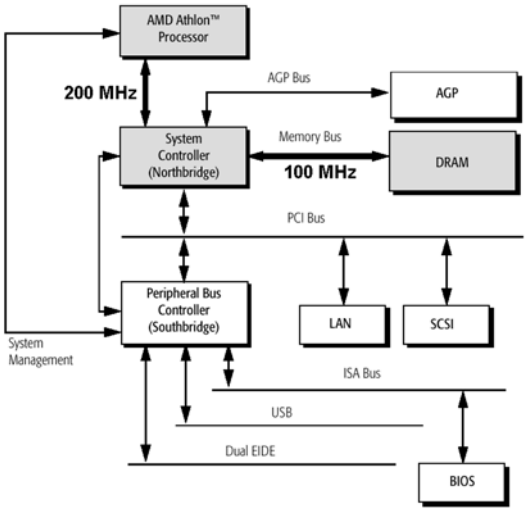


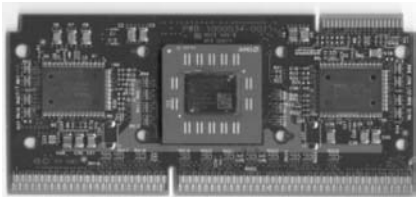
Diagrama del funcionamiento de un equipo con un Athlon

Actualmente, el *Athlon* también soporta el multiprocesamiento y es, justamente, en este ramo donde el bus *EV6* demuestra todo su potencial. Imaginemos un sistema donde tenemos dos procesadores *Pentium III*. Los 2 procesadores están encajados en la misma placa base y, consecuentemente, conectados al mismo *chipset*. El bus *GTL+* es un bus compartido, y esto significa que el mismo bus de 100 o 133 MHz es compartido por los dos procesadores. Esto significa que sólo uno de los dos procesadores podrá usar el bus cada vez. Si uno necesitara acceder a un dato de la memoria y otro necesitara enviar datos a la tarjeta gráfica, el segundo tendrá que esperar a que el primero termine su tarea antes de poder iniciar la suya. Esto causa una gran pérdida de tiempo y disminuye el rendimiento del sistema conforme más procesadores se van añadiendo, una posibilidad permitida, en cambio, por el *Pentium III Xeon*.

El bus *EV6*, por su parte, es un bus punto a punto. En él, cada procesador tiene su bus exclusivo de comunicación con el chipset. Esto permite que en un sistema con varios procesadores *Athlon* trabajando en paralelo, cada uno pueda acceder a los demás componentes del equipo en el momento deseado, sin tener que esperar su turno. Otra ventaja es que usando la comunicación punto a punto entre los procesadores y el *chipset*, el nivel de ruido electromagnético pasa a ser más bajo, lo que añade una ventaja más desde el punto de vista del rendimiento.

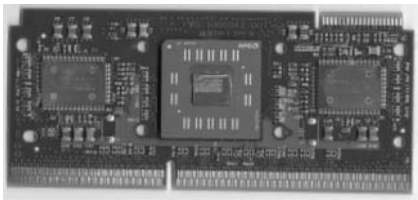
K7 vs K75

Las primeras versiones del *Athlon*, de 500, 550, 600, 650 y 700 MHz se fabricaron utilizando la vieja técnica de fabricación de 0.25 micrones, la misma utilizada en el *K6-2*. Esta primera generación utilizó el *core K7*, el proyecto original del *Athlon*.



Un Athlon con la tecnología de 0.25 micrones

A partir de finales del año 1999, la casa AMD empezó a producir el *Athlon* basándose en una arquitectura de 0.18 micrones. Fueron lanzadas entonces nuevas versiones hasta la barrera de 1 GHz. Esta arquitectura de 0.18 micrones obtuvo el nombre de códgio *K75*.



Un Athlon con la tecnología de 0.18 micrones

La memoria caché L2

Tanto los *Athlon K7* como los *Athlon K75* tenían una limitación, que era el uso de memoria caché externa, trabajando a una fracción de la frecuencia del procesador. Además de no ser una buena opción desde el punto de vista del rendimiento, todavía encarecía más los procesadores, pues el fabricante pasaba a tener que producir (o comprar) los módulos externos de memoria caché.

La casa AMD no fabricaba memorias, sólo procesadores, y por eso se veía obligada a comprar los módulos de memoria caché a otros fabricantes. Haciendo esto, quedó limitada a utilizar los productos que éstos tenían para venderle. El problema era que ninguna compañía tenía, en la época, la tecnología adecuada para producir módulos de memoria caché capaces de trabajar a más de 350 MHz a un precio competitivo.

Si la caché *L2* utilizada en estas versiones antiguas del *Athlon* trabajaba, a lo sumo, a 350 MHz y la casa AMD lanzó versiones del *Athlon K75* de hasta 1 GHz, naturalmente la caché *L2* trabajaba a una fracción de la frecuencia del procesador, el mismo caso que en el *Pentium II*. Las versiones de hasta 700 MHz del *Athlon* traían una memoria caché *L2* que trabajaba a la mitad de la frecuencia del procesador. Las versiones de 750, 800 y 850 MHz traían la caché *L2* operando a sólo 2/5 de la frecuencia del procesador, mientras que en las versiones de 900, 950 y 1 GHz la caché *L2* operaba a sólo 1/3 de la frecuencia del procesador.

3.4.8 Athlon Thunderbird

La versión original del *Athlon* fue lanzada poco después del *Pentium III core Katmai*. Como los dos procesadores utilizaban una memoria caché *L2* que trabajaba a la mitad de la frecuencia del procesador, la pugna se mantuvo bastante equilibrada, con el *Pentium III* ganando en algunas aplicaciones y el *Athlon* en otras. A pesar de que en el cómputo general el *Athlon* ganaba por un margen estrecho, el puesto de procesador más rápido acababa siendo conquistado por el procesador con una mayor frecuencia de trabajo.

Pero el *Athlon* comenzó a quedarse atrás después de que la casa Intel lanzase las nuevas versiones del *Pentium III*, basadas en el *core Coppermine*, debido a su memoria caché *L2* más lenta. Mientras que en un *Pentium III* de 900 MHz la caché *L2* operaba a la misma frecuencia del procesador, en un *Athlon* antiguo también de 900 MHz, la memoria caché *L2* operaba a sólo 300 MHz, 1/3 de la frecuencia del procesador. Pero esto cambió con el lanzamiento del *Athlon Thunderbird*.

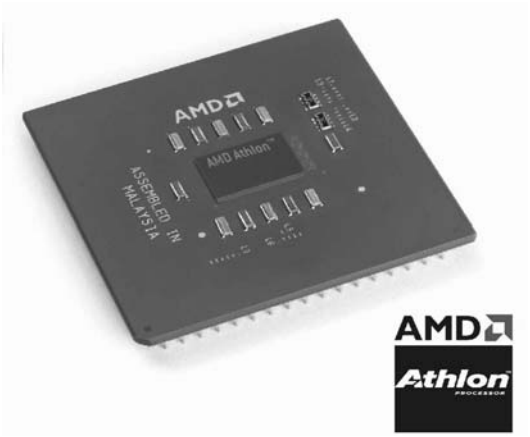
Es muy difícil encontrar un *Athlon* antiguo hoy día, pues estos procesadores dejaron de fabricarse desde hace mucho tiempo y no se llegaron a vender en un número muy grande. El *Athlon Thunderbird* ya tiene su sucesor, el *Athlon Palomino*.

Todo lo dicho hasta ahora sobre la arquitectura del *Athlon* continuaba siendo válido para el *Thunderbird*. No hubo cambios en el proyecto del procesador, sólo en la caché

L2, que pasó a ser de sólo 256 KB (contra los 512 KB) pero que pasó a operar a la misma frecuencia del procesador, como en el *Pentium III Coppermine*.

Fue en la época del lanzamiento del *Thunderbird* en la que se produjo la migración del *slot A* al *soquete A*. La casa AMD optó por hacer ese cambio de forma radical. Simplemente paró la producción de los procesadores antiguos y pasó a vender sólo el *Athlon Thunderbird*. Este movimiento fue coordinado con los fabricantes de placas base, que en la misma fecha retiraron del mercado las antiguas placas *slot A* y pasaron a vender sólo las placas *soquete A*.

AMD llegó a producir una serie del *Thunderbird* en el formato *slot A* que fue vendida a compañías como la casa Dell, que aún tenían placas base *slot A* en stock, pero fue una serie limitada que no llegó a ser vendida directamente al consumidor. No existía ningún adaptador que permitiese instalar un *Thunderbird* en una placa base slot A o un *Athlon* antiguo en una placa base *soquete A*.



Un Athlon para el nuevo soquete A

En términos de rendimiento, el *Thunderbird* superaba a un *Pentium III Coppermine* de la misma freeuencia en la mayoría de las aplicaciones. En algunas pruebas, el *Pentium III* era más rápido pero, en general, el *Thunderbird* era superior a pesar de ser más barato.

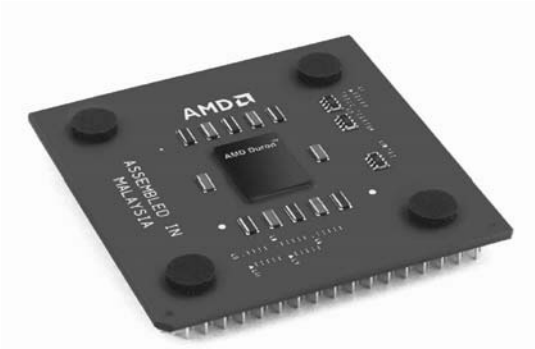
3.4.9 AMD Duron

El *Duron* es el actual procesador de bajo coste de la casa AMD, el sustituto de los procesadores *K6-2* y competidor directo del *Celeron*.

El *Duron* utiliza la misma arquitectura del *Athlon Thunderbird*, aunque sin embargo, viene con menos memoria caché. Mientras que el *Athlon Thunderbird* viene con 256 KB de caché *L2*, el *Duron* viene con sólo 64 KB de caché *L2*, aunque también *full-speed*. Sin embargo, a pesar de la menor cantidad de caché *L2*, el *Duron* lleva una enorme memoria caché *L1* de 128 KB, totalizando 192 KB de caché total, una cantidad

mayor que el *Celeron*, que tiene 32 KB de caché *L1* y 128 KB de caché *L2*, totalizando 160 KB de caché total.

En referencia a la memoria caché, el *Duron* aporta una ventaja más en relación al *Celeron*. En el *Duron*, la cache *L2* es exclusiva, esto significa que los datos almacenados en la caché *L1* y en la caché *L2* son diferentes. Entonces realmente tenemos 192 KB de datos depositados en ambas memorias caché. En el *Celeron*, la caché es inclusiva, esto significa que los 32 KB de la caché *L1* son siempre copias de los datos almacenados en la caché L2. Esto significa que en la práctica, sólo tenemos 128 KB de datos almacenados en ambas memorias caché. Todos los *Duron* utilizan el *soquete A*, pues la casa AMD comenzó su producción después de heber hecho la migración e incluido la memoria caché *L2* en el propio núcleo del procesador.



El procesador Duron de la casa AMD

El *Duron* viene sorprendiendo en términos de rendimiento, ganando por un gran margen a un *Celeron* de la misma frecuencia, presentando un rendimiento muy similar al de un *Athlon* con la arquitectura antigua (con la caché *L2* a la mitad o 2/5 de la frecuencia del procesador). Lo mejor de todo es que a pesar de que el rendimiento ya es más que convincente, el *Duron* es más barato que un *Celeron* de la misma frecuencia, y naturalmente mucho más que un *Pentium III/4* o un *Athlon*.

Claro que comparado con un *Athlon Thunderbird*, el *Duron* pierde en todos los aspectos (aunque por un pequeño margen), pues el *Duron* no es más que un *Athlon Thunderbird* con menos memoria caché. A pesar de esto, los dos procesadores son fabricados en fábricas separadas y existe una pequeña diferencia en la técnica de fabricación de cada uno.

Tradicionalmente, los procesadores utilizaban aluminio en los filamentos que interconectaban los transistores. El aluminio es un material fácil de tratar, que no reacciona con el silicio. Por otro lado, el aluminio no es un conductor tan bueno como el cobre, platino u otros materiales más "nobles". El cobre es un buen sucesor, pues es un material barato y que puede ser aplicado a través de las reacciones químicas usadas para construir un procesador, al contrario del oro, por ejemplo. Pero, el problema de utilizar cobre en lugar del aluminio es que el cobre reacciona con el silicio, haciendo que

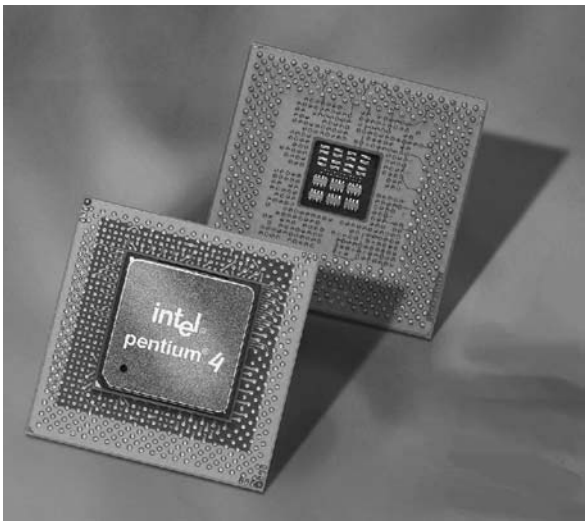
el procesador se vuelva inservible. Entonces, la casa IBM desarrolló una técnica que permitió usar una finísima capa de aluminio entre el silicio y el filamento de cobre, aislando los dos materiales. El uso del cobre permite que el procesador sea capaz de trabajar a frecuencias más altas y mantenerse estable trabajando a temperaturas más elevadas.

La casa AMD usa esta técnica de producción en todos los *Athlons Thunderbird*. Por otro lado, los *Duron* se fabrican en una fábrica separada que utiliza la vieja técnica de producción de los filamentos de aluminio. De momento, el uso del cobre era una exclusividad de la casa AMD, ya que Intel aún no usaba la técnica ni en el *Pentium III* ni en las primeras versiones del *Pentium 4*. Esta es una forma más de abaratar los costes de producción del *Duron*.

El *Duron* que está basado en el *core Thunderbird* es conocido por *Spitfire*. Ya existen versiones del *Duron* basadas en el *core Palomino*, que son conocidas por *Duron Morgan*. El *Duron Morgan* lleva una compatibilidad parcial con las instrucciones *SSE*, y por eso aportó una pequeña ganancia de rendimiento en cada ciclo de reloj en comparación con las versiones anteriores. El *core Morgan* es usado en los *Duron* de 1 GHz en adelante.

3.4.10 Pentium 4

El lanzamiento del *Pentium 4* se produjo a finales del año 2000 en versiones de 1.4 y 1.5 GHz. Actualmente ya existen versiones muy rápidas e inclusive una nueva arquitectura más avanzada que el *core Willamette* usado en estas versiones iniciales. Pero, manteniendo la idea de explicar la evolución de los procesadores, vamos a empezar comentando las características del procesador y de las versiones iniciales para después comentar las versiones más actuales.



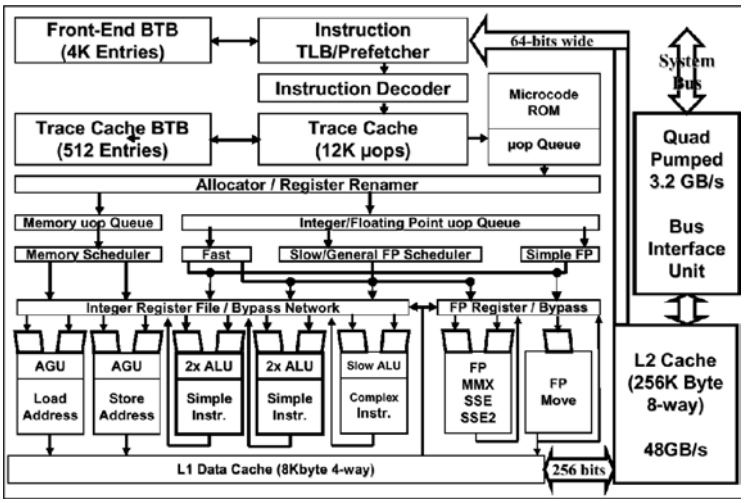
El procesador Pentium 4 de la casa Intel

Un punto interesante sobre las primeras versiones del *Pentium 4* es que el único *chipset* disponible durante casi un año, el i850 de la propia casa Intel, sólo soportaba memorias *Rambus*, lo que obligaba a cualquier interesado en adquirir un *Pentium 4* a adquirir también módulos de memoria *Rambus*. Este tipo de memoria era inicialmente muy caro.

Con el tiempo, el precio de las memorias *Rambus* fue bajando, pero este tipo de memoria aún es más caro que las memorias *SDRAM* o *DDR*. Por suerte, actualmente ya existen placas base para *Pentium 4* con soporte para las memorias *DDR* y para las memorias *SDRAM* comunes.

Arquitectura

El primer punto a tener en cuenta sobre el *Pentium 4* es que el aumento de la frecuencia de operación no significa una ganancia automática de potencia. Un *Pentium 4* de 1.5 GHz no era un 50% más rápido que un *Pentium III* de 1 GHz. Una cosa es el número de ciclos por segundo que es capaz de ejecutar el procesador y otra es lo que éste consigue procesar en cada ciclo.



El diagrama del funcionamiento interno de un Pentium 4

Para entender los puntos fuertes y débiles del *Pentium 4* nada mejor que empezar analizando la arquitectura interna del procesador. La casa Intel bautizó la nueva arquitectura del Pentium 4 como *NetBurst*. El nombre no tiene nada que ver con el rendimiento en redes o en *Internet*, pero intenta ilustrar los nuevos recursos del procesador, así como dar un aire de superioridad.

La arquitectura *NetBurst* está compuesta por 4 componentes principales: *Hyper Pipelined Technology*, *Rapid Execution Engine*, *Execution Trace Cache* y un bus de 400 MHz. Vamos a ver más detalladamente cada una de estas 4 tecnologías.

Hyper Pipelined Technology

Esta es la característica más destacada del *Pentium 4*. El *Pipeline* es un recurso que divide el procesador en varios niveles que trabajan simultáneamente, dividiendo el trabajo de proceso de las instrucciones. A partir del *486*, todos los procesadores utilizan este recurso. El Pentium III poseía 10 niveles, el *Athlon* poseía 11 niveles, mientras que el *Pentium 4* salió con nada menos que 20 niveles, de ahí el nombre *Hyper Pipelined*.

El uso de *Pipeline* permite que el procesador sea capaz de procesar varias instrucciones al mismo tiempo, ya que cada nivel se encarga de una parte del procesamiento. Mientras más niveles, menor será el procesamiento ejecutado en cada uno de ellos. En el caso del *Pentium 4*, cada nivel del *Pipeline* sólo procesa la mitad de lo procesado por un nivel del *Pentium III*, haciendo que teóricamente el resultado final sea el mismo, ya que en compensación tiene el doble de niveles.

El uso de más niveles permite que el procesador trabaje a frecuencias muy altas, ya que cada nivel ejecuta menos procesamientos. El problema, en este caso, es que los procesadores actuales ejecutan varias instrucciones simultáneamente, mientras que los programas son una secuencia de instrucciones. El *Pentium 4* procesa tres instrucciones en cada ciclo. Si las instrucciones siguientes no dependen del resultado de la primera, como una secuencia de sumas de varios números por ejemplo, el procesador no tendrá ningún problema para resolverlas de forma rápida. Sin embargo, siempre que exista una opción de toma de decisión, donde el procesador primero necesite resolver una instrucción para saber qué camino debe escoger a continuación, entra en escena el recurso de la ejecución especulativa, donde mientras se resuelve la primera instrucción el procesador escoge uno de los caminos posibles para ir "adelantando tiempo" mientras no sabe cual deberá seguir. Si al terminar de resolver la primera instrucción el procesador percibe que escogió el camino correcto, simplemente continua a partir de allí. En caso contrario, si el procesador hubiese errado, entonces tendrá que dejar de lado todo el trabajo hecho y tomar el otro camino, perdiendo tiempo.

El *Pentium 4* pierde en ese punto, pues tarda el doble de ciclos de reloj para procesar la primera instrucción, ya que es procesada en 20 niveles, en contra de los 10 del *Pentium III*. Esto significa que en cada toma de decisión errónea se perderán por lo menos 20 ciclos de procesamiento. Esto significa que, en principio, el *Pentium 4* es más lento que un *Pentium III* de la misma frecuencia, pudiendo, en compensación, trabajar a frecuencias mucho más altas. Todas las demás modificaciones realizadas por la casa Intel, explicadas a continuación, sirvieron como paliativos para intentar disminuir la pérdida de rendimiento contraída con el mayor número de niveles del recurso de *Pipeline*.

Además de la pérdida de rendimiento, otro efecto colateral del uso de más niveles de *Pipeline* es el hecho de hacer que el procesador sea mayor y más complejo y, fatalmente, más caro de producir. El Pentium 4 de 0.18 micrones mide 217 milímetros cuadrados, casi el doble que el *Athlon Thunderbird* que mide 120 milímetros cuadrados

Execution Trace Cache

El uso de la caché *L1* en el *Pentium 4* fue innovador. El *Pentium III*, por ejemplo, tenía 32 KB de caché *L1* dividida en 2 bloques de 16 KB cada uno, uno para instrucciones y el otro para datos. El *Athlon* tenía 128 KB de caché *L1*, también dividida en dos bloques. El *Pentium 4*, por su parte, sólo tiene 8 KB de caché para datos. Sin embargo, trajo dos innovaciones que compensaban esta aparente deficiencia. La primera es que gracias al reducido tamaño, la pequeña caché de datos tiene un tiempo de latencia menor, o sea, que es más rápido que la caché *L1* que encontrábamos en un *Pentium III* y en un *Athlon*. Desde el punto de vista de los desarrolladores de Intel, esta fue la mejor relación en términos de rendimiento.

A pesar de que la caché L1 siempre trabaja a la misma frecuencia del procesador, en el *Athlon* la caché *L1* tenía un tiempo de latencia de 3 ciclos, en el *Pentium III* la caché *L1* tenía un tiempo de latencia de 2 ciclos, mientras que en el *Pentium 4* la caché *L1*, a pesar de ser muy pequeña, sólo tiene un tiempo de latencia de 1 ciclo. Es casi tan rápida como los registros del procesador. Podemos notar que la velocidad de la memoria caché es inversamente proporcional al tamaño. Por eso, es por lo que la enorme caché *L1* del *Athlon* no representaba una ganancia de rendimiento muy grande en comparación con la pequeña caché *L1* del *Pentium III*. El procesador necesita visitar menos veces la memoria *RAM* ya que la caché es capaz de almacenar más datos pero, por otro lado, pierde más tiempo en cada acceso a la caché.

Por otro lado, la caché de instrucciones fue sustituida por el *Execution Trace Cache*, que en vez de almacenar instrucciones, almacena directamente las instrucciones ya decodificadas y listas para que puedan ser procesadas. Esto garantiza que la caché sólo tenga un ciclo de latencia, o sea, que el procesador no pierde tiempo alguno al utilizar los datos almacenados en la *Trace Cache*, al contrario de lo que acontecía en el *Pentium III*, donde se perdían por lo menos dos ciclos en cada lectura, el tiempo necesario para decodificar la instrucción.

Bus de 400 MHz

Para intentar competir con el bus *EV6* del *Athlon*, que operaba de 100 a 133 MHz con dos transferencias por ciclo, lo que en la práctica resultaba en unas frecuencias de 200 y 266 MHz respectivamente, el *Pentium 4* contó con un bus que trabajaba a 100 MHz pero con 4 transferencias por ciclo, lo que en la práctica equivale a 400 MHz.

El bus controla la velocidad de comunicación entre el procesador y el *chipset*. Un bus más rápido no significa una ganancia de rendimiento, aunque sin embargo, sí que un bus insuficiente causaría una pérdida de rendimiento, haciendo que el procesador no consiguiese comunicarse con los demás componentes a la velocidad adecuada.

Como el *Pentium 4* tenía una caché L1 muy pequeña, el procesador acababa siendo muy dependiente de la velocidad de acceso a la memoria *RAM*, pues los accesos eran muy frecuentes. Esta característica sirvió de justificación para la decisión de la casa

Intel de utilizar memorias *Rambus* en las primeras versiones del *Pentium 4*. Este tipo de memoria es capaz de operar a frecuencias muy altas. Los primeros modelos del *Pentium 4* utilizaban memorias *Rambus PC-800* que operaban a 200 MHz y realizaban 4 transferencias por ciclo, a buen seguro más rápidas que las memorias *PC-100* o *PC-133* que equipaban a los *Athlon* y *Pentium III* de la época.

Aún hasta hace poco, las placas para *Pentium 4* con soporte para memorias *SDRAM* comunes presentaban un rendimiento bastante inferior a causa del acceso más lento a la memoria *RAM*. Sólo las placas con soporte para las memorias *DDR* eran capaces de presentar batalla a las placas con memorias *Rambus*.

Rapid Execution Engine

Todo procesador actual está dividido en dos componentes básicos, las unidades de ejecución de enteros y las unidades de punto flotante. La parte que procesa las instrucciones que envuelven a los números enteros es la responsable de la mayor parte de las instrucciones y del rendimiento del procesador en las aplicaciones comunes, mientras que las unidades de punto flotante son las responsables del procesamiento de las instrucciones que incorporan valores complejos, usados por los juegos y las aplicaciones gráficas y de renderización.

La *Rapid Execution Engine* del *Pentium 4* consiste en un refuerzo para las unidades de ejecución de enteros del procesador. El *Pentium 4* posee un total de 5 unidades de procesamiento de enteros, dos *ALU*, que procesan las instrucciones más simples, dos *GLU*, encargadas de leer y escribir los datos y una tercera *ALU*, encargada de decodificar y procesar las instrucciones complejas, que aunque en menor cantidad, son las que toman más tiempo de procesador. Este conjunto de 5 unidades de ejecución de enteros es semejante al del *Pentium III*, sin embargo, como diferencia, en el *Pentium 4* tanto las dos *ALU* encargadas de las instrucciones simples como las dos *GLU* encargadas de las lecturas y escrituras son dos veces más potentes.

Según la casa Intel, las cuatro unidades operan a una frecuencia dos veces superior a la del procesador, lo que sugiere que en un *Pentium 4* de 1.5 GHz, operan a 3 GHz. Sin embargo, en realidad, cada unidad pasó a ser compuesta por dos unidades que trabajan en paralelo. Con esto, las unidades son capaces de procesar dos instrucciones por ciclo, pero la frecuencia de operación continúa siendo la misma. Este es un recurso que debería parecer maravilloso, pero existe un pequeño detalle que elimina buena parte de la ganancia que sería de esperar en este nuevo esquema. A pesar de que las dos *ALU* de instrucciones simples sean más rápidas, intentando compensar la pérdida de rendimiento acarreada por los 20 niveles del *Pipeline* del *Pentium 4*, la *ALU* de instrucciones complejas no tuvo la misma evolución. Esto significa que al pasar a usar 20 niveles de Pipeline, esta tercera *ALU* se volvió más lenta que la misma en el *Pentium III*. Entonces nos encontramos con un escenario donde las instrucciones simples son procesadas de forma rápida pero las instrucciones complejas quedan encalladas en la tercera *ALU*, causando una gran pérdida de rendimiento.

En el coprocesador aritmético el escenario es aún más complicado, pues a pesar de que las unidades de ejecución han perdido rendimiento debido al *Pipeline* de 20 niveles no hubo ningún avance para equilibrar la balanza, como sí ocurrió en las unidades de enteros. Por el contrario, el coprocesador aritmético encogió, pues fueron eliminadas dos de las unidades de ejecución, una de las que procesaba las instrucciones *MMX* y una de las que procesaba las instrucciones *SSE*.

En vez de evolucionar como era de esperar, el coprocesador aritmético del *Pentium 4* se hizo más frágil que el del *Pentium III*, conllevando a un escenario como mínimo curioso. Mientras que en la época del *Pentium II* y del *K6*, la casa AMD competía con un procesador que a pesar de poseer un buen rendimiento en las aplicaciones de oficina era literalmente inferior en los juegos y aplicaciones gráficas, ahora tenemos un escenario semejante con el *Pentium 4* y el *Athlon*, aunque con los polos invertidos. La casa Intel atacó con un procesador que es potente en las operaciones de enteros, pero débil en las operaciones en punto flotante.

SSE2

Las *Double Precision Streaming SIMD Extensions* del *Pentium 4* son 144 nuevas instrucciones de punto flotante de doble precisión. Tienen la misma función que la que tenían las instrucciones *SSE* del *Pentium III* y las *3D-Now!* del *Athlon*: mejorar el rendimiento del procesador en las aplicaciones de punto flotante. La diferencia es que las instrucciones del *Pentium 4* son más poderosas que las de los conjuntos anteriores y pueden ser utilizadas en más situaciones, lo que puede salvar al *Pentium 4* en el caso de que las aplicaciones estén bien optimizados para las nuevas instrucciones.

La casa AMD anunció que su próxima generación de procesadores de 64 bits soportarán el conjunto de instrucciones *SSE2*. En el momento de su aparición, el *Pentium 4* tenía su exclusividad. Es necesario recordar que el *Pentium 4* mantiene la compatibilidad con las instrucciones *SSE* del *Pentium III*.

Acceso a la memoria

A pesar de tener la desventaja de utilizar las caras memorias *Rambus*, el *Pentium 4* estaba (o aún está, ya que el uso de memoria *Rambus* aún es una opción) indiscutiblemente bien posicionado desde el punto de vista del rendimiento en el acceso a la memoria. Visitando simultáneamente dos módulos *RIMM* tenemos un bus de datos de 3.2 GB/s usando módulos *PC-800*, lo que se corresponde con tres veces el acceso permitido por los módulos de memoria *PC-133* comunes hasta la época.

Por un lado, esto ayuda bastante al procesador en las aplicaciones dependientes de la velocidad de acceso a la memoria, como los programas de edición y compresión de vídeo y algunos juegos. Por otro lado, es necesario usar los módulos en pares. Si quisiéramos 512 MB de memoria, tendríamos que usar obligatoriamente dos módulos de 256 MB de la misma marca y velocidad.

3.5 Los nuevos procesadores

A partir de este punto veremos los procesadores más actuales, lanzados a partir de finales del año 2002.

3.5.1 Las nuevas evoluciones

En un corto espacio de tiempo todos los procesadores que acabamos de ver sacaron nuevas versiones. En algunos casos, como por ejemplo el *Duron*, las nuevas versiones fueron totalmente compatibles con las placas antiguas, mientras que en otros casos, como en el caso del *Celeron Tualatin*, era necesario cambiar la placa base en la mayoría de los casos.

Todo este cambio puede ser un poco confuso si no se estuvo en contacto con las noticias de la época del cambio, y por eso dedicaremos las siguientes páginas a explicar lo que realmente cambió.

3.5.2 Intel vs AMD

La casa Intel ha adoptado una política bastante agresiva en lo referente a los precios pero la casa AMD aún continúa vendiendo sus procesadores más baratos. En realidad, las dos vienen manteniendo una guerra de precios desde hace algún tiempo. Siempre que Intel ha anunciado una disminución en los precios, AMD también ha bajado los precios de sus procesadores pocos días después. Es casi una cuestión de supervivencia para AMD conseguir vender sus procesadores más barato que Intel, ya que ésta tiene su marca mejor establecida.

Hasta hace poco, Intel vendía los procesadores *Celeron*, *Pentium III* y P*entium 4*, mientras que la AMD vendía el *Duron*, el *Athlon* (en sus versiones *Thunderbird* y XP) y el *Athlon MP*, que es el primer procesador *Athlon* con soporte para el multiprocesamiento. El *Duron* que ya hemos visto a fondo alcanzó la marca de los 1.1 GHz, un frecuencia muy elevada por el bajo precio que tenía. Las nuevas versiones del *Duron*, tanto en la versión de 1.1 GHz como los de 1.0 GHz y 950 MHz de producción más reciente ya usaban el *core Morgan*, que compartía la misma arquitectura del *Palomino*, sólo con menos memoria caché. La principal novedad era el soporte para las instrucciones *SSE* del *Pentium III*, lo que garantizaba un rendimiento de un 5 a un 10% superior al de las versiones anteriores. El *Athlon* también continuó evolucionando. Los modelos anteriores, basados en el *core Thunderbird* llegaron a la respetable marca de 1.4 GHz, mientras que el *Athlon XP*, basado en el core *Palomino*, ya llegó a los 1.53 GHz en las primeras versiones.

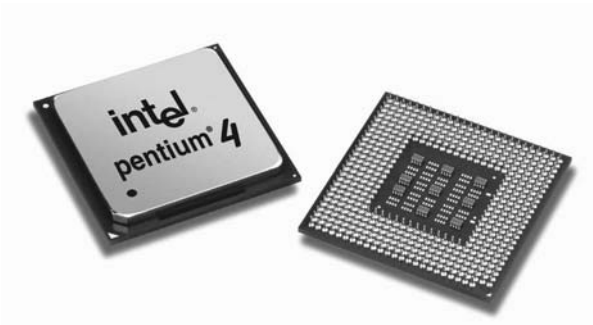
Por el lado de la casa Intel nos encontramos con una situación un poco más compleja. Tanto el *Celeron* como el *Pentium III* aún se producían con una arquitectura de 0.18 micrones. Ambos debían ser apartados en breve, cuando tenían que ser sustituidos por el *Tualatin*, que era un *Pentium III* de 0.13 micrones, capaz de trabajar a frecuencias más altas. Aparecieron versiones del *Pentium III Tualatin* (*Pentium III-S*) de 1.13

y 1.2 GHz con 512 KB de memoria caché, pero eran muy caros. La casa Intel pretendía continuar bajando los precios de los procesadores *Pentium 4*, hasta el punto de que costasen más baratos que los *Pentium III* y mantener el *Celeron* como una opción de bajo coste.

Es ahí donde entró el nuevo *Celeron*, basado en el *core Tualatin*, que tenía 256 KB de memoria caché y era producido, inicialmente, en versiones de 1.0, 1.1 y 1.2 GHz. El *Celeron Tualatin* era prácticamente igual que un *Pentium III*, con la única diferencia que el *Celeron Tualatin* utilizaba un bus de 100 MHz, en vez de los 133 MHz de las versiones más rápidas del *Pentium III*.

El *Pentium 4* es otro que se reencarnó en una nueva arquitectura. Los procesadores vendidos hasta finales del año 2001 usaban la arquitectura *core Willamette* de 0.18 micrones y 256 KB de memoria caché. A partir del año 2002 se empezaron a vender los procesadores con *core Northwood*, que ya utilizaban una arquitectura de 0.13 micrones y traían 512 KB de memoria caché *L2*.

Es preciso recordar, que a pesar de que llevasen más memoria caché, los nuevos *Pentium 4* no eran necesariamente más rápidos que los anteriores, pues pasaron a utilizar placas base basadas en el *chipset* i845 (u otros *chipsets* similares), con soporte para las memorias *SDRAM* comunes o memorias *DDR*. A pesar de que el cambio garantizaba una caída en los precios de las placas base y, principalmente, en el precio total de los PCs basados en el *Pentium 4*, ya que no era necesario usar memorias *Rambus*, el rendimiento usando memorias *SDRAM* comunes era inferior, a pesar de la mayor cantidad de memoria caché.



El Pentium 4 para el soquete 478

Otra novedad es que, finalmente, se empezaron a vender las placas base y los procesadores *Pentium 4* con el nuevo encaje *soquete 478*. A pesar de tener más contactos, este encaje era menor que el *soquete A* del *Athlon* y que el *soquete 370* del *Pentium III*. Según Intel, el soquete 478 era el soquete definitivo para el *Pentium 4*, que acomodaría a todos los procesadores que serían lanzados al mercado en adelante. Esto daba un buen margen para ampliaciones, pues la casa Intel pretendió llegar a lanzar procesadores *Pentium 4* de hasta 4.0 GHz basados en la arquitectura de 0.13 micrones.

Vamos a ver ahora, más detalladamente lo que cambió en las nuevas arquitecturas, y como diferenciar los nuevos procesadores de los más antiguos.

3.5.3 Spitfire vs Morgan

El *core Spitfire* fue la primera versión del *Duron* de la casa AMD. Con sus 128 KB de caché *L1* y 64 KB de caché *L2* (contra los 256 KB del *Athlon),* el *Duron* siempre fue capaz de superar fácilmente a un *Celeron* de la misma frecuencia, a pesar de quedarse siempre un poco por detrás del *Athlon* y del *Pentium III*. La ventaja era que el *Duron* siempre fue mucho más barato, ideal para los equipos de bajo coste. Con el *core Spitfire*, el *Duron* fue lanzado con una frecuencias de hasta 950 MHz pero, al comenzar a aproximarse a los límites de la arquitectura, la AMD mejoró su proyecto, llegando al *core Morgan*, usado en los *Duron* de 950 MHz y en todos los de 1.0 GHz y superiores.

La principal novedad de la nueva arquitectura era la compatibilidad con las instrucciones *SSE* del *Pentium III* (que la AMD llamó *3D-Now! Professional*), lo que trajo pequeñas ganancias de rendimiento en la mayoría de las aplicaciones. El *core Morgan* también fue proyectado para consumir menos electricidad y disipar menos calor, pero para compensar, los nuevos *Duron* usaron una tensión más elevada (1.75 voltios contra los 1.6 voltios de los anteriores) y operaron a frecuencias más altas. Sumando los tres factores, los nuevos procesadores consumen más energía que los antiguos y la única ventaja acaba siendo el mejor rendimiento.

Gracias a las instrucciones *SSE* y a algunas mejoras más en el proyecto del procesador consiguió ser de un 5 a un 7% más rápido (comparando dos procesadores de la misma frecuencia) que el *core Spitfire* usado en los modelos anteriores. Una duda frecuente sobre el *Duron Morgan* es sobre la compatibilidad con las placas base del momento. A pesar de que la casa AMD no alteró la señalización del procesador, era necesario actualizar el *BIOS* de la placa base para que ésta fuese capaz de activar el soporte a las instrucciones SSE e identificar correctamente el procesador durante la inicialización del sistema. De lo contrario, el procesador funcionaba pero no podíamos aprovechar la ganancia de rendimiento proporcionada por las nuevas instrucciones.

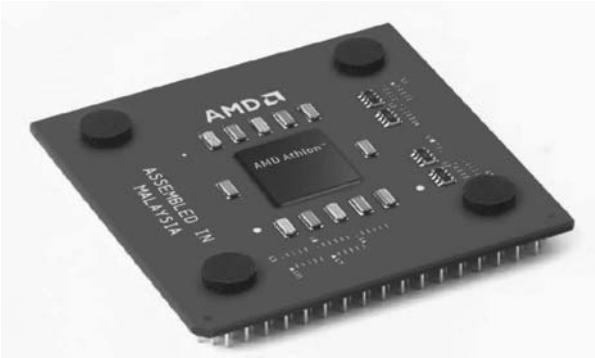
Teóricamente, el *Duron Morgan* funcionaba en cualquier placa base soquete A, pues usaba un bus de 100 MHz y una tensión de 1.75 voltios. Pero, en la práctica, algunos usuarios se quejaron que algunas placas base antiguas no arrancaban con los nuevos procesadores. Era una situación semejante a la que se produjo con el lanzamiento del *Celeron* con caché algunos años atrás, cuando algunas placas base no arrancaban con el nuevo procesador. Era necesario conseguir un *Pentium II* o *Celeron* antiguo para conectar el equipo y actualizar el *BIOS* y utilizar el nuevo *Celeron* después. Esto sólo era aplicable a las placas base antiguas. Los modelos fabricados algunos meses después ya venían con el *BIOS* actualizado.

El hecho de que los *Duron* con *core Morga*n fuesen compatibles con las placas base antiguas, reforzaba la idea que el *Athlon XP*, basado en el *core Palomino*, también lo

era. Claro que en el caso del *Athlon XP* era necesario que la placa base soportase un bus de 133 MHz.

3.5.4 Thunderbird vs Palomino

No fue sólo el *Duron* el que evolucionó, el *Athlon* también entró en una nueva fase con el *core Palomino*. En realidad, el *Morgan* era un subproduto del trabajo hecho en el *Palomino*, y por eso las arquitecturas son tan semejantes. El *Athlon Palomino* también consumía un poco menos de energía y tenía un rendimiento un poco superior gracias a la compatibilidad con las instrucciones *SSE*. Pero, la principal novedad era el soporte al multiprocesamiento. No fue sin ton ni son que los procesadores con *core Palomino* fueron vendidos como *Athlon MP* (*MP* de *Multi Processing*).



El Athlon MP con core Palomino

Tanto el *core Morgan* como el *core Palomino* consumían cerca de un 7% de energía menos que los proyectos anteriores. El *Athlon MP* usaba la misma tensión que el *Athlon Thunderbird*, y por eso consume menos. Por su parte, el *Morgan* usaba una tensión un 10% más alta. Eso anulaba la economía de usar el nuevo *core* e incluso conllevaba un pequeño aumento. Usar una tensión más alta es un recurso usado por los fabricantes para que los procesadores sean capaces de alcanzar frecuencias más altas.

Otro punto destacable es que tanto el *Palomino* como el *Morgan* trajeron finalmente una vieja reivindicación de los interesados en los procesadores AMD, un sistema de protección térmica. Fue incluso común oír hablar de procesadores *Athlon* y *Duron* quemados por sobrecalentamiento, ya sea que se le olvidase al dueño instalar o conectar el ventilador o por utilizar un ventilador inadecuado.

El nuevo sistema implantado en los procesadores *Athlon* y *Duron* con *core Palomino* consiguió que el procesador se parase al alcanzar una cierta temperatura, preservando el resto del equipo. Sin embargo, este sistema realizaba las lecturas con intervalos de 1 segundo. A final de cuentas, la protección sólo funcionaba si instalábamos un ventilador con una capacidad insuficiente para el procesador, o si el ventilador dejaba de funcionar. Si el procesador estaba mal conectado al ventilador, o no se hubiese instalado correctamente, el procesador aún podía quemarse. Este es un punto impor-

tante: jamás utilicemos un *Athlon* o *Duron* sin un ventilador aunque sólo sea para entrar en el *BIOS*, ya que el procesador se quemará antes de que el diodo sea capaz de desconectar el procesador.

Detalles sobre el core Palomino

Cuando el *Pentium 4* fue sacado al mercado hubo casi un consenso de que a pesar de que el *Pentium 4* existía en versiones que trabajaban a frecuencias muy altas, el rendimiento del procesador dejaba un poco que desear. Sin embargo, conforme se fueron incorporando optimizaciones para el *Pentium 4*, los resultados obtenidos comenzaron a mejorar un poco.

El *Athlon Thunderbird* por su parte, así como el *Duron*, presentaba un rendimiento muy bueno, además de tener un precio atractivo. Sin embargo, una desventaja de esta plataforma era que ambos consumían mucha más electricidad que los procesadores *Pentium III* y *Celeron* de la misma frecuencia. Esto conllevó como efecto colateral a un mayor calentamiento, lo que obligaba al usuario a usar un ventilador más poderoso.

La principal ventaja del *Palomino*, muy enfatizada por la casa AMD desde que el proyecto comenzó a divulgarse, son una serie de mejoras que redujeron el consumo eléctrico y disipación térmica del procesador en comparación con un *Thunderbird* de la misma frecuencia. El rendimiento también mejoró, nada escandaloso, sólo de un 3 a un 16% dependiendo de la aplicación, según informaciones de la propia AMD.

SSE

El *SSE* ya no era exclusividad del *Pentium III* y *Pentium 4*. La AMD añadió unas nuevas 52 instrucciones que permitieron una compatibilidad parcial con las instrucciones *SSE* del Pentium III, que eran 70 en el total. Se sabe que varias de estas instrucciones eran diferentes a las que se encontraban en el *SSE* del *Pentium III*, y por eso podía ser que a pesar de ser en menor número, las instrucciones del *Athlon* tuviesen una eficiencia muy parecida. De cualquier modo, incluso una implementación parcial ya debía mejorar un poco el rendimiento del *Athlon* en las aplicaciones optimizadas para el juego de instrucciones *SSE*, disminuyendo la ventaja del Pentium III en las pocas aplicaciones donde aún era mejor.

Faltaba aún la compatibilidad con el juego de instrucciones *SSE2* del *Pentium 4*, lo que sólo debía ser implementado en el AMD *ClawHammer* (o *K8*), el primer procesador de 64 bits destinado al mercado doméstico y lanzado a partir del 2004. Con este refuerzo, el juego de instrucciones 3D-Now! pasó a llamarse *3D-Now! Professional*.

Data Prefetch y aumento de las direcciones en el TLB

El *Data Prefetch* es un recurso que permitía cargar los datos que iba a necesitar el procesador para concluir sus próximas instrucciones. Aún operando a la misma frecuencia del procesador siempre existía alguna pérdida de tiempo al acceder a la memoria

caché L1 y el *Data Prefetch* consiguió disminuir el número de accesos necesarios, conllevando una pequeña ganancia de rendimiento.

Otro pequeño avance fue un aumento en las direcciones disponibles en el *TLB* (*Translation Lookside Buffer*). El TLB es una pequeña área de memoria que almacena las direcciones necesarias para que el procesador pueda recoger los datos de la memoria *RAM*, si los mismos no los ha encontrado en las memorias caché. El procesador no tiene como almacenar directamente todas las direcciones de la memoria *RAM*, pues son muchas. En vez de eso, sólo se almacenan las más usadas.

El problema surge cuando el procesador necesita acceder a un dato cualquiera en la memoria *RAM*, cuya dirección no está cargada en el *TLB*. En este caso se pierde una verdadera eternidad, pues primero será necesario cargar la dirección, para después realizar el acceso a la memoria. El aumento en el número de direcciones disponibles en el *TLB* disminuye la posibilidad de estas pérdidas, permitiendo una pequeña ganancia más de rendimiento.

PowerNow!

El PowerNow! fue originalmente implantado en los procesadores *K6-2* y *K6-3* destinados a los ordenadores portátiles. Se trata de un recurso de ahorro de energía que permite alternar la frecuencia de operación del procesador según su nivel de utilización. Los cambios se realizan en cuestión de nanosegundos, siendo completamente transparentes. Este es un recurso bastante útil para economizar energía, pero que sólo fue usado en los *Athlon Palomino* destinados a portátiles, que son vendidos como *Athlon 4*.

El *PowerNow!* ofrecería una protección térmica mucho mejor que la del diodo térmico que equipa los *Athlon Palomino* para los equipos de sobremesa, evitando que los procesadores puedan quemarse al ser conectados sin un ventilador.

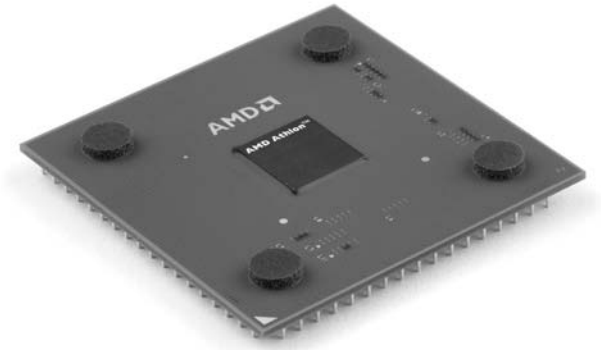
3.5.5 Athlon XP

La casa AMD tiene un problema. A pesar de que el *Athlon* tiene un rendimiento superior al de un *Pentium 4* de la misma frecuencia, la mayoría de los usuarios simplemente no saben nada de eso. Con eso, la AMD se ve obligada a vender, por ejemplo, los *Athlon* de 1.33 y 1.4 GHz a precios iguales o más bajos que el *Pentium 4* de 1.4 GHz, aunque los *Athlon* tienen un rendimiento próximo al de un *Pentium 4* de 2.0 Ghz, que puede costar 3 o 4 veces más. A pesar de las críticas, es innegable que la casa Intel acertó de lleno desde el punto de vista comercial, construyendo un procesador capaz de alcanzar frecuencias muy elevadas, a pesar de perder en una comparación frecuencia por frecuencia.

Aunque la casa AMD tradicionalmente trabaja con márgenes más pequeños que la casa Intel, es difícil creer que serán capaces de continuar vendiendo tan baratos sus procesadores tope de línea. Ya que no es posible producir *Athlon* que trabajen a fre-

cuencias tan altas como el *Pentium 4*, AMD decidió poner en práctica una estrategia diferente, intentando venderlos en relación a su rendimiento. Decidieron resucitar el viejo índice *Pr*, ya usado en los antiguos *5x86*, *K5* y *Cyrix 6x86*. El formato, que la casa AMD venía intentando establecer con la ayuda de otros fabricantes, fue bautizado como *True Performance Initiative*, e intentaba crear un conjunto de pruebas y *benchmarks* de varios fabricantes y tener en consideración el rendimiento del procesador en varias aplicaciones, y que pudiese servir de medida fiable del rendimiento total de los procesadores.

De momento, el patrón aún no está definido, por eso la casa AMD definió los índices de rendimiento basándose en una serie de *benchmarks* muy populares, manteniendo aún una cierta holgura en relación al *Pentium 4*. La intención era que el *Athlon* pudiera vencer con holgura las versiones equivalentes del *Pentium 4*. Para completar el proceso, el *Athlon Palomino* cambió de nombre y fue lanzado como *Athlon XP* en vez de *Athlon 4*, que era el nombre ya utilizado en las versiones para ordenadores portátiles.



El Athlon XP de la casa AMD

El *Athlon XP* fue lanzado al mercado en versiones de 1.33, 1.4, 1.46 y 1.5 GHz, vendidas en base a su rendimiento en relación al *Pentium 4*. Por ejemplo, la versión de 1.33 GHz era vendida como modelo 1500+, indicando que tenía un rendimiento igual o superior al de un *Pentium 4* de 1.5 GHz. La versión de 1.4 GHz era vendida como modelo 1600+, la de 1.46 GHz como 1700+, la de 1.5 GHz como 1800+. Inmediatamente después fue lanzada la versión de 1.6 GHz como el modelo 1900+.

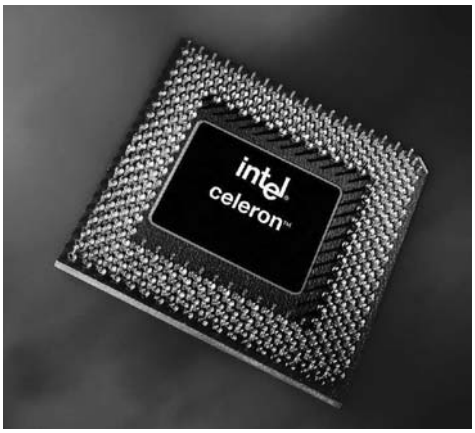
Desde el punto de vista del rendimiento, la nueva numeración era más que correcta, pues salvo algunas excepciones, un *Athlon XP* de 1.4 GHz batía con holgura a un *Pentium 4* de 1.6 GHz. Además, en muchas de las aplicaciones de prueba llegaba a batir incluso a un *Pentium 4* de 2.0 GHz.

Felizmente y para que no se creasen confusiones con esta nueva nomenclatura, las placas base con el *BIOS* actualizado eran capaces de reconocer correctamente a los nuevos procesadores y mostraban tanto el índice de rendimiento como la frecuencia real de trabajo.

Coppermine vs Tualatin

Desde el punto de vista de la tecnología de producción, la casa Intel estaba un paso por delante de la casa AMD. Tanto el *Tualatin* como el *Northwood* ya utilizaban una tecnología de 0.13 micrones, mientras que el *Morgan* y *Palomino* de la casa AMD aún utilizaban una arquitectura de 0.18 micrones. Esto representaba una diferencia muy grande tanto en términos de consumo eléctrico, como en términos de las frecuencias de trabajo que eran capaces de alcanzar los procesadores. La casa AMD lanzó sus procesadores de 0.13 micrones entre finales de 2002 y principios de 2003.

Originalmente, el *core Tualatin* debía ser usado para producir los procesadores *Pentium III* de 1.13 GHz en adelante. De hecho, llegaron a ser lanzados *Pentium III* de 0.13 micrones operando a 1.13 y 1.20 GHz. Pero, la casa Intel no estaba muy interesada en mantener estos procesadores, temiendo que pudieran entorpecer las ventas del *Pentium 4*. Al final acabaron usando el *core Tualatin* para desarrollar versiones más rápidas del *Celeron*, con los mismos 256 KB de caché que en el *Pentium III*, sólo con la diferencia de usar un bus de 100 MHz, en vez del de 133 MHz del *Pentium III*. En compensación, el nuevo *Celeron* incluía un recurso nuevo, llamado *Data Prefetch Logic*, capaz de mejorar un 5% el rendimiento del procesador.



El Celeron Tualatin

Gracias a las frecuencias de operación más altas y a la mayor caché, los nuevos *Celeron* fueron más rápidos que los antiguos. Finalmente, el nuevo *Celeron* disponía de las condiciones necesarias para poder competir con el *Duron*, a pesar de ser todavía más caro. Pero, en compensación, algunos de los cambios en la señalización del procesador hicieron a los *Tualatin* incompatibles con las placas base antiguas. Los nuevos procesadores *Tualatin* utilizaban un nuevo bus (el *AGTL*) en contraste con el bus *GTL+* utilizado por los procesadores *Pentium II*, Pentium III y *Celeron* antiguos.

Otro punto incómodo era que las placas base destinadas al *Tualatin* ya no eran compatibles con los antiguos *Celeron PPGA* (de los 366 a los 533 MHz), que ya utilizaban el *soquete 370* pero aún utilizaban una arquitectura antigua y una tensión de 2.0 vol-

tios, nuevamente a causa de la diferencia en las señalizaciones. Sin embargo, existía una compatibilidad total con todos los *Pentium III* y *Celeron* con *core Coppermine.*

El mayor problema con el *Celeron Tualatin* era la falta de opciones de actualización y ampliación. Al comprar una placa base *soquete 370* y un *Tualatin*, nuestra única opción de ampliación es adquirir una versión más rápida del *Celeron*. Comprando un *Pentium 4 Northwoo*d y una placa base *soquete 478* tendremos, por lo menos, la posibilidad de usar un *Pentium 4* de 3.0 GHz o más, aprovechando la misma placa base.

3.5.6 Celeron Tualatin

Debemos tener en cuenta que la casa Intel no guardaba el secreto de que su prioridad era popularizar la arquitectura del *Pentium 4*. Existieron incluso especulaciones sobre el futuro lanzamiento de una nueva versión *Celeron* del *Pentium 4*, que no sería nada más que un *Pentium 4* con un nombre distinto.

Intel sólo pretendía vender procesadores *Pentium 4* de 0.13 micrones, con 512 KB de memoria caché y un bus de 133 MHz (con 4 transferencias por ciclo, 533 MHz en la práctica). Los *Pentium 4* de 0.18 micrones continuarían produciéndose ya en el nuevo formato para placas *soquete 478*, pero con las mismas características, incluyendo los 256 KB de caché, un bus de 100 MHz, etc. Para diferenciar claramente los dos modelos, Intel utilizó el nombre del bueno y viejo *Celeron* en las versiones producidas con la antigua arquitectura.

Se produjo entonces una nueva ramificación en la línea de los procesadores. El *Celeron* era lanzado en versiones de hasta 1.8 GHz. El *Pentium 4*, por su parte, continuaría alcanzando frecuencias cada vez más altas, gracias a la nueva arquitectura.

Willamette vs Northwood

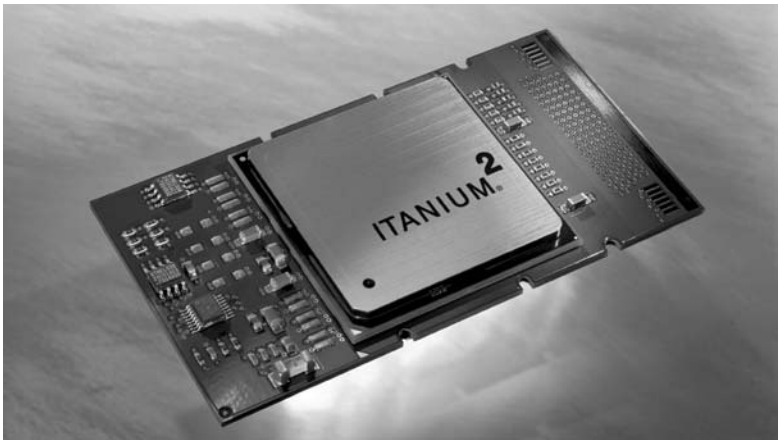
El *Willamette* no era más que el *Pentium 4* que hemos visto hasta ahora, con unas frecuencias de hasta 2.0 GHz y producido en una arquitectura de 0.18 micrones. El *core Northwood* fue usado en los nuevos modelos, que utilizaban el encaje *soquete 478* y traían 512 KB de memoria caché.

3.5.7 Intel Itanium

Durante años se oyó hablar mucho acerca del Intel *Merced*. Incluso existieron muchas previsiones de lanzamiento, pero siempre fueron más rumores que verdades. Después de cambiar hasta el nombre del proyecto, finalmente el *Itanium* fue lanzado al mercado con cerca de 3 años de retraso en marzo del año 2001. Inicialmente, el *Itanium* no fue destinado al mercado doméstico, pero sí al mercado de servidores, donde competía principalmente con los procesadores *RISC* disponibles.

El *Itanium* representó el primer procesador de 64 bits lanzado por la casa Intel, que utilizaba un nuevo conjunto de instrucciones, el *IA-64*, que era incompatible con el que existía, lo que permitió a la casa Intel eliminar toda la carga del legado que acom-

pañaba a sus procesadores desde el *8088*. Para mantener la compatibilidad del *Itanium* con las aplicaciones de 32 bits, que todavía son la gran mayoría, la casa Intel incluyó un sistema de emulación vía hardware, que permitía que el *Itanium* ejecutase estas aplicaciones, pero con un rendimiento bastante menor. El *Itanium* sólo era una muy buena opción para servidores y otras áreas donde fuese posible ejecutar aplicaciones de 64 bits.



El Itanium 2 de la casa Intel

Actualmente ya existen versiones de 64 bits de *Linux* y de *Windows*, pero no están disponibles muchas aplicaciones. Esto acaba limitando el campo de trabajo del *Itanium* a los ambientes donde es posible recompilar los programas para ejecutarlos en la nueva arquitectura. Este es el caso de muchas aplicaciones científicas, pero no es el caso de la mayoría de los programas comerciales.

La arquitectura

La arquitectura del *Itanium* era muy distinta a la del *Pentium 4*. En vez de ser proyectado con altas frecuencias de trabajo, el *Itanium* siguió otra idea, la de producir un procesador capaz de ejecutar muchas operaciones por ciclo y que fuese capaz de presentar un buen rendimiento incluso trabajando a frecuencias más bajas.

Las primeras versiones del *Itanium*, lanzadas en marzo de 2001, operaban a sólo 733 y 800 MHz. El *Itanium* era capaz de procesar entre 8 y 20 instrucciones por ciclo, dependiendo del tipo de instrucciones utilizadas por la aplicación. En comparación, un *Pentium 4* sólo era capaz de ejecutar de 3 a 8 instrucciones por ciclo, mientras que un *Athlon* ejecutaba entre 6 y 9 instrucciones. Esto muestra que al comparar un *Itanium* de 800 MHz, un *Athlon* de 1.3 GHz y un *Pentium 4* de 1.8 GHz, estamos hablando de procesadores de más o menos el mismo nivel de rendimiento.

Además del gran número de instrucciones por ciclo, el *Itanium* tenía una arquitectura de caché bastante agresiva, con una memoria caché *L1* bastante rápida, pero de sólo 32 KB (dividida en dos bloques, para datos e instrucciones), una caché *L2* también

muy rápida, de sólo 96 KB, pero en compensación tenía una caché *L3* de 2 a 4 MB, dependiendo de la versión del procesador. Esta combinación le permitía disponer de un gran número de datos almacenados en la enorme caché *L3*, al mismo tiempo que era capaz de realizar los accesos a una gran velocidad, gracias a la combinación de la caché *L1* y *L2*.

El *Itanium* utilizó un encaje bastante raro, una combinación de un *soquete* (con los pins en el centro del procesador) y un segundo conector en forma de *slot*. En el momento de instalar el procesador, primero se ensamblaba en el *slot* y después se ensamblaba en el *soquete*. El *Itanium* disipaba bastante calor, por eso se solían utilizar ventiladores realmente grandes.

La memoria caché *L3* operaba a la misma frecuencia del procesador pero estaba formada por dos módulos separados, que se instalaban cerca del núcleo del procesador. Este era el principal motivo para utilizar este encapsulado caro, en vez de diseñar un procesador en formato de *soquete*, como el *Pentium 4*. A continuación tratamos algunos de los recursos del *Itanium*:

Arquitectura EPIC: abreviatura de *Explicitly Parallel Instruction Computing*, consistía en una nueva arquitectura, ni *RISC* ni *CISC*, donde el procesador podía procesar varias instrucciones de forma simultánea.

Los procesadores actuales poseen varias unidades de ejecución que trabajan de forma paralela. Sin embargo, los programas no son optimizados para este recurso y, por el contrario, envían una instrucción cada vez. El decodificador de instrucciones y la unidad de control tienen la tarea de ordenar las instrucciones, de modo que las unidades de ejecución permanezcan ocupadas durante la mayor parte del tiempo.

Un procesador *EPIC* ya trabajaba de forma diferente. En vez de varias unidades de ejecución separadas teníamos una única unidad de ejecución capaz de procesar varias instrucciones de forma simultánea. En este caso, teníamos la posibilidad de obtener un rendimiento bastante superior, pero al mismo tiempo, existía la necesidad de que los programas fuesen optimizados para la nueva arquitectura.

Procesador de 64 bits: el hecho de que el nurvo *Itanium* fuese un procesador de 64 bits conllevó como ventajas la posibilidad de manipular 8 Bytes de datos cada vez, lo que permitió la ejecución de instrucciones más complejas y poderosas en un único ciclo de reloj y la posibilidad de acceder a mucha más memoria.

Los procesadores de 32 bits que tenemos actualmente son capaces de acceder a 2^{32} bits de memoria, lo que corresponde a 4 GBytes. Mucha gente ya utiliza 512 MB o 1 GB de memoria *RAM* y, en los servidores de red, esta cantidad puede pasar fácilmente a los 2 GBytes o más. Como podemos ver, los 4 GBytes de memoria estaban cerca de quedarse pequeños. Al ser un procesador de 64 bits, el *Itanium* era capaz de acceder a 2^{64} bits de memoria, lo que corresponde a cerca de 2 millones de TBytes de memoria *RAM*.

Registros: dentro del procesador, tenemos varias pequeñas área de memoria ultrarápida, que almacenan los datos e instrucciones que se están procesando. Los datos son transferidos de la memoria caché a los registros y procesados a continuación. Mientras más registros, mejor será el rendimiento del procesador. El *Itanium* tenía un total de 384 registros, siendo 128 para un uso general, 128 para los cálculos de punto flotante y 128 más para el recurso de ejecución especulativa.

Distribución de la señal de reloj: internamente, un microprocesador está formado por varios subcomponentes y la señal de reloj, recibida de la placa base, es la encargada de que todos trabajen de forma sincronizada. Este es uno de los motivos por el que un microprocesador debe ser pequeño. Como la señal de reloj es una señal eléctrica, ésta tarda un cierto tiempo para recorrer todo el procesador. En principio, este tiempo puede parecer muy pequeño, pero cuando hablamos en billones de ciclos por segundo, como en los procesadores actuales, cualquier fracción de tiempo marca una diferencia.

El *Itanium* fue el mayor procesador (en términos de espacio físico) producido por la casa Intel. Para minimizar el retraso causado por la distancia física que debía ser recorrida por la señal de reloj, el *Itanium* poseía varios nodos de distribución, una especie de atajos, que hacía que la señal alcanzase las áreas más remotas del procesador de forma más rápida.

3.5.8 AMD Hammer

Actualmente, los procesadores de 64 bits ya son una realidad, pero a principios de siglo ya se estaba de acuerdo en que los procesadores de 64 bits serían realmente el futuro, sólo faltaba saber cuándo y cómo se iba a realizar esa transición. Pues ahora estamos en el momento exacto, a mediados del año 2005, con la aparición de los sistemas operativos de 64 bits y la consagración de los procesadores de 64 bits.

Entre las ventajas de los procesadores de 64 bits están el soporte para más memoria *RAM*, un mayor poder de procesamiento, ya que son procesados 64 bits de datos en cada ciclo, una mejor capacidad para manejar las grandes bases de datos, capacidad para generar claves de encriptación mucho más avanzadas, etc.

La cuestión de la memoria *RAM* es probablemente la más crítica, pues los procesadores de 32 bits actuales, como el *Pentium 4* y el *Athlon*, sólo son capaces de direccionar 4 GB de memoria *RAM*. Esto aún puede parecer mucho para un PC doméstico, pero pasará a ser una gran limitación a partir de los próximos años, como ya pasa actualmente en los servidores.

Hasta es posible que un *chip* de 32 bits visite más memoria, siempre que sea utilizada la segmentación de direcciones, donde el procesador divida la memoria disponible en bloques de hasta 4 GB y los visita de uno en uno. El problema es que este último recurso conlleva una gran pérdida de rendimiento y, además, no trae los demás recursos disponibles en un procesador de 64 bits.

La casa Intel optó por hacer una migración con su *Itanium*, proyectado para sólo soportar el nuevo conjunto de instrucciones de 64 bits, el *IA-64*. Es verdad que el *Itanium* posee un sistema de emulación que le permite ejecutar también las aplicaciones de 32 bits, pero al hacer esto, se pierden casi tres cuartas partes del rendimiento.

Por su parte, la casa AMD optó por un camino diferente al desarrollar su procesador de 64 bits. La idea era que si los procesadores de la época, todos procesadores de 32 bits, eran compatibles con *DOS* y otros sistemas proyectados para ser ejecutados en procesadores de 16 bits, por qué no crear un procesador de 64 bits que continuase siendo compatible con los programas de 32 bits, sin que existiese ninguna pérdida de rendimiento.

El Athlon Hammer de 64 bits

Para conseguir eso, el *Hammer* tiene dos modos de operación. En el *Legacy Mode* el procesador es compatible con todos los programas que tenemos actualmente donde, según la casa AMD, el *Hammer* será más rápido que cualquier procesador de 32 bits. Ya en el *Long Mode* el procesador asume su verdadera identidad como procesador de 64 bits, ejecutando las nuevas aplicaciones de 64 bits y que aprovecharán todos sus recursos.

La verdad es que en el *Hammer* las instrucciones de 64 bits son implementadas como extensiones. Todo el legado que viene desde el *8088* continúa presente, lo que puede, como mínimo, aumentar bastante el número de transistores del *chip* en relación a los que serían necesario sin este legado, o incluso perjudicar su rendimiento en las aplicaciones de 64 bits.

Sin embargo, personalmente creo que mantener el soporte integral a las instrucciones de 32 bits (16 bits también, si necesitamos ejecutar aplicaciones para *DOS* o *Windows 3.1*) es la mejor elección para un procesador proyectado para sustituir al *Athlon* como procesador doméstico de la casa AMD ,y al mismo tiempo, competir en el mercado de los procesadores para servidores.

Los recursos

Un recurso de destaca en el *Hammer* es la inclusión de un controlador de memoria dentro del propio procesador. Esto aumenta un poco el número de transistores y el coste pero, por otro lado, trae interesantes ventajas. En el diseño convencional, donde el controlador de memoria forma parte del *chipset* de la placa base, las señales salen del procesador, hacen una escala en el *chipset* y después llegan a la memoria RAM. En el *Hammer*, la señal todavía necesita pasar por el controlador de memoria, pero en compensación, el controlador opera a la misma frecuencia que el resto del procesador, y no a la misma frecuencia que la placa base. Esto resulta en unos tiempos de acceso más bajos y una ganancia considerable en el rendimiento.

El controlador de memoria del *Hammer* utiliza memorias *DDR*, con soporte tanto a los accesos de 64 bits (como en la mayoría de los sistemas actuales), como a los accesos de 128 bits, una opción que queda en manos del proyectista de la placa base. La casa AMD también divulgó que el *Hammer* soporta el uso de un controlador de memoria externo, si es necesario añadir soporte a la otra tecnología de memoria (por ejemplo *Rambus*) o en el caso de que alguien consiga desarrollar un controlador de memoria superior (improvable, pero posible).

Otra novedad importante es el soporte a las instrucciones *SSE2*, soportadas por el *Pentium 4*. Los procesadores de la casa AMD derivados del *core Palomino* (*Athlon XP*, *Duron Morgan*, etc.) ya traen soporte a las instrucciones *SSE* del Pentium III, un conjunto mucho más limitado que el implantado en el *Pentium 4*. Las nuevas instrucciones le permiten al *Hammer* pelear en igualdad de condiciones con el *Pentium 4* y el *Itanium*, ya que cada vez más programas están optimizados para este conjunto de instrucciones.

Un detalle interesante es que a pesar del buen rendimiento prometido, el *Hammer* no será capaz de operar a frecuencias de trabajo muy superiores a la de los Athlon de 0.13 micrones. Eso demuestra que la casa AMD no pretende seguir el camino de la casa Intel con el *Pentium 4*, añadiendo más niveles de *pipeline* y otros recursos para permitir que el procesador sea capaz de operar a frecuencias más altas, aunque con ello el rendimiento no crezca en la misma proporción.

El *Hammer* tiene 12 niveles de *pipeline*, en lugar de los 10 niveles del *Athlon* del momento, 10 niveles del *Pentium III* y los 20 niveles del *Pentium 4*. Podemos afirmar que existe una relación entre el número de niveles del *pipeline* y la frecuencia de operación que el procesador es capaz de alcanzar. En compensación, también existe una gran pérdida de rendimiento en cada ciclo de reloj al utilizar muchos niveles de *pipeline*. Por eso es por lo que un *Pentium 4* de 2.0 GHz acaba perdiendo en comparación con un *Athlon XP* de 1.533 MHz en la mayoría de las aplicaciones.

Además, el *Hammer* tiene el doble de unidades de ejecución: 6 unidades para enteros, 6 unidades para lectura y grabación y 6 unidades más para el coprocessador aritmético, 18 en total.

Esto ya es el suficiente para concluir que el *Hammer* consume algo próximo al doble de la electricidad de un *Athlon* antiguo de la misma frecuencia. Haciendo un cálculo aproximado, un *Hammer* de 3.0 GHz producido con una arquitectura de 0.13 micrones con el uso de *SOI* (*Silicon On Insulator*, una tecnología desarrollada por IBM que disminuye el consumo eléctrico del procesador y permite que sea capaz de operar a frecuencias más altas) consumirá entre 200 y 220 vatios.

Pero volviendo al requisito del rendimiento, el hecho de doblar el número de unidades de ejecución no significa que un *Hammer* sea dos veces más rápido que un *Athlon* de 32 bits de la misma frecuencia. Parte de la ganancia se pierde a causa del aumento en el número de niveles de *pipeline* y otra gran parte se pierde en ciclos ociosos del procesador. Considerando que la gran mayoría de los programas han sido compilados para ejecutarse en procesadores capaces de ejecutar una única instrucción en cada ciclo, es muy complicado mantener 18 unidades de ejecución trabajando durante todo el tiempo.

Ha existido una ganancia de rendimiento entre un 30 y un 40% en la mayoría de las aplicaciones, teniendo en cuenta que el controlador de memoria es bastante más rápido. Considerando el enorme consumo eléctrico y el precio del procesador, no era una buena idea pensar en adquirir uno inmediatamente después del lanzamiento.

Dos características más que no podíamos dejar de comentar son la memoria caché y el soporte para el multiprocesamiento. La casa AMD divulgó que existirán versiones del *Hammer* con hasta 1 MB de caché *L2*. La memoria caché *L1* continúa siendo la misma usada en el *Athlon* de 32 bits, con 128 KB en total, dividida en dos bloques de 64 KB, para datos e instrucciones.

Finalmente, el *Hammer* también aportó un soporte para el multiprocesamiento bastante perfeccionado. Como cada procesador tiene su propio controlador de memoria, cada uno dispone de sus propios módulos de memoria *RAM*, en comparación con los anteriores, donde el mismo bus de conexión con la memoria estaba dividido entre todos los procesadores instalados en la placa base. Esto comporta una gran ganancia de rendimiento en los servidores donde se utilizan 4 u 8 procesadores.

3.5.9 La famosa ley de Moore

Durante la década de los años 70, Gordon Moore, el presidente de Intel lanzó una profecía, que decía que el poder de procesamiento de los procesadores se doblaría cada 18 meses. Esta profecía se hizo tan real que acabó erigiéndose en la famosa ley de Moore, y realmente se viene manteniendo hasta los días de hoy

Como ya sabemos, un procesador está formado por transistores. Mientras más transistores tenga el *chip* del procesador, más operaciones por ciclo será capaz de ejecutar.

Sin embargo, al mismo tiempo, cuanto mayor sea el área ocupada por el procesador, su frecuencia de operación será menor, pues con los transistores más alejados unos de

los otros la señal tardará más tiempo para ser transmitida y, además de eso, tendremos una pérdida de señal e interferencias. Es justamente por ello que los *chips* son tan pequeños.

Recientemente, la casa Intel anunció sus investigaciones para producir una nueva familia de *chips* con transistores de sólo 0.07 micrones. Hasta hace unos años atrás, se creía que sería imposible producir transistores tan pequeños. Para hacernos una idea, con sólo 0.07 micrones por transistor, cada uno de los tres filamentos que compone cada transistor, sólo tendrá 3 átomos de cobre de grosor.

Según la casa Intel, los chips de 0.07 micrones deben comenzar a producirse a finales de 2005. Hasta entonces tendremos los chips de 0.13 y 0.10 micrones, que han permitido un gran salto en el poder de procesamiento de los *chips*. Los primeros *chips* de 0.10 micrones llegaron al mercado a mediados de 2004, lo que permitió una nueva familia de *chips* operando a frecuencias cercanas a los 4 GHz, pero con potencial para llegar hasta los 6 o 7 GHz. A partir de 2005 ya hemos tenido una fuerte presencia de los procesadores de 64 bits, principalmente del modelo adaptado al *Hammer* que es más adecuado al mercado doméstico, aunque también apareció el *Pentium 4* en su versión de 64 bits.

Finalmente, si Intel consigue cumplir sus promesas, tendremos a partir de finales de 2005 los primeros *chips* de 0.07 micrones, que tal vez alcancen la marca de los 10 GHz. De ahí en adelante aún tendremos, por lo menos, 3 generaciones más de procesadores. El secreto para conseguir producir transistores tan pequeños es la técnica de producción usada. Hasta hoy día se ha usado la litografía óptica, que consiste en usar un láser para marcar el silicio. A partir de los 0.13 micrones los fabricantes pasaron a utilizar luz ultravioleta (*Extreme Ultra Violet*) que ofrece una precisión mucho mayor.

3.6 El futuro

Si ya estamos impresionados con lo que los ordenadores actuales son capaces de hacer, todavía nos vamos a sorprender más con lo que creemos que está por venir en los próximos años.

3.6.1 La era de los transistores

El día 11/06/2001 la casa Intel anunció la creación del menor transistor del mundo, midiendo sólo 0.02 micrones. La noticia tuvo su repercusión, principalmente entre la comunidad científica. El tamaño de los transistores está directamente relacionado con la velocidad y el coste de los circuitos electrónicos. Cuanto menores son los transistores, más rápidos y baratos son.

Pero, dejando de lado el aspecto económico, el transistor de 0.02 micrones, o 20 nanómetros, como se prefiera, es una gran hazaña desde el punto de vista de la ingeniería, algo considerado imposible algunos años atrás.

Las posibilidades realmente serían muchas con los transistores de 0.02 micrones y los fabricantes tendrían munición para mantener la ley de Moore unos 15 años más.

Después puede ser que consigan desarrollar transistores aún menores, utilizando nanotubos, o algún otro material que sustituya al silicio, pero aún es muy pronto para hablar del final de la era de los transistores. También puede ser que las investigaciones en el ramo de los ordenadores cuánticos avancen hasta el punto de convertirlos en productos económicamente factibles.

3.6.2 La promesa de los procesadores cuánticos

Los procesadores cuánticos parecen ser realmente el futuro de los ordenadores. La arquitectura actual, que consiste en construir procesadores usando transistores, llegará fatalmente a su límite dentro de los siguiengtes años. Entonces, será necesaria una arquitectura más eficiente.

La pregunta es: por qué no sustituir los transistores por átomos? Los procesadores cuánticos tienen tanto potencial hasta el punto de poder realizar en pocos segundos lo que los procesadores actuales no podrían hacer ni en millones de años.

Realmente, los proyectistas de los microprocesadores vienen haciendo un gran trabajo. En las tres últimas décadas, desde la aparición del Intel *4004*, el primer microprocesador del mundo, hemos tenido un aumento alucinante en la velocidad de procesamiento. Para tener una idea, el *8088*, el procesador usado en el *XT* lanzado en el año 1979 tenía un poder de procesamiento estimado de sólo 0.25 *megaflops*, o sea, sólo 250.000 operaciones por segundo. Un *Pentium* 100 ya procesaba 200 *megaflops*, 200 millones de operaciones, mientras que un *Athlon* de 1.1 GHz procesaba casi 10 *gigaflops*, 40 mil veces más rápido que el *8088*.

El problema es que todos los procesadores actuales tienen la misma limitación: están compuestos por transistores. La solución para producir *chips* cada vez más rápidos ha sido disminuir cada vez más el tamaño de los transistores. Los primeros transistores, que surgieron en la década de los años 60, eran más o menos del tamaño de la cabeza de una cerilla, mientras que los actuales miden sólo 0.07 micrones (1 micrón equivale a 0.001 milímetros).

Sin embargo, estamos aproximándonos a los límites físicos de la materia; para que continúen avanzando será necesario abandonar el uso de los transistores y cambiar a alguna estructura más eficiente. El futuro de los procesadores parece estar en manos de los procesadores cuánticos.

La idea

La idea de usar átomos para procesar datos no es nueva, existen investigaciones en este sentido desde los inicios de la década de los años 80, pero sólo eran teorías libres y que actualmente comienzan a tomar forma.

En un procesador cuántico tenemos átomos en vez de transistores. En vez de bits tenemos bits cuánticos, o *qubits*. La idea fundamental es que en un átomo, la rotación de cada electrón corresponde a un pequeño movimiento magnético, que puede ser controlado si el átomo se encuentra situado en una superficie lo suficientemente sensible.

Una peculiaridad interesante es que mientras un transistor sólo permite dos estados, o sea, conectado o desconectado, cada *qubit* puede tener cuatro estados diferentes. Dentro de un átomo, los electrones pueden girar no sólo en el sentido horario o contrario, también pueden girar en los dos sentidos de forma simultánea. Esta es una característica que aún no es muy entendida, pero ya se utiliza en los primeros ordenadores cuánticos. Entonces, tenemos un total de 4 estados posibles, lo que permite que cada *qubit* procese dos bits de forma simultánea.

Esto permite ampliar exponencialmente la capacidad de los procesadores cuánticos, ya que dos *qubits* corresponden a 4 bits, 3 *qubits* corresponden a 8 bits y 5 *qubits* corresponden a 32 bits. Entonces, 10 *qubits* serán suficientes para 1024 bits, mientras que 20 *qubits* corresponden a más de un millón de bits. Esta puede ser la gran clave para aumentar de forma inimaginable tanto la potencia de los procesadores como la capacidad de los dispositivos de almacenamiento.

Aplicaciones

El extraordinario poder de procesamiento que pueden alcanzar los procesadores cuánticos sería extremadamente útil tanto para las investigaciones científicas como para las aplicaciones comerciales de realidad virtual e inteligencia artificial, que a buen seguro se pondrán de moda a lo largo de este siglo.

Los juegos podrían ser suficientemente reales para contener varios personajes que realmente interaccionen con el jugador, conversando y reaccionando de acuerdo con las acciones del jugador. Un ordenador cuántico podría controlar centenares de estos personajes en tiempo real. El reconocimiento de voz y de los gestos ya sería algo trivial.

A buen seguro, tendríamos gigantescos avances en prácticamente todos los campos. Finalmente podríamos tener códigos de encriptación realmente seguros, investigaciones en gigantescas bases de datos usando algoritmos inteligentes, la transmisión de datos podría alcanzar velocidades del orden de los TBytes por segundo usando fibras ópticas de alta densidad y *routers* cuánticos, capaces de lidiar con esta cantidad de información.

3.6.3 Nanotubos

Ya que los transistores de silicio tienen fecha para alcanzar el límite de su evolución y los procesadores cuánticos aún están en fase de idealización, la casa IBM está investigando nuevas tecnologías que puedan sustituir al silicio en la producción de los procesadores.

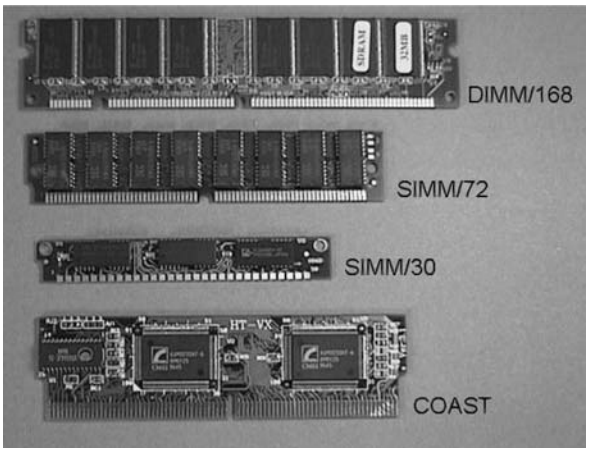
Primero anunció una tecnología prometedora, los *nanotubos*, cavidades microscópicas hechas en un bloque de carbono que podrían utilizarse para construir transistores menores que los actuales. Las investigaciones continuaron avanzando y hace poco anunciaron la creación de los primeros transistores usando esta tecnología.

Según los proyectistas de IBM, usando los *nanotubos* es teóricamente posible construir transistores de hasta 5 nanómetros, 0.005 micrones, menores que los 0.07 micrones actuales, o los 0.02 micrones que la casa Intel pretende alcanzar hasta finales de la presente década. Teóricamente, estos transistores permitirán la producción de unos *chips* con frecuencias por encima de 1 TeraHertz.

4. La memoria RAM

Si el procesador es el principal componente de cualquier ordenador, la memoria *RAM* es su principal herramienta de trabajo. Desde una simple calculadora hasta un gran servidor, no existe ningún tipo de ordenador que no utilice memoria *RAM*. El procesador utiliza la memoria *RAM* para almacenar los programas y los datos que están en uso y queda imposibilitado para trabajar si no dispone de por lo menos una cantidad mínima de ella. De hecho, en los ordenadores actuales, la velocidad de acceso a la memoria *RAM* es uno de los principales factores que marcan el rendimiento del equipo, de ahí la vital importancia del uso de la memoria caché.

Las siglas *RAM* significan *Ramdom Access Memory* o memoria de acceso aleatorio. Este nombre es más que adecuado, pues la principal característica de la memoria *RAM* es la capacidad de suministrar datos ya grabados en ella, con un tiempo de respuesta y una velocidad de transferencia centenares de veces superior a la de los dispositivos de memoria de masa, como el disco duro.



Algunos de los tipos de módulos de memoria RAM

Otra característica destacada de la memoria *RAM* es el hecho de ser volátil: necesita constantemente recibir corriente eléctrica para no perder los datos almacenados. Por ejemplo, si estamos escribiendo una carta en *Word* y, de pronto, se produce un pico

de tensión y el ordenador se reinicia sin darnos tiempo de guardar la carta en el disco duro, perderemos todo el trabajo no guardado.

Existen tipos de memoria *RAM* no volátiles, como por ejemplo las memorias *Flash*, que son utilizadas en los ordenadores de mano, móviles, cámaras digitales, etc. El problema de las memorias *Flash* es el precio. En un futuro, es posible que algún tipo de memoria no volátil sustituya de forma definitiva a la memoria *RAM*. La casa IBM viene desarrollando las memorias *MRAM*, que almacenan datos en forma de señales magnéticas. La promesa es que además de que conserven los datos grabados durante años, sean tan rápidas y eficaces como las memorias actuales. Normalmente, las promesas de los fabricantes no son muy fiables.

Ya que vamos a continuar usando memoria *RAM* volátil durante algunos (o varios) años más, estudiaremos su historia y las tecnologías de memoria disponibles en la actualidad. Antiguamente, los módulos de memoria *RAM* eran muy caros. Pocos años atrás, pagábamos el equivalente a más de 40 euros por cada MByte. En pleno inicio de la década de los 90, el precio de la memoria llegó a batir la marca de los 100 euros por cada MByte.

Con la memoria *RAM* costando esta fortuna, es comprensible que la gran mayoría de PCs *386* y *486* vinieran equipados con sólo 4 u 8 MBytes de memoria, aunque la cantidad recomendable para ejecutar sistemas gráficos como *Windows 95* o el XFree86 de *Linux* era de por lo menos 16 MB. Por suerte, en los últimos años se produjo una caída vertiginosa en el precio de las memorias. Así como los procesadores evolucionan incorporando más transistores, operando a frecuencias más altas y costando cada vez más baratos, con el paso del tiempo los fabricantes de memorias han conseguido producir *chips* de memoria con transistores cada vez más pequeños. Con esto, es posible aumentar la cantidad de memoria en cada *chip* y abaratar el precio de cada MByte.

Volviendo al caso de los procesadores, un *486* sólo tenía 1.200.000 transistores, mientras que un *Athlon Thunderbird* tiene alrededor de los 37 millones. Aún así, un *Athlon* cuesta menos de lo que costaba un *486*. La clave para este salto es la miniaturización de los transistores.

En la memoria *RAM*, cada transistor representa un bit. Un módulo de memoria con 64 MBytes está formado por cerca de 512 millones de transistores. Conforme los fabricantes desarrollan nuevas tecnologías, pasan a ser capaces de producir transistores más pequeños. Con eso, es posible incluir más transistores y, consecuentemente, más memoria en cada *chip*. Los primeros *chips* de memoria RAM *tenían* sólo 1 Kb (un Kbit y no un KByte), mientras que actualmente no se producen *chips* con menos de 128 Mb (Megabits). Juntando 8 *chips* tenemos un módulo de memoria con 128 MB, juntando 16 tenemos un módulo de 256 MB.

El bajo precio actual justifica el uso de, como mínimo, 128 o 256 MB de memoria *RAM*. Algunos usuarios de aplicaciones más pesadas ya van más lejos, definiendo los 512 MB como el mínimo requerido. La cantidad y la velocidad de las memorias son más

importantes que la propia velocidad del procesador, principalmente para los que suelen trabajar con varios programas abiertos al mismo tiempo, o trabajan con archivos pesados, imágenes, vídeo, etc.

La ecuación es bastante simple. Cualquier programa o archivo en uso necesita ser almacenado en la memoria *RAM*. El procesador comienza a usar la memoria *RAM* que es razonablemente rápida, la suficiente para que (con la ayuda de la memoria caché) el procesador pueda usar todo su potencial.

Pero, qué pasa cuando se agota la memoria *RAM*? Lo más normal es ver que las aplicaciones se continúan ejecutando de forma muy lenta, en lugar de que el ordenador muestre un mensaje de falta de memoria. Esto sucede gracias a la memoria virtual.

La memoria virtual sirve para permitir que el procesador simule más memoria *RAM* creando un archivo en el disco duro y almacenando en él los datos que no caben en la memoria. El problema es que el procesador procesa billones de instrucciones por segundo, mientras que un disco duro rápido tiene tiempos de acceso en torno a las 10 milésimas de segundo.

Como el procesador no puede trabajar cuando no dispone de datos para procesar, necesita esperar hasta que el disco duro pueda entregarle los datos que deben ser procesados. De nada sirve disponer de un procesador muy rápido si, por la falta de memoria *RAM*, no puede desarrollar todo su potencial debido al uso de memoria virtual, quedándose limitado al rendimiento del disco duro.

4.1 El funcionamiento

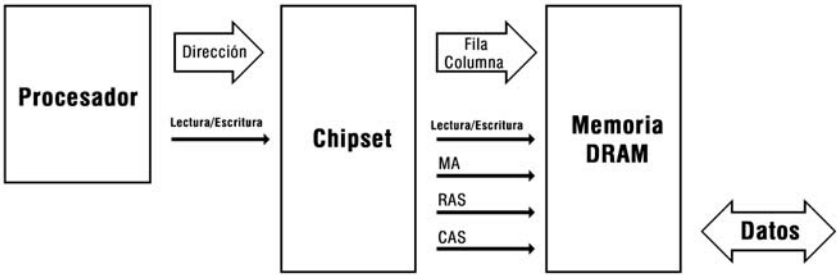
Los *chips* de memoria *RAM* poseen una estructura bastante simple. En cada bit se almacenará un 1 o un 0. Para ello, disponemos de un minúsculo *capacitor*; cuando el *capacitor* está cargado eléctricamente tenemos un bit 1 y cuando está descargado tenemos un bit 0. Para cada *capacitor* existe un transistor, encargado de la lectura del bit almacenado en su interior y lo transmite al controlador de la memoria. La memoria *RAM* es volátil justamente debido a que el *capacitor* pierde su carga de forma muy rápida, después de pocas milésimas de segundo.

La producción de los *chips* de memoria es muy similar a la producción de los procesadores: también se utiliza un *waffer* de silicio como base y un láser para su marcado. La diferencia es que los *chips* de memoria están básicamente compuestos de una sola estructura básica: el conjunto *capacitor/transistor*, que es repetida algunos millones de veces, mientras que los procesadores están formados por estructuras más complejas.

Debido a esta simplicidad, un *chip* de memoria es más barato de producir que un *chip* para un procesador. Por ejemplo, un módulo de 128 MBytes tiene poco más de 1 billón de transistores, casi 40 veces más que un procesador Pentium III Coppermine. A pesar de esto, el módulo de memoria es más barato.

4.2 El acceso a los datos

Para leer y escribir los datos en la memoria, así como para controlar todo el tráfico de datos entre la memoria y los demás componentes del equipo, se emplea el llamado controlador de memoria, que forma parte del *chipset* localizado en la placa base.



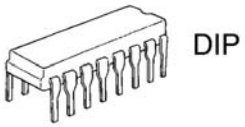
Esquema simple del funcionamiento de un equipo con memoria DRAM

Para facilitar el acceso a los datos, dividimos los módulos de memoria en filas y columnas. Para visitar un determinado transistor (ya sea para escribir o leer los datos), el controlador de memoria primero genera el valor *RAS* (*Row Address Strobe*), el número de la fila a la que pertenece el transistor, generando enseguida el valor *CAS* (*Collum Address Strobe*), que corresponde al número de la columna.

4.3 Formato

Los *chips* de memoria son placas de silicio, que necesitan ser encapsulados en una estructura más resistente antes de ser transportados y ensamblados en la placa base. Así como tenemos varios tipos de encapsulado para los procesadores, (*SEPP* y *PPGA* por ejemplo), también existen varios formatos de módulos de memoria.

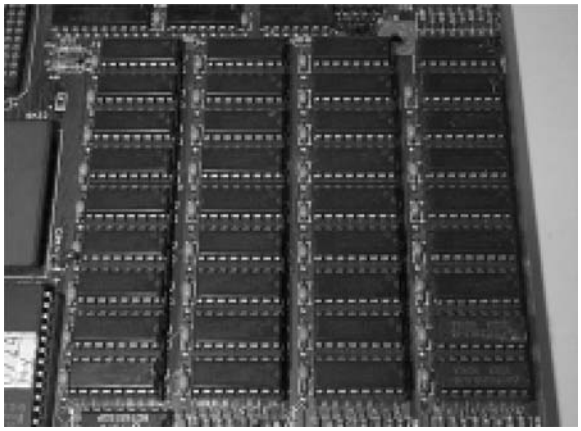
Inicialmente, los *chips* son encapsulados en módulos *DIP*, que los protegen y facilitan la disipación del calor generado por los *chips*. Por su parte, los módulos *DIP* son soldados en placas de circuito, formando los módulos de memoria. Existen 3 tipos de módulos de memoria que se han estandarizado a lo largo del tiempo: los módulos *SIMM* de 30 vías, los módulos *SIMM* de 72 vías y los módulos *DIMM* de 168 vías. En la actualidad empiezan a surgir los módulos *RIMM* y *DIMM* de 184 vías.



Un módulo DIP que luego será encajado en un módulo mayor

4.3.1 Los módulos DIP

Los módulos *DIP* son "encapsulados" de plástico o de cerámica que se encargan de proteger los *chips* de memoria y facilitan la disipación del calor generado durante su funcionamiento y, además, hacen más accesibles el encaje y la soldadura. El encapsulado *DIP* también es usado en otros tipos de componentes.

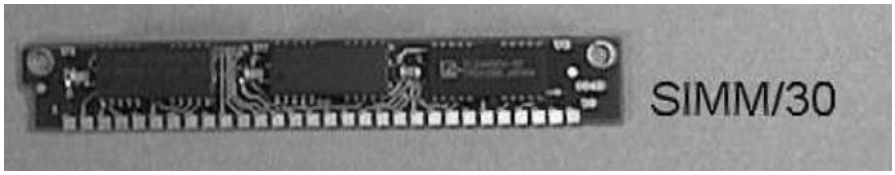


Módulos DIP soldados en la placa base

En los PCs antiguos, principalmente los *XT*, *286* y los primeros *386*, los módulos *DIP* estaban soldados directamente en la placa base o, en algunos casos, ensamblados individualmente en soquetes disponibles de la placa. Este era un sistema anticuado, que conllevaba varias desventajas, al dificultar las posibles ampliaciones de memoria o la sustitución de los módulos defectuosos.

4.3.2 Los módulos SIMM de 30 vías

Fue una cuestión de tiempo hasta que apareció una alternativa más práctica, capaz de facilitar la instalación incluso para los usuarios inexpertos. Los módulos de memoria son pequeñas placas de circuito donde son soldados los módulos *DIP* y, posteriormente, son ensamblados en los soquetes disponibles de la placa base.



Un módulo de memoria SIMM de 30 contactos

Los primeros módulos de memoria creados fueron llamados módulos *SIMM*, siglas que significan *Single In Line Memory Module*, justamente porqué existe una única vía de contactos. A pesar de que también existan contactos en la parte posterior del módulo, sólo sirven como una extensión de los contactos frontales, existiendo sólo para

aumentar el área de contacto con el soquete. Examinando el módulo, veremos un pequeño orificio en cada contacto, que sirve justamente para unificar los dos lados.

Los primeros módulos *SIMM* tenían 30 vías y eran capaces de transferir 8 bits en cada ciclo. Estos módulos fueron utilizados en los equipos equipados con los procesadores *386* y *486*, y fueron fabricados con varias capacidades, los más comunes fueron los módulos de 512 KB, 1 MB y 4 MB, aunque también han existido módulos de hasta 16 MB, raros y muy caros para la época.

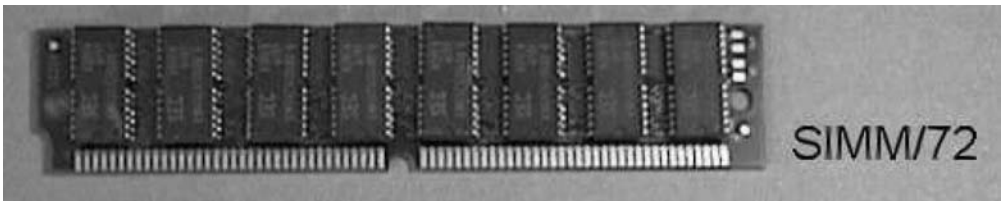
Como el *386* y el *486* eran procesadores que accedían a la memoria usando palabras de 32 bits, era preciso combinar 4 módulos de 30 vías para formar cada banco de memoria. Básicamente, el procesador visitaba los 4 módulos como si se tratase un sólo módulo, obteniendo los 32 bits que necesitaba en cada uno de los accesos.

Entonces, teníamos que usar estos módulos en cuartetos: 4 o 8 módulos de memoria. En los ordenadores equipados con procesadores *386SX* sólo eran necesarios 2 módulos, ya que el *386SX* accedía a la memoria usando palabras de 16 bits.

4.3.3 Los módulos SIMM de 72 vías

A pesar de ser más prácticos que los *chips DIP*, los módulos *SIMM* de 30 vías aún tenían bastantes inconvenientes, ya que era preciso usar 4 módulos idénticos para formar cada banco de memoria.

Para solucionar este problema, los fabricantes crearon un nuevo tipo de módulo de memoria *SIMM*, de 32 bits, que poseía 72 vías. Ese tipo de memoria fue usado en los equipos *486* más modernos y sirvió de patrón para los equipos *Pentium*, dejando de ser utilizados después de la aparición de los módulos de 168 vías, los más usados en la actualidad.



Un módulo SIMM de 72 vías o contactos

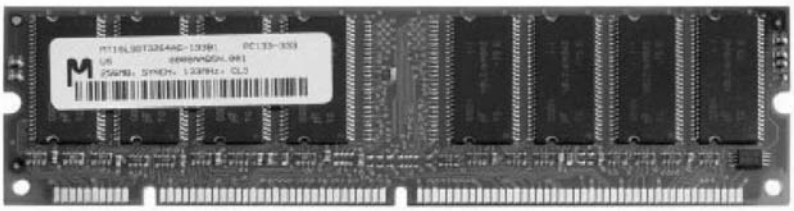
En vez de cuatro módulos, sólo era preciso un módulo *SIMM* de 72 vías para formar cada banco de memoria en los equipos con procesadores *486*. Como el Pentium accedía a la memoria usando palabras de 64 bits, eran necesarios 2 módulos *SIMM* de 72 vías en cada banco de memoria.

En realidad, después del *Pentium*, prácticamente todos los procesadores visitan la memoria usando palabras de 64 bits. A pesar de que los procesadores *Pentium II*, *Pentium III*, *Celeron*, *Athlon*, *Duron*, etc. han sido procesadores de 32 bits, acceder a 64 bits de la memoria ayuda a mejorar el rendimiento general del equipo.

El procesador es más rápido que la memoria *RAM*, que después de esperar varios ciclos a ser accedida, lo mejor es recoger la mayor cantidad de datos posibles y guardarlos en la memoria caché. Naturalmente, los datos serán procesados en bloques de 32 bits. No es casualidad que casi dos tercios de los transistores de un *Pentium III Coppermine* son usados por las memorias caché *L1* y *L2.*

4.3.4 Los módulos DIMM de 168 vías

Al contrarío de los módulos *SIMM* de 30 y 72 vías, los módulos *DIMM* poseen contactos en ambos lados del módulo, lo que justifica su nombre, *Double In Line Memory Module* o módulo de memoria con dos líneas de contactos. Como los módulos DIMM trabajan con palabras binarias de 64 bits, un único módulo es suficiente para llenar un banco de memoria en un equipo *Pentium* o superior. Por ejemplo, si queremos instalar 64 MB de memoria en un *Pentium III* sólo será necesario comprar un módulo *DIMM* de 64 MB.



Un módulo DIMM de 168 contactos

Actualmente, los módulos *DIMM* de 168 vías son los únicos que se continúan fabricando. Difícilmente podremos encontrar módulos de memoria de 72 vías, o placas base nuevas que los admitan, sino son de segunda mano.

4.4 Los bancos de memoria

Un procesador de la familia *x86* puede trabajar con cualquier cantidad de memoria, a partir de un mínimo de 64 KB. Sin embargo, es preciso que exista un bus de datos suficiente. Cualquier cantidad de memoria que atienda a las exigencias del procesador en cuanto al ancho de bus es conocida por banco de memoria.

En un *Pentium*, cada banco de memoria debía ser capaz de permitir el acceso de 64 bits de datos por vez, mientras que en los procesadores *386* y *486* era necesario un bus de 32 bits. Si el módulo de memoria posee un bus de datos más estrecho que el necesario, es necesario combinar dos o más módulos para formar cada banco de memoria.

Como los módulos *SIMM* de 30 vías poseían un bus de sólo 8 bits, eran necesarios 4 módulos para formar un banco de memoria en un equipo con un procesador *386* o *486*. Si estos módulos antiguos llegaran a ser utilizados en equipos *Pentium*, serían necesarios 8 módulos para formar cada banco de memoria. Los módulos *SIMM* de 72

vías ya poseían un bus de 32 bits, y con un único módulo era suficiente para completar un banco de memoria en un *486*, y eran necesarios 2 módulos para completar los 64 bits exigidos por un *Pentium*.

Finalmente, los módulos *DIMM* poseen un bus de 64 bits, siendo necesario un sólo módulo para formar un banco de memoria en los ordenadores equipados con procesadores *Pentium* o posteriores.

Dentro de un banco de memoria todos los módulos son accedidos al mismo tiempo, como si se tratase de un único módulo. Por eso, es necesario que todos los módulos de memoria sean capaces de responder a las llamadas del controlador de memoria de forma sincronizada. La más mínima falta de sincronía entre los módulos causaría una inestabilidad del sistema, que podría conllevar a bloqueos del mismo.

Por eso, es altamente recomendable que siempre se utilicen módulos idénticos dentro de un mismo banco de memoria (misma marca, capacidad, tiempo de acceso, etc.). Generalmente, en una placa base encontramos dos o tres bancos de memoria, que son numerados a partir del 0 (banco 0, banco 1, banco 2, etc.). No existe ningún problema en usar módulos de memoria distintos en los diferentes bancos.

Una curiosidad es que algunas placas base para procesadores *Pentium* pueden trabajar con un sólo módulo *SIMM* de 72 vías. En este caso, la placa engaña al procesador, realizando dos accesos consecutivos de 32 bits cada uno, y entregando los datos al procesador de una sola vez. A pesar de que esto funcione, este esquema reduce bastante la velocidad del equipo, pues la velocidad de acceso a la memoria se reduce a la mitad.

4.5 Tecnologías

Las memorias también contribuyeron, en parte, al desarrollo de los procesadores. Desde las primeras memorias de inicios de la década de los años 80 hasta las memorias producidas en la actualidad se ha usado la misma estructura básica, formada por un *capacitor* y un transistor para cada bit de datos. Sin embargo, se realizaron mejoras en la forma de organización física y en la forma de acceso que permitieron que se pudiese aumentar considerablemente la velocidad de acceso.

También fue posible aumentar la velocidad de acceso a los datos depositados en la memoria a través del aumento del ancho del bus de datos. El primer PC original sólo era capaz de leer 8 bits en cada ciclo de reloj, mientras que cualquier *Pentium* es capaz de leer 64 bits en cada ciclo: 8 veces más.

Durante estas dos décadas han existido distintas tecnologías de memoria, comenzando por las memorias regulares, usadas en los *XT* y algunos *286*, que evolucionaron hacia las memorias *FPM*, usadas en los *386* y *486*, y luego hacia las memorias *EDO*, usadas en los últimos *486* y en los *Pentium*. Estas tres tecnologías fueron sustituidas por las memorias *SDRAM*, usadas en los últimos PCs con procesadores *Pentium* y

Pentium MMX y sirvieron de patrón a partir del *Pentium II* y del *K6-2*. Las memorias *SDRAM* ya están llegando a su límite. Actualmente ya disponemos de las memorias *DDR* y *Rambus*. Inmediatamente tendremos también las memorias *DDR 2*.

4.5.1 Las memorias regulares

Las memorias regulares o "comunes" fueron el primer tipo de memoria usado en los PCs. En este tipo antiguo de memoria, el acceso se realizaba enviando primero la dirección *RAS* y enseguida la dirección *CAS*. Una forma de acceso simple, que era adecuada con la tecnología de la época pero que no ayudaba mucho en términos de rendimiento de la memoria.

Este tipo de memoria fue fabricado con velocidades de acceso a partir de los 150 nanosegundos (billonésimas de segundo), más que suficientes para soportar la frecuencia de 4.77 MHz del primer PC original. Posteriormente, se desarrollaron versiones con tiempos de acceso de 120, 100 y 80 nanosegundos para ser utilizadas en los equipos con procesadores *286*. Las memorias regulares sólo se encuentran en forma de módulos *DIP*, y fueron utilizadas en los equipos *XT*, *286* y en algunos de los primeros *386*.

4.5.2 Las memorias FPM

La primera mejora significativa en la arquitectura de las memorias vino con el *FPM* (*Fast Page Mode*) o modo de acceso rápido. La idea es que al leer un archivo grabado en la memoria, los datos están grabados de forma secuencial en la mayoría de los casos. En ese caso, no sería preciso enviar la dirección *RAS* y *CAS* para cada bit que debe leerse, sino enviar la dirección *RAS* (fila) una vez y, enseguida, enviar varias direcciones *CAS* (columna). Debido a este nuevo método de acceso, las memorias *FPM* consiguieron ser cerca de un 30% más rápidas que las memorias regulares.

A pesar de que ya no se fabrican desde hace bastante tiempo fueron utilizadas en los equipos *386*, *486* y en los primeros *Pentium*. Encontrábamos las memorias *FPM* en forma de módulos *SIMM* de 30 o 72 vías y con unos tiempos de acceso de 80, 70 y 60 nanosegundos, siendo las de 70 nanosegundos las más comunes. Los tiempos de acceso representan el tiempo que pasa hasta que la memoria puede disponer del dato requerido. Cuánto más bajos sean los tiempos, más rápidas serán las memorias.

Instaladas en una placa base que trabajase con un bus de 66 MHz, los intervalos de espera de las memorias *FPM* podían ser de hasta 5-3-3-3, lo que significa que el procesador tenía que esperar cinco ciclos de la placa base para que la memoria efectuase la primera lectura de datos y 3 ciclos más para las siguientes lecturas. El primer acceso tardaba más tiempo, pues en él era preciso enviar las dos direcciones, mientras que en los subsiguientes accesos sólo era necesario enviar las direcciones de las columnas.

Los tiempos de espera de las memorias pueden configurarse en el *Setup* del *BIOS*, a través de la opción de las opciones de configuración de la memoria. En las placas base para procesadores *486* era común que existiesen opciones como 4-3-3-3 o 3-2-2-2, que

representaban el tiempo de acceso a la memoria. Debemos recordar que los procesadores *486* trabajaban con bus de datos de 33 o 40 MHz. Con menos accesos por ciclo, era natural que cada acceso a la memoria tardase menos ciclos de la placa base. En las placas base para procesadores *Pentium* las opciones *slow*, *normal* y *fast* eran lo más común, que sustituían a los valores numéricos.

4.5.3 Las memorias EDO

Las memorias *EDO* (*Extended Date Output*) fueron creadas en el año 94 y trajeron una mejora muy significativa en el modo de acceso a los datos. Además de mantenerse el modo de acceso rápido de las memorias *FPM* se realizaron algunas modificaciones para permitir algún pequeño truco, a través del que podía iniciarse un acceso a los datos antes de que terminase el anterior, permitiendo aumentar la velocidad de los accesos. El nuevo modo de acceso permitía que las memorias *EDO* trabajasen con tiempos de acceso de sólo 5-2-2-2 en una placa base con un bus de 66 MHz, con una ganancia del 25%.

A pesar de ser un modo anticuado, este tipo de memoria aún es muy usado en algunos ordenadores que todavía existen, siendo fabricado en velocidades de 70, 60 y 50 nanosegundos, con un predominio de los módulos de 60 nanosegundos. Las memorias *EDO* se encuentran en módulos *SIMM* de 72 vías, existiendo también algunos casos raros de memorias *EDO* en forma de módulos *DIMM*.

Las mejoras en la arquitectura de las memorias *EDO* hicieron que fuesen incompatibles con las placas base equipadas con *chipsets* más antiguos. Básicamente, sólo las placas base para procesadores *Pentium* y algunas placas base para *486* con *slots PCI* aceptaban trabajar con memorias *EDO*. También existían placas base "tolerantes", que funcionaban con memorias *EDO* a pesar de que no eran capaces de aprovechar el modo de acceso más rápido y, finalmente, las placas incompatibles, que no llegaban ni a inicializarse en el caso de tener memorias *EDO* instaladas.

Todos los módulos de 30 vías son de memorias *FPM*, mientras (con excepción de algunos módulos antiguos) todos los de *168* vías son de memorias *SDRAM*. La confusión sólo existe con los módulos de 72 vías, que pueden ser memorias *EDO* y memorias *FPM*. Como casi todos los módulos de 70 nanosegundos son memorias *FPM*, y casi todos los módulos de memorias *EDO* son de 60 nanosegundos, podemos observar las inscripciones del módulo para saber el tipo de memoria usada.

4.5.4 Las memorias BEDO

Las memorias *BEDO* (*Burst Extended Data Output*) utilizan una especie de *pipeline* para permitir accesos más rápidos. En un bus de 66 MHz, las memorias *BEDO* son capaces de funcionar con tiempos de acceso de 5-1-1-1, casi un 30% más rápido que las memorias *EDO* convencionales. El aspecto más interesante es que el coste de producción de las memorias *BEDO* es prácticamente el mismo que los de las memorias *EDO* y *FPM*.

El mayor impedimento para la popularización de las memorias *BEDO* fue la falta de soporte por parte de los *chipsets* Intel, que sólo soportaban memorias *EDO* y SDRAM. Al final, las sucesoras de las memorias *EDO* fueron las memorias *SDRAM*, que a pesar de ser un poco más caras, ofrecían un rendimiento superior al de las memorias *BEDO* y tenían compatibilidad con todos los *chipsets* modernos.

4.5.5 Las memorias SDRAM

Tanto las memorias *FPM* como las memorias *EDO* eran asíncronas, esto significa que trabajaban a su propio ritmo, independientemente de los ciclos de la placa base. Eso explica por qué las memorias *FPM* que fueron proyectadas para funcionar en placas base para procesadores *386* o *486* (que trabajaban con un bus de 25, 30, 33 o 40 MHz), funcionaban sin problemas en las placas base para procesadores *Pentium*, que operaban a 66 MHz. Realmente, las memorias continuaban operando a la misma velocidad, lo que realmente era distinto eran los tiempos de espera que pasaban a ser más altos. Así, en vez de responder cada 2 ciclos de la placa base, por ejemplo, pasaban a responder cada 3 o 4 ciclos, funcionando normalmente.

Las memorias *SDRAM* (*Synchronous Dynamic RAM*) por su parte, son capaces de trabajar de forma sincronizada con los ciclos de la placa base, sin tiempos de espera. Esto significa que las memorias *SDRAM* siempre realizan una lectura por ciclo. Independientemente de la velocidad del bus al que está conectada, los tiempos de acceso podrán ser de 5-1-1-1. Observemos que sólo a partir del segundo ciclo, la memoria es capaz de mantener un acceso por ciclo, el primer acceso continúa siendo igual de lento que en las memorias *EDO* y *FPM*, consumiendo 5 ciclos de reloj. Como es preciso que la memoria *SDRAM* sea lo suficientemente rápida para trabajar al mismo ritmo que la placa base, es posible encontrar versiones de memorias *SDRAM* con tiempos de acceso entre los 15 y 6 nanosegundos.

Para determinar la velocidad máxima de operación de un módulo de memoria *SDRAM*, basta dividir 1.000 por su tiempo de acceso: por ejemplo, un módulo con un tiempo de acceso de 15 nanosegundos sólo podrá trabajar a 66 MHz, ya que 1000/15 = 66. Otra con un tiempo de acceso de 12 nanosegundos ya podrá trabajar a 75 MHz o hasta a 83 MHz, ya que 1000/12 = 83. Para confirmar la validez de este cálculo, basta dividir 1 segundo por 83 millones de ciclos de la placa base y obtendremos justamente los 12 nanosegundos de su tiempo de acceso.

Justamente el hecho de trabajar de forma sincronizada con los ciclos de la placa base hace que las memorias *SDRAM* sean más rápidas que todas sus antecesoras. Un módulo con un tiempo de acceso de 12 nanosegundos consigue ser cerca de un 30% más rápido que los módulos de memorias *EDO* de 60 nanosegundos a 66 MHz (5 + 1 + 1 + 1 = 8 ciclos por 4 accesos a la memoria *SDRAM* contra los 5 + 2 + 2 + 2 = 11 ciclos por 4 accesos a la memoria *EDO*) y cerca de un 50% más rápido a 83 MHz (5 + 1 + 1 + 1 = 8 accesos a la memoria *SDRAM* contra los 6 + 3 + 3 + 3 = 15 accesos a la memoria *EDO*).

Si se utiliza un bus de 100 MHz (en este caso necesitaríamos memorias *PC-100*), las memorias *EDO* serían casi 2.5 veces más lentas (5 + 1 + 1 + 1 = 8 accesos frente de los 7 + 4 + 4 + 4 = 19 accesos).

Al no trabajar de forma sincronizada con el reloj de la placa base, las memorias *FPM* y *EDO* podrían trabajar con cualquier frecuencia de bus: 100, 133, 200 MHz, o hasta más, si los tiempos de espera fuesen configurados correctamente. Sin embargo, mientras más alta es la velocidad del bus, mayores serían los tiempos de espera y peor sería el rendimiento de las memorias. Por eso, no se suelen utilizar memorias *EDO* o *FPM* en frecuencias de bus superiores a los 75 MHz.

4.5.6 Las memorias PC-100

El patrón de las memorias *PC-100*, desarrollado por la casa IBM, consiste en una serie de especificaciones que fijan el proceso de fabricación de memorias capaces de funcionar establemente en placas base que trabajan a 100 MHz. Teóricamente, cualquier módulo de memoria *SDRAM* con un tiempo de acceso por debajo de los 10 nanosegundos podría funcionar a 100 MHz, pues 1000/10 = 100. El problema es que, a pesar de que algunos módulos de memoria *SDRAM* antiguos (llamados *PC-66*, ya que tenían el funcionamiento garantizado a sólo 66 MHz) ofrezcan tiempos de acceso de 10 nanosegundos, muchas veces poseen un tiempo de latencia muy alto, fallando cuando son obligadas a funcionar a 100 MHz.

A pesar de que los fabricantes encontrasen dificultades iniciales para producir las memorias *PC-100*, con la proliferación de los procesadores que trabajaban con un bus de 100 MHz, como las versiones de 350, 400, 450 y 500 MHz del *Pentium II* y los *K6-2* de 300, 350, 400 y 450 MHz, se estandarizaron las memorias *PC-100*, y más recientemente las *PC-133* y *PC-150*.

4.5.7 Las memorias PC-133

Con la evolución de las técnicas de fabricación, fue posible producir módulos de memoria *SDRAM* con tiempos de acceso cada vez menores. A pesar del lento proceso de evolución comparado con el tiempo de evolución de otros componentes, fue posible producir memorias *SDRAM* capaces de funcionar por encima de los 133 MHz.

Estas memorias, llamadas *PC-133* son utilizadas por las versiones con un bus de 133 MHz del *Pentium III* y del AMD *Athlon*, y dado que permiten un acceso más rápido a los datos, contribuyen en el aumento del rendimiento global del equipo.

> **Nota**: debemos remarcar que las memorias *PC-133* funcionan sin problemas en las placas base con un bus de 66 o 100 MHz, así como las memorias *PC-100* trabajan a 66 MHz. Existe una frecuencia máxima de trabajo, pero también son soportadas las frecuencias menores. Inclusive, podemos mezclar módulos *DIMM* con diferentes tiempos de acceso en la misma placa base, aunque debemos utilizar una frecuencia de bus compatible con el módulo más lento.

Existen algunos casos de incompatibilidades entre algunas marcas o modelos de módulos de memoria y algunos modelos específicos de placas base, así como en algunas combinaciones de módulos de marcas diferentes.

Por eso, es posible que algunas combinaciones no dejen funcionar al ordenador, pero debería bastar el intercambio de los módulos de memoria por otros de una marca diferente. También existen algunos casos de placas base antiguas que son incompatibles con módulos de memoria *PC-100* o *PC-133*, o módulos de más de 64 MB. Más adelante y ya no hace tanto tiempo, pasaron a ser comunes los módulos de memoria *PC-150*, capaces de trabajar a 150 MHz.

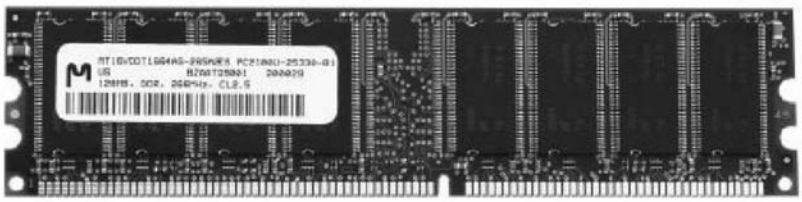
4.6 Las nuevas tecnologías

Gracias a la multiplicación de la frecuencia de reloj, los procesadores trabajan a frecuencias cada vez más altas que las usadas por la placa base y, consecuentemente, por la memoria. Como ejemplo el caso de un *Celeron* a 700 MHz. En él, la placa base y la memoria trabajan a sólo 66 MHz, el multiplicador usado por el procesadores es 10.5x. O el caso de un *Athlon* a 1.4 GHz que utiliza memorias a 133 MHz, con un multiplicador de, nuevamente, 10.5x.

Incluso con la ayuda de las memorias caché, el rendimiento de la memoria *RAM* está comenzando a limitar severamente el rendimiento del procesador, sobre todo en los juegos y aplicaciones donde se manipulan grandes cantidades de datos. Actualmente, tenemos básicamente dos sucesoras de las memorias *SDRAM*: las memorias *DDR-SDRAM* y las memorias *Direct Rambus*. Ya se han ensayado los pasos de la sucesora de las memorias *DDR*, que serán llamadas *DDR 2*.

4.6.1 Las memorias DDR-SDRAM

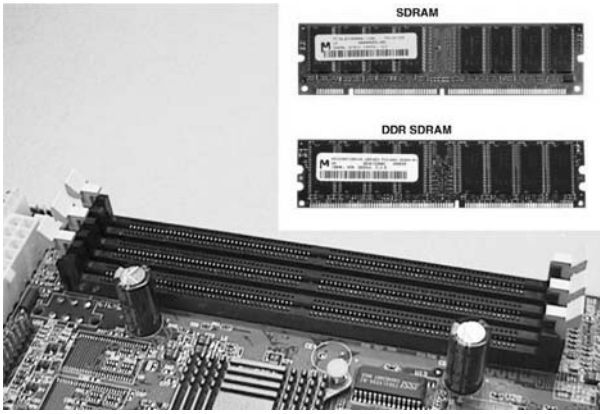
La denominación *DDR* (*Double Data Rate*) enfatiza en la principal característica de las memorias *DDR*: permitir dos transferencias de datos en cada ciclo de reloj, un esquema semejante al usado en el bus *AGP 2x*. Mientras que en un módulo de memoria *SDRAM PC-100* se transfieren 64 bits en cada ciclo de reloj, resultando una tasa de transferencia de 800 MB/s, en un módulo de memoria *DDR-SDRAM*, también de 100 MHz, tenemos dos transferencias de 64 bits en cada ciclo de reloj, alcanzando una tasa de transferencia de 1.6 GB/s, simplemente el doble.



Un módulo de memoria DDR-SDRAM

El punto fuerte de las memorias *DDR* es el precio, ya que producidas en gran cantidad cuestan poco más que las memorias *SDRAM*. En realidad, muchos fabricantes ya ofrecen las memorias *DDR* al mismo precio que las *SDRAM*, lo que volcará la tendencia en muy poco tiempo.

Los módulos de memoria *DDR-SDRAM* son parecidos a los módulos *DIMM* de memorias *SDRAM* tradicionales, sólo la posición de la hendidura es diferente, para evitar que un módulo *DDR* pueda ensamblarse en un *slot DIMM* común.



Las diferencias entre los módulos de memoria SDRAM y DDR

4.6.2 Las memorias Rambus

Las memorias *Direct Rambus* (*RDRAM*) permiten un bus de datos de sólo 16 bits de ancho, en oposición a los 64 bits utilizados por los módulos de memoria *SDRAM*, soportando en compensación, unas frecuencias de bus de hasta 400 MHz con dos transferencias en cada ciclo de reloj (como el *AGP 2x*), lo que en la práctica equivale a una frecuencia de 800 MHz. Trabajando a 400 MHz con dos transferencias en cada ciclo de reloj, su velocidad máxima, las memorias *Rambus* permiten una tasa de transferencia total de 1.6 GB/s.

Al contrario de las memorias *DDR* y *SLD* que sólo son evoluciones de las memorias *SDRAM*, las memorias *Direct Rambus* tienen una arquitectura completamente nueva, que exige mayores modificaciones en los *chipsets* destinados a trabajar con ellas, derivando en unos mayores costes de desarrollo y producción.

Los propios módulos de memoria *Rambus* son aún bastante más caros que los módulos de las memorias *SDRAM*. Claro que existe una tendencia de caída en los precios si aumenta la producción de este tipo de memoria, pero aún pueden pasar unos años hasta que lleguemos a un nivel próximo al del precio de las memorias *SDRAM*.

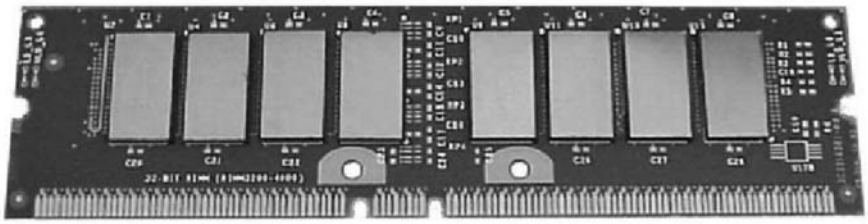
Debido a esto, muchos especialistas no creen que esta tecnología de memoria pueda ser realmente factible, y apuntan a las memorias *DDR* como una alternativa mucho más práctica. A pesar de todo, las memorias *Rambus* son las únicas nativamente

soportadas por los *chipsets* i820, i840 y i850 (que antes del i845 era el único *chipset* disponible para los procesadores *Pentium 4*).

Un problema que disminuye la velocidad del acceso a los datos en las memorias *Rambus* es el hecho de que los módulos trabajen con sólo 16 bits, mientras que los procesadores actuales se comunican con la memoria usando palabras de 64 bits. Para superar esta limitación es preciso que el controlador de memoria (situado en el *chipset* de la placa base) realice varios accesos consecutivos a los módulos de memoria y entregue los datos al procesador en forma de palabras de 64 bits.

Cuando es necesario escribir datos se realiza el proceso inverso; el controlador de memoria recibe los datos del procesador en forma de palabras de 64 bits, y éste realiza varios accesos consecutivos a los módulos de memoria para poder almacenar cada palabra de 64 bits.

Este procedimiento de conversión aumenta los tiempos de latencia, o sea, el tiempo que el procesador tiene que esperar entre cada operación de lectura o escritura de datos, retardando el acceso a la memoria. Es por esto, que a pesar de que los módulos de memorias *Rambus* son más rápidos, la ganancia de rendimiento en relación a las memorias *SDRAM* es pequeña. El ancho de banda total es mayor, pero en compensación se pierde más tiempo entre cada acceso a la memoria.



Un módulo RIMM de memoria Rambus

Los módulos de memorias *Rambus* son llamados *Rambus Inline Memory Modules* o *RIMM*. Los módulos *RIMM* son muy parecidos a los módulos *DIMM*, pero en general vienen con una protección de metal sobre los chips de memoria, que también sirve para facilitar la disipación del calor, ya que los módulos *RIMM* se calientan bastante debido a la alta frecuencia de trabajo.

4.6.3 Memorias DDR vs memorias Rambus

Existen básicamente tres factores que determinan la velocidad de un módulo de memoria. La primera es, naturalmente, la frecuencia de trabajo, la segunda es el bus de datos, o sea, cuantos datos son transferidos en cada ciclo y, finalmente, tenemos el tiempo de latencia, el número de ciclos que el módulo tarda en empezar a transferir datos a partir del momento que son solicitados y el intervalo a esperar antes de un nuevo acceso a la memoria

Por ejemplo, un módulo de memoria *SDRAM PC-100* típico trabaja a 100 MHz, o sea, realiza 100 millones de transferencias por segundo. En cada transferencia de datos son transmitidos 64 bits de datos mientras que el tiempo de latencia es de 2 o 3 ciclos, dependiendo del módulo. Con esto, tenemos un módulo capaz de transferir, en condiciones ideales 800 MB/s, pero en la práctica resulta ser menos.

Por su parte, un módulo de memoria *DDR PC-2100* trabaja a 266 MHz (133 MHz con dos transferencias por ciclo), también transmite 64 bits de datos en cada transferencia y tiene (como en un módulo típico) un tiempo de latencia de 2 o 3 ciclos. Con esto, tenemos un módulo capaz de transferir en condiciones ideales 2.1 GB/s, aunque también resulta ser menos en la práctica.

Un módulo de memoria *Rambus PC-800*, que es un tipo muy común, trabaja a 800 MHz aunque, sin embargo, sólo transfiere 16 bits de datos en cada ciclo. Podemos observar que a pesar de realizar más transferencias de datos por segundo, el módulo transfiere menos datos por ciclo, sólo 1/4 de lo transmitido por un módulo *DDR*. Con esto, tenemos un módulo capaz de transferir 1.6 GB/s. Es el doble de lo que transmite un módulo de memoria *PC-100*, pero es menos de lo que transmite un módulo *DDR*.

Debido a la arquitectura utilizada, el tiempo de latencia en un módulo de memoria *Rambus* es mayor que en un módulo de memoria *SDRAM* o *DDR*, de 4 a 5 ciclos, dependiendo del módulo. Esto hace que el módulo tarde más tiempo en iniciar cada transferencia de datos, lo que perjudica el rendimiento, especialmente cuando el procesador trabaja con varios archivos pequeños, situación muy común en un servidor de bases de datos, por ejemplo.

4.6.4 Cómo funcionan las memorias DDR

Como ya hemos visto, las siglas *DDR* provienen de *Double Date Rate*. Las siglas indican la capacidad de las memorias *DDR* de transmitir datos dos veces en cada ciclo, una transferencia en el inicio del ciclo del reloj y una segunda transferencia al final del ciclo. Por ejemplo, un módulo *DDR* de 266 MHz no trabaja a 266 MHz, sólo a 133 MHz y sin embargo, como realiza dos transferencias en cada ciclo, el rendimiento es equivalente al que alcanza un módulo de memoria *SDRAM* de 266 MHz.

El punto más interesante es que un módulo *DDR* es esencialmente un módulo de memoria *SDRAM PC-100* o *PC-133* común con sólo una ligera modificación en el circuito de acceso. Los módulos de memorias *SDRAM* y *DDR* son divididos de forma lógica en bancos. Dentro del módulo de memoria cada banco concentra una cierta parte de las direcciones de memoria disponibles. Cada banco se divide nuevamente en varias líneas que, finalmente, se dividen en columnas. La combinación de las direcciones de banco, línea y columna corresponden a la localización de los datos. Dentro de cada banco de memoria, sólo una línea puede permanecer activa cada vez, pero nada impide que sea realizado más de un acceso de forma simultánea, incluso a direcciones diferentes.

Para llevar esto a cabo no es necesario cambiar la estructura del módulo, sólo es necesario realizar algunos cambios en el circuito de acceso dentro de los módulos y en el controlador de memoria de la placa base. Ya existen en el mercado tecnologías de memoria que permiten 4 accesos por ciclo. Teóricamente, serían posibles tecnologías aún más rápidas, pero sin embargo, cuanto mayor es la cantidad de datos transferidos por ciclo, mayor es el nivel de ruido electromagnético.

Posiblemente, la única desventaja de las memorias *DDR* sobre las memorias *SDRAM* es la incompatibilidad de los módulos. Al comprar una placa base que soporte memorias *DDR*, sólo podremos usar memorias *DDR* y viceversa. Existen algunas placas base híbridas, que traen, por ejemplo, dos soquetes para memorias *SDRAM* comunes y dos encajes para memorias *DDR*, permitiendo escoger el tipo de memoria que queremos utilizar.

En el momento de escribir este capítulo del libro existen dos tipos de memorias *DDR*, las *PC-1600* y las *PC-2100*, que trabajan respectivamente a 100 y 133 MHz, siempre con dos transferencias de datos por ciclo. Los números se refieren a las capacidades teóricas de transmisión de datos de los módulos, respectivamente de 1600 y 2100 MB/s.

4.6.5 La necesidad de las memorias DDR

El punto fuerte de las memorias *DDR* es el gran apoyo de la industria, que empezó a ganar fuerza después del lanzamiento de los *chipsets DDR* para el *Athlon* y para el *Pentium III*. El primer chipset para un *Pentium 4* que soporta memorias *DDR* es el i845, ya seguido de muchos otros.

La ganancia de rendimiento con el uso de memorias *DDR* varía en función de la aplicación que usemos. En aplicaciones que manipulan pequeñas cantidades de datos, como por ejemplo las aplicaciones generales de ofimática, los programas de compresión de audio y vídeo, búsqueda, etc. la ganancia es muy pequeña, ya que la cantidad de datos que necesita la aplicación pueden ser tratados fácilmente por un módulo *PC-133* común. Sin embargo, en aplicaciones que manipulan grandes cantidades de datos, como por ejemplo las grandes bases de datos, los juegos 3D y los programas de ingeniería o programas científicos presentan grandes ganancias de rendimiento con el uso de las memorias *DDR*. Esto teniendo en cuenta los procesadores actuales, un *Athlon*, un *Pentium 4*, etc.

Conforme los procesadores vayan evolucionando, el uso de memorias *DDR* traerá ganancias de rendimiento cada vez más tangibles, ya que se usarán multiplicadores cada vez más altos, aumentando el abismo entre la velocidad del procesador y la velocidad de la memoria.

Un ejemplo significativo es el *Celeron*. En sus primeras versiones a 333, 366, 400 MHz, el *Celeron*, aún usando un bus de 66 MHz, rivalizaba en rendimiento con los *Pentium II* de 350 y 400 MHz, que ya utilizaban un bus de 100 MHz, simplemente porque, a 366 MHz, el *Celeron* aún no era penalizado por la memoria más lenta.

Sin embargo, conforme fueron apareciendo versiones más rápidas del *Celeron*, el bus de 66 MHz comenzó a limitar severamente su rendimiento. Entonces, la casa Intel decidió lanzar al mercado el *Celeron* de 800 MHz usando un bus de 100 MHz. La diferencia de rendimiento entre un *Celeron* a 766 MHz (que usa bus de 66 MHz) y un *Celeron* a 800 MHZ (que usa bus de 100 MHz) llegaba a ser de casi un 20% en algunas aplicaciones, aunque la frecuencia del procesador es casi la misma.

Finalmente, el *Celeron* a 766 MHZ usa un increíble multiplicador de 11.5x, mientras que el *Celeron* a 800 MHZ usa un multiplicador de sólo 8x, perdiendo mucho menos tiempo para acceder a la memoria *RAM*.

4.6.6 Como funcionan las memorias Rambus

La gran diferencia entre un módulo de memoria *Rambus* y uno de memoria *DDR* y, justamente lo que permite a los módulos *Rambus* alcanzar frecuencias de trabajo altísimas, es el hecho de ser divididos en un número mucho mayor de bancos de memoria. Con más bancos es posible mantener más bancos transfiriendo datos al mismo tiempo, obteniendo frecuencias de trabajo más elevadas. Esto lleva a la segunda propiedad que es, al mismo tiempo, una ventaja y una desventaja: el hecho de que los módulos sólo transmiten 16 de datos bits en cada ciclo, mientras que los módulos de memoria *DDR* transmiten 64 bits.

Esta limitación surge en el transcurso de la frecuencia de trabajo más elevada. Si cada vía de datos realiza más transferencias de datos por segundo, consecuentemente también generará más ruido electromagnético, o sea, generará más interferencias. Con esto, las vías de datos necesitarán estar más distantes unas de las otras. Ya que el espacio en un módulo de memoria está muy restringido, la única solución fue disminuir el número de vías de datos, de 64 a sólo 16.

Por otro lado, esto puede ser una ventaja desde el punto de vista de los fabricantes de placas base, pues si los módulos de memoria poseen sólo 16 vías de datos, es preciso usar menos pistas en la placa base, lo que significa menos costes de desarrollo y producción. Existe, inclusive, la posibilidad de aumentar el número de vías, de 16 a 32 y acceder al mismo tiempo a dos módulos. Con esto se dobla la velocidad de acceso a la memoria. Cada módulo transmite los datos (teóricamente) a 1.6 GB/s. Entonces, dos módulos juntos transmitirán a 3.2 GB/s. Este sistema es usado en el *Pentium 4*, donde es necesario usar los módulos *Rambus* a pares.

El uso de más bancos de memoria conlleva un efecto colateral, que es la necesidad de usar un circuito de control más complejo y más caro. Este circuito de control es uno de los factores que hace que los módulos *Rambus* sean más caros y, además, es el culpable de los tiempos de latencia más elevados. Esta organización más compleja y el circuito de control más lento hacen que los accesos tarden mucho más tiempo. El primer acceso a la memoria tarda mucho más que en un módulo *DDR*, a pesar de que los siguientes tarden sólo un ciclo.

Esto explica que un *Pentium III* en una placa base con el *chipset* i820, que usa memorias *Rambus*, sea más lento en muchas aplicaciones que un *Pentium III* que utiliza memorias *PC-100* comunes. A pesar de transferir más datos por ciclo, los módulos *Rambus* tardan más en empezar a transferir los datos, lo que conlleva más ciclos de procesamiento desperdiciados.

Las aplicaciones que realizan varios accesos a la memoria transfiriendo un número pequeño de datos en cada acceso acaban presentando un rendimiento inferior del que presentarían con memorias *PC-100*, mientras que las aplicaciones que manipulan grandes cantidades de datos salen beneficiadas con el uso de las memorias *Rambus*.

4.7 ECC y paridad

Por buena que sea la calidad, todos los tipos de memoria son sensibles a errores, que pueden ser causados por incontables factores, que van desde las interferencias momentáneas a defectos físicos en los mismos módulos de memoria. Un dato adulterado puede causar diversos efectos colaterales.

Para aumentar el grado de fiabilidad de los sistemas se crearon métodos de diagnóstico y corrección de errores. Tal vez, en un equipo doméstico, un sistema de corrección de errores no sea muy importante, pues un error en la memoria causaría el bloqueo de la máquina como máximo. Sin embargo, en aplicaciones críticas, como en un banco, cualquier error de procesamiento puede causar graves perjuicios. Actualmente, los métodos usados para la detección de errores son la paridad y el *ECC* (*Error Correcting Code* o código de corrección de errores), que se basan en técnicas totalmente distintas.

La paridad es un método más antiguo, que solamente es capaz de identificar alteraciones en los datos depositados en las memorias, sin posibilidades de hacer cualquier tipo de corrección. La paridad consiste en la adición de un bit más en cada Byte de memoria, que pasa a tener 9 bits, teniendo este el último la función de diagnosticar alteraciones en los datos.

En la paridad, la operación de comprobación de los datos es bien simple: son contados el número de bits 1 de cada Byte. Si el número es par, el bit de paridad coge un valor 1 y en caso de que sea impar, el bit asume un valor 0. Cuando los datos son requeridos por el procesador, éstos son verificados por el circuito de paridad que verifica si el número de bits 1 corresponde al valor depositado en el 9 bit. Si se constata una alteración en los datos se envía un mensaje de error al procesador.

Este método no es 100% eficaz, pues no es capaz de detectar la alteración de un número de bits que consiga mantener la paridad. Por ejemplo, si dos bits cero se alteran y se vuelven bits uno, el circuito de paridad no notaría la alteración en los datos. Felizmente, es muy remota la posibilidad de que se produzca la alteración de dos o más bits al mismo tiempo.

El uso de la paridad no hace que el ordenador sea más lento, pues los circuitos responsables de la comprobación de los datos son independientes del resto del sistema. Su único efecto es el encarecimiento de las memorias, que en vez de tener 8 bits por Byte, pasan a tener 9.

Antiguamente, casi no se fabricaban memorias sin paridad. Las memorias *EDO* y *SDRAM* actuales, sin embargo, presentan un buen nivel de fiabilidad, lo que hace dispensable el uso de la paridad. De hecho, pocos fabricantes de memorias producen todavía memorias con el 9 bit.

Para los sistemas destinados a operaciones críticas, fue desarrollado el *ECC*, un método de diagnóstico muy eficiente al ser capaz de identificar los errores en los datos y corregirlos a través de algoritmos especiales. En una memoria con *ECC* encontramos 1, 2 o hasta 3 bits más por cada Byte de memoria. Cuanto mayor es la cantidad de bits destinados al *ECC*, más complejos serán los códigos almacenados y mayor será la posibilidad de corregir un eventual error.

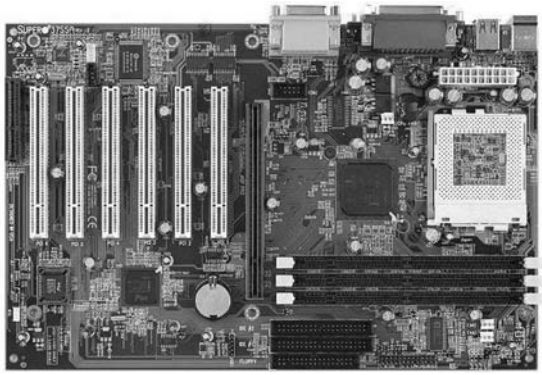
A pesar de no ser muy usado en memorias *RAM*, justamente debido a la buena fiabilidad de las memorias actuales, el *ECC* es un componente obligatorio en los discos duros y otros componentes, pues el deterioro de los datos es mucho más común.

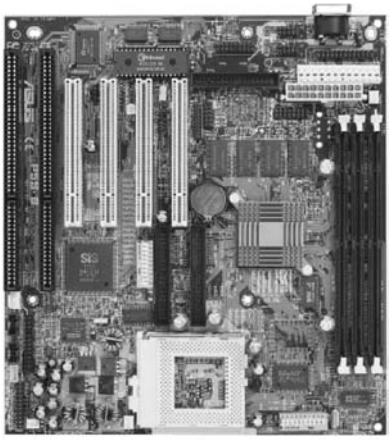
5. La placa base

La función de la placa base es suministrar los medios para que el procesador pueda comunicarse con los demás componentes del equipo con la mayor velocidad y eficacia posibles. El soporte para las nuevas tecnologías, las posibilidades de ampliación y, hasta cierto punto, el propio rendimiento del equipo están condicionados por las característica de la placa base de nuestro equipo.

5.1 Los formatos de placas base

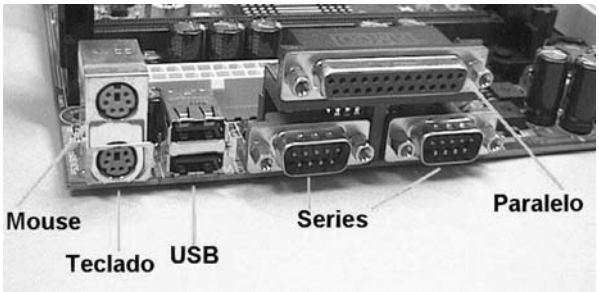
Es posible encontrar placas base con el formato *AT*, formato que se viene utilizando desde los primeros PCs de los años 80 y ya un poco anticuado, y con el formato *ATX*, más actual. Básicamente, los dos formatos se diferencian por el tamaño: las placas base del formato *ATX* son mayores, lo que permite a los fabricantes crear placas base con una disposición más racional de los distintos componentes, evitando que queden amontonados. En las siguientes imágenes podemos observar que la mala posición del procesador en la placa base *AT* impide que ensamblemos tarjetas de expansión más largas en algunos de los *slots PCI*. Las cajas (torres) para las placas base *ATX* también son más grandes, lo que además de garantizar una mejor ventilación interna hace más fácil su montaje y manipulación.





Una placa base ATX (arriba) y una placa base AT(derecha)

Otra ventaja es que en las placas *ATX* los puertos serie y paralelo, así como los conectores para el teclado, puertos *USB* y *PS/2* forman un panel en la parte trasera de la placa, minimizando los posibles problemas derivados de un mal contanto. Algunas placas base con el sonido y la tarjeta de red *onboard* también ofrecen los conectores para estos periféricos en la parte trasera.



Los conectores de la parte trasera de una placa base ATX

A pesar de que aún podemos encontrar a la venta placas base de ambos formatos, las placas *AT* ya son muy raras de encontrar, siendo completamente sustituidas por las placas base *ATX*.

5.1.1 Variaciones del formato ATX

Además del formato *ATX* tradicional, existen dos variaciones distintas del mismo formato, llamadas *micro ATX* y *WATX* (o *Wide ATX*). Estas dos variaciones difieren del formato *ATX* original sólo en el tamaño. El formato *micro ATX* es un formato menor de placa base, más o menos del tamaño de una placa base *AT*, que se usa en placas base de bajo coste. Generalmente, como estas placas incluyen pocos componentes, acaba saliendo más barato la producción de las placas base de un formato menor. Las placas base *micro ATX* pueden ser usadas sin problemas en torres *ATX* convencionales, pero una torre *micro ATX* no acepta una placa base del formato *ATX* tradicional, debido a su reducido tamaño.

Una placa base microATX

El formato *WATX*, por su parte, se usa en placas base destinadas a servidores. Generalmente, disponen de dos o cuatro encajes para procesadores y para varios módulos de memoria. Al poder disponer de muchos componentes, estas placas base son bastante mayores que las placas base *ATX* normales, pudiendo ser encajadas sólo en torres *WATX*.

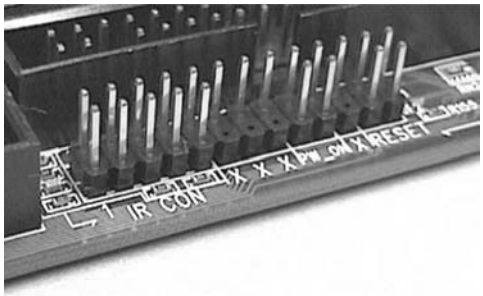
Una de las principales desventajas de las placas *micro ATX* es el hecho de que disponen de menos *slots* de expansión (generalmente 4 *slots*), distribuidos en 1 *slot AGP* y 3 *slots PCI*, o 1 *AGP*, 2 *PCI* y 1 *ISA*, mientras que las placas *ATX* disponen, normalmente, de 7 *slots*. Justamente por esto, el formato *micro ATX* es el usado en las placas base que ya vienen con la tarjeta de video y sonido *onboard*.

5.2 La fuente de alimentación

Además del tamaño y de la disposición más práctica de las conexiones de los puertos serie, paralelos, *PS/2* y *USB*, otra gran diferencia del formato *ATX* con el formato *AT* es la fuente de alimentación. Mientras que en el formato *AT* la fuente de alimentación se limita a enviar corriente, o interrumpir el suministro cuando se pulsa el botón de encendido, en el formato *ATX* se utiliza una fuente de alimentación más inteligente. La fuente de alimentación del formato *ATX* recibe órdenes directamente desde la placa base, lo que permite la introducción de nuevos recursos, como la posibilidad de desconectar el ordenador directamente desde el sistema operativo sin la necesidad de pulsar el botón de apagado, programar el ordenador para que se conecte o desconecte en un horario programado, etc.

Una fuente de alimentación con todos sus cables

El propio funcionamiento del botón de encendido y apagado de una torre *ATX* también es distinto. En primer lugar, el botón no está conectado a la fuente, como en el caso del formato *AT*, pero sí que está conectado al conector *ATX Power Switch*, un conector de dos contactos de la placa base, que queda próximo a los conectores de las luces del panel de la torre. También es distinto el comportamiento del botón cuando es pulsado. Estando el equipo encendido, un sólo toque en el botón hace que el equipo entre en modo de suspensión. Para cortar el suministro eléctrico de verdad, es necesario mantener el botón pulsado más de 4 segundos.



Los conectores de la placa base para el panel de la torre

Otra diferencia entre los dos formatos reside en el conector de la fuente de alimentación. En las placas base *AT*, el conector posee 12 contactos que reciben tensiones de 5 y 12 voltios de la fuente de alimentación. En las placas base *ATX*, el conector es un poco diferente, poseyendo 20 contactos y recibiendo una tensión de 3.3 voltios.

5.3 Los componentes de la placa base

Independientemente de su formato o modelo, básicamente siempre encontramos los mismos componentes en una placa base. Tenemos las ranuras de expansión (llamados *slots*) *ISA*, *PCI*, *AGP* y *AMR*, para el encaje de las tarjetas gráfica, de sonido, módems internos y otros periféricos, soquetes para el encaje de los módulos de memoria y también del procesador; puertos serie y paralelo, controladora de la unidad de disquetes, interfaces *IDE*, conectores para el teclado y para la fuente de alimentación, puertos *USB*, reguladores de tensión y, es claro, el *BIOS* y el chipset. Normalmente encontraremos un diagrama en las primeras páginas del manual de nuestra placa base. Este diagrama es útil a la hora de montar un ordenador, pues nos permite localizar fácilmente los encajes y *jumpers* de la placa base con facilidad.

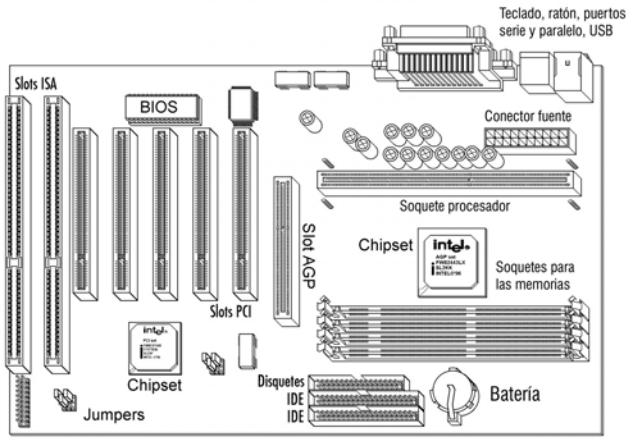


Diagrama de una placa base

La placa base propiamente dicha no está formada por una única placa de circuito impreso, sino que es, en realidad, un conjunto formado por varias placas prensadas. Cada una de estas placas contiene algunos de los contactos necesarios para conectar los distintos componentes, y en varios puntos tenemos los contactos que realizan la comunicación entre ellos.

Esta técnica llamada *MPCB* o *Multiple Layer Printed Circuit Board* (placa de circuito impreso con varias capas), exige una tecnología punta y desarrollar el proyecto de forma muy cuidadosa, pues un mínimo error en la posición de los componentes o contactos puede generar problemas eléctricos o interferencias, poniendo en riesgo toda la estabilidad del sistema. La calidad del proyecto y las técnicas de producción usadas en la fabricación de las placas de circuito impreso son algunos de los factores que diferencian las buenas placas base de las placas base más baratas, de inferior calidad.

5.3.1 Chipset

Con toda seguridad, el chipset es el componente más importante de la placa base, pues es el encargado de todo el flujo de datos entre el procesador, la memoria y los demás componentes del equipo. Los buses *ISA*, *PCI* y *AGP*, así como las interfaces IDE, puerto paralelo y los puertos serie están controlados por el *chipset*.

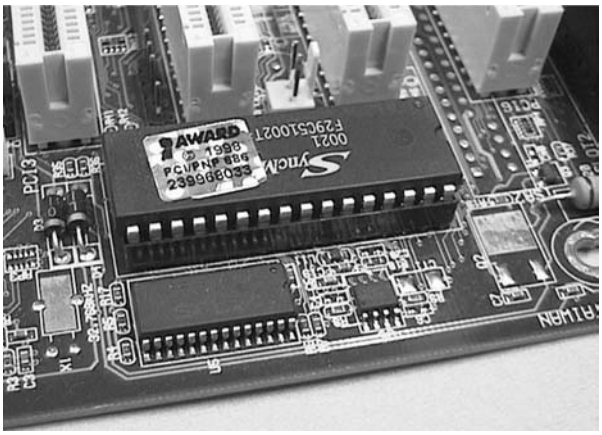
El chipset de la placa base

Internamente, el *chipset* está compuesto por varios *chips* pequeños, uno para cada función de las muchas que ejecuta. Tenemos un *chip* controlador para las interfaces *IDE*, otro para las memorias, etc. De ahí el nombre de *chipset*, o conjunto de *chips*.

Existen varios modelos de *chipsets*, cada uno con recursos diferentes. En las placas base con un *soquete 7*, por ejemplo, son utilizados los *chipsets* i430FX, i430HX, i430VX e i430TX, fabricados por la casa Intel, además de *chipsets* de otros fabricantes. En las placas *súper 7*, predominan los *chipsets* Apollo MVP 3, Aladdin V y Sis, mientras que en las placas base para *Pentium II* encontramos, básicamente, los chipsets i440FX, i440LX, i440EX e i440BX. Más adelante, y a lo largo de los demás capítulos, veremos más detallemente algunas de las distintas funciones de los diversos *chipsets*.

5.3.2 BIOS

BIOS significa *Basic Input Output System*, o en español, sistema básico de entrada y salida. El *BIOS* es la primera capa de software del sistema, un pequeño programa que tiene la función de iniciar el equipo. Durante el proceso de inicialización, el BIOS es el encargado de reconocer los distintos componentes de hardware instalados, dar la señal de arranque y proveer las informaciones básicas para el correcto funcionamiento del sistema.



Chip del BIOS en la placa base

El *BIOS* está grabado en un pequeño *chip* instalado en la placa base. Cada modelo de *BIOS* está personalizado para cada modelo específico de placa base, funcionando mal en un modelo inadecuadode placa base.

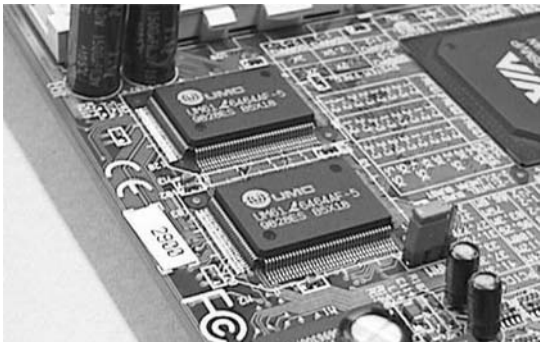
Cuando inicializamos el sistema, el *BIOS* realiza un cuenteo de la memoria disponible en el sistema, identifica los dispositivos *Plug & Play* instalados en el equipo y realiza una comprobación general de los diversos componentes instalados. Este procedimiento es llamado *POST* y está destinado a verificar si existe algún tipo de error con algún componente, además de verificar si ha sido instalado algún dispositivo nuevo. Solamente después del *POST*, el *BIOS* entrega el control del equipo al sistema operativo. En ese momento es cuando aparece el mensaje *Iniciando Windows XP*, o cualquier otro, dependiendo del sistema operativo instalado.

Después de terminar el *POST*, el *BIOS* emite un informe con la información sobre el hardware instalado en el ordenador. Este informe es una manera fácil y rápida de verificar la configuración del ordenador. Para paralizar la imagen el tiempo suficiente para conseguir leer el informe, basta con pulsar la tecla <Pausa> del teclado.

Si se encuentra algún problema durante el *POST*, el *BIOS* emite señales sonoras (pitidos), indicando que existe algún problema. Vamos a estudiar el *BIOS* con más profundidad y examinar las configuraciones del *Setup* más adelante, en el capítulo dedicado a la configuración del *BIOS*.

5.3.3 Memoria caché L2

La memoria caché se empezó a utilizar a partir de los equipos *386*. Inicialmente, la memoria caché formaba parte de la placa base, estando formada por algunos *chips* soldados en ella. A partir del *486*, integraron una pequeña cantidad de memoria caché en el propio núcleo del procesador, pero se continuó usando memoria caché en la placa base. Entonces, apareció la distinción entre la memoria caché *L1* integrada en el procesador y la memoria caché *L2* que formaba parte de la placa base.



Memoría caché en la placa base

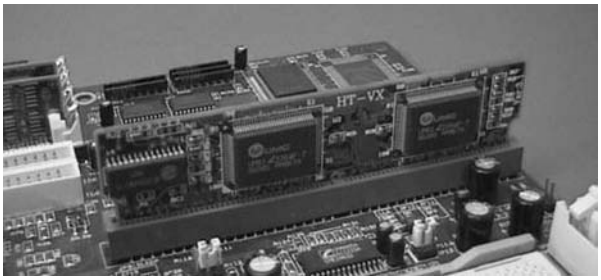
Con el desarrollo de las técnicas de producción los procesadores pasaron a utilizar multiplicadores cada vez mayores, haciendo que la memoria caché *L2* integrada en la placa base se volviese cada vez más ineficiente, ya que la memoria *RAM* trabajaba a 66 o 100 MHz, la misma frecuencia de la placa base, mientras el procesador operaba a una frecuencia mucho mayor.

En ese punto se produjo el segundo cambio en la historia de la memoria caché: integrar también la memoria caché *L2* en el procesador, lo que permitió a la memoria caché funcionar siempre a la mitad de la frecuencia del procesador (como en el *Pentium II*) o, incluso, integrar memoria caché capaz de trabajar a la misma frecuencia del procesador (como en el *Pentium III Coppermine*).

Como ya disponemos de la memoria caché suficiente en cantidad y velocidad, integrada en el procesador, ya no es necesaria su integración en la placa base. A excepción de las placas base con *soquete 7*, usadas en los procesadores *K6-2*, *K6-3* y los procesadores *soquete 7* antiguos, ningún modelo de placa base actual lleva memoria caché *L2*.

Sin embargo, cuatro o cinco años atrás, en la época de los *Pentium MMX* y AMD *K6* el escenario era bien distinto. A nadie se le pasaba por la cabeza comprar una placa base sin por lo menos 512 KB de memoria caché *L2*.

Algunas placas base un poco más antiguas, no venían con ningún tipo de memoria caché, pero en su lugar llevaban un encaje para un módulo *COAST* (*Cache On a Stick*). En este caso, el módulo de memoria caché debía ser adquirido de forma separada. Los módulos *COAST* eran difíciles de encontrar y también eran razonablemente caros.



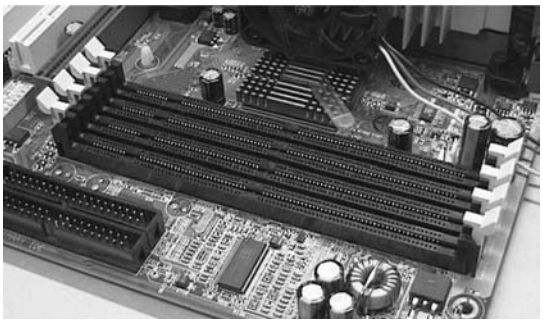
Un módulo COAST de memoria caché

A pesar de que ya no se fabriquen placas base con *soquetes* para módulos *COAST*, es posible que nos encontremos con una de ellas al trabajar en un equipo bastante antiguo. También existen casos de placas base con falsos *chips* de memoria caché. En vez de módulos de memoria caché encontramos soldados en la placa base encapsulados huecos, con la inscripción Write Back en bajo relieve. Durante el *POST*, la supuesta memoria caché también es identificada como *Write Back*, a pesar de no existir ninguna memoria caché.

Este tipo de trampa fue muy utilizada en las placas base más baratas, principalmente las fabricadas entre los años 94 y 97. Para reconocer una placa base de este tipo es suficiente con verificar si está estampada la inscripción *Write Back* en los módulos de memoria caché, o si la memoria caché es identificada en el informe del *POST* como *Write Back*.

5.3.4 Encajes para los módulos de memoria

El uso de módulos de memoria, en forma de módulos de *30*, *72* y *168* vías, facilita mucho la vida. En la época de los equipos *XT* y *286*, los *chips* de memoria tenían que ser ensamblados en la placa base uno a uno, lo que comportaba un enorme trabajo. Ahora, un módulo de memoria puede ser ensamblado en pocos segundos sin ninguna dificultad.



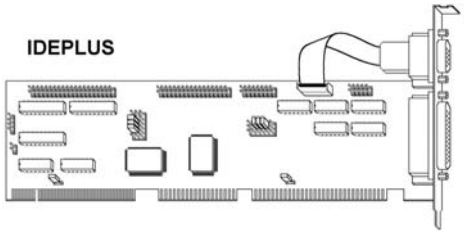
Soquetes para la memoria RAM

Los módulos de *30* y *72* vías ya se encuentran en desuso desde hace un buen tiempo. Actualmente, utilizamos sólo módulos de *168* vías de memoria *SDRAM* (y algunos

pocos de memoria *VC-SDRAM*), así como módulos de memoria *Rambus*. Muchas placas base también aceptan memorias *DDR-SDRAM* desde hace un tiempo.

5.3.5 Interfaces de disco

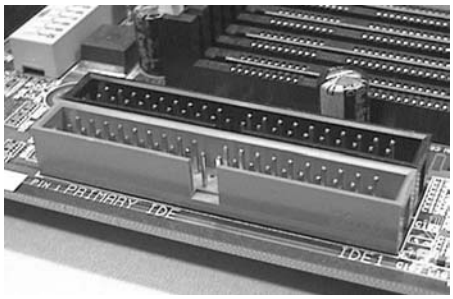
En placas base antiguas, las interfaces *IDE* y de la unidad de disquetes, así como los puertos paralelos y de impresora, formaban parte de una tarjeta llamada *SuperIDE*, que se conectaba a un *slot* disponible de la placa base. Existieron varios modelos de esas tarjetas o placas. Algunas traían sólo una interfaz *IDE*, otras traían dos. Existían tarjetas que utilizaban un slot ISA y otras que utilizan un *slot VLB*.



Una tarjeta de expansión SuperIDE

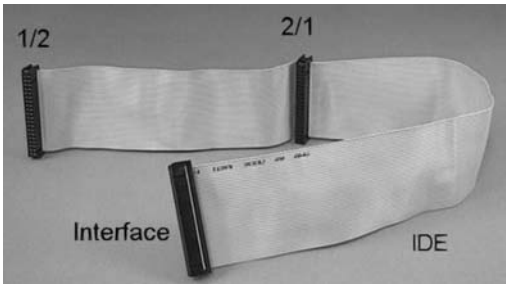
Usar una tarjeta separada para proveer recursos que todo el mundo necesitaba sólo servía para aumentar el precio de los equipos, y crear problemas en caso de un mal contacto. Por eso, a partir de finales de la era de los *486*, estos puertos pasaron a estar integrados en la propia placa base.

Cada interfaz *IDE* situada en la placa base permite la conexión de dos discos duros, unidades de *CD-ROM/DVD-ROM*, unidades *ZIP* o cualquier otro dispositivo *IDE*. Como tenemos dos puertos en cada placa base, podemos conectar hasta 4 dispositivos *IDE*. La controladora de disquetes permite la instalación de hasta dos unidades y, a pesar de que uno de los puertos serie puede estar ocupado por el ratón, aún nos queda un puerto serie libre para la conexión de un módem externo o cualquier otro dispositivo. A pesar de disponer de un único puerto de impresora, podemos compartirlo entre varios dispositivos con la ayuda de un conmutador, un dispositivo simple, que permite la conexión de 2 o 3 dispositivos en un único puerto, teniendo una llave que permite alternar entre uno y otro.



Conectores IDE de la placa base

Para conectar los discos duros y la unidad de disquetes a la placa base usamos cables *flat*. Los cables destinados a los discos duros poseen 40 vías, mientras que los destinados a la unidad de disquetes poseen sólo 32 vías. Cada cable posee tres conectores, siendo uno para la conexión a la placa base y los otros dos permiten la conexión de dos discos duros en cada interfaz. Los discos duros e interfaces *IDE UDMA 66* en adelante utilizan un cable *IDE* de 80 vías, donde 40 son usadas para datos y las otras 40 funcionan como hilos tierra, atenuando las interferencias. A pesar de que los cables *IDE* de 40 vías tradicionales funcionan en las interfaces I*DE UDMA 66*, su uso perjudica el rendimiento del puerto, ya que los cables antiguos no son adecuados para transferencias de datos a más de 33 MB/s.



Los cables flat para las interfaces IDE

También existen cables *IDE* con sólo 2 conectores (en este caso permitiendo el uso de un único disco duro), y cables para unidades de disquetes con 4 conectores además del conector de la placa base, siendo 2 para unidades de 3 y 1/2 y 2 para unidades de 5 y 1/4. En este caso, a pesar del mayor número de conectores, continúa existiendo la limitación de 2 unidades de disquete en cada puerto. Los cables *IDE*, el cable para la unidad de disquetes, así como los cables serie y paralelo vienen junto con la placa base.

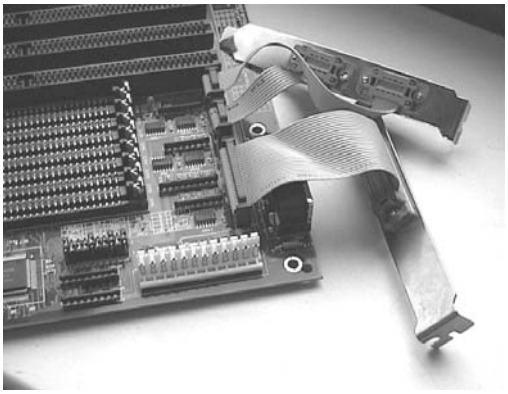
5.3.6 Puertos paralelo y serie

Tanto los puertos serie como los puertos paralelos, o de impresora, son puertos de comunicación que comparten el canal de datos del bus *ISA*. En principio, el funcionamiento de ambos puertos es muy parecido. Se usan algunos contactos para la transmisión de datos y otros para el control del flujo y la comprobación.

La principal diferencia es que en un puerto serie sólo se usa un contacto para la transmisión de datos, transmitiendo los bits de uno en uno, en serie, de ahí el nombre de puerto serie. En los puertos paralelos se usan ocho vías de datos, permitiendo el envío de 8 bits cada vez, lo que los hace mucho más rápidos que los puertos serie. En el caso de las placas base que no traen slots *ISA* y, consecuentemente no poseen este bus, los puertos se conectan directamente al bus *PCI*.

Los primeros puertos paralelos sólo eran capaces de transmitir datos, y no de recibir, restringiendo su uso a la conexión de impresoras. Posteriormente, se crearon otros for-

matos para los puertos paralelos, que además de ser más rápidos, permitieron la comunicación bidireccional, como el *ECP* y el *EPP*. Los puertos serie también evolucionaron. Los primeros puertos serie eran capaces de transmitir datos a una velocidad de sólo 9.600 bits por segundo, mientras que los más recientes pueden transmitir a hasta 115.000 bits por segundo.

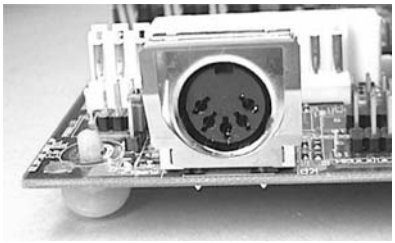


Puertos serie y paralelo en una placa base AT

Con excepción de las placas *ATX*, que poseen el panel de estos puertos en su parte posterior, también usamos cables *flat* en las salidas paralela y serie. En una extremidad tenemos el conector para ser ensamblado en la salida correspondiente de la placa base y en la otra una chapa metálica que debe ser fijada a la torre. En este caso, el cable sólo funciona como una extensión que facilita la conexión de los dispositivos. Los cables serie y paralelo, así como los cables IDE, vienen con la placa base, siendo los cables serie especialmente importantes, pues existen varias combinaciones de posiciones de los hilos internos en este tipo de cable, haciendo que muchas veces el cable serie de una placa no funcione en otra.

5.3.7 Conector del teclado

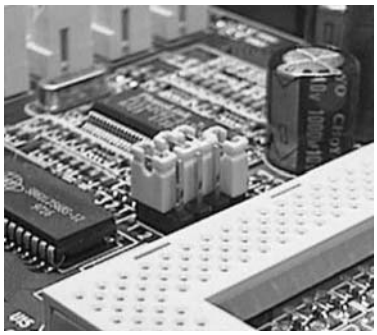
En placas base *AT* se utilizaba un conector *DIN* para la conexión del teclado. Ya en las placas base *ATX*, lo más común es el uso de un conector *mini-DIN*. A pesar de la diferencia en tamaño, ambos ensamblajes son eléctricamente compatibles, lo que permite el uso de un adaptador para la conexión de un teclado con encaje *mini-DIN* en un conector *DIN* y viceversa sin problemas.



El conector del teclado de una placa base

5.3.8 Jumpers

Los *jumpers* son pequeñas piezas plásticas, metalizadas internamente para permitir el paso de la corriente eléctrica, siendo ensamblados en contactos metálicos situados en la placa base o en varios otros tipos de placas y tarjetas. Los *jumpers* permiten el paso de la corriente eléctrica entre dos contactos, funcionando como una especie de interruptor.



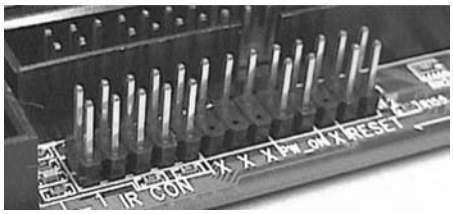
Jumpers de la placa base

Las distintas alternativas en la posición de anclaje de los *jumpers* permiten programar varios recursos de la placa base, como el voltaje, el tipo y la velocidad del procesador y de la memoria, además de otros recursos. Al montar un equipo, los *jumpers* de la placa base deben estar correctamente configurados, de lo contrario, podemos estropear algunos componentes en casos extremos.

En las placas base modernas, los *jumpers* son un componente más en vías de extinción, pues la mayoría de las placas actuales son *jumperless*, o sea, no poseen ningún *jumper*, realizando toda la configuración de las funciones de la placa base a través del *CMOS Setup* del *BIOS*.

5.3.9 Conectores para el panel de la torre

En una de las extremidades de la placa base encontraremos un conjunto de encajes que se destinan a la conexión de las luces y botones del panel frontal de la torre. Aquí se conectaban los botones *reset*, *turbo*, el *keylock* y los *leds* de corriente y funcionamiento del disco duro. No todas las placas base poseen estos conectores. Por ejemplo, la llave del teclado y el botón *turbo* ya no se usan, justamente porque ya no tienen ninguna utilidad.



Conectores de la placa base para el panel frontal

El botón *turbo* de la torre servía para aumentar/disminuir la frecuencia del procesador. Esto era bastante útil en la época de los equipos *XT* y *286*, donde algunos programas para ser ejecutados en equipos lentos sólo funcionaban bien con este botón presionado. Sin embargo, estos programas ya no se usan desde hace mucho tiempo y no existe ningún motivo para que alguien quiera ralentizar su equipo.

La llave del teclado (*keylock*) ya no se usa, ya que hoy existen medios menos anticuados para proteger el equipo de espías, como las contraseñas a nivel del *BIOS* y del mismo sistema operativo.

5.3.10 Seguimiento de funciones

Un recurso que viene siendo cada vez más utilizado en las placas base actuales es el seguimiento de algunas funciones, como la temperatura del procesador, la velocidad de rotación del ventilador, la temperatura del *chipset*, la corriente eléctrica enviada por la fuente de alimentación, etc. Los recursos soportados dependen de la placa base, pero el objetivo es siempre el mismo: prevenir el equipo ante posibles daños.

Si, por casualidad, el ventilador del procesador presenta algún defecto y el procesador empieza a sobrecalentarse, sonará una alarma alertándonos sobre el problema. Si, por casualidad, la fuente de alimentación empieza a enviar corrientes mayores, o menores, que las ideales para la placa base sonará la alarma avisándonos antes de que ocurra algo más grave.

Las informaciones suministradas por el monitor de seguimiento pueden ser vistas a través del *BIOS*, o desde el mismo *Windows* a través de un programa de seguimiento. Existen varios ejemplos de programas pero, casi siempre, las placas base que soportan el uso de este recurso traen un programa compatible grabado en el CD de los *drivers* que las acompañan.

El seguimiento se obtiene a través de la adición de un *chip* especial y de sensores de temperatura en la placa base. Existen varias soluciones disponibles y vamos a presentar algunos de los modelos de *chips* disponibles en el momento de la redacción del libro:

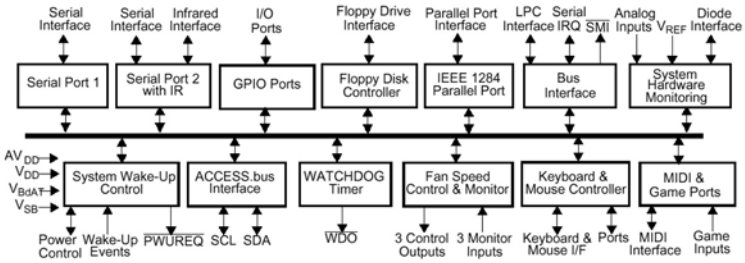
Winbond 83781D: entre los chips en uso este es el más simple, pero todavía es usado por varios fabricantes. El 83781D permite monitorizar 7 tensiones (5 positivas y 2 negativas), 3 sensores de temperatura (el posicionamiento de los sensores queda a cargo del fabricante de la placa base) y es capaz de monitorizar la rotación de hasta 3 ventiladores. La precisión de los sensores, así como su tiempo de respuesta, depende de la calidad de los sensores utilizados por el fabricante de la placa base.

Winbond 83782D: permite monitorizar 9 tensiones, 3 sensores y 3 ventiladores. Su principal ventaja sobre el 83781 es que permite monitorizar la temperatura del procesador a través del diodo térmico colocado en todos los procesadores Intel, a partir del *Pentium II* de 350 MHz. Este diodo es más sensible que la mayoría de los sensores usa-

dos en las placas base, y conlleva la ventaja adicional de ya venir colocado en el procesador, dispensando el uso de componentes adicionales.

Winbond 83783S: esta es la versión más barata. Tiene los mismos recursos que el 83782D, con excepción de que sólo monitoriza 3 tensiones, contra las 9 del anterior.

Winbond 83627HF: este es un circuito *super I/O* con recursos de seguimiento semejantes a los del 83782D. El circuito *super I/O* está en el *chip* de la placa base que controla los puertos serie y paralelo, así como la controladora de la unidad de disquetes. Algunos *chipsets*, como el Via Apollo Pro 133 traen el *chip súper I/O* incorporado, pero en la mayoría de los casos, el fabricante de la placa base necesita comprar el *chip* por separado. En este caso, un *chip super I/O* con recursos de seguimiento es una buena opción para abaratar costes.



El diagrama de un chip super I/O

Winbond 83697HF: una versión de bajo coste del 83627HF, con menos recursos pero con un precio menor.

5.4 Buses

Los buses son puertas a través de las cuáles el procesador puede comunicarse con los demás componentes del equipo, como las tarjetas gráficas. Hablando de las tarjetas gráficas, podemos notar que todas las tarjetas gráficas modernas son conectadas en los *slots AGP* y que tarjetas de sonido y módems antiguos casi siempre usan los slots ISA.

Esto es así porque las tarjetas de sonido y los módems son periféricos relativamente lentos, para los cuales el lento bus *ISA* ya es suficiente. Sin embargo, las tarjetas gráficas necesitan un bus mucho más rápido, motivo por el cual utilizan un *slot AGP*.

5.4.1 ISA

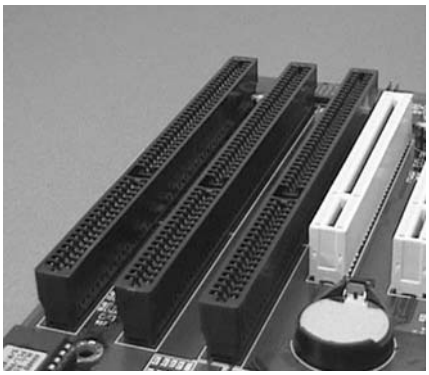
Los procesadores *8088*, usados en los equipos *XT*, se comunicaban con los demás periféricos usando palabras binarias de 8 bits. Para trabajar de forma conjunta con estos procesadores fue creado el bus *ISA* de 8 bits. Este bus funcionaba usando palabras binarias de 8 bits y trabajaba a una frecuencia de 8 MHz, permitiendo una transferen-

cia de datos a una velocidad de 8 MBytes por segundo, velocidad más que suficiente para un procesador tan lento como el *8088*.

5.4.2 ISA de 16 bits

Los procesadores *286* ya se comunicaban con los demás periféricos usando palabras de 16 bits. Para acompañar esta mejora del procesador se creó una extensión para el bus *ISA* de 8 bits, formando el *ISA* de 16 bits. Este nuevo bus, así como el procesador *286*, trabajaba con palabras de 16 bits a una frecuencia de 8 MHz, permitiendo una transferencia total de 16 MB/s.

Los periféricos *ISA* se usan desde la época del *286*, pero en realidad, este patrón ya existe desde 1981. La verdad es que el *ISA* ha durado tanto tiempo porque el bus de 16 MBytes por segundo era suficiente para asumir periféricos lentos como los módems y algunas tarjetas de sonido, haciendo que los fabricantes de estos periféricos continuaran produciendo periféricos *ISA* prácticamente hasta hoy.



Slots ISA

Como existía una gran demanda por parte del mercado, los fabricantes no tenían otra alternativa que combinar slots ISA y *PCI* en sus placas base, lo que servía para aumentar los costes de producción. Con la popularización de los módems y las tarjetas de sonido PCI, finalmente se abrió el camino para poder enterrar el bus *ISA* de forma casi definitiva.

5.4.3 MCA

Con el surgimiento de los procesadores *386*, que trabajaban usando palabras binarias de 32 bits, se hizo necesario la creación de un bus más rápido que el *ISA* para el uso de periféricos más rápidos, como las tarjetas gráficas. Entonces, la casa IBM creó el bus *MCA*, que funcionaba con palabras de 32 bits y a una frecuencia de 10 MHz, siendo 2.5 veces más rápido que el bus *ISA* de 16 bits.

Sin embargo, el bus *MCA* tenía un pequeño inconveniente al ser patentado por la casa IBM, y era que solamente ella podía usarlo en sus ordenadores. Los demás fabrican-

tes, sin otra elección, se vieron obligados a producir equipos con procesadores *386* equipados con sólo slots *ISA*.

Esto era un gran inconveniente, pues a pesar de que *386* era un procesador increíblemente rápido para la época, acababa siendo subutilizado por el lento bus *ISA*, ya que todos los demás componentes eran accedidos a la velocidad del bus *ISA*. A pesar de que poseían un procesador rápido para la época (el *386*), estos equipos eran tan rápidos como un *286* en términos de velocidad de acceso a los discos y demás componentes. Estos equipos eran llamados *AT 386*, pues a pesar de que utilizaban procesadores *386*, también utilizaban el mismo tipo de bus utilizado por los equipos *AT 286*.

A pesar de ofrecer recursos sorprendentes para la época, como el *Bus Mastering* y soporte para el *Plug & Play* (fue el primero bus que soportaba estos recursos), el bus *MCA* no consiguió popularizarse debido a su alto coste, a su incompatibilidad con el bus *ISA* y, principalmente, por ser una arquitectura cerrada, cayendo en desuso con el surgimiento de los buses *EISA* y *VLB*.

5.4.4 EISA

Este nuevo bus fue la respuesta de los demás fabricantes al bus *MCA*. Con el objetivo de ser compatible con el bus *ISA*, el nuevo bus *EISA* también funcionaba a 8 MHz, aunque sin embargo, trabajaba con palabras binarias de 32 bits, totalizando una tasa de transferencia de 32 MB/s, siendo dos veces más rápido que su antecesor. El bus *EISA* también ofrecía soporte al *Bus Mastering* y al *Plug & Play*, con la misma eficiencia que la del bus *MCA*.

Una de las grandes preocupaciones de los fabricantes durante el desarrollo del bus *EISA* fue mantener la compatibilidad con el bus *ISA*. El resultado fue un *slot* con dos líneas de contactos, capaz de acomodar tanto tarjetas *EISA* como tarjetas *ISA* de 8 o 16 bits. Una tarjeta *EISA* utilizaba todos los contactos del *slot*, mientras que una tarjeta *ISA* sólo utilizaba la primera línea de contactos. Naturalmente, el bus *EISA* era capaz de reconocer si la tarjeta instalada era *ISA* o *EISA*.

La complejidad del bus *EISA* conllevó un alto coste de producción, lo que dificultó su popularización. De hecho, pocas placas base llegaron a producirse con slots *EISA* y pocas tarjetas de expansión se desarrollaron para este bus. Así como el *MCA*, el bus *EISA* es un bus desaparecido en la actualidad.

5.4.5 VLB

Lanzado en 93 por la Video Electronics Standards Association (una asociación de los principales fabricantes de tarjetas gráficas), el bus *VLB* era mucho más rápido que el bus *EISA* o el bus *MCA*, siendo utilizado por las tarjetas gráficas y las controladoras de disco, las principales perjudicadas por los buses lentos. Con el bus *VLB*, los discos duros podían comunicarse con el procesador usando toda su velocidad y se posibilitó la creación de tarjetas gráficas mucho más rápidas.

Como antes, existió la preocupación de mantener la compatibilidad con el bus *ISA*, de modo que los *slots VLB* son compuestos por 3 conectores. Los dos primeros son idénticos a un *slot ISA* común, pudiendo ensamblar en ellos una tarjeta *ISA*, mientras que el tercero estaba destinado a las transferencias de datos a las altas velocidades permitidas por el bus *VLB*.

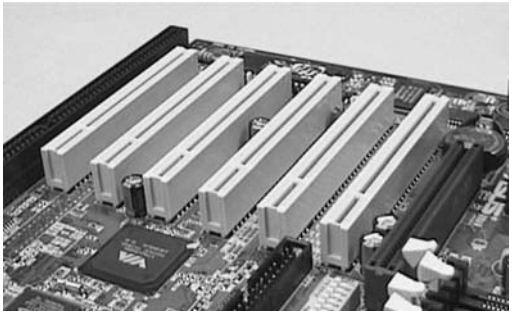
El bus *VLB* funcionaba a la misma frecuencia de la placa base, o sea, en un *486 DX-2 50*, donde la placa base funcionaba a 25 MHz, el *VLB* también funcionaba a 25 MHz. En una placa base para un *486 DX-4 100* que funcionaba a 33 MHz, el *VLB* también funcionaba a 33 MHz. Es necesario recordar que el *VLB* era un bus de 32 bits.

Las desventajas del *VLB* eran la falta de soporte al *Bus Mastering* y *Plug & Play*, además de unas altas tasas de utilización del procesador y limitaciones eléctricas, que sólo permitían un máximo de 2 o 3 *slots VLB* en cada equipo. Esto no llegaba a ser una gran limitación, pues generalmente sólo se utilizaban una tarjeta gráfica y una tarjeta *Super-IDE* con este bus.

Debido al alto rendimiento, a un bajo coste y, principalmente, debido al apoyo de la mayoría de los fabricantes, el *VLB* se hizo rápidamente un formato de bus para placas base *486*. Como el *VLB* fue desarrollado para trabajar en conjunto con los procesadores *486*, no llegaron a ser desarrolladas placas base para procesadores *Pentium* con este bus, pues la adaptación generaba demasiados costes, además de la existencia de problemas de incompatibilidad.

5.4.6 PCI

Creado por la casa Intel, el bus *PCI* era tan rápido como el bus *VLB*, sin embargo era más barato y mucho más versátil. Otra ventaja era que al contrario del bus *VLB*, el bus *PCI* no era controlado por el procesador y sí por una controladora dedicada, incluida en el *chipset*. Además de disminuir la utilización del procesador, esto permitía que el bus *PCI* fuese utilizado con cualquier procesador, sin implicar ninguna modificación.



Slots PCI

En una placa base *soquete 7*, el bus *PCI* funcionaba a la mitad de la frecuencia de la placa base, pudiendo funcionar a 25 MHz, 30 MHz o 33 MHz, dependiendo de la frecuencia de bus utilizada por la placa base. Por ejemplo, funcionando a 33 MHz el bus

PCI permitía una transferencia de datos a 133 MB/s, permitiendo 120 MB/s a 30 MHz y 100 MB/s funcionando a 25 MHz. En un *Pentium 75*, donde la placa base funcionaba a 50 MHz, el bus *PCI* funcionaba a 25 MHz; en un *Pentium 120* funcionaba a 30 MHz y en un *Pentium* 100, 133, 166, 200 o 233 funcionaba a 33 MHz. En las pocas placas base para procesadores *486* equipadas con *slots PCI*, el bus *PCI* trabajaba a la misma frecuencia que el bus de la placa, o sea: 25, 33 o 40 MHz.

En el caso de las placas base que trabajaban a 100 MHz, la frecuencia del bus *PCI* era de 1/3 de la frecuencia de la placa base, es decir 33 MHz. Incluso en las placas base que trabajaban a 133 MHz, el bus *PCI* mantenía sus 33 MHz, funcionando a 1/4 de la frecuencia de la placa base.

Además del bajo coste y de la alta velocidad, el bus *PCI* poseía otras ventajas como el soporte al *Plug & Play*: los nuevos periféricos instalados en los *slots PCI* eran reconocidos y configurados de forma automática a través del trabajo conjunto del *BIOS* y de un sistema operativo con soporte para *Plug & Play*, como *Windows 95/98* y posteriores. Posteriormente, todos los periféricos más rápidos, tarjetas gráficas y controladoras de disco duro usaron casi obligatoriamente el bus *PCI*. Los componentes más lentos, como las tarjetas de sonido y los módems podían encontrarse en versiones *ISA*, a pesar de que cada vez más encontrásemos estos componentes en versiones *PCI*.

Bus Mastering

A pesar de que los buses *MCA* y EISA ya ofrecían soporte para el *Bus Mastering*, la popularización de este nuevo recurso se produjo con la aparición del bus *PCI*. El *Bus Mastering* es un recurso capaz de aumentar el rendimiento general del sistema, al permitir que los dispositivos visiten directamente la memoria *RAM*, mejorando la velocidad de las transferencias y dejando libre al procesador para que pueda ejecutar otras tareas. El *Bus Mastering* fue implementado de una forma bastante completa en el bus *PCI*, haciendo posible que varios componentes utilicen el bus de forma simultánea, dividiendo los recursos del sistema.

Como las interfaces *IDE* también están conectadas al bus *PCI*, podemos utilizar el *Bus Mastering* en conjunto con los discos duros, haciendo que la tasa de utilización del procesador disminuya durante las transferencias de datos. A pesar de que la tasa de transferencia del disco duro no aumentó, el sistema quedaba más libre y ya no se quedaba casi paralizado durante los accesos al disco duro. *Windows 98* ya era capaz de instalar automáticamente los *drivers* de *Bus Mastering* tanto para discos duros *Pio Mode 4* como *UDMA*. Generalmente, los *drivers* del *Bus Mastering* están incluidos en el CD de *drivers* que acompaña a la placa base, también pudiendo encontrarlos en la *Web* del fabricante de la placa.

Plug & Play

Traducido al pie de la letra, *Plug & Play* significa conectar y usar. El objetivo de este recurso es hacer que el equipo sea capaz de reconocer y configurar automáticamente

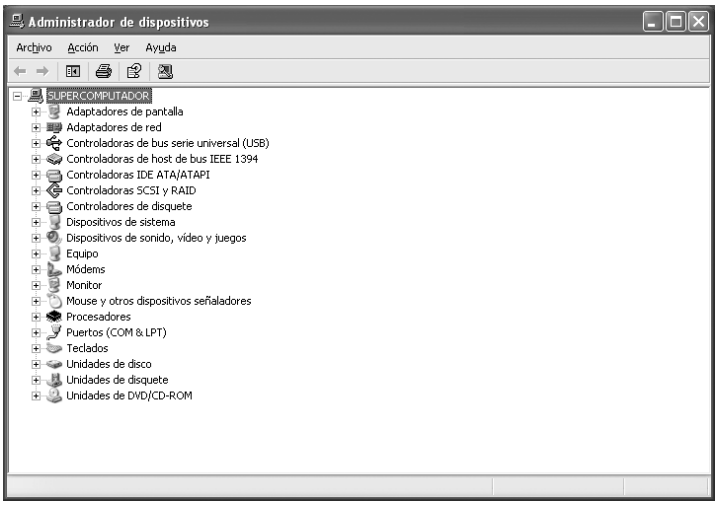
cualquier periférico instalado, reduciendo el trabajo del usuario a prácticamente sólo al ensamblaje del nuevo componente.

A pesar de ser una idea antigua se popularizó solamente hace pocos años. La dificultad es que además de un bus compatible también es necesario el soporte por parte del *BIOS*, del sistema operativo y también por parte del periférico para que todo funcione correctamente.

El proceso empieza durante la inicialización del equipo. El *BIOS* envía una señal de interrogación a todos los periféricos instalados en el equipo. Un periférico *Plug and Play* es capaz de responder a esta señal, permitiendo al *BIOS* reconocer los periféricos *Plug and Play* instalados.

El paso siguiente es la creación de una tabla con todas las interrupciones disponibles y realizar la asignación de cada una de ellas a los dispositivos. El sistema operativo entra en acción inmediatamente después, debiendo ser capaz de trabajar de forma conjunta con el *BIOS*, recibiendo la información sobre la configuración del sistema y suministrando todo el software de bajo nivel (en forma de *drivers* de los dispositivos) necesario para que los dispositivos puedan ser utilizados por los programas.

La información sobre la configuración de la distribución de los recursos entre los distintos periféricos es grabada en un área del *CMOS* llamada de *ESCD*. Tanto el *BIOS* (durante el *POST*) como el sistema operativo (durante la inicialización), leen esta lista, y en caso de no encontrar ningún cambio en el hardware instalado, mantiene sus configuraciones. Esto permite que el sistema operativo (desde que es compatible con el *Plug and Play*) pueda alterar las configuraciones en caso necesario. A partir de *Windows 95/98*, el propio usuario podía cambiar las configuraciones del sistema a través del *Administrador de dispositivos*, una vez encontrado en el icono *Sistema* dentro del *Panel de control*.

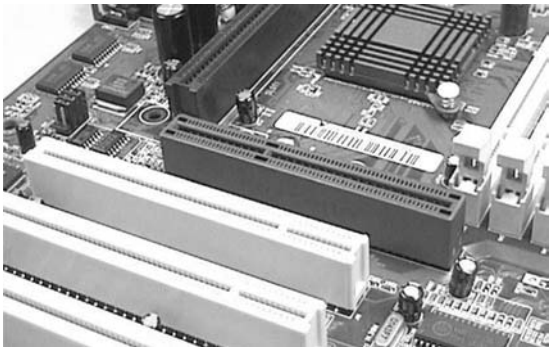


Los dispositivos instalados en el sistema en el Administrador de dispositivos

5.4.7 AGP

El *AGP* es un bus relativamente nuevo, hecho a la medida para las tarjetas gráficas más modernas. El bus *AGP* fue creado en base a las especificaciones del *PCI 2.1* y trabaja al doble de la velocidad del bus *PCI*, o sea, a 66 MHz, permitiendo transferencias de datos de 266 MB/s, contra los 133 MB/s permitidos por el bus *PCI*.

Además de la velocidad, el bus AGP permite que una tarjeta gráfica pueda acceder directamente a la memoria *RAM* para almacenar texturas. Este es un recurso muy utilizado en las tarjetas 3D, donde la tarjeta gráfica usa la memoria *RAM* para almacenar las texturas que son aplicadas sobre los polígonos que componen la imagen tridimensional. A pesar de que esto también es posible utilizando el bus *PCI*, en este caso los datos tendrían que pasar por el procesador, perjudicando el rendimiento general del equipo. Originalmente, el bus *AGP* fue concebido para equipar las placas base para *Pentium II* y *Pentium III*. Sin embargo, muchos fabricantes también lo usaron en las placas base *soquete 7* y *slot A*.



Slot AGP

Es importante no confundir bus con *slot*. Por ejemplo, en una placa base generalmente tenemos 3 o 4 *slots PCI*. Sin embargo, todos ellos comparten el mismo bus de 133 MB/s. El bus es la carretera que permite la comunicación con el procesador, y es compartido por todos los periféricos conectados a este bus. Los *slots* sólo son los medios de conexión.

Los 16 MB/s del bus *ISA*, por ejemplo, son compartidos por todos los periféricos conectados en los slots ISA, por los puertos serie y paralelo y por la controladora de disquetes. El bus *PCI* es compartido por todos los periféricos *PCI*, por las interfaces *IDE* y también por las controladoras *SCSI* que estén conectadas en los *slots PCI*.

Sin embargo, el bus *AGP* sólo es utilizado por la tarjeta gráfica, lo que en el caso de tarjetas rápidas, como las tarjetas 3D, acaba marcando una diferencia importante de rendimiento. Si tenemos varios discos duros en un mismo equipo, equipado con una tarjeta gráfica rápida, los 133 MB/s del bus *PCI* acaban siendo insuficientes, perjudicando el rendimiento de los periféricos conectados al bus. En este caso, el uso de una tarjeta gráfica *AGP* es muy recomendable.

A pesar de que el bus *AGP* también puede ser utilizado por las tarjetas gráficas 2D, su uso no conlleva ninguna ventaja, pues estas tarjetas no usan la memoria *RAM* para almacenar texturas y no son lo suficientemente rápidas para sacarle el adecuado provecho a la mayor velocidad del bus *AGP*.

Además del *AGP* estándar tenemos también el bus *AGP 2x*, donde, a pesar de que el bus continúa trabajando a 66 MHz, se realizan dos transferencias de datos en cada ciclo de reloj, equivaliendo en la práctica, a una frecuencia de 133 MHz, lo que equivale a una tasa de transferencia de nada menos que de 533 MB/s. Los nuevos chipsets traen soporte para el *AGP 4x*, que mantiene los 66 MHz del bus *AGP* y *AGP 2x*, pero permite cuatro transferencias de datos en cada ciclo de reloj, lo que corresponde a una frecuencia de 266 MHz en la práctica, resultando una tasa de transferencia de 1066 MB/s, más que el bus actual entre la memoria y el procesador.

5.4.8 AGP Pro

El bus *AGP Pro* es en realidad un *slot AGP 4x* con 48 contactos más, 20 en un lado y 28 más en el otro. Estos contactos adicionales son usados para aumentar la capacidad de suministro eléctrico del *slot*.

Existen dos tipos de *slots AGP Pro*: el *AGP Pro50* y el *AGP Pro110*. El nombre indica la capacidad de suministro eléctrico de ambos formatos: el *AGP Pro50* está certificado para suministrar hasta 50 vatios, mientras que el *AGP Pro110* puede suministrar hasta 110 vatios. Los *slots AGP Pro* aún no son muy comunes, pero terminarán convirtiéndose en un estándar dentro de poco tiempo, ya que muchas tarjetas gráficas sólo vendrán en este formato y no podrán ser usadas en placas base con los *slots AGP* comunes.

5.4.9 USB

Poco tiempo atrás sólo podíamos contar con los puertos serie y paralelo para la conexión de los dispositivos externos, como impresoras y ratones. Pero, disponiendo sólo de dos puertos serie y un puerto paralelo teníamos unos recursos de expansión bastante limitados. Además de eso, la velocidad de estas interfaces deja bastante que desear en la actualidad.

El *USB* fue la tentativa de crear un nuevo patrón para la conexión de periféricos externos. Sus principales armas son la facilidad de uso y la posibilidad de conectar varios periféricos en un mismo puerto *USB*. Con excepción del *PCMCIA*, el *USB* es el primer bus para los equipos PC realmente *Plug & Play*. Podemos conectar los periféricos con el equipo encendido, siendo suficiente suministrar el *driver* del dispositivo para que todo funcione, sin tener que reiniciar el equipo. La controladora *USB* también es suficientemente inteligente para percibir la desconexión de un periférico.

A partir del *chipset* i430VX (lanzado en el año 96) todos los *chipsets* ya ofrecen soporte para el bus *USB* y, prácticamente, todas las placas base disponen de dos puertos *USB* como mínimo.

Topología

Podemos conectar hasta 127 periféricos en fila en un único puerto *USB*, o sea, conectando el primero periférico a la salida *USB* de la placa base y conectando los demás al anterior de la fila.

La controladora *USB* del equipo es el nodo raíz del bus. A este nodo principal podemos conectar otros nodos llamados *hubs*. Un *hub* no es más que un concentrador que dispone de más conexiones, siendo 7 el límite por *hub*. El *hub* posee permiso para suministrar más niveles de conexión, lo que permite conectar más *hubs* al primero, hasta alcanzar el límite de 127 periféricos permitidos por el puerto *USB*.

Existen dos tipos de conectores USB, llamados conector A y conector B. El conector A es el conector usado para la conexión al puerto de la placa base, mientras que el conector B es el utilizado para los periféricos.

Los conectores USB y el cable

Rendimiento

Cada puerto *USB* permite una tasa de transferencia de 12 Mbps, o cerca de 1.5 MB/s, cerca de 100 veces más que la permitida por un puerto serie, y un poco más que la permitida por un puerto paralelo *ECP*. El nuevo *USB 2.0* permite tasas de transferencia de hasta 60 MB/s.

Esta velocidad es suficiente para periféricos como impresoras, escáneres, unidades *ZIP* externas, módems e interfaces de red de 10 Mbps. El problema es que los 12 Mbps son compartidos entre todos los periféricos conectados al mismo puerto. Si queremos utilizar dos periféricos *USB* que consuman mucho ancho de banda, como una grabadora de CDs externa y una interfaz de red, sería importante instalar cada uno de ellos en un puerto distinto de la placa base.

5.4.10 PCMCIA

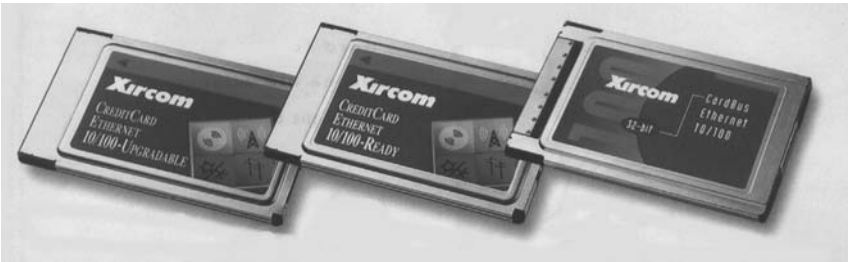
Este bus se utiliza principalmente en ordenadores portátiles y ordenadores de mano, donde la mayoría de las veces, es el único modo de poder conectar tarjetas de expan-

sión. La principal ventaja de los dispositivos *PCMCIA* es el tamaño: todos tienen unas dimensiones menores que las de una tarjeta de crédito, sólo son más gruesos. Actualmente, es posible encontrar cualquier tipo de dispositivo en forma de tarjeta PCMCIA: módems, tarjetas de sonido, tarjetas de red, tarjetas de memorias *SRAM* y memorias *Flash* e, incluso, discos duros removibles.

Existen 4 tipos de tarjetas *PCMCIA*, conocidas por tarjetas tipo 1, 2, 3, y 4. Las tarjetas tipo 1 fueron las primeras, son muy finas, más o menos del grosor de cuatro tarjetas de crédito apiladas. Este formato sólo se usa para tarjetas de memoria.

Las tarjetas del tipo 2 ya son las más comunes, tienen cerca de 7 mm de grosor, el suficiente para contener los circuitos de prácticamente cualquier tipo de periférico. Las tarjetas del tipo 3 son usadas, generalmente, para albergar disco duros removibles. Estos discos tienen el mismo funcionamiento que los discos duros *IDE* o *SCSI* aunque miniaturizados. Las tarjetas del tipo 3 tienen 1.5 cm de grosor aproximadamente.

Los tres formatos de tarjeta son ensamblados en el mismo *soquete PCMCIA* tipo 2. Este encaje acomoda perfectamente una tarjeta del tipo 2 y utiliza un sistema de resortes para poder acomodar tarjetas del tipo 1. Las tarjetas del tipo 3 son proyectadas para ensamblarse en un conjunto de dos encajes del tipo 2 (ocupando el espacio de ambas), la configuración de *slots* más común en los ordenadores portátiles. Estas tarjetas no se pueden ensamblar si el ordenador portátil, o de mano, sólo posee un encaje simple del tipo 2.



Tarjetas PCMCIA

Por su parte, la tecnología de las tarjetas tipo 4 fue desarrollada y patentada por la casa Toshiba, y esto significa que sólo ella puede utilizarla en sus productos.

Actualmente, se viene usando bastante también el término *PC-Card* en el lugar del término *PCMCIA*. Ambos términos son sinónimos, y si oímos hablar de una tarjeta *PC-Card* del tipo 2 por ejemplo, se trata de una tarjeta *PCMCIA* del tipo 2.

El bus *PCMCIA* es totalmente *Plug & Play*, como el bus *USB*. Usando un sistema operativo *Plug and Play*, como *Windows 98/2000/XP* o *Windows CE*, incluso podemos conectar las tarjetas *PCMCIA* con el equipo encendido y serán reconocidas automáticamente. Además de esto, los periféricos tienen la ventaja de gastar menos energía y de ser fácilmente transportables.

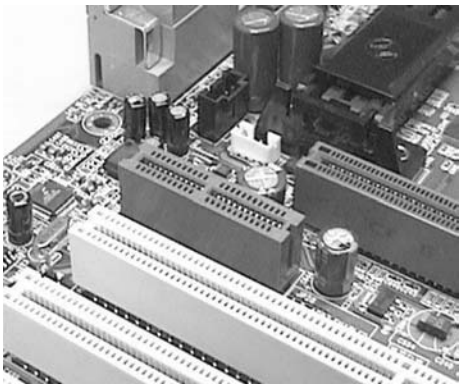
Generalmente, sólo encontraremos los *soquetes PCMCIA* en equipos portátiles, ya que a pesar de su practicidad, estos periféricos son extremadamente caros.

Otro tipo de periférico *PCMCIA* muy usado son las tarjetas de memoria. Éstas permiten almacenar y transportar datos, sirviendo como medio de almacenamiento removible. Las tarjetas pueden ser de memoria *Flash*, que conserva los datos grabados durante años sin problemas, como de memoria *SRAM*, que en este caso lleva incorporada una batería de reloj para mantener los datos. Ambos tipos de memoria son muy rápidos y se estan haciendo muy populares.

5.4.11 Slots AMR

Las siglas *AMR* son la abreviatura de *Audio Modem Riser*. Se trata de un formato de bus que permite el encaje de tarjetas de sonido y módems controlados vía software.

El *slot AMR* se parece a un *slot AGP*, pero sólo tiene 1/3 del tamaño de éste. El objetivo es permitir la creación de componentes extremadamente baratos para que sean usados en equipos de bajo coste.



Slot AMR

La ventaja es clara, el precio. La desventaja es el hecho de que estos componentes son controlados vía software, lo que consume bastantes recursos del procesador, haciendo que el equipo sea más lento. Usando, al mismo tiempo, un módem y una tarjeta de sonido *AMR* en un *Pentium III 800*, por ejemplo, la pérdida de rendimiento es de más del 10%.

Como el *chip* controlador se encuentra incorporado en el propio *chipset* de la placa base, las tarjetas de sonido y módems *AMR* contienen un número extremadamente reducido de componentes, básicamente las entradas y salidas de sonido, el *CODEC* y, en el caso de los módems, el *Relay* (el componente que permite el acceso a la línea telefónica).

A pesar de esto, el bus *AMR* no tuvo mucho éxito, pues no ofrece soporte al *Plug & Play*, lo que dificultaba la instalación de los dispositivos por parte de los usuarios y

sólo admitía tarjetas de sonido y módems, dejando aparte a las tarjetas de red y otros dispositivos.

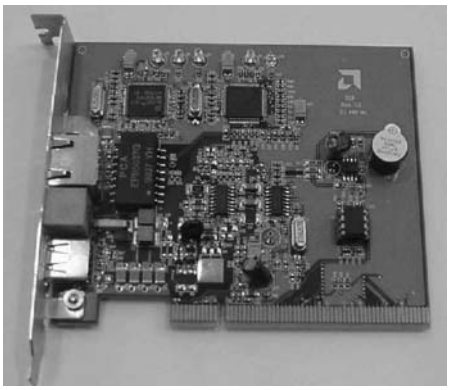
Para resolver estas limitaciones, varios fabricantes se reunieron para desarrollar el bus *ACR*, un formato abierto que sustituyera al bus *AMR* con más ventajas, manteniendo siempre el bajo coste.

5.4.12 ACR

Este bus es el sucesor del bus *AMR*, un formato desarrollado por la asociación de varios fabricantes. Los *slots ACR* se parecen a un *slot PCI* invertido. En realidad, los fabricantes optaron por aprovechar el mismo encaje para abaratar los costes, pero las semejanzas se terminan aquí, ya que cambian la posición y la señalización eléctrica del *slot*.

Los *slots ACR* son *Risers* para la conexión de tarjetas de sonido y módems de bajo coste, como los *slots AMR*. Muchas placas base actuales llevan un slot *ACR*, pero los fabricantes evitan desarrollar placas base con dos o más slots *ACR* para no disminuir el número de slots *PCI*.

La principal ventaja del bus *ACR* sobre el bus *AMR* es que el *Riser* del *ACR* permite la conexión de prácticamente todo tipo de dispositivos, desde módems y tarjetas de sonido controlados vía software, hasta tarjetas de red, módems *ADSL* o *RDSI*, tarjetas de sonido y módems controlados vía hardware, etc.



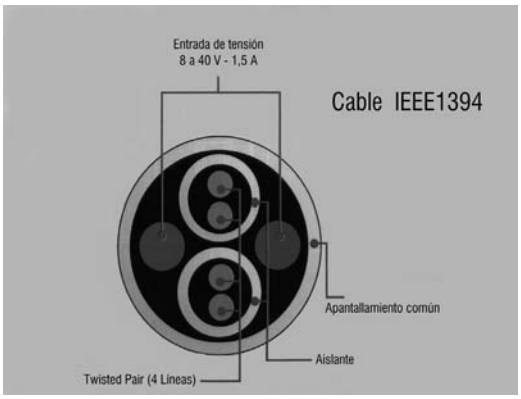
Una tarjeta ACR

5.4.13 IEEE 1394 (Fireware)

El *IEEE 1394* es un formato de interfaz relativamente nuevo que tiene varias características en común con el *USB*, pero con la ventaja de ser más rápido, permitiendo transferencias de datos a 400 MB/s, contra los 1.5 MB/s del USB. Este formato fue desarrollado por la casa Sony, que lo utiliza en varios aparatos de audio y vídeo, así como en algunos equipos portátiles. Un dato importante es que el *IEEE 1394*, o *Fireware* como es más conocido, es un formato abierto.

Los posibles usos para el *Fireware* son muchos, ya que puede ser utilizado para la conexión de cámaras digitales, impresoras, dispositivos de audio, creación de redes locales de alta velocidad e incluso para la conexión de discos duros externos.

La principal arma del *Fireware* es su simplicidad. Al tratarse de un bus serie, tanto las controladoras como los cables son muy baratos de producir. El cable utilizado está compuesto por sólo 3 pares de hilos, dos pares para la transferencia de datos y el último para el suministro eléctrico. Los conectores son pequeños, semejantes a los conectores *USB* y los *chips* controladores, que se encuentran incorporados en los periféricos, son baratos. Como en el caso del *USB* tiene soporte para la conexión en caliente*f*, o sea, es posible conectar y desconectar los periféricos con el equipo encendido.



Una sección del cable Fireware

La forma de conexión de los periféricos también es muy parecida con lo que tenemos en el *USB*. Tenemos la posibilidad de conectar hasta 63 periféricos en cada puerto *Fireware* con el uso de *hubs*, y cada segmento puede tener 4.5 metros como máximo. El bus *Fireware* también puede ser usado para interconectar varios equipos, siendo una opción barata de red local. En este caso, son permitidos 16 equipos como máximo, siempre que se respete el límite de los 4.5 metros por cable.

A pesar de los buenos pronósticos, todo lo que tenemos en el presente son muchas promesas. A buen seguro, el bus *Fireware* es un bus muy prometedor debido a sus diversas ventajas. Sin embargo, aún existe una cierta resistencia por parte de los fabricantes para adoptar este formato, debido a las grandes inversiones. Para la conexión de discos duros tenemos como opciones las interfaces *IDE* y *SCSI*; para la conexión de periféricos lentos, como impresoras y escáneres existe el *USB*; para la confección de redes locales ya existe también todo un conjunto de formatos en uso; lo que le deja poco espacio al *Fireware*.

Tanto *Windows 98* como *Windows 2000/XP* ofrecen el soporte necesario a los periféricos *Fireware*, pero aún existen pocas (y caras) opciones en el mercado.

5.4.14 Serial ATA

El *serial ATA* está desarrollado por la casa Intel y es un bus semejante al *Fireware*, pero sólo destinado a la conexión de discos duros.

Las interfaces *IDE* evolucionaron de una forma muy rápida en velocidad en la última década, pasando de los 3.3 MB/s permitidos por una interfaz *Pio Mode 1* a los 100 MB/s en el *UDMA 100*, aunque muy pocos creen que el formato pueda seguir evolucionando mucho más.

Entonces, el *serial ATA* sería el sucesor del *UDMA*, permitiendo transferencias de datos de hasta 800 MB/s, mucho más que lo necesario para cualquier disco duro que vaya a ser lanzado de momento.

Teóricamente, las interfaces y los discos duros del nuevo formato podrían ser más baratos que los anteriores, pues al tratarse de una interfaz serie, el *serial ATA* es más simple que las otras interfaces.

En vez de los cables de 80 vías usados por las interfaces *UDMA 66* y *UDMA 100*, el *serial ATA* utiliza cables con sólo dos pares de hilos, el primer par está destinado a la transmisión de los datos y el segundo para la alimentación eléctrica.

5.5 El presente de los buses

En 1984, la casa IBM apareció en el mercado con un ordenador revolucionario para la época, el *286*. Todavía los PC actuales conservan varias características de la época del *286*. Un ejemplo han sido los *slots ISA*, que felizmente van desapareciendo gradualmente, pero otras características antiguas todavía amenazan con continuar con nosotros durante algunos años: los puertos serie y paralelos.

Sí, incluso después del *USB*, los puertos serie y paralelos continúan siendo usados para periféricos externos. Los dos grandes problemas del *USB* ha sido la tasa de transferencia relativamente baja, 12 Mbps (o 1.5 MB/s) que equivalen a la velocidad de un puerto paralelo *ECP*.

Otro posible sustituto que ha ido encontrando dificultades es el *Fireware*, que ya es muy común en los Macs, pero que aún está encallado en los equipos PC. El *Fireware* es más rápido que el *USB*: 400 Mbps o 50 MB/s.

Recientemente, surgió un candidato más, el *USB 2.0*, desarrollado en una asociación entre las casas Intel, NEC, Philips, Lucent, Microsoft y Compaq. El *USB 2.0* resuelve los dos principales problemas del primer *USB*. En primer lugar, la velocidad ha pasado de los antiguos 12 Mbps a unos increíbles 480 Mbps o 60 MB/s, una velocidad muy próxima a la permitida por las interfaces *IDE*. La segunda ventaja es el coste: el *USB 2.0* es un formato abierto, libre de *royalties*, lo que es un gran estímulo para los distintos fabricantes de periféricos.

Con estas ventajas era de esperar que el *USB 2.0* sustituyese al *USB* rápidamente. De hecho, las primeras placas base con soporte para el *USB 2.0* salieron al mercado a principios del año 2002. El nuevo formato es compatible con todos los periféricos *USB* anteriores que trabajan a una menor velocidad de la permitida por el bus.

La gran evolución final del *USB* ha calado en el mercado y facilita bastante nuestra vida. Casi podemos jubilar los puertos serie y paralelos: lentos, limitados y que adoran entrar en conflictos con los demás periféricos del equipo.

Los primeros periféricos *USB 2.0* que llegaron al mercado fueron los escáneres de alta velocidad, grabadoras de CDs portátiles y unidades de almacenamiento en general, discos duros externos por ejemplo, seguidos por las impresoras, ratones y todo tipo de periféricos externos.

5.6 La petición de interrupción (IRQ)

Las direcciones *IRQ* son interrupciones de hardware, canales que los dispositivos pueden utilizar para llamar la atención del procesador. A pesar de que podamos ejecutar varios programas al mismo tiempo, los procesadores sólo son capaces de hacer una cosa cada vez. La multitarea surge de un intercambio muy rápido entre las aplicaciones abiertas, dando la impresión de que todas se están ejecutando al mismo tiempo.

Pero, qué pasa cuando el procesador está ocupado procesando cualquier dato y tecleamos algo en el teclado, el módem necesita transmitir datos al procesador, o cualquier cosa parecida? En este punto entran en juego las direcciones *IRQ*. El procesador, al ser avisado a través de uno de estos canales, detiene inmediatamente lo que esté haciendo y le presta atención al dispositivo, volviendo al trabajo interrumpido inmediatamente después.

Si no fueran por las direcciones *IRQ*, el procesador no sería capaz de leer las letras tecleados, ni las pulsaciones del ratón, etc. La mala noticia es que una misma *IRQ* no puede compartirse entre dos dispositivos y sólo existen 16 direcciones disponibles, que no pueden ser expandidas, o sea, tenemos que conformarnos con lo que tenemos (actualmente muchas placas *PCI* pueden compartir la misma IRQ pero, en realidad, sólo un dispositivo, el controlador *PCI*, usa el canal de interrupción).

Pero mirándolo por el lado bueno, podría ser peor y ya lo fue. En el XT de 16 bits sólo existían 8 direcciones *IRQ*, numeradas del 0 al 7. La explicación es simple, el *XT* usaba el procesador *8088*, que a pesar de ser internamente un procesador de 16 bits, usaba un bus de sólo 8 bits para comunicarse con los periféricos. Con esto, sólo tenía 8 *IRQs* y en un *XT*, las 8 direcciones *IRQ* disponibles quedaban generalmente configuradas de la siguiente manera:

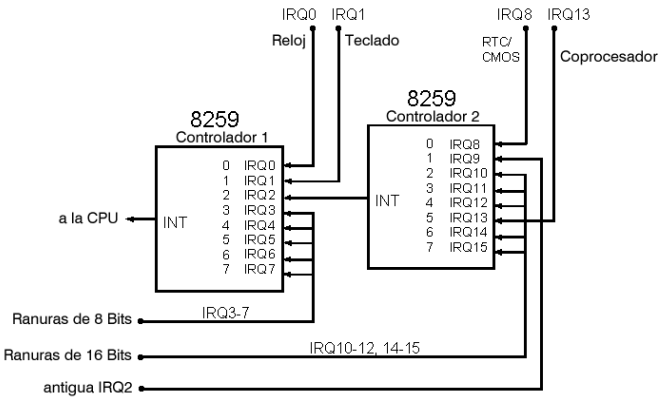
0 Señal del reloj de la placa base

1 Teclado

2 Libre
3 COM 2
4 COM 1
5 Disco duro
6 Unidad de disquetes
7 Puerto paralelo

El número de la dirección *IRQ* también indica su prioridad, comenzando del 0 que es la que tiene la prioridad más alta. No es sin ton ni son que la *IRQ* 0 esté ocupada por la señal del reloj de la placa base, pues es éste el encargado de sincronizar el trabajo de todos los componentes, inclusive el procesador. Inmediatamente después aparece el teclado, que ocupa la *IRQ* 1. Notemos que el teclado es el dispositivo con el nivel de prioridad más alto, para evitar que se pierdan las teclas pulsadas. Esto puede parecer innecesario ya que un procesador actual procesa billones de operaciones por segundo y difícilmente alguien teclea más de 300, o tal vez 400, teclas por minuto, pero, en la época del *XT*, las cosas no eran así de rápidas. Seguidamente vienen los demás periféricos, como los puertos serie y la unidad de disquetes. La *IRQ* 2 quedaba libre para la instalación de otro periférico.

A partir del *286,* hubo una evolución en este esquema, pues finalmente los PC pasaron a tener 16 direcciones *IRQ*, numeradas del 0 al 15. Como casi todas las evoluciones en la familia de los PCs, fue necesario mantener la compatibilidad con el formato anterior, para que las placas *XT* pudiesen funcionar en los PCs *286* y en adelante. De esta forma, resolvieron mantener el controlador de *IRQs* original para que todo continuase funcionando de la misma manera que antes y, simplemente, añadiron un segundo controlador para obtener las 8 direcciones nuevas. Este segundo controlador pasó a estar conectado en la *IRQ* 2, que solía estar libre. Todas las peticiones de interrupción de los periféricos conectados en las direcciones entre la 8 y la 15, supervisadas por el segundo controlador, primero pasan por la *IRQ* 2, para después llegar al procesador.



Esquema de la distribución de las interrupciones

Podemos observar dos cosas en este esquema de distribución de las *IRQs*. En primer lugar, la *IRQ* 2 ya no podía ser utilizada por ningún periférico. Si configurábamos un módem para usar la *IRQ* 2, ésta era remapeada a la *IRQ* 9. O sea, en la práctica, no teníamos 16 direcciones *IRQ*, sólo 15. En segundo lugar, como el segundo controlador está conectado a la *IRQ* 2, todas las tarjetas que utilizasen las direcciones de la 8 a la 15, tendían prioridad sobre las que usaban las *IRQs* de la 3 a la 7, pues, dede el punto de vista del procesador, se encontraban conectadas a la *IRQ* 2.

En un PC con 16 direcciones *IRQ*, esta es la configuración de más común:

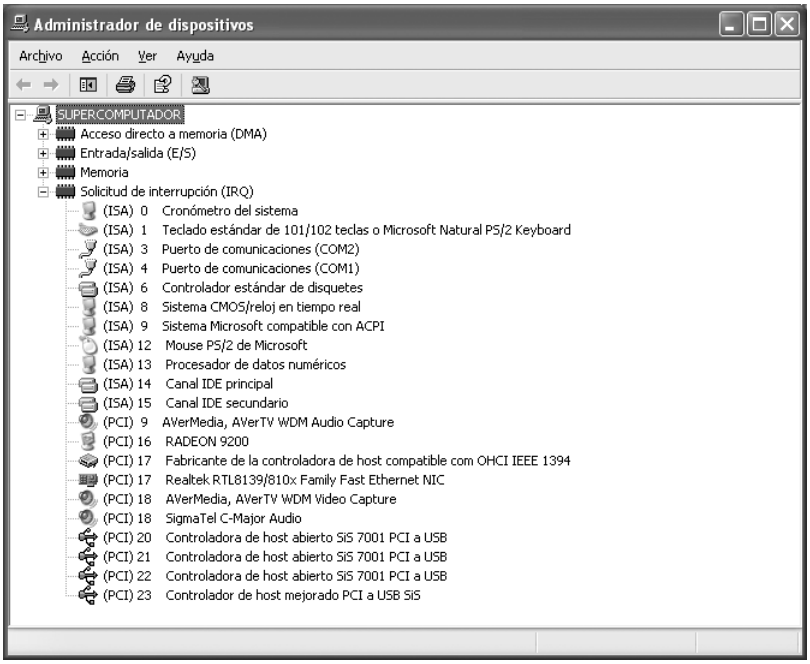
IRQ 0	Señal del reloj de la placa base (fijo)
IRQ 1	Teclado (fijo)
IRQ 2	Conmutador de IRQs (fijo)
IRQ 3	Puerto serie 2
IRQ 4	Puerto serie 1
IRQ 5	Libre
IRQ 6	Unidad de disquetes
IRQ 7	Puerto paralelo (impresora)
IRQ 8	Reloj del CMOS (fijo)
IRQ 9	Tarjeta gráfica
IRQ 10	Libre
IRQ 11	Controladora USB
IRQ 12	Puerto PS/2
IRQ 13	Coprocesador aritmético
IRQ 14	IDE primario
IRQ 15	IDE secundario

Notemos que con sólo la tarjeta gráfica y los dispositivos de la placa base ya nos quedamos con sólo dos direcciones *IRQ* disponibles. Al añadir una tarjeta de sonido y un módem quedan ocupadas todas las direcciones, sin margen de ampliación para una tarjeta de red o una tarjeta *SCSI*, por ejemplo. La solución sería deshabilitar los dispositivos que no se estuvieran usando, como por ejemplo, el segundo puerto serie, el puerto *PS/2*, la controladora *USB*, etc. Pero, de cualquier forma, inmediatamente llegaríamos otra vez al punto en el que ya no habría nada para deshabilitar.

Este problema empezó a hacerse cada vez más evidente a medida que empezaron a aparecer más y más periféricos. Los únicos periféricos de fábrica en un *286* eran el monitor y el teclado, todo el resto era opcional. Hoy en día, es raro encontrar un PC a la venta sin por lo menos una tarjeta de sonido, módem y *CD-ROM/DVD-ROM* y una tarjeta de red. Esto sin hablar en las tarjetas de captura de vídeo, de una segunda tarjeta de red necesaria para compartir la conexión de banda ancha a *Internet*, etc.

Felizmente, cuando se desarrolló el bus *PCI* pensaron en este problema, incluyendo el recurso de *PCI Steering*, que permite que dos, o más periféricos *PCI*, compartan la misma dirección *IRQ*. En este caso, la controladora *PCI* pasa a actuar como un puente entre los periféricos y el procesador. La controladora recibe todas las peticiones de interrupción y las encamina hacia el procesador y, al recibir las respuestas, las encamina hacia los dispositivos correspondientes. Como la controladora es la única que se encuentra directamente conectada al procesador es posible ocupar sólo una dirección *IRQ*. No todos los dispositivos *PCI* soportan trabajar de esta manera, pero la gran mayoría funciona sin problemas. Además del bus *PCI*, otros buses usados actualmente permiten compartir una única *IRQ* entre varios periféricos. El bus *USB* es otro buen ejemplo, ya que la controladora ocupa una única *IRQ,* que luego es compartida entre todos los puertos *USB* y todos los dispositivos conectados a ellos. Aunque nuestra placa base tenga 6 puertos *USB* y los utilicemos todos, sólo estaremos ocupando una única dirección *IRQ*.

Si sólo utilizáramos periféricos *USB* podríamos deshabilitar todos los demás puertos de la placa base: los dos serie, el paralelo y el *PS/2*. En este caso, 4 direcciones *IRQ* quedarían libres. Otro ejemplo son las controladoras *SCSI*, donde es posible conectar hasta 15 dispositivos, entre discos duros, *CD-ROM*, *DVD-ROM*, grabadoras, etc. ocupando una única dirección *IRQ*.



Las interrupciones de mi equipo

Los puertos *IDE* de la placa base utilizan dos *IRQs*, ya que tenemos dos puertos. Pero, si sólo disponemos de un único disco duro y no pretendemos usar el *IDE* secundario,

siempre existe la posibilidad de deshabilitarlo. En realidad, la escasez de direcciones *IRQ* es un problema mucho menor hoy día. Antiguamente era preciso configurar manualmente la dirección que se iba a usar en cada tarjeta de expansión y no había forma de compartir una *IRQ* entre dos periféricos. Un *jumper* en el lugar equivocado era suficiente para que la tarjeta de sonido y el módem entrasen en conflicto con alguien y dejasen de funcionar. Hoy en día, todas las tarjetas nuevas ya son *PCI*, lo que significa que soportan el *Plug & Play*. Es suficiente ensamblarlas para que el *BIOS* les atribuya automáticamente una dirección *IRQ*. Usando sólo tarjetas *PCI*, sólo deberíamos tener conflictos en el caso de que todas las *IRQs* estén ocupadas.

Pero no todo está resuelto. A pesar de que los conflictos sean muy raros al usar sólo tarjetas *PCI*, aún están en uso muchos PCs antiguos con tarjetas *ISA*. En este caso, las cosas pueden complicarse un poco. Existen dos tipos de tarjetas *ISA*: las que soportan *Plug & Play* y las tarjetas anteriores. Las primeras, que comenzaron a aparecer en la época del *486*, son configuradas automáticamente por el *BIOS*, pero no son capaces de compartir una *IRQ* como las tarjetas *PCI*. El modo *Plug & Play* de estas tarjetas *ISA* no es tan seguro como el de las tarjetas *PCI*, y pueden entrar en conflicto existiendo *IRQs* libres. En este caso, podremos cambiar manualmente la dirección que debe usar la tarjeta a través del *Administrador de dispositivos* de *Windows*.

También existen las tarjetas *ISA* anteriores o *legacy ISA*, en las que es necesario configurar las direcciones *IRQ* manualmente, a través de *jumpers* o usando un software que acompaña a la tarjeta. Una vez configurada la tarjeta para que use una dirección, la placa entrará en conflicto con cualquier dispositivo que intente usar la misma dirección.

Para complicarlo un poco más, el *BIOS* no reconoce a las tarjetas *legacy ISA*, pudiendo asignar la misma dirección a alguna tarjeta *Plug & Play*, provocando un conflicto. Para evitar este problema, es necesario reservar las direcciones *IRQ* y *DMA* ocupadas por los periféricos *legacy ISA* a través de la opción *PNP/PCI Setup* del *Setup* del *BIOS*. La opción *default* no sirve para reservar las direcciones, dejándolas libres para el uso de cualquier dispositivo *Plug and Play*. Para poder reservar una dirección es necesario cambiar la opción. Debemos tener presente de reservar sólo las direcciones necesarias, pues al reservar una dirección, ésta deja de estar disponible para las tarjetas *Plug & Play*. Si reservamos muchas direcciones tendremos problemas de conflictos, incluso con varias *IRQs* libres.

Existe un punto interesante sobre las tarjetas *ISA*, que es el hecho de que existan tarjetas *ISA* de 8 bits y tarjetas *ISA* de 16 bits. Es fácil diferenciarlas, pues las tarjetas de 8 bits sólo usan la primera parte del anclaje. Las tarjetas de 8 bits usan exactamente el mismo bus que era usado en el *XT*, lo que significa que estas tarjetas sólo reconocen las *IRQs* de la 0 a la 7.

Notemos que toda la configuración de las direcciones hecha por el *BIOS* es almacenada en un área del *CMOS*, llamado *ESCD*. En el caso de que, por ejemplo, el *BIOS* no reconozca una tarjeta recién instalada, aunque ésta sea *Plug & Play* y existan direccio-

nes libres, y se instala en la misma dirección de otras tarjetas que no soportan el *PCI Steering*, provocando un conflicto, o cualquier otro problema parecido, existe la opción de borrar el *ESCD*. Esto provocará que el *BIOS* necesite reconfigurar las direcciones de todas las tarjetas, partiendo de cero. Para limpiar el *ESCD* es necesario acceder al *Setup* del *BIOS* y activar la opción *Force Update ESCD* o *Reset Configuration data*. No debemos preocuparnos, pues no será necesario reinstalar las tarjetas, sólo pueden cambiarse las *IRQs* usadas por algunas.

5.6.1 Conseguir más IRQs

A pesar de que existen raras excepciones, todos los dispositivos del equipo necesitan enviar información al procesador. Para ello, cada uno necesita un canal de *IRQ*, que funciona como una especie de avisador para llamar la atención del procesador. En un PC básico, con un módem, una placa de sonido y una impresora, las *IRQs* disponibles son suficientes y de hecho aún acostumbran a sobrar una o dos *IRQs*. Cualquier conflicto de hardware no surgiría por la falta de *IRQs* en este caso, sino por la falta de experiencia del usuario que no supo configurarlo.

El problema es que en hoy día se usan más accesorios, como tarjetas de captura de vídeo, tarjetas de red, tarjetas *SCSI*, *RAID* o ambas, dispositivos *USB* u otros dispositivos que ocupan los puertos serie, varios discos duros y una lista de dispositivos casi ilimitada. Sin embargo, continuamos con las mismas 16 *IRQs* disponibles, conllevando que en un PC más equipado el usuario se encuentre con una falta crónica de direcciones *IRQ* para la instalación de dispositivos. Y sin un *IRQ* disponible no funciona nada de la nueva tarjeta. Manteniendo todas las configuraciones por defecto (*default*) del *Setup* (en una placa base actual) y dejando a *Windows* que detecte todos los dispoitivos, sin alterar nada, nos encontraremos con un escenario parecido al de la lista siguiente:

IRQ 0	Señal del reloj de la placa base (fijo)
IRQ 1	Teclado (fijo)
IRQ 2	Puente de IRQs (fijo)
IRQ 3	Puerto serie 2
IRQ 4	Puerto serie 1
IRQ 5	Libre
IRQ 6	Unidad de disquetes
IRQ 7	Puerto paralelo (impresora)
IRQ 8	Reloj del CMOS (fijo)
IRQ 9	Tarjeta gráfica
IRQ 10	Libre
IRQ 11	Controladora USB
IRQ 12	Puerto PS/2

IRQ 13 Coprocesador aritmético
IRQ 14 IDE primario
IRQ 15 IDE secundario

A partir de este punto, para instalar nuevos dispositivos tendremos que racionalizar el uso de las direcciones *IRQ*, desactivando dispositivos que no son necesarios. Por ejemplo, manteniendo la configuración que tenemos (placa base, memoria, procesador, disco duro, unidad de disquetes, *CD-ROM/DVD-ROM*, tarjeta gráfica, monitor, teclado, ratón *PS/2*, tarjeta de sonido y módem) podríamos deshabilitar varios dispositivos que están ocupando direcciones *IRQ*:

- **Ambos puertos serie**: estamos usando un ratón *PS/2* y no tenemos ningún dispositivo que necesite un puerto serie.
- **Puerto paralelo**: de momento no tenemos ninguna impresora, escáner u otro dispositivo que use el puerto paralelo.
- **USB**: no tenemos instalado ningún periférico USB.

Con sólo una rápida inspección, ya ha sido posible liberar cuatro direcciones que antes estaban ocupadas. Todos estos dispositivos pueden ser desactivados a través del *Setup* del *BIOS*.

Veamos otro posible problema. Imaginemos que ahora vamos añadir al PC una tarjeta de red, una tarjeta *SCSI*, una impresora, un escáner *USB* y una tarjeta de captura de vídeo. En primer lugar, tendríamos que reactivar el puerto paralelo a causa de la impresora y el puerto *USB* a causa del escáner. La tarjeta de red y la tarjeta *SCSI* ocuparían entonces las otras dos *IRQs* que estaban libres. Estaríamos entonces, otra vez, sin direcciones libres, cuando todavía nos falta instalar la tarjeta de captura de vídeo.

En este caso existirían varias opciones: el ratón, que está ocupando el puerto *PS/2* podría sustituirse por un ratón USB, permitiendo desactivar el puerto *PS/2* y liberar la *IRQ* 12, desactivar el puerto de la unidad de disquetes (sustituyéndolo por una unidad *USB*), liberando la *IRQ* 6, desactivar el canal *IDE* secundario de la placa base, para instalar el *CD-ROM/DVD-ROM* como esclavo del disco duro (liberando la *IRQ* 15).

> **Nota:** Debemos recordar que en el mercado existen adaptadores que permiten transformar un puerto USB en un puerto paralelo.

Llevando a cabo estas acciones todavía sería posible instalar otros periféricos nuevos, hasta que nuevamente nos quedáramos sin direcciones *IRQ*.

5.6.2 PCI IRQ Activated By

Ya debemos haber oído que dos o más tarjetas *PCI* pueden compartir la misma dirección *IRQ*, pero en la opción *PCI/Plug and Play Setup* del *Setup* del *BIOS* puede estar la respuesta de porque esto funciona o no en algunas placas base.

A pesar de que desde sus primeras versiones el bus *PCI* permite el reparto de *IRQs*, los primeros periféricos *PCI* no eran compatibles con este recurso. Por otro lado, todos los periféricos *PCI* actuales son compatibles para compartir la misma dirección *IRQ*.

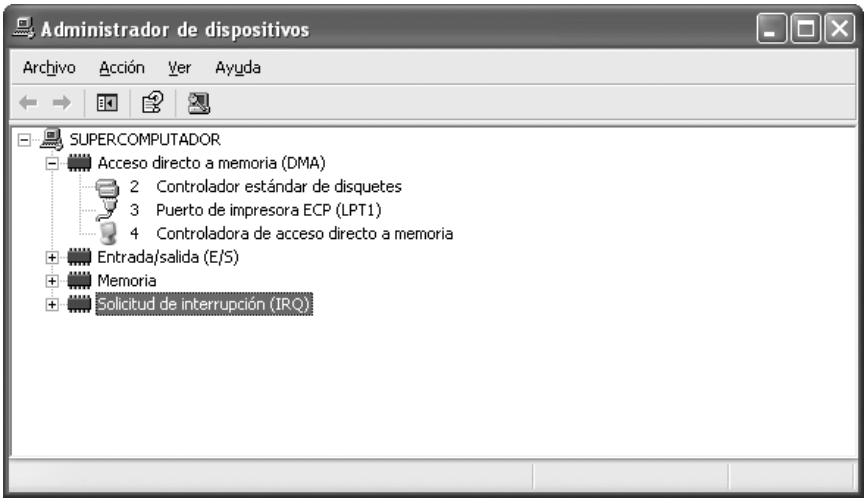
Esta opción posee dos alternativas, *Edge* y *Level*. Si estamos configurando el *Setup* de un equipo antiguo, montado hace 3 o 4 años, entonces es recomendable mantener la opción en *Edge*, ya que probablemente el PC tendrá tarjetas *PCI* incompatibles con este recurso. La opción *Edge* desactiva el reparto de *IRQs*, lo que evitaría posibles problemas. Sin embargo, si estamos configurando un PC actual deberíamos elegir la opción *Level*, que habilita el reparto de *IRQs*.

5.7 DMA (Acceso directo a memoria)

El *DMA* (*Direct Memory Access*) permite mejorar el rendimiento general del equipo, permitiendo que los periféricos transmitan datos directamente a la memoria, ahorrando pasar por el procesador.

Existen 8 canales *DMA* y, como pasa con las peticiones de interrupción, dos dispositivos no pueden compartir el mismo canal *DMA*, ya que de lo contrario se produciría un conflicto. Los 8 canales *DMA* están numerados del 0 al 7. En los canales del 0 al 3 la transferencia de datos se realiza con 8 bits y en los restantes con 16 bits. El uso de palabras binarias de 8 bits en los primeros 4 canales *DMA* es para mantener la compatibilidad con los periféricos más antiguos.

Justamente, por ser muy lentos, los canales *DMA* sólo son utilizados por periféricos lentos, como la unidad de disquetes, la tarjeta de sonido y los puertos paralelos *ECP*. Los periféricos más rápidos, como los discos duros, utilizan el *Bus Mastering*, una especie de *DMA* mejorado.



Los canales DMA de mi equipo

El canal 2 del *DMA* está usado por el controlador de disquetes. Una tarjeta de sonido necesita generalmente de dos canales *DMA*, uno de 8 bits y otro de 16 bits, usando generalmente los canales *DMA* 1 y 5. El canal *DMA* 4 está reservado para la placa base. Nos quedamos entonces con los canales 3, 6 y 7 libres. Si el puerto paralelo se configura para trabajar en modo *ECP*, también necesitará un canal *DMA*, que acostumbra a ser el canal 3.

5.8 Los componentes integrados

Las placas base con el sonido, módem y red *onboard* vienen ganando cada vez más espacio en el mercado informático. La principal ventaja de estas placas es el bajo coste. Sale mucho más barato comprar una placa base con componentes integrados (*onboard*) que comprar la placa base más los componentes por separado.

En todas las placas base con componentes *onboard*, podemos deshabilitar de forma individual los componentes integrados a través de *jumpers* o del *Setup* del *BIOS* y sustituirlos por las tarjetas convencionales (en caso de quema o ampliación), siempre que existan *slots* de expansión disponibles.

En la mayoría de las placas base con componentes *onboard* es posible sustituir los componentes sin problemas. Sólo existe el viejo problema de la falta de *slot AGP* para la tarjeta gráfica, ya que el vídeo *onboard* ocupa el bus *AGP* en la mayoría de las placas base, quedando libres los *slots PCI* para conectar otras tarjetas gráficas. Como siempre, existen buenas y malas placas base con los componentes *onboard*.

Excluyendo el rendimiento de los componentes *onboard* (ya que la tarjeta gráfica que viene integrada en una placa base nunca será tan rápida como una placa 3D de gama alta), el rendimiento mostrado por un equipo construido con una placa base con los componentes *onboard* (considerando una placa base de buena calidad) quedará muy próximo al rendimiento de un equipo montado con una placa base sin ningún componente *onboard*. El vídeo *onboard* también le quita algo de rendimiento al procesador, pues comparte el bus de la memoria con el procesador.

Ya en la actualidad, la mayoría de las placas base a la venta ya vienen con sonido y módem *onboard*, pues es más barato integrar estos componentes en la placa base o en el propio *chipset* que comprarlos por separado. Sólo las placas base para el mercado de alto rendimiento vienen sin estos componentes.

5.9 Entendiendo los nombres y códigos

La placa base es el componente más importante de cualquier PC. El procesador, las memorias, la tarjeta gráfica 3D pueden ser fácilmente sustituidos pero, para eso, será necesario que la placa base soporte los nuevos componentes. Acabamos de ver que se están haciendo cada vez más comunes las placas base con componentes *onboard*, a

causa del bajo coste, pero que pueden traer trampas en términos de las posibilidades para una ampliación futura.

Existe una noción general de calidad, basada en la marca de las placas. Por ejemplo, casi todo el mundo está conforme que las placas de la casa Asus tienen una buena calidad. Las placas base producidas por la casa Intel tienen fama de ser estables.

Pero, sólo rotular las placas base por parte del fabricante está muy lejos de ser lo ideal. Finalmente, incluso fabricantes de prestigio producen placas base de bajo coste con todo *onboard*, y también existe una gran diferencia de calidad y recursos entre diferentes modelos de placas base del mismo fabricante. Surge entonces otro problema, ya que no siempre es fácil diferenciar los incontables modelos de placas base del mismo fabricante.

A pesar de que a primera vista los nombres de código de las placas base parezcan bastante confusos, existe una lógica básica detrás de las designaciones. En el caso de las placas base de la casa Asus todo empieza con dos letras, que designan la familia de procesadores que soporta la placa. Enseguida vienen una o dos letras más que indican el *chipset* usado, aunque muchas placas incluyen nomenclaturas adicionales que indican algún tipo de recurso adicional que posee la placa base.

Por ejemplo, la placa base Asus *CUV4X-V* está destinada a procesadores *Pentium III/Celeron Coppermine* (*CU*), y está basada en un *chipset* Vía (*V*), posee un *slot AGP 4x* (*4X*) y lleva vídeo *onboard* (*-V*), en este caso un *chipset* de vídeo Savage 4, integrado en el *chipset* Vía SavagePro PM133. Basta sólo entender las siglas y convenciones que la casa Asus usa para nombrar sus placas base.

A pesar de citar aquí sólo modelos de la casa Asus, la idea es que esta información nos ayude a entender los códigos usados por cualquier fabricante de una forma más fácil.

5.9.1 El procesador

En todas las placas base de la casa Asus, las primeras letras indican la familia de procesadores que soporta la placa base:

P2 Viene de *Pentium II*. Todas las placas base que empiezan con *P2*, como *P2B* o *P2B-F* están destinadas a procesadores *Pentium II*. Naturalmente, son placas base antiguas con encaje *slot 1*, y que ya no se fabrican. Algunas placas *P2*, también soportan procesadores *Pentium III*, inclusive los *Coppermine*, pero siempre con algunas limitaciones, como por ejemplo no poder utilizar procesadores que usen un bus de 133 MHz. Tampoco podemos esperar ningún tipo de soporte por parte de la casa Asus, ya que estas placas ya no se fabrican y ellos quieren que compremos placas nuevas.

P3 Si *P2* indica que las placas base están proyectadas para procesadores *Pentium II*, las placas base que comienzan con *P3* están destinadas a los procesadores *Pentium III*. Pero una pequeña observación es que todas las placas *P3* usan un enca-

je *slot 1*. Esto las hace versátiles en términos de actualización y ampliación, ya que usando un adaptador también es posible procesadores para un *soquete 370* en ellas.

CU La casa Asus usa el *CU* para todas las placas base *soquete 370* para procesadores *Pentium III Coppermine*. Debemos notar que ninguna de estas placas base soporta procesadores *Pentium II*, pues los *Pentium II* sólo existieron en formato de *slot 1*, y no existe ningún adaptador de *slot 1* a *soquete 370*, sólo el contrario.

TU Viene de *Tualatin* que era la nueva generación de los procesadores *Pentium III* de 0.13 micrones. El *TU* es usado en todas las placas base *soquete 370* que tienen soporte oficial para los *Pentium III Tualatin*.

K7 *K7* es el nombre de código del *Athlon*. Aprovechando la familiaridad de los usuarios, la casa Asus usa el código *K7* en el nombre de todas las placas base para procesadores AMD *Athlon* con encaje *slot A*.

A7 Estas son las placas para procesadores *Athlon* y *Duron* con *soquete A*. El *A7* enfatiza justamente en el encaje, *soquete A*, y sirve para distinguir estas placas de las antiguas placas base *slot A*.

P4 Este es fácil, viene de *Pentium 4*. Hasta el momento, todas las placas Asus para *Pentium 4* comienzan con *P4*.

P5 Este era el nombre de código de la arquitectura del *Pentium* antiguo (*Pentium I*). La casa Asus aprovechó esta nomenclatura para dar nombre a sus últimas placas *soquete 7*, que soportaban toda la familia *Pentium*, así como los Cyrix y los *K6*, *K6-2* y *K6-3*.

P/I Estas son placas muy antiguas La *P* viene de *Pentium* y la *I* de Intel. Pero eso no ayuda mucho, pues estas placas pueden ser para procesadores *Pentium Pro*, o incluso para placas *soquete 7* para *Pentium I* o *MMX*. En ese caso, son los dos números siguientes los que indican la clase de la placa. Por ejemplo, *P/I-P5* significa que se trata de una placa base para procesadores *Pentium I*, mientras que *P/I-P6* indica que la placa base es para procesadores *Pentium Pro*.

ME Estas son las primeras placas base Asus con el encaje *soquete 370*. Estas pioneras soportaban procesadores *Celeron PPGA*, de 366 a 533 MHz. Algunas de las placas base *ME* también soportan procesadores *Coppermine*.

5.9.2 El chipset

Inmediatamente después de la designación del procesador, viene la designación del chipset soportado por la placa base:

V Todas las placas con la *V* en el nombre usan *chipsets* de la casa Vía. La *V* es genérica y puede referirse a cualquiera de los *chipsets* de la casa Vía. Por ejemplo, la placa base *CUV4X* usa el chipset Vía ProSavage PM133, mientras la *CU V4X-D* usa el Vía 694XDP. La *V* siempre viene después del código principal de la placa base destinado al procesador, *P2*, *P3* o *CU* que acabamos de ver en el apartado anterior.

BX Mientras que la *V* viene de Vía, el *BX* también se refiere al *chipset*, pero en este caso no dice nada respecto a la marca del *chipset*, sino directamente al modelo, en el caso el i440BX. El *BX* es probablemente el *chipset* más famoso de la historia. El *BX* fue lanzado en la época del *Pentium II* y fue el primer *chipset* para los procesadores *Pentium II* que soportaba el bus de 100 MHz.

El *BX* se reveló como un proyecto tan bueno desde el punto de vista del rendimiento que los fabricantes lo continuaron utilizando, añadiéndole recursos que el *chipset* original no tenía, para construir placas base destinadas a los procesadores más modernos. Controladoras externas añaden soporte al *UDMA 66* o *UDMA 100*, o al mismo *RAID*.

Reguladores de voltaje y los *BIOS* actuales garantizan compatibilidad con los procesadores *Coppermine*, el soporte para el bus de 133 MHz es obtenido vía *overclock* (ya que originalmente el *BX* sólo soportaba un bus de 100 MHz). Ejemplos de placas base con el famoso *chipset BX* son las placas *CUBX*, *CUBX-Y* y *CUBX-L*.

B También indica que la placa base usa el *chipset* i440-BX, pero es usado en las placas más antiguas, como en *P2B* y *P3B-F*. Una vez más es bueno recordar que todas las *P2* o *P3* son *slot 1*, mientras todas las *CU* son *soquete 370*.

A Viene de Ali, y sirve para indicar las placas base con *chipsets* de esta compañía, como en *A7A-266* (*soquete A*, *chipset* Ali MAGIK1 M1647 y soporte para un bus de 266 MHz) o la *CUA* (*Pentium II Coppermine*, *chipset* Ali Aladdin TNT2).

SI Viene de SiS y nuevamente indica el fabricante del *chipset*, en este caso la casa SiS, como en las placas base *CUSI-M* y *CUSI-FX*, ambas con el *chipset* SiS 630E.

S Una S sola, en una placa *soquete A* (en las placas *soquete 370* para *Pentium III/Celeron* se usa *SL*, que significa otra cosa como veremos a continuación) también indica que la placa posee un *chipset* de la casa SiS. Un ejemplo es la placa base *A7S-VM*.

SL Son las últimas placas base para *Pentium III/Celeron*, con *soquete 370*, con el *chipset* i815 y slot *AGP 4x*. Todas las placas base *SL* tienen el *soquete 370*, eso significa que todas son *CU* (*Coppermine*) o *TU* (*Tualatin*). Ejemplos de placas base con el *chipset* i815 son las *CUSL2-C*, *CUSL2-M* o la *TUSL2-C*.

C Son las placas para *Pentium III/Celeron* que utilizan el antiguo *chipset* i820, tanto las que utilizan memorias *Rambus* como las que usan el *chip* MTH y soportan memorias *SDRAM PC-100/133*. Ejemplos de estas placas base son las *CUC2* y *P3C-L*.

C2 Se usa de la misma manera que la *C* que acabamos de ver.

W Indican las placas base con el *chipset* i810, que tiene video *onboard*, pero no soporta el *slot AGP*. Estas placas ya no se fabrican, siendo sustituidas por las placas con el *chipset* i815, que también tiene vídeo *onboard*, pero que ofrece soporte al bus *AGP 4x*. Ejemplos de placas base con el chipset i810 son la *CUW* y la *P3W-E*.

5.9.3 Otros recursos adicionales

Después de los procesadores que soporta y del *chipset* usado, el resto del nombre de las placas base sirve para indicar si la placa posee algún recurso adicional, como, por ejemplo, sonido o red *onboard,* soporte para un bus de 133 MHz, *slot AGP 4x*, el formato de la placa (*ATX* o *mini-ATX*), etc.

Por ejemplo, la diferencia entre una placa base *CUSL2* y una *CUSL2-M* es que la primera viene en formato *ATX* y con 6 *slots PCI*, mientras que lasegunda (la *M* viene de *mini*) viene en formato *mini-ATX* y, al ser menor, sólo posee 3 *slots PCI.* Existen más diferencias entre los dos modelos de placa base, pero la principal diferencia es el formato en este caso.

4X Indica que la placa base tiene un slot *AGP 4x*. El *slot* puede ser tanto un *AGP 4x* convencional como un *AGP Pro*, el *4X* es genérico. El *4X* siempre va inmediatamente después de las dos letras del procesador y las letras del *chipset*, como en *P3V4X*, *CUV4X-D* o *CUV 4X-DLS*. Naturalmente, no todas las placas base con un bus *AGP 4x* tienen el *4X* en el nombre, como por ejemplo la *TUSL2-C* (donde es obvio que la placa soporta *AGP 4x*, ya que está basada en el *chipset* i815), el *4X* sólo se usa cuando puede existir cualquier duda.

V Una *V* después del código principal, como en *CUV4X-V* indica que la placa base tiene vídeo *onboard*.

M Indica que la placa base viene en el formato *mini-ATX*.

VM Indica que además de tener vídeo *onboard,* la placa base viene en el formato *mini-ATX*. Las placas base *mini-ATX* son menores que las *ATX* y sólo poseen 4 *slots*: o 4 *slots PCI*, o 1 *slot AGP* y 3 *slots PCI*. Este formato se usa normalmente en las placas base de bajo coste, que ya vienen con componentes *onboard*.

L La *L* viene de *LAN* o red. Indica que la placa base viene con una tarjeta de red *onboard*.

S Al contrario de lo que puede parecer a primera vista, la *S* no significa *sound* pero sí *SCSI*, indicando que la placa base posee una interfaz *SCSI onboard*. Son muy pocos los modelos de la casa Asus con este recurso y siempre son placas caras.

LS *SCSI* e interfaz de red *onboard*, como en la placa base *P3C-LS*.

D Indica que la placa soporta dos procesadores. Como en el caso de las placas con *SCSI*, son siempre placas caras, destinadas a servidores. La *D* viene de *Dual*.

DSL Estas son siempre las placas de gama alta de la casa Asus destinadas a servidores. Poseen soporte para dos procesadores (*D*) y traen interfaces de red (*L*) y *SCSI* (*S*) integradas.

E La *E* viene de *Enhanced* o perfeccionado. Indica que la placa base posee algún recurso adicional que no forma parte del *chipset*, a diferencia de otros modelos basados en el mismo *chipset*.

C La *C* viene de *Complete*, o completa. Estas placas base vienen en formato *ATX*, con 5 *slots PCI* y un *slot AGP*, y además traen sonido *onboard* opcional.

FX *FX* viene de *Flex-ATX*. Este es un formato de placa menor que el *mini-ATX* con espacio para sólo dos *slots* de expansión. La idea es una placa de bajísimo coste, con video, sonido y red *onboard* y un o dos *slots* más para que el usuario pueda añadir algun dispositivo más. Felizmente, la casa Asus sólo hizo tres placas base en ese formato, la *CUSI-FX*, la *CUW-FX* y la *CUWE-FX*.

133 Indica que la placa base soporta el bus de 133 MHz. Así como en el *4X*, la nomenclatura sólo se usa en modelos de placa base donde pueda existir cualquier duda.

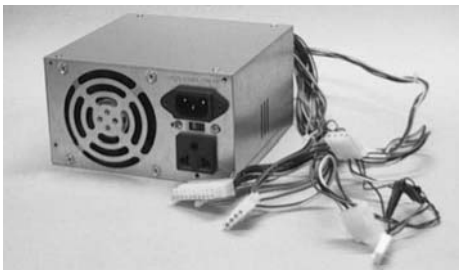
266 El 266 indica las placas base con soporte para memorias *DDR*.

5.10 El suministro de energía

Para que el equipo funcione adecuadamente es necesario que la fuente de alimentación sea capaz de suministrar la electricidad suficiente y de forma ininterrumpida.

Comparados con los equipos que teníamos hace tres o cuatro años, los equipos actuales son mucho más exigentes en el requisito de la alimentación eléctrica. Esto implica la necesidad de fuentes de alimentación mejores y más poderosas que las que necesitábamos hasta hace poco tiempo. Si la fuente no fuese capaz de suministrar la energía suficiente, el equipo se colgaría muy a menudo o no llegaría ni a encenderse.

Actualmente, prácticamente todo consume más electricidad: los discos duros y *CD-ROM/DVD-ROM* (al poseer mayores velocidades de rotación), la memoria *RAM*, etc. Actualmente, también se suelen usar más accesorios, lo que también consume más electricidad, como varios ventiladores dentro del equipo para mejorar la ventilación, grabadoras de *CDs/DVDs*, etc.



La fuente de alimentación y sus cables

Sin embargo, el mayor problema es la pareja procesador y tarjeta gráfica. Una fuente *ATX* tiene tres salidas independientes: una de 12V destinada al disco duro, *CD-ROM/DVD-ROM*, unidad de disquetes y otros periféricos del mismo género, una de 5V destinada a la placa base y a los periféricos conectados a ella (tarjeta gráfica, tarjeta de sonido, etc.) y una de 3.3V destinada al procesador y a la memoria *RAM*.

Dentro de la fuente de alimentación tenemos circuitos separados para el suministro de cada una de estas tensiones. Esto significa que la fuente es capaz de suministrar una corriente X a 12V, una corriente Y a 5V y una corriente Z a 3.3V.

Sin embargo, las fuentes de alimentación son vendidas según la capacidad total, o sea X, Y y Z sumadas, lo que da los 250 o 300 vatios. Algunas veces, la fuente de alimentación trae especificadas las capacidades máximas de cada salida, pero no siempre.

5.10.1 SAIs y estabilizadores

Existen picos de tensión, parpadeados, apagones, etc. Infelizmente, el sistema eléctrico está muy lejos de ser seguro y estable. A veces, trabajando en el equipo se ve de repente que las luces parpadean, el equipo se reinicia y perdemos todo lo que estábamos haciendo. Lo peor de todo es que un sistema eléctrico inestable pone al hardware del equipo en riesgo. El componente que más sufre es el disco duro, que con el tiempo empieza a presentar varios sectores defectuosos, seguido por las memorias y la placa base, que también pueden dañarse con facilidad.

También son comunes los casos de fuentes de alimentación quemadas y, en casos más extremos, incluso la pérdida total del equipo, literalmente quemado debido a algún rayo que haya caído en las proximidades. Existen dos accesorios destinados a proteger el equipo de ese tipo de desastres, los estabilizadores y los SAIs.

Estabilizadores

Los estabilizadores servirían, en teoría, para atenuar los picos de tensión, como los provocados por rayos, protegiendo al equipo pero sin ofrecer ninguna protección contra la pérdida de datos o algún reinicio cuando parpadea la luz. El gran problema es que los estabilizadores baratos no sirven absolutamente para nada. Un buen estabilizador sí que puede ofrecernos la protección contra los rayos, pero sólo cuando esté conectado a tierra, de lo contrario tampoco serviría de nada. Una forma de intentar diferenciar los estabilizadores es por el peso, ya que los estabilizadores baratos pesan menos que los buenos ya que se les eliminan componentes esenciales.

Hilo a tierra

La instalación de un hilo a tierra es relativamente simple. Para ello, debemos comprar una barra de cobre de unos 2.5 metros y clavarla en el suelo por completo. Para facilitar el proceso, podemos comenzar haciendo un agujero pequeño usando la propia barra de cobre.

Una vez tenemos la barra de cobre clavada en el suelo tiramos un hilo hasta la toma a la que conectaremos el equipo. Debemos usar un hilo rígido, de un espesor 4 o 6. Antes de instalarlo deberíamos probarlo usando una bombilla de 100 vatios. Conectamos el positivo de la bombilla en el positivo de la toma y el negativo en el hilo tierra que acabamos de montar. Si la bombilla se enciende con normalidad, entonces el hilo tierra estará bien instalado. Escogemos una toma de las que tienen tres conectores y conectamos el hilo tierra en el tercer polo de la toma justamente al hilo tierra. A partir de este momento, nuestro equipo estará más protegido.

SAIs

Los *SAIs* son, de lejos, los componentes ideales para proteger nuestro equipo ya que como acabamos de ver, la mayoría de los estabilizadores no ofrecen una gran protección. La gran ventaja de usar un *SAI* es tener garantizado el suministro continuo de electricidad Aunque la luz parpadee o el suministro se corte, podremos continuar trabajando hasta que se agoten las baterías del *SAI*, teniendo el tiempo suficiente para guardar los documentos y desconectar tranquilamente el equipo.

Existen dos tipos de *SAIs*, los *online* y los *offline*. Los primeros, los *online*, son mejores, pues en ellos la batería se alimenta de forma continua y el equipo es alimentado directamente por la batería, teniendo un suministro 100 % estable. Un SAI online también realiza la tarea de cualquier buen estabilizador, que es mantener un suministro completamente estable.

Distintos formatos de SAI

En los *SAIs offline* la energía de la toma es enviada directamente hacia el equipo, usando la batería sólo cuando se corta la corriente, sin ofrecer una protección tan completa como el los *SAIs online*.

Para diferenciar los dos tipos de *SAIs*, basta con realizar una mirada a las luces. Un *SAI* generalmente viene con dos luces, *Red* y *Battery*. La primera indica que la energía de la toma está siendo desviada hacia el equipo (*offline*), mientras que la segunda indica que se está usando la batería (online). Si la luz *Red* se quedase siempre encendida, apagándose sólo cuando se cortase el suministro de electricidad, entonces nos encontraríamos con un *SAI offline*.

Autonomía

En general, un *SAI* empieza a emitir pitidos cuando se corta el suministro de energía. Conforme la batería se va quedando sin carga los pitidos se van haciendo cada vez más frecuentes, indicándonos que debemos empezar a desconectar el equipo. Si ignoramos las advertencias de los pitidos, el *SAI* continuará pitando hasta que se agoten las baterías por completo.

Generalmente, lo ideal es una autonomía de 15 minutos, que en general será tiempo suficiente para terminar lo más urgente y guardarlo todo antes de poder desconectar el equipo sin problemas.

Si por ejemplo, tenemos un equipo con un *Penium 4* de 1.000 MHz, con una tarjeta gráfica 3D, dos discos duros, *CD-ROM/DVD-ROM* y una grabadora de CDs, 128 MB de *RAM* y un monitor de 17 pulgadas, lo más indicado será un *SAI* que esté por encima de 1 KVA.

En el mercado existen algunos modelos de baja autonomía, de sólo 250 o 300 VA, cuyas baterías duran alrededor de los 5 minutos. Particularmente no es recomendable la compra de *SAIs* que estén por debajo de los 600 VA, pues la diferencia de precio es muy pequeña. Muchos modelos poseen una salida para baterías externas, que permiten aumentar la autonomía del aparato.

Muchos SAIs vienen con interfaces inteligentes. En estos casos, debemos conectarlo a una de las salidas serie del equipo e instalar el software que lo acompaña. Con ello, podremos programar el *SAI* para que desconecte el equipo automáticamente en caso de que se corte el suministro, guarde los documentos, etc. Este es un recurso útil sólo si dejamos el equipo conectado cuando salimos o en el caso de un servidor.

5.11 La torre

Además de la fuente de alimentación, la propia torre merece una cierta atención en la hora de la compra. En este caso, el problema principal en este caso está relacionado con la capacidad de ventilación. Los componentes que tenemos actualmente, en especial el procesador y la tarjeta gráfica, generan mucho calor. El ventilador instalado sobre el procesador sólo disipa el aire caliente generado por él mismo, aire que se esparce dentro de la torre y acaba calentando los demás componentes y el propio procesador. Es ahí donde entra en juego la capacidad de la torre para retirar el aire caliente de dentro de la caja y hacer entrar aire frío.

Distintos formatos de torres

Cuanto mayor sea esta capacidad, más baja será la temperatura que se almacenará en el procesador, tarjeta gráfica, disco duro, etc. y mayor será la vida útil de los componentes. Otro punto clave es el aumento de la estabilidad del equipo, que está interrelacionada directamente con la temperatura.

6. Los discos duros

El *Hard Disk*, o sencillamente disco duro, es un sistema de almacenamiento de alta capacidad, que al no ser volátil, se destina al almacenamiento de archivos y programas. A pesar de no parecerlo a simple vista, el disco duro es uno de los componentes que envuelve más tecnología entre los componentes que componen un PC. En este capítulo vamos a examinar la puesta en marcha de los discos duros, tanto a nivel físico como a nivel lógico.

Sin duda, el disco duro es uno de los componentes que más ha evolucionado a lo largo de la historia de la computación. El primer disco duro fue construido por la casa IBM en 1957, y estaba formado por un conjunto de 50 discos de 24 pulgadas de diámetro, con una capacidad total de 5 MBytes, algo impensable para la época. Comparado con los discos duros actuales, este disco pionero costaba una verdadera fortuna: unos 35.000 dólares de la época.

Un disco duro con la etiqueta y sus especificaciones

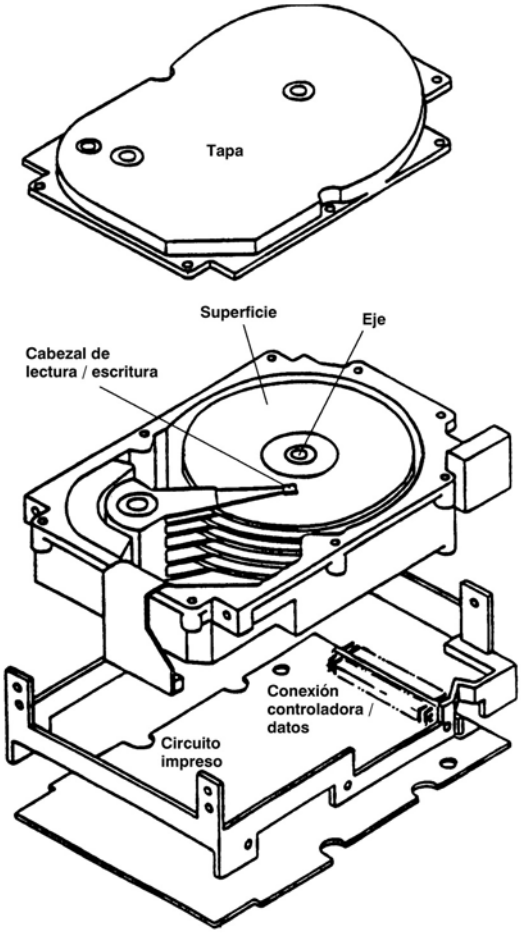
Sin embargo, a pesar de ser extremadamente caros en un inicio, los discos duros se convirtieron en populares en los sistemas corporativos, pues suministraban un medio rápido de almacenamiento de los datos.

Este primer disco duro fue llamado *RAMAC 350* y, posteriormente apodado *Winchester*, término usado todavía hoy en día para designar a los discos duros de cualquier especie.

Con el paso del tiempo, los discos duros fueron creciendo en capacidad, disminuyendo en tamaño, ganando en fiabilidad y volviéndose más baratos. Los primeros discos duros usados en microcomputadoras, a inicios de la década de los años 80, estaban compuestos por discos de 5.25 pulgadas de diámetro, tenían capacidades de entre 5 y 20 MBytes y costaban a partir de los 1.500 dólares, mientras que hoy en día, cerca de 25 años después, es posible encontrar discos de 80 GBytes o más, por menos de 300 euros.

6.1 El funcionamiento

Dentro del disco duro, los datos son grabados en discos magnéticos, llamados *platters* (del inglés). El nombre de "disco duro" viene del hecho de que los discos internos son láminas metálicas extremadamente rígidas. Cada *platter* está compuesto por dos capas.



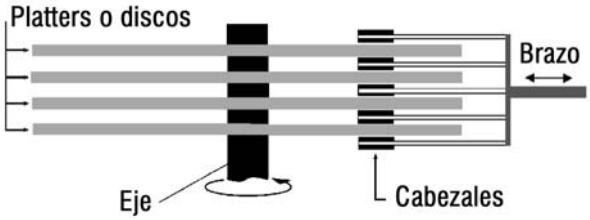
La estructura básica de un disco duro

La primera capa es llamada de sustrato y no es más que un disco metálico, generalmente de aluminio. Este disco se pule en salas limpias,para que quede perfectamente plano. Para poder almacenar los datos, este disco es recubierto por una segunda capa, ahora de un material magnético.

La aplicación de la capa magnética se realiza en los dos lados del disco y puede ser llevada a cabo de dos formas distintas. La primera se llama *electroplating* y es muy parecida a la electrólisis usada en los baños de oro usados en bisutería. Esta técnica no consigue una superficie muy uniforme y, por ello, sólo fue usada en los discos duros antiguos, los de menos de 500 MBytes. Actualmente, la técnica usada es más precisa, conocida por *sputtering* y usa una tecnología semejante a la usada en las soldaduras de los transistores de los procesadores.

Como la capa magnética sólo tiene alguna micrones de espesor, se recubre con una fina capa protectora, que ofrece protección contra pequeños impactos. Esta capa es importante, pues a pesar de que los discos son encapsulados en salas limpias, internamente contienen aire con una presión semejante a la del ambiente. Como veremos más adelante, no sería posible que funcionase un disco duro en el caso de que estuviese en el vacío.

Los discos duros se cierran herméticamente, a fin de impedir cualquier contaminación que venga del medio externo. Sin embargo, nunca es posible mantener un ambiente 100% libre de partículas de polvo. Un pequeño daño en la capa protectora no interfiere en el proceso de lectura/escritura hecho de forma magnética.



La estructura interna de un disco duro

Los discos son montados en un eje de aluminio, que debe ser lo suficiente sólido para evitar cualquier vibración de los discos, posibles por las altas rotaciones. Éste es otro componente del disco duro que también pasa por un proceso de pulimento, ya que los *platters* deben quedar perfectamente anclados y alineados.

Finamente, tenemos el motor de rotación, responsable de mantener una rotación constante. El motor es uno de los mayores responsables de la durabilidad del disco duro, pues la mayoría de los fallos graves provienen de él. Los discos duros más antiguos utilizaban motores de 3.600 rotaciones por minuto, mientras que actualmente, son utilizados motores de 5.600 y 7.200 rpm, que pueden llegar a más de las 10.000 rpm en los modelos más caros. La velocidad de rotación es uno de los principales factores que determinan el rendimiento de un disco duro.

Para leer y escribir los datos en el disco se usan cabezales de lectura electromagnéticos (*heads* en inglés) que están anclados en un brazo móvil (*arm*), lo que permite el acceso a la totalidad del disco. El brazo de lectura es una pieza triangular de aluminio, pues necesita ser ligero y resistente. Un dispositivo especial, llamado "actuador", o "*actuator*" en inglés, coordina el movimiento de los cabezales de lectura.

En los primeros discos duros, se usaban antiguos motores de paso para mover los distintos brazos y cabezales de lectura. Sin embargo, además de ser muy lentos, eran muy susceptibles a los problemas de un mal alineamiento, además de no ser muy fiables. Los discos duros contemporáneos ya utilizan un mecanismo más sofisticado para esta tarea, el *actuator*, compuesto por un dispositivo que actúa a través de atracción y repulsión electromagnética. Básicamente tenemos dos electroimanes, uno a cada lado del brazo móvil.

Alterando la intensidad de la corriente eléctrica y, consecuentemente, la potencia de cada imán, se mueven el brazo y los cabezales de lectura. Este sistema es mucho más rápido y fiable que los antiguos motores de paso.

Otro dato interesante es la manera como los cabezales de lectura leen los datos, sin tocar en la capa magnética. Si tuviéramos la oportunidad de ver un disco duro abierto, veríamos que con los *platters* parados, los cabezales de lectura están levemente presionados en dirección a los *platters*, tocándolos con una cierta presión. Sin embargo, cuando los *platters* giran a una alta velocidad de rotación se forma una especie de colchón de aire. Este colchón de aire repele al cabezal de lectura, haciendo que quede siempre a algunos micrones de distancia del *platter*.

Notemos que mientras el disco duro está apagado los cabezales de lectura quedan en una posición de descanso y sólo salen de esa posición cuando los discos duros ya están girando a la velocidad de rotación máxima. Para prevenir accidentes, los cabezales de lectura vuelven a la posición de descanso siempre que no se están leyendo datos, a pesar de que los discos continúen girando.

Es justamente por esto que a veces, al producirse un pico de tensión o apagarse el equipo mientras el disco duro está funcionando, aparecen sectores defectuosos. Al cortarse el suministro de energía, los *platters* paran de girar y se deshace el colchón de aire, haciendo que los cabezales de lectura puedan tocar los discos magnéticos.

Para disminuir que ocurra este tipo de accidente, en los discos duros modernos se instala un pequeño imán en uno de los lados del *actuator* que se encarga de atraer a los cabezales de lectura a la posición de descanso cada vez que se corta el suministro eléctrico (tecnología llamada de *auto-parking*).

La capa de protección de los discos magnéticos también ofrece alguna protección contra los impactos, pero aun así, a veces se producen daños, resultando en uno o varios sectores defectuosos. Por esto, siempre es bueno desconectar el equipo cuando aparece la pantalla de que ya podemos apagar el equipo con seguridad de *Windows*.

6.1.1 La tarjeta controladora

Todo el mecanismo de puesta en marcha del disco duro, el movimiento de los cabezales de lectura, la velocidad de rotación, la lectura y escritura de los datos, el envío y el recibo de datos a través del puerto *IDE*, etc. es coordinado por la tarjeta controladora. En los discos duros antiguos, la tarjeta controladora era una tarjeta separada, conectada a un *slot ISA* y enchufada al disco duro por dos cables de datos. Este sistema era ineficiente, pues la distancia volvía susceptible la comunicación a distintas interferencias y a la corrupción de los datos.

A partir de la aparición de los discos duros *IDE*, la tarjeta controladora pasó a formar parte del propio disco duro. Fue un paso muy lógico, pues la tarjeta controladora necesita ser construida conforme a la arquitectura física del disco duro y jamás funcionaría en otro modelo.

Además de practicidad, este avance permite una comunicación de datos mucho más eficiente ya que se usan cables más cortos. Es por ello que no hablamos de *controladora IDE* y sí de *interfaz IDE*, pues sólo funciona como un medio de comunicación ya que forma parte del propio disco duro.

A pesar de ser pequeña, la tarjeta controladora de un disco duro actual es mucho más sofisticada que un equipo antiguo entero (uno *286* por ejemplo), ya que posee más poder de procesamiento y hasta más memoria, en forma de memoria caché o *búfer*. Los discos duros llevan un mínimo de 2 MBytes de memoria caché, que almacena los datos accedidos, disminuyendo bastante el número de lecturas realizadas. Los datos almacenados en la memoria caché pueden ser transferidos casi instantáneamente, usando toda la velocidad permitida por la interfaz *IDE*, mientras que el acceso a los datos grabados en los discos magnéticos necesitaría mucho más tiempo.

6.1.2 El diámetro de los platters

El tamaño de los discos magnéticos determina el tamaño físico del disco duro. Actualmente se utilizan discos de 3.5 pulgadas de diámetro, pero también es posible encontrar algunos modelos más antiguos de 5.25 pulgadas.

Estos discos más grandes, sin embargo, no son una buena opción, pues son mucho más lentos y más antiguos que sus "hermanos" menores. Eso se debe a varios factores, ya que teniendo *platters* mayores no se consigue hacerlos girar a una velocidad muy alta, ocasionando lentitud en el acceso a los datos grabados. Debido a que la superficie de los *platters* es mayor, los cabezales de lectura tardan más tiempo para localizar los datos, justamente debido a la mayor distancia que debe ser recorrida.

Existen también discos de 2.5 pulgadas destinados a los ordenadores portátiles, debido a su reducido tamaño y a su bajo consumo de electricidad. Existen también discos miniaturizados destinados a aparatos menores, como ordenadores de mano, cámaras digitales, etc.

6.1.3 Materiales utilizados

Existen investigaciones para encontrar materiales más baratos que el aluminio usado actualmente, pero que a la vez sean lo suficientemente rígidos, lo que podría abaratar sustancialmente los discos duros. Hace tiempo que los grandes fabricantes vienen investigando, la mayoría de las veces con compuestos de vidrio o plástico.

La casa IBM fue la pionera con los discos de vidrio, con su *Deskstar 75GXP*. Esta tecnología desarrollada por la casa IBM ofrece algunas ventajas respecto a los discos de aluminio tradicionales, ya que el vidrio es un material más duro y, justamente por ello, los *platters* son más estables a las altas rotaciones. Sin embargo, los discos de la casa IBM todavía son más caros que los modelos equivalentes con discos de aluminio.

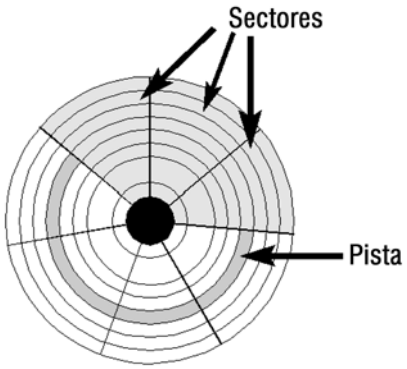
La casa Samsung viene trabajando en discos de compuestos plásticos, intentando producir discos duros de bajo coste. Sin embargo, al contrario que en el caso del vidrio, el plástico es un material muy poco resistente y los discos son mucho más susceptibles a deformaciones a altas temperaturas y altas rotaciones.

6.1.4 Pistas, sectores y cilindros

Para organizar el proceso de escritura y lectura de los datos en el disco duro la superficie de los discos es dividida en pistas y sectores. Las pistas son círculos concéntricos, que empiezan al final del disco y se van volviendo menores conforme se acercan al centro. Cada pista recibe un número de direccionamiento que permite su localización.

La pista más externa recibe el número 0 y las siguientes reciben los números 1, 2, 3 y así en adelante. Para facilitar todavía más el acceso a los datos, las pistas se dividen en sectores, que son pequeños trozos donde son almacenados los datos, siendo cada sector de 512 Bytes.

Un disco duro actual posee unos 900 sectores en cada pista (el número varía en función de la marca y del modelo), poseyendo siempre más de 3.000 pistas.

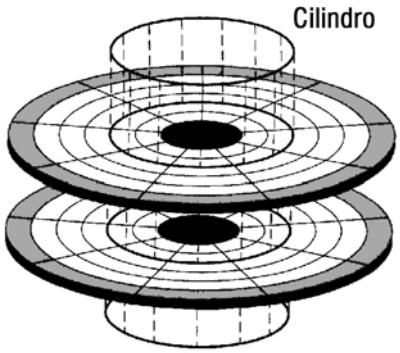


Los sectores y pistas del platter de un disco duro

Para definir el límite entre una pista y otra, así como donde termina un sector y donde empieza el próximo, se usan marcas de direccionamiento, pequeñas áreas con una señal magnética especial, que orientan al cabezal de lectura, permitiendo a la controladora del disco duro localizar los datos deseados en cada momento. En los discos duros *IDE* estas marcas sólo son hechas una vez durante el proceso de fabricación del disco, y no se pueden borrar vía software. Existen algunos programas como el *Norton Calibrate*, que prometen un formateo físico no destructivo, grabando de nuevo las marcas de orientación, lo que según los manuales mejora la fiabilidad del disco duro. Sin embargo, la amplia mayoría de los discos duros no permiten este tipo de regrabación de las marcas.

Además de las pistas y los sectores, también tenemos las distintas caras de los *platters*. Un disco duro está formado internamente por varios discos apilados (*platters*), siendo lo más común el uso de 2 o 3 discos. Por regla general, sólo los discos duros de gran capacidad utilizan 4 o más *platters*. Así como en un disquete, podemos usar los dos lados del disco para grabar datos, cada lado pasa entonces a ser llamado cara. En un disco duro con 2 *platters* por ejemplo, disponemos de 4 caras. Como una cara se encuentra aislada de la otra, en un disco duro tenemos varios cabezales de lectura, uno por cada cara.

A pesar de poseer varios cabezales de lectura en un disco duro, éstos no se mueven independientemente, pues todos son parte de la misma pieza metálica, conocida por brazo de lectura. El brazo de lectura es una pieza triangular, que puede moverse horizontalmente. Ya que todos los cabezales de lectura siempre estarán en la misma pista de sus respectivos *platters*, dejamos de llamarlas pistas y pasamos a usar el término cilindro. Un cilindro es el conjunto de pistas con el mismo número en los distintos *platters* del disco duro.



Los cilindros de un disco duro

6.1.5 Zoned Bit Recording (ZBR)

La pista más externa de un disco duro posee más del doble de diámetro que la pista más interna y, consecuentemente, posee una mayor capacidad para almacenar más datos. Sin embargo, en los primeros discos duros, así como en los disquetes, todas las

pistas del disco, independientemente de su diámetro, poseían el mismo número de sectores, haciendo que en las pistas más externas, los sectores ocupasen un espacio mucho más grande que los sectores de las pistas más internas.

Tenemos entonces un gran espacio de disco desperdiciado, pues hay que nivelar por abajo, haciendo que todas las pistas posean el mismo número de sectores permitido por las pistas más internas, acabando por desperdiciar enormes cantidades de espacio en las primeras pistas del duro.

El recurso de *Zoned Bit Recording* permite variar la cantidad de sectores por pista, en función del diámetro de la pista que va a ser dividida, permitiendo una organización más racional del espacio en disco y permitiendo aumentar la densidad de la grabación de los datos.

La cantidad de sectores de cada pista se define durante el formateo físico del disco duro, hecho al final de su proceso de fabricación.

6.1.6 Densidad

Para crear un disco duro de mayor capacidad podemos usar más *platters* en el mismo disco duro, usar *platters* de un tamaño mayor o aumentar la densidad de la grabación de los *platters* ya existentes.

Aumentar la cantidad de *platters* dentro del disco duro, de 3 a 6 por ejemplo, sólo aumentaría la capacidad total del disco duro, pero no su rendimiento. En el caso de aumentar el tamaño de los *platters* de 3.5 a 5.25 pulgadas por ejemplo, también aumentaría la capacidad total del disco duro, pero la velocidad de acceso a los datos se vería bastante comprometida.

Siendo así, la manera más eficiente de aumentar la capacidad de los discos duros es aumentando la densidad de los distintos *platters*. Aumentar la densidad significa conseguir grabar más datos en el mismo espacio físico. Entonces, podemos disponer de más pistas en el mismo disco y cada pista puede pasar a disponer de más sectores, permitiendo grabar más datos en un disco duro del mismo tamaño físico.

Sin embargo, aumentando la densidad de los *platters* aparecen varios problemas. Disminuyendo el espacio ocupado por cada bit en el *platter* se disminuye su señal magnética. Entonces necesitamos de un soporte físico de mayor calidad para que los datos se mantengan estables en el disco. También necesitamos desarrollar un cabezal de lectura más sensible, así como perfeccionar los mecanismos del movimiento de los brazos de lectura.

A pesar de estas dificultades, los fabricantes están consiguiendo desarrollar increíbles tecnologías que están permitiendo aumentar la densidad de los *platters* de una forma espectacular, permitiendo que además de discos más veloces, exista una caída vertiginosa en el precio por GByte.

6.1.7 Lectura y escritura

Un *platter* de un disco duro es recubierto por una capa magnética extremadamente fina. En realidad, cuanto más fina es la capa de grabación, mayor será su sensibilidad y, consecuentemente, mayor será la densidad de grabación permitida.

Los primeros discos duros, así como los discos usados a inicios de la década de los años 80, utilizaban la misma tecnología magnética utilizada en los disquetes, llamada "*coated media*", que además de permitir una baja densidad de grabación, no era muy durable. Los discos duros de inicios del siglo XXI ya utilizan un soporte físico laminado (*plated mass media*); un soporte físico más denso, de una calidad muy superior y que permite una enorme capacidad de almacenamiento.

El cabezal de lectura y escritura de un disco duro funciona como un electroimán, estando compuesto por una bobina de hilos que envuelven un núcleo de hierro. Este electroimán es extremadamente pequeño y preciso, hasta el punto de ser capaz de escribir datos en pistas que miden menos de una centésima de milímetro.

Cuando se están grabando los datos en el disco duro, el cabezal utiliza su campo magnético para organizar las moléculas del óxido de hierro de la superficie de escritura, haciendo que los polos positivos de las moléculas queden alineados con el polo negativo del cabezal y, consecuentemente, que los polos negativos queden alineados con el polo positivo del cabezal. En este caso, se usa la vieja y famosa ley de que los polos opuestos se atraen.

Como el cabezal de lectura y escritura del disco duro es un electroimán, su polaridad puede alternarse constantemente. Con los *platters* girando continuamente y variando la polaridad del cabeza de escritura, también variamos la dirección de los polos positivos y negativos de las moléculas de la superficie magnética. Según la dirección de los polos, tenemos un bit 1 o 0.

Para grabar las secuencias de bits 1 y 0 que forman los datos, la polaridad magnética del cabezal es modificada algunos millones de veces por segundo, siempre siguiendo ciclos bien determinados. En el disco, cada bit está formado por una secuencia de varias moléculas. Cuanto mayor es la densidad del disco, se usarán menos moléculas para almacenar cada bit y tendremos una señal magnética más débil. Entonces necesitamos de un cabezal magnético mucho más preciso.

En el momento de leer los datos almacenados en el disco, el cabezal de lectura capta el campo magnético generado por las moléculas alineadas. La variación entre las señales magnéticas positivas y negativas genera una pequeña corriente eléctrica, que corre a través de los hilos de la bobina. Cuando la señal llega a la tarjeta lógica del disco duro, es interpretada como una secuencia de bits 1 y 0.

Viéndolo así, el proceso de almacenamiento de los datos en discos magnéticos parece ser simple. A pesar de que en los discos duros modernos se hayan incorporado varios perfeccionamientos, el proceso básico sigue siendo el mismo.

6.1.8 Formateo

Para poder usar un disco duro, primero debemos darle formato. Formatear significa dividir, de forma lógica, el disco en sectores direccionables, permitiendo que los datos puedan ser grabados y leídos, a posteriori, de forma organizada.

El formateo del disco es un asunto relativamente complicado tanto que, muchas veces, los mismos profesionales del tema tienen sus dudas. La primera cosa a entender es que existen dos tipos de formateo: el formateo físico, o formateo de bajo nivel, y el formateo lógico.

Ya en el inicio de este capítulo hemos hablado de la organización del disco en pistas, sectores y cilindros. Esta organización es necesaria para que se puedan leer y grabar los datos en el disco.

La división del disco en pistas, sectores y cilindros se conoce como formateo de bajo nivel, o formateo físico. Los discos más antiguos, del formato *ST-506* y *ST-412* (que ya hace más de una década que dejaron de usarse, al ser substituidos por los discos de los formatos *IDE* y *SCSI*), eran más simples que los actuales, permitiendo que el formateo físico fuese realizado por el propio usuario a través del *Setup*. Inclusive, estos discos necesitaban ser periódicamente reformateados físicamente. Eso era así por un simple problema: al ser leídos por el cabezal de lectura, los sectores del disco se calentaban y se expandían, y al enfriarse se contraían rápidamente.

Esta expansión y contracción de la superficie del *platter*, acababa por alterar la posición de las pistas, causando una desalineación y dificultando la lectura de los datos por parte del cabezal magnético, siendo necesario un nuevo formateo físico para que las pistas, sectores y cilindros volviesen a sus posiciones iniciales.

Para empeorarlo aún más, en esos discos obsoletos se utilizaba un motor de paso para mover los cabezales electromagnéticos que, al no ser completamente precisos, siempre acababan causando alguna desalineación.

Los discos duros *IDE* y *SCSI*, usados actualmente, ya son mucho más complejos, siendo casi imposible determinar la disposición de las pistas, sectores y cilindros para posibilitar un formateo físico. Tampoco tienen el problema de una desalineación, de modo que el formateo físico sólo se realiza una vez en la fábrica.

Cualquier intento indebido de formatear físicamente un disco duro moderno no surtirá efecto, pudiendo en algunos casos raros hasta inutilizar el disco. Concluyendo, todos los discos duros *IDE* o *SCSI* no necesitan ser formateados físicamente por el usuario.

Existen algunos programas, como *Ontrack Disk Mánager* o *Maxtor Low Level Format*, que son usados por algunos usuarios para realizar formateos físicos. En realidad, en mayor medida, estos programas son herramientas de diagnóstico y de corrección de errores, en la misma línea del *Scandisk*, sólo con algunos recursos más, que che-

quean el disco duro marcando los sectores defectuosos, también permitiendo visualizar muchos otros errores lógicos en el disco y corregirlos.

Por último, tenemos algunos programas antiguos, así como la opción *Low Level Format* encontrada en el *Setup* del *BIOS* de placas base antiguas, destinada a formatear físicamente los antiguos discos duros *MFM* y *RLL*. Usando discos duros *IDE* o *SCSI*, este tipo de formateo no funciona. Como mucho, se borra el Defect Map y el sector de arranque del disco duro, deshaciendo el formato lógico del disco y causando la pérdida de los datos grabados, pero sin modificar el formato físico.

El interior de un disco duro

Algunas personas intentan usar placas base más antiguas, que poseen la opción de formateo a bajo nivel para "formatear físicamente" sus discos duros *IDE*, a fin de eliminar los sectores dañados del disco. Este procedimiento, además de poder causar daños o la inutilización total del disco duro, no conlleva ninguna ventaja.

Un sector dañado es un pequeño fallo en la superficie magnética del disco duro, donde no se pueden grabar los datos con seguridad. Estos daños en la superficie del *platter* pueden surgir debido a algún golpe o al desgaste del medio magnético, lo que acostumbra a ocurrir en los discos duros muy usados. Cuando ejecutamos alguna utilidad de diagnóstico del disco duro, como *Scandisk* que acompaña al sistema *Windows*, se comprueban todos los sectores del disco duro y aquéllos que están dañados, son marcados como defectuosos en un área reservada del disco, llamada *Defect Map*, para que no se utilicen más. Los sectores dañados son conocidos también como "*badblocks*". Estos sectores son marcados como defectuosos al presentar una tendencia a la corrupción de los datos grabados. Intentar borrar el *Defect Map* sólo haría que estos sectores fuesen nuevamente accesibles por el sistema operativo. Este intento desesperado sólo disminuiría la fiabilidad del disco

6.2 El sistema de archivos

Después del formateo físico, llevado a cabo por el propio fabricante del disco duro en las etapas finales de la producción, nos encontramos con un disco duro dividido en pis-

tas, sectores y cilindros, toda la infraestructura básica para permitir que el cabezal de lectura pueda leer y escribir los datos.

Sin embargo, para que este disco duro puede ser reconocido y utilizado por el sistema operativo, es necesario realizarle un nuevo tipo de formateo, el formateo lógico. Éste consiste en escribir en el disco duro la estructura del sistema de archivos utilizado por el sistema operativo.

Un sistema de archivos es un conjunto de estructuras lógicas y de rutinas que le permiten al sistema operativo controlar el acceso al disco duro. Los diferentes sistemas operativos usan diferentes sistemas de archivos. Los sistemas operativos son los encargados de organizar el espacio del disco duro, de modo que permita almacenar y acceder a los datos de la manera más eficiente, de acuerdo con los recursos, limitaciones y objetivos del sistema.

Los distintos sistemas operativos existen para cubrir diferentes propósitos. *Windows 98*, por ejemplo, está destinado básicamente al uso doméstico, teniendo como prioridad la facilidad de uso y la compatibilidad. Los sistemas operativos basados en *Unix* tienen como prioridad la estabilidad y la seguridad. Al tener propósitos tan diferentes, estos sistemas operativos usan diferentes métodos para organizar los datos en el disco, para atender de la mejor forma posible sus objetivos.

Para formatear un disco que va a ser utilizado por *Windows*, por ejemplo, necesitamos arrancar el sistema desde un disquete de arranque, y ejecutar el programa *FDISK*, seguido del comando *FORMAT C:* (o la letra de unidad que debe ser formateada).

Otros sistemas operativos incluyen, a veces, asistentes que orientan al usuario sobre el formateo lógico del disco durante el proceso de instalación. Las distribuciones actuales del *Linux* poseen una opción de realizar las particiones de forma automática durante la misma instalación. Para los que prefieren realizar las particiones del disco duro de forma manual, existe un programa muy fácil de usar. En *Windows 2000*, el programa para realizar las particiones es igual de fácil de usar, sólo con la diferencia de que se ejecuta en modo texto durante la instalación.

Los sistemas de archivos más usados actualmente son: *FAT16*, compatible con el *DOS* y todas las versiones de *Windows*, *FAT32* compatible sólo con *Windows 98/2000* y *Windows XP*, *NTFS* compatible con *Windows NT* y *Windows XP*, *EXT2* usado por *Linux* y el *HPFS* compatible con *OS/2* y las versiones antiguas de *Windows NT*.

6.2.1 FAT16

Este es el sistema de archivos utilizado por *MS-DOS*, incluyendo el *DOS 7.0*, y por *Windows 95*, siendo compatible también con *Windows 98* y la mayoría de los sistemas operativos utilizados en la actualidad. Hoy en día, el sistema *FAT16* sirve para todo, pues los discos duros formateados con este sistema pueden accederse sin dificultad en *Windows 2000/XP*, *Linux* y en otros sistemas operativos.

Sin embargo, a pesar de su versatilidad, el sistema FAT16 posee muchas limitaciones. El sistema de archivos adopta 16 bits para el direccionamiento de los datos, permitiendo un máximo de 65.526 *clusters*, que no pueden ser mayores de 32 KB. Esta es la mayor limitación del sistema *FAT16*: como sólo podemos tener 65.526 *clusters* con un tamaño máximo de 32 KB en cada uno, utilizando este sistema de archivos sólo podemos crear particiones de como máximo 2 GBytes. En el caso de tener un disco duro mayor, será necesario dividirlo en dos o más particiones.

El sistema operativo reconoce cada partición como un disco duro distinto: en el caso de tener dos particiones, por ejemplo, la primera aparecerá como *C:* y la segunda como *D:*, exactamente como si tuviésemos dos discos duros distintos instalados en el equipo.

En *Linux*, las particiones aparecen dentro de la carpeta */dev*. El primer disco duro (el master del *IDE* primario) aparece como */dev/hda*, el segundo disco duro aparece como */dev/hd*b, el tercero como */dev/hdc* y así en adelante. En el caso de que los discos duros estén divididos en varias particiones, el número de la partición aparecerá justo después del nombre del disco duro. Por ejemplo, si el primer disco duro (*/dev/hda*) está dividido en tres particiones, las particiones aparecerán como */dev/hda1*, */dev/hda2* y */dev/hda3*.

Imaginemos ahora que tenemos un equipo con *Linux* instalado y decidimos instalar un segundo disco duro, formateado con el sistema *FAT16*. El disco duro aparecerá como */dev/hdb* y la partición aparecerá como */dev/hdb1*. Para acceder a él, sólo necesitamos abrir el terminal y pasarle el comando: *mount /dev/hdb1 /win -t vFAT*. Con este comando le estamos indicando a *Linux* que nuestro segundo disco duro está formateado con el sistema de archivos *FAT* y que queremos acceder a él a partir de la carpeta */win*. Después de pasarle el comando, basta acceder a la carpeta /win para tener acceso a todos los archivos del disco duro. Se quisiéramos, podemos cambiar la carpeta */win* por otra carpeta que creamos más conveniente. Para crear carpetas, basta usar el comando *mkdir*, como por ejemplo *mkdir nueva_carpeta*.

Este es el proceso manual para crear particiones que funciona con cualquier distribución actual de *Linux* y con las antiguas. Las distribuciones actuales siempre incluyen utilidades gráficas que permiten crear particiones sin tener que usar los comandos que acabamos de ver.

Continuando con la descripción de cómo funciona el sistema *FAT16*, un *cluster* es la menor unidad de destino de archivos reconocida por el sistema operativo. Recapitulando, en el sistema *FAT16* sólo podemos tener 65.526 *clusters* por partición. Este límite existe debido a que cada *cluster* tiene una dirección única, a través de la cuál es posible localizar donde está almacenado un dato determinado. Un archivo grande se graba en el disco duro fragmentado en varios *clusters*, pero un *cluster* no puede contener más de un archivo. Como no es posible tener más de 65.526 *clusters* en una partición, es posible tener clusters de menor tamaño para no desperdiciar tanto espacio, sólo en el caso de crear particiones más pequeñas.

Tamaño de la partición	Tamaño de los clusters en FAT16
Entre 1 y 2 GB	32 KBytes
Menos de 1 GB	16 KBytes
Menos de 512 MB	8 KBytes
Menos de 256 MB	4 KBytes
Menos de 128 MB	2 KBytes

Justamente, debido al tamaño de los *clusters*, no es recomendable usar el sistema *FAT16* para formatear particiones de más de 1 GB. En caso contrario, con *clusters* de 32KB, el desperdicio de espacio será excesivo. De cualquier forma, aunque el desperdicio de espacio no sea el problema, el límite de 2 GB ya es suficiente motivo para no usar este sistema de archivos en cualquier PC actualizado.

Windows NT puede crear y utilizar particiones *FAT16* con clusters de 64 KB, lo que permite la creación de particiones *FAT16* de hasta 4 GB. Sin embargo, este no es un buen argumento, pues con *clusters* tan grandes el desperdicio de espacio será enorme. Sólo *Windows NT 4.0* y algunos programas para formatear discos, como *Partition Magic* de la casa Power Quest (www.powerquest.com) son capaces de crear estas particiones y sólo *Windows NT/2000* es capaz de accederlas correctamente. *Windows 98* puede acceder a estas particiones pero tendremos algunos problemas, como programas que informarán incorrectamente del espacio libre en el disco. Pero, según Microsoft, no existe ningún peligro de pérdida de datos.

La versión *OSR/2* de *Windows 95* (también conocido como *Windows "B"*), aportó un nuevo sistema de archivos, llamado *FAT32*, y que se siguió utilizando en *Windows 98* y soportado por *Windows 2000/XP* y *Linux*. A propósito de esto, el sistema *FAT32* es el sistema de archivos más utilizado mientras los PCs con *Windows 98/ME* sean mayoritarios.

6.2.2 FAT32

Una evolución natural del antiguo *FAT16*, el sistema de archivos *FAT32* utiliza 32 bits para el direccionamiento de cada *cluster*, permitiendo *clusters* de sólo 4 KB. El tamaño máximo de una partición con el sistema *FAT32* es de 2.048 GBytes (2 Terabytes), lo que permite formatear cualquier disco duro actual en una única partición.

Cuando convertimos una partición *FAT16* a *FAT32*, lo más normal es que consigamos entre un 15 y un 30% de disminución en el espacio ocupado en disco. El problema es que varios sistemas operativos antiguos, incluyendo *Windows NT 4.0* y *Windows 95*, no son capaces de acceder a las particiones formateadas con *FAT32*. Felizmente, los sistemas operativos más actuales, como *Windows 2000/XP* y versiones recientes de *Linux*, ya ofrecen soporte para este sistema. El único problema, aparte de la menor compatibilidad en sus inicios, fue que la desfragmentación del disco, sea cual fuese el programa usado, era un poco más lenta debido al mayor número de *clusters*. Por

ejemplo, usando el *Desfragmentador de disco* de *Windows 98*, la desfragmentación de un disco puede tardar más de una hora. Otros programas como *Norton Speed Disk* ya son más rápidos, a pesar de que la desfragmentación tarda más que en los discos con el sistema *FAT16*.

A pesar del uso de direcciones de 32 bits para cada *cluster* permite usar *clusters* de 4 KB en particiones grandes, por cuestiones de rendimiento se estableció que, por defecto, los *clusters* de 4 KB sólo se usarían en particiones de hasta 8 GB. Por encima de esto, el tamaño de los *clusters* variara de acuerdo con el tamaño de la partición:

Tamaño de la partición	Tamaño del cluster
Menos de 8 GB	4 KBytes
De 8 a 16 GB	8 KBytes
De 16 a 32 GB	16 KBytes
Mayor que 32 GB	32 KBytes

6.2.3 Convertir unidades de FAT16 a FAT32

En el caso de estar usando *Windows 95 OSR/2* o *Windows 98*, y nuestro disco duro está formateado con el sistema *FAT16*, podemos convertirlo al sistema *FAT32* usando algunos programas específicos. El primero es el *FDISK*, que puede encontrarse en un disco de arranque del mismo *Windows 98* o *Windows 95 OSR/2*. Para usarlo, basta inicializar el equipo usando el disquete de arranque y teclear el comando *FDISK* en el *prompt* de *DOS*.

Otra alternativa para realizar la conversión es usar el programa *Partition Magic* que consigue convertir de *FAT16* a *FAT32* sin ninguna pérdida de datos. Si estamos usando *Windows 98* la conversión será más fácil, ya que este sistema operativo viene acompañado de un conversor, que a pesar de no tener todos los recursos del programa *Partition Magic*, realiza el proceso sin ninguna pérdida de datos.

6.2.4 Estructuras lógicas en los sistemas FAT

Todos los sistemas de archivos están constituidos por varias estructuras lógicas, que permiten al sistema operativo grabar y localizar los datos grabados en el disco duro con la mayor facilidad y velocidad posibles. Estas estructuras son muy semejantes en los sistemas *FAT16* y *FAT32*. Además, la única diferencia entre los dos sistemas es que el sistema *FAT32* usa 32 bits en el direccionamiento de cada *cluster*, en contra de los 16 bits del sistema *FAT16*.

El nombre de "estructura lógica" fue otorgado porque las estructuras son a nivel de sistema operativo, es decir, no alteran la estructura física del disco (pistas, sectores, cilindros, etc.), ni la manera como la controladora del disco duro lo utilizará, sino sólo la manera en la que el sistema operativo detectará y utilizará el disco.

Dado que todas estas estructuras son creadas a nivel lógico (software), no existe ningún problema en reformatear un disco en el que instalamos *Windows 98* para que podamos instalar *Linux, OS/2* o cualquier otro sistema operativo, basta tener el programa de formateo adecuado.

Sector de arranque

Durante el arranque, justo después de ejecutar las comprobaciones del *Post*, el próximo paso del *BIOS* es cargar el sistema operativo. Independientemente del sistema de archivos que estemos usando, el primer sector del disco duro es reservado para almacenar la información sobre la localización del sistema operativo, que permiten al *BIOS* iniciar su ejecución.

En el sector de arranque se registra qué sistema operativo está instalado, con qué sistema de archivos fue formateado y qué archivos deben leerse para inicializar el equipo. Generalmente, también son grabados mensajes de error. Un único sector de 512 Bytes puede parecer poco, pero es suficiente para almacenar el registro de arranque debido a su pequeño tamaño. El sector de arranque también es conocido como "pista *MBR*", "pista 0", etc.

La existencia de un sector de arranque es justamente lo que maraca la diferencia entre un disco de arranque y un disco sin sistema. Si grabamos los archivos de inicialización en un disquete o en un disco duro vacío, usando el comando *Copy* o arrastrándolos a través del *Explorador de Windows*, no conseguiremos inicializar el equipo a través de él, pues a pesar de tener todos los archivos necesarios, el *BIOS* no será capaz de encontrarlos debido a la inexistencia de un sector de arranque. Para crear un disquete de arranque o convertir el disco duro en un disco de arranque, deberemos usar el comando *SYS* seguido de la letra de la unidad. En este caso, además de copiar los archivos del sistema, se creará el sector de arranque.

Algunos tipos de virus son capaces de instalarse en el sector de arranque, que se revela un sitio ideal para alojar estos programas destructivos, pues como este sector es leído cada vez que se enciende el equipo, el virus siempre se cargará en la memoria junto con el sistema operativo. En este caso, el virus no eliminará ni con el formateo del disco duro, ya que usando el comando *Format* no reescribimos el sector de arranque.

Para acabar con este tipo de virus la solución ideal es usar el comando *FDisk /MBR*, después de formatear el disco, para rescribir el sector de arranque. Después de esto, bastará llevar a cabo la reinstalación del sistema operativo o usar el bueno y viejo comando *SYS* para reescribir el sector. Obviamente, el formateo del disco duro sólo es necesario en el caso de no poseer ningún buen antivirus en el equipo.

FAT

Después de que el disco duro sea formateado y dividido en *clusters*, algunos sectores son reservados para guardar la *FAT* (*File Allocation Table* o "tabla de destino de archi-

vos"). La función de la *FAT* es servir como índice, almacenando la información sobre cada *cluster* del disco. A través de la *FAT* el sistema operativo sabe si una determinada zona del disco está ocupada o no, y puede localizar cualquier archivo almacenado.

Cada vez que se graba o borra un archivo , el sistema operativo modifica la *FAT*, manteniéndola siempre actualizada. La *FAT* es tan importante que, además de la tabla principal también se guarda una copia de seguridad, que se usaría si la tabla principal estuviese dañada. Una curiosidad es que, cuando formateamos un disco duro usando el comando *FORMAT* no se borra ningún dato, sólo la *FAT* principal es substituida por una tabla en blanco.

El conjunto *Norton Utilities* posee una utilidad, llamada *Rescue Disk*, que permite almacenar una copia de la *FAT* en disquetes. En el caso de que nuestro disco duro sea formateado accidentalmente por un virus, o por cualquier otro motivo, podremos restaurar la *FAT* con la ayuda de estos disquetes, volviendo a tener acceso a todos los datos, como si no hubiese ocurrido nada.

Aunque no poseamos una copia de la *FAT*, es posible recuperar los datos usando otra utilidad del paquete *Norton Utilities*, llamada *Diskedit*, que permite acceder directamente a los *clusters* del disco, y (con algún trabajo) recuperar los datos importantes. Estos sólo son dos ejemplos, ya que actualmente existen una infinidad de programas que permiten crear copias de seguridad de la *FAT* y recuperar los datos de los discos duros formateados de forma accidental.

Directorio raíz

El directorio raíz ocupa algunos sectores del disco, justo después de los sectores ocupados por la *FAT*. Cada archivo o directorio del disco duro posee una entrada en el directorio raíz, con el nombre del archivo, la extensión, la fecha de cuando fue creado o cuando fue realizada la última modificación, el tamaño en Bytes y el número de *cluster* donde empieza el archivo.

Un archivo pequeño puede ser almacenado en un único *cluster*, mientras que un archivo grande es repartido y almacenado ocupando varios *clusters*. En este caso, al final de cada *cluster* existirá un marcador indicando el próximo *cluster* ocupado por el archivo. En el último *cluster* ocupado por el archivo existe un código que marca el fin del mismo. Cuando suprimimos un archivo, sencillamente se remueve su entrada del directorio raíz, haciendo que los *clusters* ocupados por él parezcan vacíos para el sistema operativo.

Además del nombre, cada archivo recibe también una extensión de hasta tres caracteres, como *EXE*, *DOC*, etc. A través de la extensión, el sistema operativo sabe con qué programa debe abrir cada archivo. La extensión no tiene ninguna influencia sobre el archivo, sólo determina como será visto por el sistema operativo.

Después de la extensión, existe un Byte más reservado para el atributo del archivo, que puede ser "sólo lectura", "oculto", "sistema", "volumen", "directorio" o "archivo".

El atributo permite indicarle al sistema operativo y demás aplicaciones cómo se debe soportar el archivo.

El atributo "sólo lectura" indica que el archivo no puede ser modificado o suprimido. Si intentamos suprimir o modificar un archivo de sólo lectura recibiremos un mensaje de error. El atributo "sistema" posee una función parecida, indicando que, además de ser oculto, el archivo es utilizado por el sistema operativo.

Para indicar que un archivo no debe ser visto en circunstancias normales por ningún usuario, usamos el atributo "oculto". Para poder ver los archivos ocultos, deberemos usar el comando *DIR /AH* en *DOS*, o marcar la opción *Mostrar todos los archivos y carpetas ocultos* en la ficha *Ver* del cuadro de diálogo *Opciones de carpeta*, al que accedemos a través la opción *Herramientas/Opciones de carpeta* de cualquier ventana de *Windows*.

De todos los atributos, el más importante es el atributo de "directorio", pues permite la existencia de subcarpetas. Las carpetas, incluso vacías, son vistas por el sistema operativo como archivos. Dentro de este archivo quedan almacenadas informaciones sobre el nombre de la carpeta, los atributos, la posición de la carpeta en el árbol de directorios e informaciones sobre qué archivos o subcarpetas están almacenados en la carpeta, así como la localización de estos archivos en el disco.

Desde ya hace tiempo, los archivos con el atributo "directorio" pasaron a ser vistos por el sistema operativo como carpetas, pero el intento de convertir un archivo cualquiera en carpeta no saldría bien, pues a pesar de que las carpetas también son archivos en esencia, éstas poseen un formato específico.

Una curiosidad sobre las subcarpetas es que sólo empezaron a ser soportadas a partir de la versión 2.0 del *DOS*. Los usuarios de *DOS 1.0* tuvieron que convivir durante algún tiempo con un sistema que sólo permitía almacenar archivos en el directorio raíz, con la consecuente limitación de los 512 archivos en el disco duro.

Finalizando, el atributo "archivo" indica un archivo que raramente es modificado, o que es una copia de seguridad de algún archivo importante. Muchos programas de copia de seguridad verifican este atributo cuando realizan una copia de seguridad incremental (cuando sólo son guardados los archivos que se han modificado desde la última copia de seguridad). En este caso, el programa de copia de seguridad retira el atributo tras guardar el archivo. Al ser modificado por algún otro programa, el archivo recibe el atributo de nuevo, permitiendo al programa de copia de seguridad saber qué archivos han sido modificados.

Para modificar los atributos de un archivo a través del *Explorador de Windows* es suficiente pulsar en él con la ayuda del botón derecho del ratón y seleccionar la opción *Propiedades* para abrir la ventana de propiedades.

Para modificar los atributos a través del *DOS* basta usar el comando *ATTRIB*. Los signos usados en la sintaxis del comando *ATTRIB* son los siguientes:

```
+ : para activar un atributo.
- : para desactivar un atributo.
R : sólo lectura.
A : archivo.
S : sistema.
H : oculto.
```

Desfragmentar el disco duro

Cuando un archivo es borrado, los *clusters* que ocupaba quedan libres. Al grabar un nuevo archivo en el disco, el sistema operativo empezará a grabarlo en el primer *cluster* que encuentre libre, continuando en los próximos *clusters* que estén libres, aunque estén muy distantes unos de los otros.

Este procedimiento genera un fenómeno llamado fragmentación de archivos, que disminuye mucho la velocidad de acceso a los datos. A diferencia de otros sistemas de archivos más modernos, el sistema *FAT* no posee ningún mecanismo que impida o, al menos, disminuya la fragmentación. Por esto, es recomendable hacer una desfragmentación del disco duro una vez a la semana. Si los archivos se encuentran grabados en clusters correlativos, los archivos se leerán mucho más rápido, aumentando mucho el rendimiento global del equipo.

Mientras el *Desfragmentador de disco* de *Windows 95* se contentaba en sólo desfragmentar los archivos, los desfragmentadores comerciales como *Norton Speed Disk* organizaban los archivos de forma secuencial, en un orden que mejoraba la velocidad de lectura, considerando la posición en el disco, frecuencia y el orden en el que los archivos eran accedidos, etc.

El *Desfragmentador de disco* de *Windows 98* y *Windows 2000/XP* ya incluye recursos semejantes, haciendo un trabajo casi tan bueno como el desfragmentador del paquete de utilidades *Norton Utilities*. Sin embargo, el *Desfragmentador de disco* de *Windows* sigue siendo más lento, y puede tardar horas para desfragmentar una partición muy grande, o muy fragmentada.

6.2.5 VFAT

La *FAT16* usada por el sistema operativo *DOS* posee una grave limitación en cuanto al tamaño de los nombres de archivo, que no pueden tener más que 11 caracteres, siendo 8 para el nombre del archivo y 3 para la extensión. El límite de sólo 8 caracteres es un gran inconveniente.

Sabiamente, Microsoft decidió eliminar esta limitación en *Windows 95*. Para conseguir sobrepasar esta barrera y, a la vez, continuar usando la *FAT16*, evitando así los costes de desarrollo y los problemas de incompatibilidad generados por la adopción de un nuevo sistema de archivos (*Windows 95* original sólo era compatible con el sis-

tema *FAT16*), se optó por remendar el sistema de archivos *FAT16* con un nuevo sistema de archivos llamado *VFAT*.

A través del VFAT, los archivos con nombres largos son grabados en el directorio raíz respetando el formato 8+3 (ocho letras para el nombre y una extensión de hasta 3 caracteres), siendo el nombre verdadero almacenado en una zona reservada. Si tuviéramos dos archivos, llamados "Reunión anual de 2003" y "Reunión anual de 2004", por ejemplo, se grabarían en el directorio raíz como "Reunio~1" y "Reunio~2".

Si el disco fuese leído desde *DOS*, el sistema sólo accedería al nombre simplificado. Leyendo el disco desde *Windows 95*, sería posible acceder a las zonas ocultas del *VFAT* y ver los nombres completos de los archivos.

6.2.6 Problemas con la FAT

Como la *FAT* es un registro muy dinámico y que necesita ser actualizado tras cualquier modificación en los archivos grabados en el disco duro, es normal que a veces surjan algunos problemas, aunque fáciles de resolver en la gran mayoría de los casos.

Si, por ejemplo, un programa modifica cualquier archivo, aumentando su tamaño y haciendo que ocupe más *clusters* y el sistema se cuelga sin que el programa tenga la oportunidad de hacer las modificaciones necesarias en la *FAT*, tendremos *clusters* ocupados por el archivo que se mostrarán al sistema como libres. Entonces, cualquier programa podría escribir en estos *clusters* dañando el archivo anteriormente grabado.

Los errores que acostumbramos a ver más frecuentemente son: agrupaciones perdidas (Lost Clusters), archivos cruzados (*Cross-Linked Files*), archivos o directorios inválidos (*Invalid Files or Directories*) y errores en la FAT (*Allocation or FAT Errors*).

Todos estos errores ocurren a nivel lógico, es decir, sólo son corrupciones en los datos grabados causadas por el propio sistema operativo u otros programas, y no debido a una mala puesta en funcionamiento del disco duro. Estos problemas son fáciles de resolver usando programas de diagnóstico, como *Scandisk* o *Norton Disk Doctor*.

Agrupaciones perdidas: las agrupaciones perdidas ("agrupación" es la traducción de la palabra "*cluster*") no son otra cosa que clusters marcados en la *FAT* como ocupados, pero que en realidad no almacenan ningún archivo o fragmento de archivo.

Las agrupaciones perdidas surgen cuando se produce alguna avería cuando un programa está guardando algún archivo. En este caso, se realizan las modificaciones en la *FAT* pero el sistema (sea por un pico de tensión o por un error de Windows) se cuelga antes de que los datos sean grabados.

Un programa de diagnóstico verifica todas las entradas de la *FAT*, así como todas las cadenas de archivos durante su fase de *test*. Al encontrar un *cluster* perdido altera la *FAT* marcando el *cluster* como vacío y ofrece al usuario la opción de guardar los datos almacenados en el *cluster* en forma de archivo o, sencillamente, eliminarlos.

Archivos cruzados: cualquier archivo de mayor tamaño que un *cluster* es grabado en forma de una secuencia de *clusters*, cada uno almacenando un fragmento del archivo. Al final de cada *cluster* existe una señal indicando en qué *cluster* está grabada la continuación del archivo. En el caso de interrumpirse bruscamente la aplicación, puede que algunos archivos pasen a apuntar, de forma errónea, a clusters usados por otros archivos. Entonces nos encontramos con un o más *clusters* que son asignados a dos archivos a la vez.

Para solucionar este problema, un programa de diagnóstico borraría ambos archivos, ofreciendo antes la opción de guardar su contenido como un nuevo archivo.

Archivos o directorios inválidos: algunas veces, debido a un error del sistema, algún archivo guardado puede estar dañado. Estando corrompido su contenido, un directorio, o algún otro archivo que posee una estructura definida, se convierte en inútil.

Algunas veces, los programas de diagnóstico consiguen reparar el archivo; en otras, no queda otra opción de guardar el contenido del archivo y luego eliminarlo.

Errores en la FAT: debido a una avería, puede pasar que la propia *FAT* se corrompa. En este caso, un programa de diagnóstico podría intentar corregirla (comparando las dos copias de la *FAT*, en el caso de que el daño sea pequeño) o, sencillamente, sustituir la utilizada por la copia de seguridad.

6.2.7 FAT12

El sistema *FAT12* fue el primer sistema de archivos utilizado en PCs, antes del *FAT16*. En este arcaico sistema de archivos sólo se usaban 12 bits para formar la dirección de cada *cluster*, permitiendo un total de 4096 *clusters*. El tamaño máximo de cada *cluster* era de 4 KB, permitiendo particiones de hasta 16 MB.

En 1981, cuando se lanzó el IBM PC, 16 MB parecía ser una capacidad altamente satisfactoria, ya que en aquella época los discos duros más caros no tenían más de 10 MB, siendo lo más común el uso de discos duros de sólo 5 MB. Claro que, tratándose de informática, por mayor que parezca un límite, éste no será nunca definitivo por mucho tiempo. Rápidamente se empezaron a usar discos duros de 40, 80 o 120 MB, obligando a la casa Microsoft a crear el sistema de archivos *FAT16*, e incluirlo en la versión 4.0 del *MS-DOS*.

Al ser un sistema de archivos más simple que el *FAT16*, todavía se usa la *FAT12* en *Windows 95/98/NT/2000/XP* para formatear disquetes, donde tenemos *clusters* de 512 Bytes.

6.2.8 NTFS

El *NTFS* es un sistema de archivos más antiguo de lo que muchos piensan. Empezó a ser desarrollado a inicios de la década de los años 80, cuando el proyecto de *Windows NT* estaba en sus primeras etapas.

Desde el inicio, la idea fue crear un sistema de archivos que pudiese usarse durante décadas, aunque los discos duros evolucionasen.

Dado que el gran problema en el sistema *FAT16* era el uso de sólo 16 bits para el direccionamiento de cada *cluster*, permitiendo sólo 65.526 *clusters* por partición, el sistema *NTFS* incorporó la capacidad para direccionar los *clusters* usando direcciones de 64 bits desde el principio.

Esto permite más de 18 billones de billones de *clusters*, un número casi infinito. La única limitación pasa a ser el tamaño de los sectores del disco duro. Como cada sector posee 512 Bytes, usando el sistema *NTFS* el tamaño de cada cluster puede ser de 512 Bytes, independientemente del tamaño de la partición.

Fue, sin duda, un gran avance sobre los *clusters* de 32 KB y las particiones de hasta 2 GB del *FAT16*. Pero, existe un pequeño problema para direccionar particiones muy grandes usando *clusters* de 512 Bytes, el rendimiento. Con un número muy grande de *clusters*, el procesamiento necesario para encontrar los datos deseados pasa a ser muy grande, disminuyendo el rendimiento.

Tamaño de la partición	Tamaño del cluster
Menos de 512 MB	512 Bytes
Menos de 1 GB	1 KByte
Menos de 2 GB	2 KBytes
Por encima de 2 GB	4 KBytes

Pese a que, por defecto, se usan *clusters* de 4 KB en cualquier partición mayor a los 2 GB, podemos crear particiones con *clusters* del tamaño que deseemos a través del asistente para la creación de particiones de *Windows 2000/XP*, que puede encontrarse en la opción *Panel de control/Herramientas administrativas/Administración de equipos/ Almacenamiento/Administración de discos*. En el lado derecho de la ventana aparecerá un mapa de los discos duros instalados en el equipo, basta pulsar con el botón derecho sobre un área de espacio libre y luego en la opción *Crear partición*.

Una ventaja más del sistema *NTFS* es que los nombres de archivos y carpetas utilizan caracteres *Unicode*, al contrario del *ACSII*. El *ASCII* es el sistema donde cada carácter ocupa 1 Byte de datos, pero sólo se permiten letras, números y algunos caracteres especiales. En el *Unicode*, cada carácter ocupa 2 Bytes, lo que permite 65 mil combinaciones, las suficientes para almacenar caracteres en varios idiomas. Esto permite que usuarios de Japón, China, Taiwán y otros países que no utilizan el alfabeto occidental puedan crear archivos usando caracteres de su propio idioma, sin la necesidad de instalar los *drivers* que añadan el soporte a sus idiomas.

Otro punto importante en el que el *NTFS* es superior al sistema *FAT* es en la tolerancia a errores. En el sistema *FAT*, siempre que el sistema se bloquea o es desenchufado mientras se están actualizando archivos o directorios en el disco duro, existe una alta

posibilidad de que el sistema se vuelva inestable, con archivos cruzados, agrupaciones perdidas y los otros problemas que acabamos de ver.

En el sistema *NTFS*, se mantiene un *log* de todas las operaciones realizadas. Con esto, aunque el equipo sea desenchufado en medio de la actualización de un archivo, el sistema podrá, durante el siguiente arranque, examinar este *log* y descubrir exactamente en qué punto se paró la actualización del archivo, ofreciendo la oportunidad de corregir el problema de forma automática. Además de no necesitar perder tiempo pasando el *Scandisk*, la posibilidad de pérdidas de datos es mucho menor. También existe el recurso de "*Hot fix*", donde los sectores dañados son marcados automáticamente, sin la necesidad de usar el *Scandisk* o cualquier utilidad de diagnóstico.

6.2.9 NTFS 5

Este es el sistema de archivos utilizado por *Windows 2000/XP*. Como *Windows 2000* fue construido basándose en *Windows NT 4.0*, fue un paso natural continuar usando el mismo sistema de archivos, aunque sin embargo, con algunas mejoras como el soporte para el directorio activo, que puede ser usado en redes basadas en *Windows 2000 Server*.

El recurso más enfatizado por Microsoft es el *Encripting File System*, que permite la encriptación de los datos grabados en el disco duro, de modo que sólo su propietario pueda acceder a ellos.

Así como *Windows NT*, *Windows 2000/XP* posee un buen sistema de seguridad que, usado correctamente, sólo puede ser evitado por alguien con profundos conocimientos del sistema. Sin embargo, esta seguridad sólo impide el acceso al sistema. Alguien que tenga acceso al equipo puede burlarlo de forma fácil, sencillamente instalando el disco duro como esclavo en otro equipo.

Este recurso de encriptación es interesante, por ejemplo, para profesionales de bases de datos, que llevan datos confidenciales en sus portátiles. Es posible la encriptación del disco duro entero, como carpetas o archivos individuales.

Con *Windows 2000* instalado, éste convierte automáticamente las unidades *NTFS* a *NTFS 5*, ofreciendo la opción de convertir las unidades *FAT16* o *FAT32* sin ninguna pérdida de datos. Las unidades *NTFS 5* pueden accederse desde *Windows NT* con excepción, claro, de los directorios encriptados. Otros recursos nativos del sistema *NTFS 5* tampoco funcionarán, pero se podrá acceder a los datos sin problemas.

Desde el punto de vista del usuario doméstico, sin embargo, el recurso más interesante es la posibilidad de comprimir carpetas o archivos. En *Windows 95/98* es posible compactar una unidad de disco usando el programa *Double Space*, aunque sin embargo, sólo es posible compactar particiones de disco enteras, lo que normalmente no acaba siendo una buena opción, pues disminuye mucho la velocidad del equipo y aumenta la posibilidad de perder algún dato.

Usando *Windows 2000/XP* en una partición *NTFS* podemos juntar lo mejor de los dos mundos, sólo compactando las carpetas o archivos que no son usados frecuentemente para ganar espacio. Es posible acceder a las carpetas compactadas normalmente a través del *Explorador de Windows*, aunque el acceso a los datos será un poco más lento, pero, usado a partir de un *Pentium II 300* ni notaremos la diferencia.

Para compactar un archivo o carpeta basta pulsar en él con el botón derecho del ratón, escoger la opción *Propiedades* y pulsar el botón *Opciones avanzadas* de la ficha *General*. Ahora debemos marcar la opción *Comprimir contenido para ahorrar espacio en disco*. Para convertir una partición *FAT32* (o *FAT16*) a *NTFS*, desde *Windows 2000*, podemos usar el comando *convert*. Abrimos el *Símbolo del sistema* y tecleamos:

```
convert letra_de_unidad: /fs:ntfs
```

Si intentamos convertir la partición donde está instalado el sistema operativo, éste nos va a mostrar un mensaje de error:

```
Convert cannot gain exclusive access to the [driveletter]:,
so it cannot convert it now. Would you like to schedule it
to be converted the next time the system restarts (Y/N)?
```

Respondemos que sí y reiniciamos el equipo. La conversión empezará automáticamente. Es un proceso lento. Si fuese posible, deberíamos enchufar el equipo a un *SAI* para evitar que el equipo se reinicie en cualquier inestabilidad en la red eléctrica. Este comando realiza la conversión sin ninguna pérdida de datos, así como el conversor *FAT32* de *Windows 98* y el programa *Partition Magic*.

6.2.10 Estructuras lógicas en el sistema NTFS

Así como en el sistema *FAT*, en el sistema *NTFS* se incluyen varias estructuras lógicas en el disco duro. A pesar de que la idea es básicamente la misma, estas estructuras son muy diferentes.

Master File Table

En el sistema *NTFS*, la *MFT* (*Master File Table*) substituyó a la *FAT*, almacenando las localizaciones de todos los archivos y directorios, incluyendo los archivos referentes al propio sistema de archivos. Pero, la forma de hacerlo difiere del sistema *FAT*.

Cada entrada de archivo o directorio en la *MFT* tiene 2 KB, donde se almacenan el nombre del archivo y sus atributos. Entonces, sobra una pequeña zona de datos, en general de 1.500 Bytes (puede ser mayor o menor, dependiendo del espacio ocupado por el nombre y los atributos) que se usa para guardar el inicio del archivo.

En el caso de que el archivo sea muy pequeño, podrá ser almacenado directamente en la entrada de la *MFT*. En caso contrario, sólo se almacenarán los números de los *clusters* ocupados por el archivo.

En algunos casos, no es posible almacenar ni los atributos del archivo en la *MFT*. En este caso, los atributos serán almacenados en *clusters* vacíos del disco duro y la *MFT* sólo contendrá entradas que apuntan a ellos.

Puede parecer raro que un archivo pueda tener más de 2 KB de atributos, pero en el *NTFS* los atributos del archivo van mucho más allá que los atributos de archivo, directorio, oculto, etc. que existen en el sistema *FAT*.

Los atributos del archivo incluyen su nombre, versión, nombre *MS-DOS* (el nombre simplificado a 8 + 3 caracteres) pero, sobre todo, incluyen los permisos de acceso al archivo, qué usuarios del sistema podrán acceder a ellos y quiénes no, y un espacio reservado para la auditoría, que permite almacenar información sobre qué operaciones en las que interviene el archivo deben grabarse para que sea posible realizar una auditoría en caso necesario.

A propósito de esto, la posibilidad de establecer permisos de acceso para cada archivo o carpeta es uno de los recursos más interesantes de *Windows NT* y *Windows 2000/XP*.

Permisos de acceso

Configurar los permisos de las carpetas en *Windows 2000/XP* no es tan complicado como parece. Primero, creamos los *logins* y contraseñas de acceso de los usuarios que utilizarán el PC. En *Windows 2000* es necesario registrarse antes de poder utilizar el sistema. Para hacer esto, abrimos la opción *Cuentas de usuario* del *Panel de control*.

Seleccionamos la opción *Crear una cuenta nueva* y escribimos el nombre de usuario y luego escogemos sus permisos en el equipo, lo que determinará lo que cada usuario podrá hacer en el equipo.

Un administrador tiene acceso a todo, un usuario normal puede instalar programas y crear archivos, pero no tiene acceso a la mayor parte de las configuraciones del sistema, mientras que un usuario limitado no puede instalar programas, no puede alterar las configuraciones del sistema y sólo puede editar o suprimir sus propios archivos.

Después de estos pasos, podemos configurar los permisos de acceso de cada archivo o carpeta, que podrá (o no) ser accedida por cada usuario, tanto de forma local, como a través de la red. En el caso de que el equipo esté conectado en red y deseemos que otros usuarios tengan acceso a algunas de las carpetas o archivos de nuestro equipo, basta compartir la carpeta con la red. Para ello, basta acceder a las propiedades de la carpeta, abrir la ficha *Compartir* y marcar la opción *Compartir esta carpeta*.

Para configurar los permisos de acceso abrimos la ficha *Seguridad*. Las configuraciones sirven tanto para los accesos locales como para los accesos a través de la red. *Windows* aplica los permisos de acceso de acuerdo con el usuario que se encuentra registrado en el equipo.

Por defecto, todos los usuarios tienen acceso total a la carpeta. En el campo *Nombre* veremos los distintos usuarios del sistema y todos los permisos marcados como *Permitir*. Realizamos una pulsación en el usuario para configurar sus permisos, que aparecen en la parte de abajo de la ventana. La opción *Control total* significa que el usuario tendrá acceso total a la carpeta, podrá abrir, crear, suprimir y ejecutar los distintos archivos.

Modificar permite que el usuario altere el contenido de los archivos que están en la carpeta. *Lectura y ejecución* permite que se ejecuten los archivos que están en la carpeta. Esta opción debe quedar marcada para que los usuarios puedan abrir programas a través de la red.

Listar el contenido de la carpeta sólo es el permiso básico para ver lo que hay en la carpeta. Sin este permiso activado, un usuario sólo podrá acceder a algún archivo en caso de saber su nombre. *Leer* sólo permite abrir o copiar archivos que están en la carpeta, pero sin la posibilidad de crear nuevos archivos, suprimir o modificar nada. Sin el permiso de lectura, el usuario no podrá ni abrir la carpeta.

Escribir permite crear o copiar archivos en la carpeta. Veamos que si otorgamos el permiso de escritura, pero no el permiso de modificar, el usuario no podrá alterar los archivos de la carpeta, sólo crear algunos nuevos.

La configuración de los permisos puede ser la parte más importante de la implementación de una red basada en *Windows 2000/XP* o en el antiguo *Windows NT* a la vez que puede ser, de lejos, la más dificultosa dependiendo del número de usuarios y restricciones que tuviésemos que configurar.

Log File Service

Este servicio, *LFS*, es el mayor responsable de la tolerancia a errores del sistema *NTFS*. Tolerancia a errores significa, en este caso, no perder datos o estructuras del sistema de archivos cuando el sistema se bloquea, o hubiese cualquier otro imprevisto, o que al menos, el daño sea el menor posible.

Para eso, el sistema mantiene un *log* con todas las alteraciones hechas en el sistema de archivos. Al grabar un archivo cualquiera, primero se grabará una entrada en el *log* con los detalles sobre la operación, qué archivo está siendo grabado, en qué parte del disco, etc. y al terminar la grabación es grabada otra entrada, un *OK* confirmando que todo ha salido bien. En el caso de que se bloquee el sistema durante la grabación, en el próximo arranque será posible verificar lo que estaba haciéndose y realizar las correcciones necesarias.

Periódicamente, el sistema verifica todas las entradas del *log* y si todo está en orden suprime el antiguo *log* para evitar que el archivo ocupe demasiado espacio en el disco. El tamaño del archivo *log* varía en función de la capacidad del disco duro pero, por regla general, ocupa entre 2 y 4 MBytes.

6.2.11 Sistema EXT2

El *EXT2* es el sistema de archivos utilizado en la amplia mayoría de las distribuciones de *Linux*. En realidad, el *EXT2* ya está sobrepasado, pues ya existe su sucesor, el *EXT3*, que veremos más adelante.

El sistema de archivos de *Linux* pasó por una gran evolución desde su aparición. En realidad, en las prácticas primarias del desarrollo, *Linux* utilizaba un sistema de archivos mucho más antiguo, el *Minix filesytem*. Pero el *Minix filesytem* tenía limitaciones. Las direcciones de los bloques de datos sólo tenían 16 bits, lo que sólo permitía crear particiones de un máximo de 64 MBytes. Además, el sistema no permitía nombres de archivos con más de 14 caracteres. No es de extrañar que en poco tiempo *Linux* crease su propio sistema de archivos, el *Extended File System*, o sencillamente *EXT*, que quedó listo en abril del año 92, a tiempo de ser incluido en el *Kernel 0.96c*.

En esta primera versión, el *EXT* permitía la creación de particiones de hasta 2 GB y soportaba nombres de archivo con hasta 255 caracteres. Fue un gran avance, pero el sistema todavía estaba muy lejos de ser perfecto. Rápidamente empezaron a aparecer en el mercado discos duros con más de 2 GB y el sistema no tenía un buen rendimiento, además de que la fragmentación de los archivos era casi tan grande como en el sistema *FAT*. Como respuesta a estos problemas surgió, a principios del año 93, el *EXT2* que se convertiría en el sistema de archivos para *Linux*.

El *EXT2* conllevó el soporte a particiones de hasta 4 TBytes, mantuvo el soporte a los nombres de archivo con hasta 255 caracteres, además de varios otros recursos, que veremos a continuación. Una cosa interesante es que en *Linux* los archivos no necesitan tener una extensión.

La diferencia con *Windows* es que en *Linux* las extensiones son sólo una parte del nombre del archivo, no un ítem obligatorio. Por este motivo es posible crear archivos con varios caracteres tras el punto, o no usar punto alguno. En *Linux*, lo más importante son los atributos del archivo, ya que son éstos los que hacen con que el archivo sea ejecutable o no, por ejemplo. Esto es más seguro y conlleva una mayor flexibilidad, a pesar de ser un poco confuso al principio.

6.2.12 VFS

Linux es, probablemente, el sistema operativo que soporta un mayor número de sistemas de archivos diferentes. Además de los sistemas *EXT2*, *EXT* y *Minix*, también soporta los sistemas *FAT16* y *FAT32* de *Windows*, el *HPFS* de *OS/2*, además de varios otros sistemas como el *proc*, *smb*, *ncp*, *iso9660*, *sysv*, *affs* y *ufs*.

El secreto para toda esta versatilidad es el uso del *VFS* o *Virtual FileSystem*, un divisor de aguas entre el sistema de archivos y el *Kernel* y los programas. La función del *VFS* es asumir toda la parte administrativa, traduciendo vía software, todos los detalles y estructuras del sistema de archivos y entregando sólo los datos al *Kernel*, que a su vez, los entrega a los programas.

Gracias al *VFS*, *Linux* puede trabajar en cualquiera de los sistemas de archivos soportados. Es por ello que existen varias distribuciones de *Linux* que se pueden instalar en particiones *FAT16* o *FAT32* (como *Winlinux*, que puede ser instalado en una carpeta de la partición de *Windows*) e incluso inicializar desde el CD-ROM, sin la necesidad de instalar el sistema en el equipo (en este caso, sólo para fines educativos o para experimentar con el sistema, ya que no se pueden guardar archivos, entre otras limitaciones). Pero claro, también existen desventajas. Al ser instalado en una partición *FAT32*, *Linux* se vuelve mucho más lento, pues el *VFS* tiene que emular muchas estructuras que no existen en el sistema.

6.2.13 Soporte al NTFS

El soporte para el sistema *NFTS* todavía estaba siendo implementado en *Linux*. El mismo *Kernel 2.4.9*, que mientras escribo este apartado es la última versión estable, sólo permite el acceso de lectura. Esto significa que podemos instalar una distribución de *Linux* y acceder a un disco duro formateado como *NTFS*, pero sólo podremos leer los datos, nada de modificar o grabar nuevos archivos.

Existe un proyecto para ofrecer el soporte completo al *NTFS*, el *Linux-ntfs*. Podemos visitar la página oficial en la dirección http://sourceforge.net/projects/linux-ntfs/. Ya existe una versión estable, pero el programa todavía no está completamente desarrollado como para ser lanzado.

Personalmente creemos que no tardarán mucho para conseguir soportarlo completamente en el propio *kernel*, ya que el sistema *NTFS* es el sistema de archivos nativo de *Windows 2000* y de *Windows XP*, lo que implica que cada vez es más usado.

6.2.14 Estructuras del EXT2

Como ya hemos dicho, el sistema *NTFS* y el sistema *EXT2* utilizan estructuras muy diferentes (y hasta cierto punto mucho más complejas) que las usadas en el sistema de archivos *FAT*.

Ahora que ya hemos tratado los sistemas *FAT16*, *FAT32* y *NTFS* usados por *Windows* ha llegado el momento de ver un poco más sobre el sistema usado en *Linux*.

Boot block: es donde todo empieza, pues es aquí donde se graban todas las informaciones necesarias para inicializar el sistema.

Inodes: los *inodes* almacenan información sobre cada archivo guardado. La función de los *inodes* es muy parecida a las entradas en el *MFT* del sistema *NTFS*. Cada *inode* almacena los detalles sobre un determinado archivo, incluyendo el tipo de archivo, los permisos de acceso, la identificación del usuario dueño del archivo, la fecha de creación y la última modificación, el tamaño y, por último, los punteros a los bloques de datos donde está almacenado el archivo. Al leer cualquier archivo, el *VFS* primero lee el *inode* correspondiente para llegar al archivo.

Directorios: en el sistema *EXT2* los directorios son tipos especiales de archivos, que almacenan una lista de todos los archivos y subdirectorios subordinados. En esta tabla se almacenan sólo los nombres y los *inodes* que representan a cada uno.

Enlaces: este es un recurso bastante versátil permitido por el *EXT2*. Estos enlaces funcionan de una manera muy parecida a los enlaces usados en las páginas *Web*. Cada enlace puede apuntar a un archivo o directorio cualquiera. Al acceder al enlace, automáticamente se accede al destino. Por ejemplo, podemos crear un enlace "*CD*" dentro del directorio raíz para acceder al *CD-ROM*. Al teclear *cd /root/cd* veremos los archivos del *CD-ROM*. En realidad, los enlaces no son más que *inodes* que apuntan al archivo o directorio en cuestión. Al abrir el enlace, el *VFS* lee las instrucciones y va directo a los sectores ocupados por el archivo.

6.2.15 EXT3

El *EXT3* es la evolución del sistema *EXT2*, que a pesar de no representar un avance tan grande como fue el *EXT2* respecto al *EXT*, trae nuevos recursos importantes. Lo más importante es una mejora en el sistema de tolerancia a errores. En el *EXT3*, el sistema mantiene un "diario" de todas las operaciones realizadas. Cuando ocurre cualquier error, un reinicio o bloqueo del sistema, el sistema consulta las últimas entradas del diario, para ver exactamente en qué punto ocurrió el error y corregir el problema automáticamente, en unos segundos.

En el *EXT2*, siempre que hay un error, el sistema ejecuta el *e2fsck*, un primo del *Scandisk*, que verifica *inode* por *inode* el sistema de archivos, en busca de errores. Este reconocimiento tarda varios minutos, además de que no siempre consigue evitar la pérdida de algunos archivos. Existe la opción de configurar las entradas en el diario para aumentar la velocidad de acceso pero, a cambio, sacrificando un poco de la fiabilidad en el caso de errores, o aumentar la tolerancia a los errores a cambio de una pequeña pérdida de rendimiento.

6.2.16 HPFS

Desde el inicio de la era PC, Microsoft e IBM venían trabajando juntas en el desarrollo del *MS-DOS* y otros programas para la plataforma PC. Pero, en 1990 la casa IBM y Microsoft no se entendieron y cada una se quedó con una parte del trabajo hecho, con el que intentaron tomar el liderazgo del mercado de sistemas operativos.

Algunos cuentan que IBM se quedó con la parte que funcionaba y Microsoft con el resto, pero la verdad es que a pesar de que el sistema operativo *OS/2* era técnicamente muy superior a *Windows 95*, fue éste el que se llevó el mercado, condenando al poderoso *OS/2* casi al olvido.

A pesar de que el *OS/2* también podía instalarse sin problemas en particiones formateadas como *FAT16*, era más recomendable usarlo en particiones formateadas con el sistema *HPFS*, su propio sistema de archivos propietario. En ellas, el *OS/2* presenta

un rendimiento muy superior, debido a la forma más eficaz de organización de los datos en este sistema de archivos.

Los recursos del *HPFS* se aproximan mucho a los permitidos por el *NTFS* de *Windows NT*. Se permiten los nombres de archivo de hasta 254 caracteres incluyendo espacios, particiones de hasta 512 GB, con unidades de destino de 512 Bytes. A pesar de ser eficiente, este sistema de archivos cayó en desuso junto con el sistema *OS/2*. Actualmente, el único sistema operativo que soporta el *HPFS* es *Linux*.

6.3 Sectores Defectuosos

Uno de los problemas que más aterroriza a los usuarios es la aparición de sectores defectuosos. Los sectores defectuosos o, sencillamente, *bad clusters*, son errores físicos en los discos magnéticos, fallos en la superficie de escritura que pueden surgir debido a picos de tensión o debido al envejecimiento del mismo soporte físico.

En el primer caso, no hay motivo para desesperarse, pues cuando el cabezal de lectura está leyendo un sector y se corta la energía de repente puede ser que el sector se acabe dañando, pero en este caso basta marcar el sector usando el *Scandisk* y continuar usando el disco con total normalidad. A menos que se produzcan nuevos picos de tensión, difícilmente aparecerán nuevos sectores defectuosos.

El segundo escenario, que normalmente ocurre en los discos duros con más de 2 o 3 años de uso, es más grave. Con el paso del tiempo, y tras sucesivas lecturas, la superficie magnética de los discos empieza a deteriorarse, haciendo que periódicamente aparezcan nuevos sectores defectuosos. Para que un sector sea marcado como defectuoso no es necesario que el sector falle siempre, sólo que falle durante la prueba del *Scandisk*. Por ello, en los discos duros muy viejos es normal que se marquen nuevos sectores defectuosos cada vez que se realiza el examen de la superficie. En este caso, no es aconsejable continuar usando el disco duro, al menos para guardar los datos más importantes.

Prácticamente, todos los discos duros modernos poseen una pequeña zona reservada al final del disco, que no es usada para grabar datos, pero sí para sustituir sectores defectuosos. En este caso, al ejecutar el programa adecuado, la dirección de los *clusters* con algún defecto es modificada y pasa a apuntar a uno de los sectores de la zona reservada. El *cluster* defectuoso deja de ser visto por el sistema, pasando a usarse su sustituto. Este cambio es realizado, directamente, en las direcciones físicas de los sectores y es completamente transparente al sistema operativo.

En realidad, la mayoría de los discos duros nuevos salen de fábrica ya con algunos sectores defectuosos, que representan mínimas imperfecciones en la superficie magnética del disco. Sin embargo, antes de que salgan de la fábrica, las direcciones de los *clusters* defectuosos son alteradas, apuntando a otros *clusters* de la zona reservada, de modo que el disco duro parezca inmaculado.

Este ajuste no puede realizarse con el *Scandisk* u otro de los programas de diagnóstico, debemos usar el formateador del propio fabricante. Los fabricantes disponen estos programas gratuitamente en *Internet*.

6.4 Las particiones

Hasta ahora, hemos visto que existen varios sistemas de archivos y que, generalmente, los sistemas operativos son compatibles con sólo uno o, como máximo, dos sistemas diferentes. Cómo llevar a cabo entonces la instalación de *Windows 2000/XP* y *Linux* en el mismo disco duro?

Para resolver esta encrucijada, podemos dividir el disco duro en unidades menores, llamadas particiones, cada una pasando a ser propiedad de un sistema operativo. Entonces, podemos dividir el disco en 2 particiones, una en NTFS (para *Windows 2000/XP*) y una en *EXT3* (para *Linux*). Desde el punto de vista del sistema operativo, cada partición es una unidad separada, casi como si hubiese dos o tres discos duros instalados en el equipo.

Cada partición posee su propio directorio raíz y su propia *FAT* o su correspondiente. La información sobre el número de particiones, su localización en el disco y el espacio ocupado por cada una es almacenada en la tabla de la partición, que comparte el primer sector del disco con el sector de arranque.

Antes de formatear el disco duro, es necesario particionarlo. Lo más común (y también lo más simple) es establecer una única partición que englobe todo el disco duro, pero dividir el disco en dos o más particiones conlleva algunas ventajas, como la posibilidad de instalar varios sistemas operativos en el mismo disco, permitir una mejor organización de los datos grabados y disminuir el espacio desperdiciado (*slack*), ya que con particiones pequeñas, los *clusters* también serán menores.

Para dividir el disco en dos o más particiones, ya sean dos particiones con el mismo sistema de archivos, o cada una utilizando un sistema de archivos distinto, necesitaremos un programa formateador que soporte los formatos de archivo que vamos a usar. Un buen ejemplo es el programa *Partition Magic* de la casa Power Quest (www.powerquest.com) que soporta, prácticamente, todos los sistemas de archivos existentes hoy en día. En el caso de pretender instalar varios sistemas operativos en el mismo disco duro, también necesitaremos de un administrador de arranque (*Boot Manager*), que es un pequeño programa instalado en el sector de arranque o en la primera partición del disco, que permite inicializar con cualquiera de los sistemas operativos instalados.

Durante el arranque, el administrador de arranque nos mostrará una lista con los sistemas operativos instalados y nos preguntará con cuál deseamos inicializar. Si, por ejemplo, tenemos *Windows 2000* y *Linux* instalados y cargamos *Windows 2000* por defecto y, de golpe precisamos hacer algo en *Linux*, bastará reiniciar el equipo y cargar *Linux*, pudiendo volver al *Windows 2000* en cualquier momento.

Otra cosa a considerar al dividir el disco duro en varias particiones es la velocidad de acceso a los datos. Como el disco duro gira a una velocidad fija, el acceso a las pistas más externas, que son más extensas, es más rápido que el acceso a las pistas internas, que son más cortas. En la práctica, la velocidad de acceso a la primera pista del disco (la más externa) es cerca de dos veces más rápida que el acceso a la última (la más interna).

Como las particiones se crean de forma secuencial, a partir del inicio del disco, el acceso a la primera partición siempre será más rápido que el acceso a las demás. Por ello, es aconsejable instalar en la primera partición el sistema operativo que utilizaremos más frecuentemente. También es acertado dividir el disco en dos particiones, dejando la primera para el sistema y los programas y la segunda para guardar los archivos.

6.5 La recuperación de datos

Para muchos, la peor pesadilla sería que el disco duro del equipo de trabajo, con todos los archivos importantes, dejase de funcionar o que los datos fuesen borrados por el ataque de algún virus.

La mejor prevención contra este tipo de accidentes todavía es la buena y vieja copia de seguridad, pero, en caso de que el daño ya se haya producido o los datos ya se hayan perdido y no exista ni sombra de una copia de seguridad actualizada, a continuación mostraremos algunas indicaciones para intentar recuperar los datos.

6.5.1 Los defectos más comunes

Los discos duros actuales todavía se basan en componentes mecánicos, que no son tan fiables como los componentes electrónicos. Los problemas físicos más comunes en los discos duros son, en primer lugar, los sectores defectuosos que surgen después de picos en la energía eléctrica o por el envejecimiento del soporte físico. Los sectores defectuosos no se pueden corregir, pero pueden ser marcados para que el resto del disco duro pueda seguir usándose.

Otro defecto común, esta vez más grave, es cuando deja de funcionar el motor de rotación del disco duro. En este caso, los *platters* dejan de girar y el disco duro deja de ser reconocido por el *BIOS* y el disco duro, sencillamente, "muere".

Otra posibilidad son los errores de posicionamiento de los cabezales de lectura. Esto ocurre cuando, por cualquier motivo, la controladora no consigue detectar la posición de los cabezales de lectura sobre los discos magnéticos. Este defecto es común en los discos duros antiguos, donde las señales magnéticas que permiten la identificación ya se han debilitado. En este caso, el disco duro gira pero se oirán algunos "clicks" y no será reconocido en el *BIOS*.

A pesar de ser más raro, la culpa también puede ser de la tarjeta controladora del disco duro. En este caso, la gama de problemas es grande. Puede ser que el disco duro sen-

cillamente "muera", que los *platters* giren pero el disco duro no sea reconocido en el *BIOS*, etc. En este caso deberíamos coger una lupa y examinar la placa lógica en búsqueda de *capacitors* o *chips* quemados, contactos rotos, que puedan ser los culpables de los defectos en su funcionamiento.

6.5.2 Y ahora, ¿qué?

Sin duda, esta no es una experiencia muy agradable. Existen compañías especializadas en la recuperación de los datos de los discos duros con errores, algunas de ellas equipadas con salas limpias y otros tipos de equipamientos, pero el gran problema es el precio de este tipo de servicio, que fácilmente puede sobrepasar los 600 euros. Además, no existe la garantía de que realmente se recuperarán los datos. Si preferimos intentarlo nosotros mismos, podemos intentar seguir algunas de las siguientes indicaciones.

Los datos se habían borrado pero el disco duro funciona

El modo a través en el que los datos son grabados en el disco duro permite que pueda recuperarse prácticamente cualquier dato anteriormente borrado. En realidad, cuando borramos un archivo, ya sea a través de *DOS* o del *Explorador de Windows*, sólo es eliminada la referencia a él de la *FAT*, la tabla grabada en el inicio del disco duro que almacena la localización de cada archivo en el disco.

Como la dirección anteriormente ocupada por el archivo es marcada como vacía en la *FAT*, el sistema operativo considera vacía la parcela del disco que ocupaba el archivo. Sin embargo, no es borrado hasta que se graba un nuevo dato sobreescribiendo el contenido anterior.

El paquete *Norton Utilities* posee una utilidad, *Rescue Disk*, que permite almacenar una copia de la *FAT* en disquetes. En el caso de que nuestro disco duro sea accidentalmente formateado por un virus, o por cualquier otro motivo, podremos restaurar la *FAT* con la ayuda de estos discos, volviendo a tener acceso a todos los datos como si no hubiese pasado nada.

Además de este programa existen otros programas extremadamente especializados en la recuperación de datos, en el caso de no tener una copia de seguridad de la *FAT*. La casa Ontrack tiene el suyo, *Easy Recovery* con versiones para los sistemas *FAT16*, *FAT32*, *NTFS*, *Novel Netware*, etc. Estos programas son capaces de recuperar archivos borrados, o un disco duro entero víctima de la acción de un virus, aunque la *FAT* haya sido borrada.

El programa lo hace basándose en las informaciones del final de cada *cluster* y en estadísticas. Realmente hacen un buen trabajo, recuperando prácticamente cualquier archivo que todavía no haya sido reescrito en el disco. Estos no son programas baratos. Existe una demostración gratuita que sólo muestra los archivos que se pueden recuperar en las distintas *Webs* de los desarroladores.

En la misma categoría, también encontramos el programa *Lost and Found* de la casa Power Quest. El modo de recuperación es parecido con el usado por el E*asy Recovery* y la eficiencia también es semejante, aunque su ventaja es la de ser más barato.

Usando cualquiera de los dos programas, la oportunidad de recuperar los datos que todavía no hayan sido reescritos es muy grande, siempre en el caso de que el disco duro no tenga ningún defecto de hardware.

Otra indicación es que algunos virus pueden bloquear el acceso al disco duro, haciendo que no consigamos acceder a él ni usando un disco de arranque. En este caso, iniciaremos el sistema con un disquete de inicio y usamos el comando *FDISK /MBR*, que rescribe el sector de arranque del disco duro. Luego, deberíamos pasar una antivirus actualizado.

El disco duro no es detectado por el BIOS

Existe un truco que acostumbra a salir bien y vale la pena intentarlo. Debemos poner el disco duro dentro de un plástico herméticamente cerrado y dejarlo en la nevera durante 4 horas o más. Parece raro, pero tiene su lógica, ya que el frío altera la posición de las marcas magnéticas de los *platters* (contracción del material) y algunos componentes del disco duro funcionan mejor a bajas temperaturas. A continuación, reinstalamos rápidamente el disco duro. Existe una posibilidad de que vuelve a funcionar durante algunos minutos, tiempo suficiente para hacer una copia de los datos más importantes y jubilarlo definitivamente. Por increíble que parezca, esto también acostumbra funcionar con los *CDs* rayados que dejan de funcionar.

También deberíamos visitar la *Web* del fabricante y configurar el disco duro manualmente en el *BIOS*, dando el número de pistas, sectores y cilindros. Esto, a veces, resuelve algunos casos raros.

También podemos probar dando el arranque usando un disquete con el administrador de arranque del fabricante, ya que muchas veces es capaz de acceder a la unidad, aunque no haya sido reconocida por el *BIOS*.

Si oímos el ruido de los *platters* girando, seguido de un "click", "click", "click", significa que la tarjeta controladora está intentando posicionar los cabezales de lectura, pero no lo consigue. Este defecto es común en los discos duros antiguos. Probaremos de apoyar el disco duro sobre una mesa, en la vertical, y dar un golpecito sobre el lado superior. Un golpe muy ligero, pues como los *platters* están girando un golpe más fuerte puede dañar los discos magnéticos y entonces los datos no se van a poder recuperar nunca más.

El disco duro "murió" y no oímos nada

En este caso, existen dos hipótesis posibles, el motor de rotación no está girando o existe algún defecto grave en la tarjeta lógica. Empezaremos comprobando los voltajes de la fuente de alimentación, sólo para eliminar la posibilidad de la existencia de

defectos en la fuente. También probaremos de instalarlo en la segunda controladora IDE o en otro equipo, verificaremos los *jumpers*, el cable *IDE*, etc. sólo para eliminar estas posibilidades más banales.

Luego, examinaremos la tarjeta lógica del disco duro en busca de algún daño visible. Intentaremos descubrir también las circunstancias en las que el disco duro dejó de funcionar.

El motor de rotación del disco duro no para de golpe, es como un ventilador, que va perdiendo potencia de forma gradual. Si el problema estuviese en el motor de rotación, podemos probar de dar unos golpecitos en el disco duro pasados unos 2 o 3 segundos tras enchufar el equipo. En muchos casos, el motor vuelve a girar.

Otra cosa que a veces funciona es asegurar el disco duro y girarlo rápidamente en el sentido contrario de las agujas del reloj justo después de enchufar el equipo. Este es el sentido de rotación de los discos. En caso de tener suerte, la ley de la inercia se encargará de dar el "empujón" que necesita el motor de rotación.

En el caso de encontrar algún defecto en la tarjeta lógica, pistas rotas, *capacitors* quemados, etc. las opciones son intentar corregir el defecto o intentar encontrar otro disco duro de la misma marca y modelo y coger "prestada" la tarjeta lógica. Recuperamos los datos inmediatamente, devolvemos la tarjeta al donante y jubilamos el disco duro defectuoso.

No tenemos nada que perder, la última sugerencia

Las indicaciones siguientes sólo deben usarse tras agotar todas las anteriores, pues permitirán que el disco duro funcione algunos minutos u horas más, tiempo suficiente para realizar una copia de los datos importantes, pero a cambio, el disco duro será definitivamente condenado.

El último recurso para intentar recuperar los datos es abrir el disco duro. Intentaremos hacerlo en un ambiente con el aire lo más limpio posible. De cualquier modo, el disco duro estará condenado a partir del momento en el que lo abramos. El polvo que entre empezará a destruir, lentamente, los discos magnéticos. Pero, antes de eso tendremos el tiempo suficiente para recuperar los datos.

Una vez abierto, enchufamos los cables de la fuente de alimentación. Enchufamos el equipo y vemos cuál es el comportamiento del disco duro. Si no gira, utilizaremos un dedo para darle un empujoncito y si el motor estuviese desgastado esto hará que gire. Con esto, bastará para poder reconocerlo en el *BIOS* y recuperar los datos. Al tocarlo con el dedo, debemos hacerlo en el eje de rotación y no en los discos magnéticos.

Debemos recordar que los datos están almacenados en los discos magnéticos del disco duro, el resto del mecanismo sólo tiene la función de darnos el acceso a los datos grabados. Otra posibilidad es encontrar una unidad del mismo modelo, abrirla y cambiar los discos magnéticos.

Con esto, el disco duro "bueno" sería aprovechado para leer los datos almacenados en los *platters* del disco duro malo. Esto debería realizarse dentro de una sala limpia, debido al problema del polvo.

6.6 Eliminar datos con seguridad

Como hemos visto, es perfectamente posible recuperar datos aunque el disco duro haya sido formateado. En realidad, las técnicas avanzadas de recuperación de datos utilizadas por algunas empresas especializadas y, por regla general, por los propios fabricantes, son capaces de recuperar los datos anteriormente grabados aunque ya hayan sido sobreescritos. Es decir, aunque hayamos formateado el disco y grabado datos encima, todavía es posible recuperar los datos anteriores, a partir de sutiles vestigios dejados en la superficie magnética siempre que se ha alterado un bit.

Claro que estas técnicas exigen aparatos especiales y son extremadamente laboriosas y caras, pero no dejan de ser una posibilidad. Según la casa Ontrack, para que no exista la posibilidad de recuperar los datos, deben realizarse, al menos, 7 regrabaciones sobre los datos originales.

Esto conlleva otro problema si realmente necesitamos borrar los datos grabados en el disco duro, de manera que nadie pueda recuperarlos posteriormente. Podríamos formatear el disco y sobreescribir los datos varias veces o utilizar un programa que hace esta tarea de forma automática, como el *Data Eraser* de la casa Ontrack. La principal ventaja es que el programa realiza la regrabación en unos minutos.

6.7 Comprimir archivos

La compresión de archivos siempre ha sido un recurso muy utilizado y su origen se confunde con la propia historia de la computación. A través de la compresión, podemos aumentar considerablemente la cantidad de archivos y programas que pueden grabarse en el mismo espacio físico.

Por ejemplo, un disco duro de 200 MBytes puede guardar fácilmente 300 MB, 400 MB o hasta más de archivos una vez comprimidos. Conforme los discos duros y otras formas de almacenamiento fueron aumentando en capacidad, el uso de la compresión fue volviéndose menos necesario, pero todavía es un recurso bastante utilizado.

Comprimir archivos es un simple proceso de sustitución. Por ejemplo, cada carácter de texto ocupa 8 bits, lo que nos da un total de 256 combinaciones posibles. El conjunto de caracteres *ASCII* prevé el uso de todas las 256 combinaciones, sin embargo, por regla general, utilizamos sólo letras, números y los acentos. En una imagen BMP, con 256 colores, también usamos 8 bits para representar cada punto de color, pero en una imagen siempre tenemos grandes áreas con puntos del mismo color. En un archivo ejecutable, siempre tenemos comandos e información repetitiva. En todos los casos

tenemos información redundante que puede ser perfectamente sustituida por unos códigos menores que ocupen menos espacio.

Existen varios algoritmos de compresión, que tienen en cuenta varios tipos de sustituciones para los diferentes tipos de archivos. Sin embargo, una vez comprimido, cualquier archivo deja de ser utilizable. Para poder usarlo nuevamente debe realizarse el proceso inverso para tener nuevamente el archivo original. Existen varios tipos de compresión de archivos, cada tipo con sus ventajas y desventajas:

Compresión de archivos individuales: consiste en comprimir archivos utilizando programas como *Winzip*, *Winrar*, *Arj*, etc. Probablemente ya hemos trabajado con uno de estos programas y debemos saber que algunos archivos, como los textos e imágenes *BMP* permiten una tasa de compresión mucho más grande que otros. Esto acontece porqué estos archivos poseen una cantidad muy grande de información redundante.

El único problema es que, usando uno de estos programas para comprimir archivos, es necesario descomprimirlos antes de poder utilizarlos de nuevo. Estos programas se están usando mucho hoy en día, sobre todo en *Internet*, donde es muy práctico comprimir los archivos antes de enviarlos.

Compresión de volúmenes: al contrario de comprimir archivos individualmente es posible crear volúmenes comprimidos, usando programas como *DriveSpace* (que acompaña a *Windows 95/98*) o *Stacker*. Por regla general, se compacta una partición entera del disco duro. Todos los archivos grabados en esta partición pasan a formar parte de un volumen comprimido, un único archivo grande. En este caso, queda residente en memoria un *driver* de compresión que sirve como intérprete, comprimiendo los archivos antes de grabarlos y descomprimiéndolos conforme son leídos, entregándolos al sistema operativo en su formato original, todo ello en tiempo real.

Como los datos son grabados de forma comprimida es posible grabar un 50 o 60% más de datos. La desventaja es que al ser utilizado el procesador para comprimir y descomprimir los archivos, tenemos una disminución en el rendimiento general del equipo.

Compresión de archivos por el sistema operativo: este es el método permitido por el *Windows 2000/XP* y *Windows NT* en particiones *NTFS* para comprimir individualmente archivos o carpetas, manteniendo los datos pero ahorrando espacio en disco. Otra ventaja es que, debido a que los archivos se comprimen individualmente, no existe el riesgo de una pérdida de datos.

6.8 Interfaces

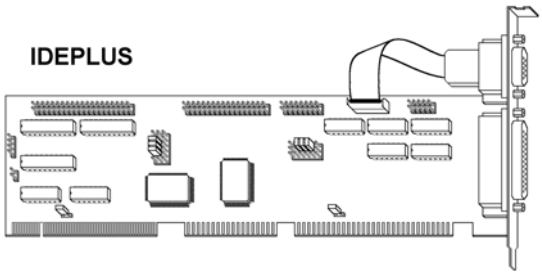
Así como una tarjeta gráfica se conecta en un *slot PCI* o *AGP*, para poder comunicarse con el resto del sistema el disco duro necesita estar conectado a alguna interfaz. Una interfaz de disco duro no es más que un medio de comunicación, una carretera por

donde puedan transitar los datos que entran y salen del disco duro. De nada sirve un disco duro muy rápido si la interfaz no permite que se comunique con el resto del sistema usando toda su velocidad.

Actualmente son usados dos formatos de interfaces de disco: el *IDE* (también llamado *ATA*) y el *SCSI*, con una mayor implementación del *IDE*.

6.8.1 IDE

Las placas base más antiguas no poseían interfaces *IDE*. En ellas, la interfaz *IDE* debía ser adquirida por separado y encajada en un *slot* vacío. Si tuviéramos la posibilidad de examinar el hardware de un *486* antiguo, veríamos una tarjeta *ISA*, *EISA* o *VLB*, que incluye la interfaz *IDE*, además de la interfaz para la unidad de disquetes, un puerto paralelo, dos puertos serie y un puerto para el joystick. Esta tarjeta era conocida como "*super IDE*".

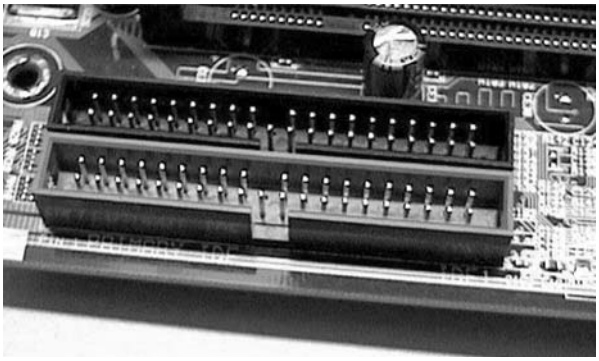


Una tarjeta super IDE

Todas las placas base actuales poseen, además de dos puertos serie y un puerto paralelo, dos interfaces *IDE* integradas, llamadas controladora primaria y controladora secundaria.

Cada controladora soporta dos dispositivos, lo que permite como máximo 4 dispositivos *IDE* en un mismo equipo. Para ello, uno de los dispositivos debe configurarse como *master* (maestro) y el otro como *slave* (esclavo), configuración que se realiza a través de *jumpers*.

El cable *IDE* posee tres encajes, uno que es enchufado a la placa base y otro para cada uno de los dos dispositivos posibles. Aunque sólo tengamos un dispositivo *IDE*, deberemos conectarlo en el conector del final del cable, nunca en el conector del medio. El motivo para esto es que, enchufando el dispositivo en el conector del medio, el cable queda sin terminación, haciendo que los datos vayan hasta el final del cable y retornen como paquetes sobrantes, interfiriendo en el envío de los paquetes correctos y causando una disminución en la velocidad de transmisión. Este fenómeno también ocurre en los cables coaxiales de red, donde se instalan terminadores en las dos puntas del cable, que absorben las transmisiones evitando los paquetes sobrantes. En el caso de los dispositivos *IDE*, el dispositivo enchufado en la punta del cable funciona como terminador.



Los conectores IDE de una placa base

Existen varios modelos de interfaces *IDE* que ofrecen diferentes modos de trabajo. Estos modos de trabajo son llamados "*Pio*" y determinan la velocidad y los recursos de la interfaz.

Las placas más antiguas, como las placas para procesadores *Pentium* que utilizan los *chipsets* FX y VX, sólo soportan el modo *Pio* 4, siendo capaces de transferir datos a 16.6 MBytes por segundo. Las placas un poco más recientes, también soportan el modo *Ultra DMA 33* o *Ultra DMA 66*.

Probablemente, ya debemos haber oído hablar del modo *Ultra DMA*, también llamado *Ultra ATA*. Este modo de trabajo conlleva varias ventajas sobre el antiguo *Pio Mode* 4, como una mayor tasa de transmisión de datos, que pasa a ser de 33 MBytes por segundo. Sin embargo, la principal ventaja del modo *UDMA* es que permite que el disco duro acceda directamente a la memoria *RAM*.

Usando el modo *UDMA*, al contrario de que el procesador tenga que transferir los datos del disco duro para la memoria *RAM* y viceversa, ahora sólo debe realizar una solicitud al disco duro para que éste mismo haga el trabajo. Es evidente que este modo de trabajo aumenta el rendimiento del sistema, pues ahorra al procesador el trabajo de los traspasos de datos, quedando libre para ejecutar otras tareas.

El *Pio Mode* 4 permite el uso del modo *Multiword DMA* 2, que también permite el acceso directo a la memoria, aunque de una forma un poco menos eficiente.

Para hacer uso de las ventajas del modo *UDMA* es necesario que el disco duro también ofrezca el soporte para esta tecnología. Todos los modelos de discos duros más recientes incluyen el soporte al modo *UDMA*, manteniendo además la compatibilidad con las controladoras más antiguas. En el caso de tener en la placa base una controladora que sólo soporte el modo *Pio* 4, el disco duro funcionará normalmente, aunque limitado a las características de la interfaz. *Windows 98* posee soporte nativo para los discos duros *UDMA*; en el caso de *Windows 95* es necesario instalar los *drivers UDMA*, que se encuentran generalmente en la carpeta "*IDE*" del *CD* de *drivers* que acompaña a la placa base.

Existen 8 modos de trabajo en las interfaces *IDE*, que van desde el *Pio Mode* 0 al nuevo *UDMA* 100, que mantiene los recursos del *Ultra DMA* soportando mayores velocidades en la transmisión de los datos.

Es preciso recordar que estas velocidades son el flujo máximo de datos permitido por la interfaz, y no se corresponde necesariamente con la velocidad de trabajo del disco.

Modo de trabajo	Tasa máxima de transferencia
PIO MODE 0	3,3 MB/s
PIO MODE 1	5,2 MB/s
PIO MODE 2	8,3 MB/s
PIO MODE 3	11,1 MB/s
PIO MODE 4	16,6 MB/s
UDMA 33	33,3 MB/s
UDMA 66	66,6 MB/s
UDMA 100	100 MB/s

La mayoría de los discos duros actuales son compatibles con el *UDMA 66/UDMA 100,* esta nueva interfaz permite tasas de transmisión próximas a las de las controladoras *SCSI*. Claro que los 66 MB/s o 100 MB/s permitidos no son necesariamente la velocidad alcanzada por el disco duro, aunque sin embargo, siempre existe una ganancia de rendimiento sobre el mismo disco duro instalado en una puerto *UDMA 33*, no tanto en las aplicaciones domésticas, pero sobre todo en las aplicaciones que trabajan con un gran flujo de datos, como es el caso de un servidor de red.

El conector de las interfaces *UDMA 66/UDMA 100* poseen los mismos 40 *pins* de los otros formatos, así como una compatibilidad retroactiva con cualquier disco duro *IDE*. Sin embargo, los cables poseen 80 vías, 40 usadas para transportar datos y 40 como tierra. Los hilos son intercalados, un tierra, otro de datos etc., y esta disposición atenúa las interferencias y permite alcanzar los 66 o 100 MB/s.

Otro requisito impuesto por el formato es que los cables no tengan más de 45 cm de largo, pues el cable actúa como antena, captando interferencias externas. Cuánto más largo es el cable, más fuertes serán las interferencias.

Para activar el modo *UDMA 66/100*, necesitamos que tanto el disco duro como la placa base sean compatibles con el modo, siendo obligatorio el uso del cable especial de 80 vías que es suministrado junto con la placa base.

Algunas veces los *drivers* necesarios para dar soporte a estos modos también se pueden instalar directamente, a través de un programa de configuración incluido en el *CD* que acompaña a la placa base. En caso de duda, basta consultar el manual de la placa base.

Más puertos IDE

Si las placas base actuales disponen de sólo dos puertos *IDE*, y cada puerto permite la conexión de dos dispositivos, entonces tendríamos un límite de 4 dispositivos *IDE* por equipo. Pero es posible instalar dos puertos *IDE* más, llamados puerto terciario y cuaternario, totalizando 4 puertos *IDE* en el mismo equipo.

Estas placas base con interfaces extra pueden adquirirse por separado, aunque sean un poco difíciles de encontrar. Para que la controladora terciaria o cuaternaria funcione, es necesario que esté configurada para funcionar en tal posición, usando una *IRQ* y dirección de *I/O* diferente de la de las dos controladoras integradas en la placa base.

Existen, sin embargo, tarjetas *IDE* más contemporáneas, generalmente *PCI*, que pueden tener su dirección de *IRQ* e *I/O* configurada a través de *jumpers*; en este caso, basta elegir una IRQ libre. Al usar el bus *PCI*, estos puertos son bastante rápidos, generalmente funcionando en *PIO MODE* 4.

A pesar de que muchas controladoras terciarias y cuaternarias no son reconocidas por el *BIOS*, podemos hacerlas funcionar sin mucha dificultad en *DOS* usando los *drivers* que acompañan a la placa base. Ya en *Windows 98* eso no será necesario, pues detecta normalmente las interfaces, bastando usar la utilidad *Agregar hardware* del *Panel de control*.

Existen casos de placas base que ya vienen con cuatro interfaces *IDE* integradas, como por ejemplo la Abit BE6, que viene con dos puertos *UDMA 33* y dos puertos *UDMA 66*.

6.8.2 RAID en discos IDE

El *RAID* siempre fue un recurso bastante usado en servidores y en ordenadores de gran porte para optimizar el acceso al disco y añadir tolerancia a errores. Actualmente, este recurso está al alcance de cualquier usuario doméstico que tenga las condiciones de comprar más de un disco duro. Es posible usar *RAID* tanto en discos duros *IDE* como en discos duros *SCSI*.

Una controladora *RAID* permite combinar varios discos duros, permitiendo aumentar el rendimiento, haciendo que varios discos trabajen como si fuesen sólo uno; y la fiabilidad, usando un sistema de reflejos.

Además de las controladoras *SCSI*, que no son una solución muy viable para un usuario doméstico ya que los discos duros *SCSI* son caros, también existen controladoras *RAID IDE* que, además de ser más baratas, permiten usar los discos duros *IDE* que existen en el mercado.

Las controladoras *RAID* acostumbran a ser una tarjeta de expansión que se debe conectar a uno de los *slots PCI* del equipo. La tarjeta sustituye a las interfaces *IDE* de la placa base y, por ello, es detectada automáticamente por el sistema operativo, ya sea

Windows 95/98 como *Windows 2000/XP* o el mismo *Linux*, facilitando mucho el proceso de instalación. La tarjeta lleva las dos salidas *IDE* normales. Cada salida permite conectar dos discos duros, lo que otorga la posibilidad de instalar hasta 4 discos duros *IDE*. A continuación pasamos a ver las distintas posibilidades.

RAID 0 (Striping)

Con esta posibilidad es posible combinar 2, 3 o 4 discos duros, que se accederán como si fuesen sólo uno, aumentando radicalmente el rendimiento del acceso al disco. Los datos grabados son fragmentados y los distintos fragmentos son repartidos por todos los discos. A la hora de realizar la lectura, los discos son accedidos a la vez.

En la práctica, tenemos un aumento de rendimiento cercano al 98% usando dos discos, al 180% usando 3 discos y muy próximo al 250% usando 4 discos. Las capacidades de los discos son sumadas. Usando 3 discos de 8 GB por ejemplo, pasaremos a tener un disco de 24 GB.

Este modo es el mejor desde el punto de vista del rendimiento, pero es malo desde el punto de vista de la fiabilidad, pues al fragmentarse los datos, en el caso de que falle sólo un disco, perderemos los datos grabados en todos los discos. Una observación importante sobre este modo es que debemos usar discos duros idénticos. Es posible usar discos de diferentes capacidades, pero el rendimiento quedará limitado al rendimiento del disco más lento.

RAID 1 (Mirroring)

Este modo permite usar 2 discos duros, en el que el segundo almacenará una imagen idéntica del primero. En la práctica, es como si tuviésemos un sólo disco duro instalado, pero en el caso de que falle el disco principal por cualquier motivo, tendremos una copia de seguridad almacenada en el segundo disco. Este es el modo ideal si deseamos aumentar la fiabilidad del sistema.

La observación sobre este modo es que al usar dos discos duros deberíamos poner cada uno en una de las dos interfaces *IDE* de la tarjeta controladora, ya que esto mejorará el rendimiento. En el caso de que los dos discos estén en la misma interfaz, como maestro y esclavo, deberíamos reiniciar el equipo en el caso de que fallase el primer disco duro (este problema ocurre con todas las controladoras *RAID IDE*). Usando un disco en cada interfaz, la controladora realizará el cambio automáticamente, sin necesidad de ningún reinicio.

RAID 10 (Mirror/Strip)

Este modo sólo puede usarse en el caso de tener 4 discos duros. Los dos primeros funcionarán en *Striping*, doblando el rendimiento, mientras que los otros dos guardarán una imagen de los dos primeros, asegurando la fiabilidad. Este modo es una combinación de los dos primeros.

Configuración

Después de insertar la tarjeta controladora en un *slot PCI* y conectar los discos duros a ella, deberemos pulsar la combinación de teclas <Ctrl> + <F> (la combinación de teclas puede variar en función del modelo de la controladora) durante la inicialización del equipo para acceder al *BIOS* de la tarjeta. Entonces tendremos acceso a la utilidad de configuración de la tarjeta *RAID*. La interfaz es semejante a la de las utilidades de configuración de las tarjetas *SCSI*.

Dentro del *Setup*, escogemos la opción de autoconfiguración y luego el modo de trabajo, entre los 3 que hemos visto anteriormente y nuestro sistema *RAID* estará listo para ser usado. En el mismo menú encontramos las opciones de deshacer el *RAID* (*Delete Array*). Recordemos que al deshacer un *RAID 0* perderemos todos los datos de los discos duros.

Usando el modo *Striping* los discos serán vistos por el sistema como si fuesen uno sólo, esto significa que realizaremos las particiones y los accesos a los discos como si tuviésemos sólo un disco duro instalado. Usando el modo *Mirroring*, desde el punto de vista del sistema operativo sólo existirá también un disco duro instalado. La propia controladora se encargará de copiar los datos en el segundo disco duro.

Por regla general, también tendremos la opción de configurar el "*stripe block*", donde es posible definir cuál será el tamaño de los bloques de datos en los que la controladora fragmentará los archivos al usar los modos *RAID 0* o *RAID 10*.

Básicamente, si se almacenasen muchos archivos pequeños, un *stripe block* de 32 KB funcionaría mejor, mientras que si se almacenasen archivos mucho más grandes, 128 KB sería un valor más adecuado. El valor por defecto de 64 KB es adecuado para la mayoría de las situaciones.

Además de las controladoras dedicadas, existe también la opción de comprar una placa base con una controladora *RAID* integrada. Esta es la opción más barata, ya que la plusvalía en el precio de la placa base es menor que el coste de una controladora separada. Pero, no debemos dejar de verificar si la placa base soporta los tres modos de *RAID*. Muchas placas sólo soportan el modo *RAID 0*. En las placas base con *RAID* encontraremos 4 interfaces *IDE*. Las dos primeras son reservadas para los discos duros en *RAID*, mientras que las otros dos son para su uso general.

Una última observación es que tanto *Windows 2000 Server* como *Windows NT Server* ya traen un servicio que permite usar estos mismos recursos, pero usando las controladoras *IDE* de la placa base.

6.8.3 SCSI

Una alternativa a las interfaces *IDE* son las controladoras *SCSI* (*Small Computer Systems Interface*). Estas controladoras son ensambladas en un *slot* disponible de la placa base.

Las controladoras y discos duros *SCSI* son superiores a los *IDE* en varios aspectos, aunque no son tan populares debido al precio. Desde el punto de vista de un usuario doméstico, las ventajas del *SCSI* no justifican su alto precio. Pero en equipos de alto rendimiento, como servidores de red, el uso del *SCSI* es casi obligatorio.

En una controladora *SCSI* podemos usar hasta 15 dispositivos de forma simultánea (ya que un *ID* es usado por la propia controladora) sin que haya ninguna degradación de rendimiento, como pasa cuando usamos más de un dispositivo *IDE* en la misma controladora.

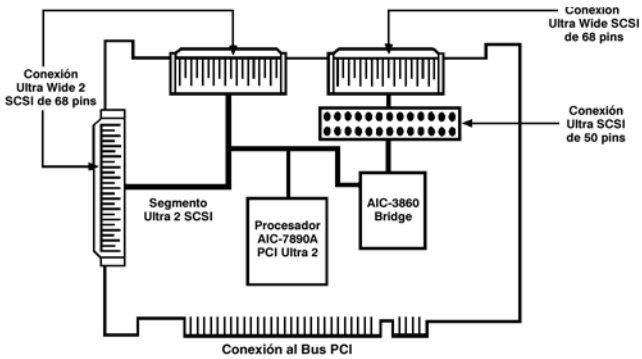
Otra gran ventaja del *SCSI* es una menor utilización del procesador cuando se accede al disco duro, porque prácticamente todo el trabajo es realizado por los propios discos duros (bajo la orientación de la controladora), y no por el procesador.

Una tarjeta controladora SCSI

Básicamente, el procesador sólo necesita informar a la controladora qué datos deben ser transferidos, donde están grabados y hacia dónde serán transferidos, para que la controladora pueda hacer el resto del trabajo, avisando al procesador cuando haya terminado. Durante este tiempo, el procesador quedará libre para ejecutar otras tareas. Aunque las interfaces *IDE UDMA* también ofrecen este recurso, éste es implementado de manera mucho más transparente y eficiente en las controladoras *SCSI*.

El primer modelo de controladora *SCSI* apareció en 1986. Era una controladora de 8 bits, que permitía un bus de datos de hasta 5 MBytes por segundo. Esta controladora antigua es llamada *SCSI 1*. En 1990, fue lanzada la segunda generación de controladoras SCSI, llamadas *SCSI 2*. Estas nuevas controladoras ya eran mucho más rápidas, incluyendo también otros recursos, como el soporte de hasta 15 dispositivos por controladora, en contra de los 7 soportados por las primeras controladoras.

Posteriormente, han sido lanzados los modelos *Ultra SCSI*, *Ultra SCSI 2* y *Ultra SCSI 3*, que permiten tasas de transmisión de datos todavía mayores, incluyendo algunos recursos nuevos.



El modo de conexión de una controladora SCSI

Además de la división en modelos, las interfaces *SCSI* también se dividen en controladoras *Narrow* y *Wide*. Las controladoras *Narrow* trabajan con tranferencias de datos de 8 bits y soportan el uso de hasta 8 dispositivos por interfaz. Las controladoras *Wide*, por su parte, trabajan con transferencias de 16 bits.

Los cables también cambian; las controladoras *Narrow* usan cables de 50 vías, mientras que las controladoras *Wide* usan cables de 68 vías. En la tabla siguiente podemos observar las distintas tasas máximas de transferencia de datos soportadas por cada modelo de controladora *SCSI*.

Modelo	Controladora Narrow SCSI	Controladora Wide SCSI
SCSI 1	5 MB/s	10 MB/s
SCSI 2 (Fast SCSI)	10 MB/s	20 MB/s
Ultra SCSI (Fast 20)	20 MB/s	40 MB/s
Ultra SCSI 2 (Fast 40)	40 MB/s	80 MB/s
Ultra SCSI 3 (Fast 80)	80 MB/s	160 MB/s

Como en los discos *IDE*, esta es la tasa de transferencia de datos permitida por la controladora, y no la velocidad de trabajo de los discos. Podremos pensar entonces, cuál es la ventaja de una controladora rápida si ningún disco es capaz de utilizar toda su velocidad.

No podemos olvidar, sin embargo, que en una interfaz *SCSI* podemos conectar hasta 15 dispositivos, entre discos duros, CD-ROMs, grabadoras de *CD/DVD*, escáneres y otros dispositivos. Eso es particularmente aplicable en los servidores, donde es común el uso de varios discos duros. En este caso, todos los periféricos compartirán el mismo bus de datos, utilizando toda su capacidad.

Otro factor es que los discos duros más rápidos son *SCSI*. Esto no significa que un disco duro es más rápido solamente por ser *SCSI*, pero las tecnologías más nuevas y

caras son utilizadas primero en los discos *SCSI*, utilizándose en los discos *IDE* después de que ya sean más baratas.

La instalación de periféricos *SCSI* también es diferente de la instalación de los periféricos *IDE*. Primero debemos comprar un cable con los encajes suficientes para el número de dispositivos que pretendemos instalar. En el dispositivo de la punta deberemos instalar un terminador, que le permite a la controladora encontrar el final del circuito. En la mayoría de los casos, el terminador es encajado en el dispositivo, pero en algunos casos basta cambiar la posición de una llave. En caso de duda, basta consultar el manual.

Para poder ser accedidos por la controladora, cada dispositivo *SCSI* posee una única dirección de identificación, o *ID*. Las controladoras *Narrow* poseen 8 direcciones, numeradas del 0 al 7, mientras que las controladoras *Wide* disponen de 16 direcciones, numerados del 0 al 15. Como la dirección 0 es usada por la propia controladora, sobran 7 o 15 direcciones para los demás dispositivos.

6.8.4 RAID con discos duros SCSI

Actualmente, el disco duro es uno de los componentes más lentos en cualquier PC. Mucho más lento que el procesador, que la memoria *RAM*, que la tarjeta gráfica, etc.

El gran problema es que debido a su naturaleza mecánica, no es posible producir discos duros más rápidos con la tecnología actual, al menos a un coste aceptable. En el futuro, este problema será resuelto con el uso de discos duros en estado sólido, donde se usará memoria *Flash* (u otra tecnología que surja en los próximos años), al contrario de los discos magnéticos. De cualquier forma, esta es una solución todavía un poco distante. Los discos duros en estado sólido ya existen para uso militar y en algunos servidores de alto rendimiento, pero su coste es muy alto.

RAID significa *Redundant Array of Inexpensive Disks*, o disposición redundante de discos baratos. La idea es un sistema en el que sean combinados varios discos duros para aumentar el rendimiento. En un nivel más complejo, el *RAID* también puede usarse para mejorar la fiabilidad del equipo, a través de reflejo o paridad. En un sistema *RAID 1*, donde tenemos dos discos duros, donde el segundo almacena una copia fiel de los datos del primero, aunque uno de los discos duros se rompa de golpe el sistema continúa intacto, funcionando como si no hubiese pasado nada.

Antes hemos visto cómo montar un sistema *RAID* usando discos duros *IDE*, ahora es el momento de explicar el sistema *RAID* usando discos duros *SCSI*. El primer paso para montar un sistema *RAID SCSI* es comprar una controladora *SCSI* con soporte para *RAID*. La mayoría de las controladoras *SCSI* actuales soportan el *RAID*.

Las controladoras se diferencian por los recursos que traen. En primer lugar, existen controladoras *SCSI* de 8 bits y de 16 bits; las de 8 bits permiten usar hasta 7 duros, mientras que las de 16 bits permiten un máximo de 15 y son dos veces más rápidas.

Otra diferencia es el formato soportado por la controladora, que dicta la velocidad de la interfaz. Las tarjetas *Ultra SCSI* (*Fast 20*) trabajan a 20 MB/s (tarjetas de 8 bits) o 40 MB/s (tarjetas de 16 bits). Las tarjetas *Ultra SCSI 2 (Fast 40*) operan a 40 MB/s (8 bits) o 80 MB/s (16 bits). Las tarjetas *Ultra SCSI 3* (*Fast 80*) son las más rápidas, trabajando a 80 MB/s o 160 MB/s.

También existen controladoras con un búfer incrustado. Este búfer funciona como una caché de disco, mejorando el rendimiento, sobre todo en las operaciones de escritura. Pueden encontrarse tarjetas con 32 MB de búfer. Naturalmente, cuanto más búfer, más cara es la tarjeta.

Resuelto el problema de la tarjeta *SCSI*, ya sólo queda elegir el modo de *RAID* que usaremos. El modo puede configurarse a través del *BIOS* de la tarjeta *SCSI*, al que se accede pulsando una combinación de teclas durante el arranque. La combinación más común es <Ctrl> + <C>. A continuación pasamos a ver los modos *RAID* disponibles en las tarjetas *SCSI*.

RAID 0 (Striping)

Es el modo que permite obtener el mejor rendimiento posible, sacrificando parte de la fiabilidad. Todos los discos pasan a ser accedidos como si fuesen una única unidad. Al ser grabados, los archivos son fragmentados entre los distintos discos, permitiendo que los fragmentos puedan leerse/grabarse a la vez. El problema es que en el caso de que cualquiera de los discos duros presente problemas, se perderán los datos almacenados en todos los discos duros, ya que cualquier archivo queda inservible en caso de que se pierda una parte del mismo.

RAID 1 (Mirroring)

Es el famoso sistema de reflejo o espejo, conseguido usando dos discos duros. Uno de ellos almacena los datos mientras el otro almacena una copia exacta de los mismos datos. En el caso de que uno de los discos duros pare, es automáticamente substituido por el otro y el sistema continúa intacto. En la mayoría de las controladoras *RAID SCSI* es posible realizar el cambio del disco duro defectuoso "en caliente", con el equipo enchufado, recurso que todavía no está disponible en las controladoras *RAID IDE*.

RAID 2

Es un modo que ya no es utilizado. El *RAID 2* consiste en incluir códigos de corrección de errores en cada *cluster* de datos grabado. Sin embargo, todos los discos duros actuales ya vienen con sistemas de corrección de errores.

RAID 3

El *RAID 3* usa un sistema de paridad para mantener la integridad de los datos. En un sistema con 5 discos duros, los 4 primeros servirán para almacenar datos, mientras que el último almacenará los códigos de paridad.

En los 4 primeros discos duros tenemos un sistema *RAID 0*, donde los datos se distribuyen entre los 4 discos y el rendimiento se ve multiplicado por 4. Sin embargo, los códigos almacenados en el 5 disco duro permiten recuperar los datos en el caso de que se pare uno de los 4 discos duros. La recuperación se realiza usando los códigos de corrección de errores combinados con los datos distribuidos en los otros discos.

Es posible aplicar el *RAID 3* en sistemas con más discos duros, y uno siempre almacenará los códigos de corrección. Claro que este sistema funciona sólo en el caso de que un único disco duro presente problemas.

Si se da la mala suerte de que dos o más discos presentan problemas a la vez, o antes de que la controladora termine la reconstrucción de los datos, se pierden todos los datos de todos los discos duros.

La paridad consiste en añadir un bit adicional a cada grupo de bits. En la memoria *RAM*, existe un bit de paridad para cada 8 bits de datos, mientras que en el *RAID 3* tenemos un bit extra para cada 4 bits de datos. Si dentro de estos 4 bits existe un número par de bits 1, entonces el bit de paridad es 0. Si existe un número impar de bits 1, entonces el bit de paridad es 1.

A continuación veamos un ejemplo para entenderlo un poco mejor.

	Bits	**Paridad**		
1	0	1	0	0
1	1	1	0	1
0	0	0	0	0
1	0	0	1	0

Veamos que gracias al bit de paridad sólo es posible saber si dentro del grupo de 4 bits existe un número par o impar de bits 1. Pero, eso ya es suficiente para recuperar cualquiera de los 4 bits perdido, siempre que se respeten dos condiciones:

a) Que sólo se pierda un bit de cada grupo.

b) Que sepamos qué bit se ha perdido.

En el *RAID 3* cada uno de los bits de cada grupo queda guardado en un disco duro. Cuando uno de los discos deja de funcionar, la controladora sabe exactamente qué bits han sido perdidos y tienen las condiciones para recuperarlos.

En la primera línea tenemos dos bits 1 y un bit 0. Si el bit de paridad es 0, significa que tenemos un número par de bits 1. Como ya tenemos dos bits 1, entonces está claro que el bit que falta es un 0. En la segunda línea tenemos dos bits 1 y un bit 0. Como el bit de paridad es 1, significa que tenemos un número impar de bits 1. Como sólo tenemos dos, significa que el bit que falta es un bit 1.

	Bits	Paridad		
1	?	1	0	0
1	?	1	0	1
0	?	0	0	0
1	?	0	1	0

Todo lo que necesita hacer la controladora para recuperar los datos es repetir este cálculo hasta recuperar todos los datos. Para un ordenador esto es muy simple.

El único problema es que cuando uno de los discos duros deja de funcionar, el equipo también queda paralizado. Sólo después de sustituir el disco duro y que la controladora recupere los datos el sistema vuelve a funcionar con total normalidad.

RAID 4

Este modo es parecido al *RAID 3*, ya que uno de los discos es dedicado a la tarea de almacenar los códigos de paridad, pero la forma como los datos son grabados en los demás discos es diferente. En el *RAID 3* los datos son divididos, y cada fragmento es guardado en un disco diferente. Esto permite ganar velocidad tanto en la grabación como en la lectura de los datos.

En el *RAID 4* los datos son divididos en bloques, pedazos mayores que en el *RAID 3*. Con esto, es posible leer varios archivos a la vez, lo que es útil en algunas aplicaciones, aunque el proceso de grabación es mucho más lento que en el *RAID 3*.

El *RAID 4* presenta un buen rendimiento en aplicaciones que necesitan leer una gran cantidad de archivos pequeños. En el *RAID 4* el tiempo de reconstrucción de los datos es mayor que en el *RAID 3*.

RAID 5

Es un sistema más que se basa en el uso de paridad para garantizar la integridad de los datos. La diferencia con el *RAID 3* es que en vez de dedicar un disco duro a esta tarea, los datos de corrección son repartidos entre los discos.

La ventaja sobre el *RAID 3* es que se alcanzan unas tasas de lectura un poco más altas, pues es posible leer los datos a partir de todos los discos de forma simultánea. Sin embargo, las grabaciones de datos son un poco más lentas.

El *RAID 5* puede implementarse a partir de 3 discos duros. A pesar de que los datos de paridad son repartidos entre los discos, el espacio equivalente al de un disco duro es dedicado a estos datos. Por ejemplo, si usamos 4 discos de 20 GB cada uno, tendremos 60 GB para datos y 20 GB para los códigos de paridad.

RAID 6

Es un modo relativamente nuevo, sólo soportado por algunas controladoras. Es semejante al *RAID 5*, aunque usa el doble de bits de paridad, garantizando la integridad de los datos en caso de fallen hasta 2 de los discos duros a la vez.

RAID 10

Este sistema combina características del *RAID 0* y *RAID 1*, de ahí el nombre. El *RAID 10* puede implementarse en sistemas con 4 discos duros o más, siendo obligatorio un número par.

La mitad de los discos almacena datos y la otra mitad almacena una copia. La mitad que almacena datos es combinada, formando un sistema *RAID 0*, aumentando el rendimiento, aunque manteniendo la fiabilidad, ya que disponemos de copias de todos los datos.

RAID 53 (5+3)

Al contrario de lo que sugiere el nombre, este modo es una combinación de los modos 3 y 0. El *RAID 53* puede implementarse en sistemas con al menos 5 discos duros. Los dos primeros discos forman un sistema *RAID 3*, con los datos distribuidos entre ellos. Los dos discos duros siguientes forman un sistema *RAID 0*, mientras el último almacena los códigos de paridad de todos.

Este sistema presenta un equilibrio entre las buenas tasas de transmisión del *RAID 3* y el buen rendimiento al leer varios archivos pequeños del *RAID 0*. Sin embargo, no existe un 100% de garantías de recuperar todos los datos en el caso de que uno de los discos duros falle. Es un modo poco usado.

6.9 Evolución

Los seres humanos estamos acostumbrados a pensar en valores que siguen un formato decimal, teniendo más facilidad para trabajar con números múltiplos de 10. Sin embargo, un ordenador trabaja con un sistema binario y, por ello, un KByte no se corresponde con 1.000 Bytes y sí con 1.024 Bytes, ya que 1.024 es la potencia de 2 más cercana a 1.000.

Un MByte se corresponde con 1.048.576 Bytes y un GByte se corresponde con 1.073.741.824 Bytes. El problema es que los fabricantes, a fin de aumentar el tamaño de sus discos, acostumbran a usar el sistema decimal para medir la capacidad de los discos duros.

Así, un disco duro vendido como disco de 4.3 GBytes tiene 4.3 mil millones de Bytes, que se corresponden a un poco más de 4 GBytes, en lugar de los 4.617.089.843 Bytes reales.

6.9.1 Las barreras de los 504 MB y los 8 GB

Un gran problema que encontramos al intentar instalar un disco duro moderno en una placa base más antigua pueden ser las limitaciones en cuánto a la capacidad del disco. En la época del *486* el límite eran los discos duros mayores de 504 MB (528 MB en la notación decimal utilizada por los fabricantes) y, en las placas base producidas hasta poco tiempo atrás existía una nueva limitación, en este caso para los discos duros mayores de 7.88 GB (8.4 GB en notación decimal).

Estas limitaciones aparecieron por la falta de visión de las personas que desarrollaron el formato *IDE* y las instrucciones INT 13h del *BIOS*, las responsables del acceso al disco, desarrolladas para ser usadas en un PC *AT*, pero que acabaron siendo perpetuadas hasta nuestros días. En aquella época, los discos duros con más de 504 MB parecían algo muy lejano.

La división del disco en *clusters*, usando cualquier sistema de archivos, permite el soporte por parte del sistema operativo. En un nivel más bajo, sin embargo, es necesario acceder individualmente a los sectores del disco duro y, para eso, cada sector debe tener una única dirección. El formato *IDE* reserva 16 bits para el direccionamiento de los cilindros (65.536 combinaciones), 4 bits para el direccionamiento de los cabezales de lectura (16 combinaciones), 8 bits para los sectores (256 combinaciones), lo que permite el direccionamiento de 256 millones de sectores. Como cada sector siempre tiene 512 Bytes, disponemos de soporte para discos duros de hasta 128 GB.

Sin embargo, el *BIOS* posee otras limitaciones para el acceso a los discos (instrucciones *INT 13h*), reservando 10 bits para el direccionamiento de los cilindros (1.024 combinaciones), 8 bits para los cabezales de lectura (256 combinaciones) y 6 bits para los sectores (63 combinaciones, pues la dirección 0 está reservada), lo que permite el direccionamiento de 1.61 millones de sectores y, en consecuencia, discos de hasta 7.88 GB ytes.

Como se usan tanto la interfaz *IDE* como las instrucciones *INT 13h* se acabaron uniendo sus limitaciones. La interfaz *IDE* reserva 16 bits para el direccionamiento de los cilindros y el *BIOS* sólo utiliza 10 bits. A su vez, el *BIOS* reserva 8 bits para el direccionamiento de los cabezas de lectura, sin embargo la interfaz *IDE* sólo puede utilizar 4. Entonces, la capacidad de direccionamiento acaba nivelándose por abajo, combinando las limitaciones de ambas partes, permitiendo el direccionamiento de discos duros de un máximo de 504 MB, el límite que teníamos para la mayoría de los equipos *486* o inferiores. Este método de direccionamiento es conocido por *CHS* (cilindro, cabezal de lectura y sector).

Limitaciones	Cilindro	Cabezal de lectura	Sectores	Tamaño máximo
IDE/ATA	16 bits	4 bits	8 bits	128 GBytes
INT 13h (BIOS)	10 bits	8 bits	6 bits	7.88 GBytes
Combinación	10 bits	4 bits	6 bits	504 Mbytes

Esta configuración es sólo lógica, ya que el modo como el *BIOS* detecta al disco duro no tiene nada que ver con la geometría física del disco duro, sino que sólo está relacionada con la tarjeta lógica del disco duro. Esta tarjeta del disco duro se encarga de convertir las direcciones lógicos del *BIOS* en las direcciones físicas reales.

En la época del *286*, donde se usaban discos duros de como máximo 20 o 40 MBytes, este límite no molestaba a nadie; pero a partir del momento en que pasamos a disponer de discos de 800 MB, 1 GB, 1.2 GB, etc. se necesitaba hacer algo.

La primera solución fue el *Extended CHS* o modo *Large*. Este formato continuaba teniendo las mismas limitaciones de la interfaz *IDE* y de las instrucciones *INT 13h*, pero usaba un pequeño truco para variar sus limitaciones.

El *BIOS* posee más direcciones para los cabezales de lectura (256 contra 16) y, sin embargo, la interfaz *IDE* posee más direcciones para los cilindros (65.536 contra 1.024). Usando el modo *Large* pasamos a utilizar un adaptador, un pequeño programa integrado en el *BIOS*, encargado de convertir direcciones. La conversión se realiza usando un simple factor multiplicativo: la interfaz *IDE* permite más direcciones para el cilindro, pero a la vez permite menos direcciones para el cabezal de lectura, entonces podemos aplicar la traducción de direcciones dividiendo el número de direcciones del cilindro y multiplicando las direcciones para los cabezales de lectura por el mismo número. Pueden usarse los números 2, 4, 8 y 16.

Si por ejemplo, instalásemos un disco duro con una geometría lógica de 3.068 cilindros, 16 cabezales y 63 sectores, usando el factor 4 pasaríamos a tener 3.086/4 = 767 cilindros, 16 x 4 = 64 cabezales y 63 sectores, y así tenemos direcciones dentro de los límites del *BIOS* y, por ello, podemos utilizar sin problemas los 1.5 GB del disco. El modo *Large* nunca fue muy utilizado, pues justo después surgió una solución mucho mejor para el mismo problema, conocida como modo *LBA*, contracción de *Logical Block Addressing* o direccionamiento lógico de bloques.

La idea es la siguiente: el formato *IDE* reserva 16 bits para el direccionamiento de los cilindros, 4 bits para el direccionamiento de los cabezales de lectura y 8 bits más para los sectores, totalizando 28 bits de direccionamiento. El modo *LBA* deja de lado el direccionamiento *CHS*, con direcciones independientes para los cilindros, cabezales y sectores, pasando a adoptar una dirección única. Los sectores pasan a recibir direcciones de forma secuencial 0, 1, 2, 3, 4 etc. como los *clusters* en el sistema *FAT*. Entonces, los 28 bits de direccionamiento permiten 228 millones de direcciones, lo que corresponde a discos duros de hasta 137 GB.

Claro que para usar el modo *LBA* es necesario que el disco duro soporte este modo de direccionamiento. Felizmente, todos los discos duros actuales soportan el modo *LBA*. Veamos que para el direccionamiento de los 137 GB permitidos por el modo *LBA* es necesario abandonar el uso de las instrucciones I*NT 13h*, lo que significa desarrollar nuevas instrucciones de acceso a la disco para el *BIOS*.

6.9.2 Dynamic Drive Overlays (DDO)

Si no existe una actualización del BIOS disponible para la placa base para soportar el modo *LBA*, existe otra solución (a pesar de no ser muy recomendable) que es instalar un *driver* traductor, llamado *Dynamic Drive Overlay*.

Este pequeño programa sustituye a las instrucciones del *BIOS*, permitiendo superar sus limitaciones. Existen programas destinados a las placas base con el límite de 504 MB como para las placas base con el límite de 7.88 GB.

Como el disco duro sólo se puede acceder con el apoyo del programa, es necesaria su instalación antes del sistema operativo. Una vez instalado, pasa a ser referenciado en el sector de arranque del disco duro y se carga siempre que se conecta el equipo.

Usar un *Dynamic Drive Overlay* conlleva algunas desventajas: no son compatibles con muchos programas de diagnóstico, no permiten el uso de los *drivers* de *Bus Mastering*, lo que perjudica el rendimiento del disco duro y del sistema. También tenemos problemas con la mayoría de los administradores de arranque, dificultando la instalación de más de un sistema operativo en el mismo disco y, por último, tenemos el problema de la desinstalación del *driver*, ya que parra ello es necesario limpiar el sector de arranque a través del comando *FDISK /MBR* y luego volver a realizar las particiones del disco.

6.9.3 Detección y corrección de errores

Al ser un soporte físico magnético, el disco duro nunca es 100% fiable por mejor que sea la calidad de sus componentes. Los discos duros actuales utilizan tecnologías que se encuentran casi en el límite de lo que podemos imaginar, guardando cada vez más información en espacios cada vez menores, con tiempos de acceso cada vez más irrisorios y con unas tasas de transferencia cada vez más altas. No es difícil pensar que en situaciones tan límites de funcionamiento un disco duro pueda presentar errores.

Obviamente, como todos los datos importantes están guardados en el disco duro, la posibilidad de errores en la lectura es inaceptable. Esto es todavía mucho más inaceptable en equipos destinados a operaciones críticas.

Para convertir los discos duros en más fiables, los fabricantes usan técnicas de corrección de errores cada vez más avanzadas. La técnica más utilizada actualmente es el *ECC* (*Error Correction Code*), la técnica también utilizada en las unidades de *CD-ROM/DVD-ROM* y en las memorias.

Usando el modo *ECC*, cuando se graba un dato en un sector se generan códigos que se graban junto a él. Además de los 512 Bytes de datos, cada sector almacena algunos Bytes más que contienen los códigos *ECC*. La creación de los Bytes de *ECC*, así como su utilización posterior, se realiza a través de la controladora del disco duro, por lo que el proceso de verificación y corrección es completamente transparente al sistema operativo y al usuario.

La cantidad de códigos que se van a usar es una decisión del fabricante, pero cuantos más códigos se usen, mayor será la posibilidad de recuperar los dados del sector en caso de producirse algún error. Actualmente, es común el uso de hasta 100 Bytes de códigos para cada sector, de lo que resultan discos prácticamente libres de errores.

Cuando un sector es leído por el cabezal de lectura, juntamente con los datos son leídos algunos de los códigos *ECC,* que verifican si los datos que se están leyendo son los mismos que se habían grabado, una técnica que recuerda un poco a la paridad usada en la memoria *RAM*.

En el caso de encontrar un error, se usan los demás códigos *ECC* para intentar corregir el problema. En la mayoría de los casos, este primer intento es suficiente para corregir el error. En caso de fracaso, se realiza un nuevo intento de lectura del sector, pues la posibilidad de que el error haya sido causado por alguna interferencia o inestabilidad momentánea es bastante grande.

Si después de estos intentos permanece el error, se usan los códigos *ECC* más avanzados, que garantizan una mayor posibilidad de éxito pero que, por otro lado, exigen mucho más procesamiento y consumen más tiempo. Si aun así no fuese posible corregir el error, nos encontramos ante un error de lectura. Entonces aparece un mensaje de error en la pantalla y el sector es marcado como defectuoso.

6.10 Rendimiento

El rendimiento del disco duro es un factor que tiene mucha influencia en el rendimiento global del sistema, determinando el tiempo de carga de las aplicaciones y archivos grandes e, incluso, la velocidad de acceso al archivo de intercambio. Para llevar a cabo la ejecución de un programa o acceder a cualquier otro dato, primero debe ser transferido del disco duro a la memoria. No se gana mucho teniendo un procesador rápido o memorias muy rápidas si, en todo momento, tienen que esperar a que los datos sean transferidos por el disco duro. Después de la cantidad de memoria *RAM* y caché, quizás el disco duro es el componente que más afecta al rendimiento del equipo, incluso rivalizando con el procesador.

Las diferentes marcas y modelos de discos duros que existen a la venta en el mercado presentan rendimientos muy distintos. No es fácil medir el rendimiento de un disco duro, pues el rendimiento del disco viene determinado por un conjunto de varios ítems y no es fácil intentar resumirlos en un único número.

Ningún programa de *benchmark* de los que existen actualmente es muy fiable para medir el rendimiento de un disco duro, pues los resultados siempre acaban siendo contaminados por el rendimiento del resto de los dispositivos del equipo. Para empeorar todavía más este cuadro, los fabricantes tienen el hábito de vender los discos duros informando de la tasa máxima de transferencia de datos de la controladora, y no del disco duro.

Esto es completamente irreal, pues una controladora *UDMA 33*, por ejemplo, transmite los datos a una velocidad de 33 MB/s, mientras que la velocidad real del disco duro generalmente está alrededor de los 10 o 15 MB/s.

Por ejemplo, otras veces sólo se informa del tiempo de acceso y muchos acaban concluyendo que el disco duro X es mejor que el disco duro Y, sencillamente por tener un tiempo de acceso de 9 milisegundos, mientras que el otro tiene un tiempo de acceso de 9.5 milisegundos.

Para esclarecer un poco mejor este asunto, vamos a analizar ahora los diferentes factores que determinan el rendimiento de un disco duro. Investigando en las *Webs* de los fabricantes de los discos duros encontraremos casi siempre estas especificaciones para decidir qué disco duro es mejor para nuestro equipo.

6.10.1 Tiempo de búsqueda (Seek Time)

Este es un factor importante en el rendimiento general del disco, pues es el tiempo que tarda el cabezal de lectura para ir de una pista del disco a otra. Este tiempo es un poco difícil de determinar con exactitud, pues el tiempo que tarda el cabezal para ir de la pista 50 a la pista 100, desplazándose 50 pistas, por ejemplo, no es 5 veces mayor que el tiempo para ir de la pista 50 a la pista 60, desplazándose 10 pistas.

Así, para ir de la pista 50 a la 60, los cabezales de lectura y escritura podrían tardar 4 milisegundos, tardando sólo 10 para ir de la pista 50 a la 100, por ejemplo. Estos valores varían en cada marca y modelo de disco duro, pero cuánto menores son los tiempos, mejor será el rendimiento.

6.10.2 Tiempo de latencia (Latency Time)

Dentro del disco duro, los *platters* magnéticos giran continuamente. Por ello, difícilmente los sectores que deben ser leídos estarán debajo del cabezal de lectura/escritura en el momento exacto de ejecutar la operación, pudiendo, en el peor de los casos, ser necesaria una vuelta completa del disco hasta que el sector deseado pase nuevamente por debajo del cabezal de lectura.

El tiempo de latencia es tan importante como el tiempo de búsqueda. Felizmente, es fácil de ser calculado con sólo dividir 60 por la velocidad de rotación del disco duro, medida en *RPM* (rotaciones por minuto), y multiplicar por 1000. Entonces tendremos el tiempo de latencia en milisegundos. Un disco duro de 5200 RPM, por ejemplo, tendrá un tiempo de latencia de 11.5 milisegundos (el tiempo de una rotación), ya que 60/5200 x 1000 = 11.5.

Generalmente, se usa el tiempo medio de latencia, que corresponde a la mitad de una rotación del disco, asumiendo que los *clusters* deseados estarán, como promedio, a medio camino del cabezal. Un disco duro de 5200 RMP tendría entonces un tiempo de latencia medio de 5.75 milisegundos.

6.10.3 Tiempo de acceso (Access Time)

El tiempo de acceso es el tiempo medio que tarda el disco para acceder a un sector localizado en un sitio aleatorio del disco. Este tiempo es una mezcla del tiempo de búsqueda y del tiempo de latencia del disco duro, significando el tiempo que tarda el brazo de lectura para llegar a una determinada pista, sumado con el tiempo que tarda el disco para girar y llegar al sector correcto. El tiempo de acceso en los discos duros modernos gira alrededor de los 10 a los 7 milisegundos (cuanto más bajo mejor).

6.10.4 Head Switch Time

Un disco duro está compuesto internamente de varios discos. Naturalmente, tenemos un cabezal de lectura para cada disco, grabándose los datos de forma distribuida entre las distintas caras de estos *platters*.

Como los cabezales de lectura son fijos, en el caso de que sea necesaria la lectura de dos archivos, uno en la cara 1 y otro en la cara 3, por ejemplo, la controladora activará el cabezal de lectura de la cara 1, leerá el primer archivo y luego activará el cabezal responsable de la cara 3, leyendo el segundo archivo.

El término *Head Switch Time* puede traducirse como "tiempo de alternancia entre los cabezales de lectura", al ser justamente el tiempo que tarda el disco para cambiar la lectura de un cabezal a otro. Este tiempo es relativamente pequeño y no tiene una influencia tan importante como el tiempo de acceso y la densidad.

6.10.5 Tasa de transferencia (Internal Transfer Rate)

De todos los factores que hemos discutido hasta ahora, probablemente este es el más complicado, pues es el que envuelve más variables. Una definición simplista sobre la tasa de transferencia interna sería la cantidad de datos que pueden ser leídos por segundo por el cabezal de lectura.

Sin embargo, existen varios factores que pueden distorsionar esta medición. Si los archivos estuviesen fragmentados, o fuese necesario leer varios archivos grabados en sectores distantes unos de los otros, la tasa de transferencia será mucho menor que al leer un archivo grande grabado de forma secuencial.

Si estamos oyendo una canción grabada en el disco duro, los datos accedidos (suponiendo que el disco duro esté desfragmentado) probablemente estarán en el disco de forma secuencial, aumentando la tasa de transferencia. Sin embargo, si accedemos a datos distribuidos de forma aleatoria por la superficie del disco, entonces la tasa será mucho menor.

Cuando existan referencias a la tasa de transferencia interna en las especificaciones de un disco duro, podemos estar seguros de que se trata de la velocidad de lectura cuando se leen sectores consecutivos.

6.10.6 Caché (Buffer)

Los discos duros actuales poseen una pequeña cantidad de memoria caché integrada en la controladora, que ejecuta varias funciones con el objetivo de mejorar el rendimiento del disco duro.

Generalmente, en la lectura de un archivo se leerán varios sectores consecutivos. La forma más rápida de hacer esto es hacer que el cabezal de lectura lea de una sola vez todos los sectores de la pista, pase a la próxima pista, luego a la próxima y así en adelante. Eso permite obtener el mejor rendimiento posible.

El problema es que en la práctica esto no es así. El sistema pide el primer sector del archivo y sólo solicita el próximo sector después de recibirlo y certificar que no existen errores. Si no hubiese ningún tipo de búfer, el cabezal de lectura del disco duro acabaría teniendo que pasar varias veces sobre la misma pista, leyendo un sector en cada paso, ya que no daría tiempo de leer los sectores de forma secuencial después de todo el tiempo perdido antes de cada petición nueva.

Gracias a la caché se resuelve este problema, pues en cada paso el cabezal de lectura lee todos los sectores próximos, independientemente de que hayan sido solicitados o no. Tras realizar la verificación de rutina, el sistema solicita el próximo sector, que al ya estar cargado en la caché se suministrará en tiempo récord. En la caché quedan almacenados también los últimos datos accedidos por el procesador, permitiendo que un dato solicitado repetidamente pueda ser reaccedido a partir de la caché, dispensando una nueva y lenta lectura de los datos por el cabezal de lectura. Este sistema es capaz de mejorar muchísimo la velocidad de acceso a los datos cuando éstos son repetitivos, cosa que tiene lugar con mucha frecuencia en los servidores de red o cuando se usa la memoria virtual.

Los datos leídos por los cabezales de lectura originalmente son grabados en la caché, y a partir de ella, se transmiten a través de la interfaz *IDE* o *SCSI*. En el caso de que la interfaz esté momentáneamente congestionada, los datos son acumulados en la caché y, luego, transmitidos de golpe cuando la interfaz queda libre, evitando cualquier pérdida de tiempo durante la lectura de los datos.

A pesar de su reducido tamaño, generalmente de 512 o 1024 KBytes, la caché consigue acelerar bastante las operaciones de lectura de los datos. Claro que cuanto mayor y más rápida fuese la caché, mayor será el aumento del rendimiento.

Además de la caché localizada en la controladora del disco duro, la mayoría de los sistemas operativos, incluyendo *Windows 95/98/2000/XP/NT*, reservan una pequeña zona de la memoria *RAM* para crear un segundo nivel de caché de disco. Como en el caso de un procesador, cuanta más caché, mejor es el rendimiento. El tamaño de la caché de disco utilizada por *Windows XP* puede ser configurada a través del icono *Sistema* del *Panel de control*. Basta acceder al sector *Rendimiento* de la ficha *Opciones avanzadas*.

El uso de la caché de disco es el factor que más distorsiona los resultados de los programas *benchmark*, pues el programa no puede saber si un dato viene de un acceso a la caché o de una lectura física. Las pruebas de disco más lentas utilizan archivos grandes para disminuir este problema, pues como la caché sólo almacena 1 o 2 MB de datos, la distorsión al trabajar con un archivo más grande sería muy pequeña.

6.10.7 Densidad

La densidad de los *platters* de un disco duro es otro factor con un impacto enorme en el rendimiento. Cuanto mayor sea la densidad, menor será el espacio a ser recorrido por el cabezal de lectura para localizar un determinado sector, pues los datos estarán más próximos unos de los otros. La densidad puede ser calculada muy fácilmente, bastando dividir la capacidad total del disco por la cantidad de cabezales de lectura (y consecuentemente el número de caras de disco).

Por ejemplo, un disco duro de 4 GBytes y 4 cabezales de lectura posee una densidad de 1 GByte por cada cara de *platter*, mientras que otro disco, también de 4 GBytes con 6 cabezales de lectura posee una densidad menor, de sólo 666 MBytes por cada cara de *platter*.

La densidad tiene una influencia directa en los tiempos de acceso y de latencia del disco duro, además, con un número menor de cabezales de lectura, el tiempo perdido con el *Head Switch* también es menor. Muchas veces encontramos en el mercado discos duros de la misma capacidad, aunque con densidades diferentes. En este caso, casi siempre el disco duro con una mayor densidad utilizará tecnologías más recientes, siendo más rápido por ello.

6.10.8 Velocidad de la interfaz

La interfaz determina la velocidad máxima de transferencia, pero no necesariamente el rendimiento del disco duro. Por regla general, la interfaz siempre es mucho más rápida que la tasa de transferencia interna alcanzada por el disco duro.

Sin embargo, en muchas situaciones, la interfaz *IDE* queda momentáneamente congestionada, dejando de transmitir datos. En estas situaciones, los datos se acumulan en el búfer del disco duro y, luego, son transmitidos de golpe cuando la interfaz queda libre.

Esto puede ocurrir en dos situaciones: cuando tenemos dos discos instalados en el mismo puerto *IDE* y los dos discos son accedidos de forma simultánea, o cuando el bus *PCI* queda congestionado (ya que los puertos *IDE* comparten los 133 MB/s con todos los demás periféricos *PCI* instalados).

En estas situaciones, tener una interfaz más rápida permitirá que los datos almacenados en la caché sean transferidos de forma más rápida. Sin embargo, en situaciones normales, el rendimiento queda limitado a la tasa de transferencia interna del disco duro.

Por ejemplo, el simple hecho de pasar a usar *UDMA 66* en el lugar de *UDMA 33*, no cambia en casi nada el rendimiento del disco duro en las aplicaciones reales, al menos mientras no tengamos discos duros capaces de mantener tasas de transferencia interna próximas a los 30 MB/s, lo que probablemente sólo debió acontecer alrededor del año 2002. El *UDMA 66/100* apareció con el objetivo de ampliar el límite de transferencia de las interfaces *IDE*, abriendo camino para el futuro lanzamiento de discos duros mucho más rápidos, que pudiesen trabajar sin limitaciones por parte de la interfaz.

6.11 Pio 4 vs UDMA 33/66 vs UDMA 100/133

Así como una tarjeta gráfica se conecta en un *slot PCI* o *AGP*, para poder comunicarse con el resto del sistema, el disco duro necesita estar conectado a alguna interfaz. Una interfaz de disco no es más que un medio de comunicación, una pista por donde pueden transitar los datos que entran y salen del disco duro. De nada sirve un disco duro muy rápido si la interfaz no permite que se comunique con el resto del sistema usando toda su velocidad. Por otro lado, una interfaz mucho más rápida que el disco duro es sencillamente un desperdicio, pues quedará ociosa demasiado tiempo.

Como ya hemos visto, actualmente se usan dos formatos de interfaces de disco: el *IDE* (también llamado *ATA*) y el *SCSI*, con una gran predominancia del *IDE*. Las placas base más antiguas no poseían interfaces *IDE*. En ellas, la interfaz *IDE* debía ser adquirido por separado y encajarse en un *slot* disponible de la placa base. Si pudiésemos examinar el hardware de un *486* veríamos una tarjeta *ISA*, *EISA* o *VLB* que incluía la interfaz *IDE*, además de la interfaz para las unidades de disquetes, un puerto paralelo, dos puertos serie y un puerto para el joystick. Esta tarjeta se llamaba s*uper IDE*.

Todas las placas base actuales poseen, además de dos puertos serie y un puerto paralelo, dos interfaces *IDE* integradas, llamadas controladora primaria y controladora secundaria. Cada controladora soporta dos dispositivos *IDE*, lo que permite un máximo 4 dispositivos *IDE* en un mismo equipo. Para esto, uno de los dispositivos debe ser configurado como *master* (maestro) y el otro como *slave* (esclavo), configuración que se realiza a través de *jumpers*.

El cable *IDE* posee tres encajes, uno que se enchufa a la placa base y otro en cada dispositivo. Aunque sólo tengamos un dispositivo *IDE*, deberemos enchufarlo en el conector del final, nunca en el conector del medio. El motivo de esto es que enchufándolo en el conector del medio, el cable quedará sin terminación, haciendo que los datos vayan hasta el final del cable y retornen como paquetes sobrantes, interfiriendo en el envío de los paquetes buenos y causando una disminución en la velocidad de transmisión. En el caso de los dispositivos *IDE*, el dispositivo enchufado en la punta del cable funciona como terminador.

Existen varios modelos de interfaces *IDE* que ofrecen diferentes modos de operación. Estos modos de operación son llamados *PIO* y determinan la velocidad y los recursos de la interfaz. Las placas base un poco más antiguas, como las placas para procesado-

res *Pentium* que utilizan los *chipsets* FX y VX, sólo soportan el modo *Pio 4*, siendo capaces de transferir datos a 16.6 MB/s. Las placas un poco más recientes, también soportan el modo *Ultra DMA 33* o el *Ultra DMA 66*.

Probablemente debemos haber oído hablar del modo *Ultra DMA*, también llamado *Ultra ATA*. Este modo de operación aporta varias ventajas sobre el antiguo modo *Pio Mode 4*, como una mayor tasa de transferencia de datos, que pasa a ser de unos 33 MB/s como mínimo. Sin embargo, la principal ventaja del *UDMA* es permitir que el disco duro pueda acceder directamente a la memoria *RAM*.

Usando el modo *UDMA*, al contrario de que el procesador tenga que transferir los datos del disco duro a la memoria *RAM*, y viceversa, sólo debe realizar una solicitud al disco duro para que él mismo haga el trabajo. Claro que este modo de operación aumenta bastante el rendimiento del sistema, pues ahorra al procesador el trabajo con las transferencias de datos, quedando libre para ejecutar otras tareas. El *Pio Mode 4* permitía el uso del *Multiword DMA 2*, que también permitía el acceso directo a la memoria aunque de una forma un poco menos eficiente.

Para poder usar las ventajas del *UDMA* es necesario que el disco duro también ofrezca soporte para esta tecnología. Todos los modelos de discos más recientes incluyen el soporte para *UDMA*, aunque sin embargo, manteniendo la compatibilidad con algunas controladoras más antiguas. En caso que tengamos en la placa una controladora que soporte sólo el modo *Pio 4*, el disco duro funcionaría normalmente, aunque limitado a las características de la interfaz usada. *Windows 98* poseía soporte nativo para discos duros *UDMA*; en el caso de *Windows 95* era necesario instalar los *drivers UDMA*, generalmente situados en la carpeta *IDE* del *CD* de *drivers* que acompañaba a la placa base.

Existen como máximo 8 modos de operación de las interfaces *IDE* en el momento de escribir este capítulo, que van desde el prehistórico *Pio Mode 0*, extremadamente lento, al *UDMA 133*, que mantiene los recursos del *Ultra DMA* soportando mayores velocidades en la transferencia de los datos. Cabe recordar que estas velocidades son el flujo máximo de datos permitido por la interfaz, no correspondiéndose necesariamente con la velocidad de operación del disco duro. Los modos de operación de las interfaces *IDE* son las que mostramos en la siguiente tabla.

Modo de operación	Máxima tasa de transferencia
PIO MODE 0	3.3 MB/s
PIO MODE 1	5.2 MB/s
PIO MODE 2	8.3 MB/s
PIO MODE 3	11.1 MB/s
PIO MODE 4	16.6 MB/s
UDMA 33	33.3 MB/s

Modo de operación	Máxima tasa de transferencia
UDMA 66	66.6 MB/s
UDMA 100	100 MB/s
UDMA 133	133 MB/s

La mayoría de los discos actuales son compatibles con el modo *UDMA 100/133*. Esta nueva interfaz permite tasas de transferencia casi similares a las de las controladoras *SCSI*. Claro que los 100 o 133 MB/s permitidos no son necesariamente la velocidad alcanzada por el disco duro.

El encaje de las interfaces *UDMA 100* posee los mismos 40 *pins* de los otros modos, así como una compatibilidad retroactiva con cualquier disco duro *IDE*. Sin embargo, los cables poseen 80 vías, 40 usadas para transportar datos y 40 como tierra. Los hilos son intercalados como en un hilo de datos. Esta disposición atenúa las interferencias y permite alcanzar los 100 MB/s. Otro requisito impuesto por el modo es que los cables no tengan más de 45 cm de largo, pues el cable actúa como antena, captando interferencias externas.

Para activar el modo *UDMA 100* necesitaremos que tanto el disco duro como la placa base sean compatibles, siendo obligatorio el uso del cable especial de 80 vías que es suministrado junto con la placa base. Finalmente, hay que instalar los *drivers* que acompañan a la placa base para activar el soporte al modo *UDMA 100* en *Windows 98/2000*, pues sólo posee soporte nativo para *UDMA 33/66*.

Además del *UDMA 100*, que acabamos de ver, existe el modo *UDMA 133*, que también es soportado por la mayoría de las placas base actuales. El modo *UDMA 133* es el resultado de un pequeño perfeccionamiento del *UDMA 100*, y trae la ventaja de permitir transferencias de datos a hasta 133 MB/s. Los requisitos para activar el modo *UDMA 133* son los mismos del modo *UMDA 100*.

6.12 La unidad de disquetes

A pesar de tener una puesta en funcionamiento parecida a la de un disco duro, una unidad de disquetes es mucho más simple, ya que es parecida a los discos duros de inicios de la década de los años 80.

El soporte físico magnético de un disquete está compuesto de óxido de hierro. Se aplica una fina capa de este material sobre un disco de plástico *mylar*, el utilizado en las cintas *K-7*, aunque un poco más tupido en el caso de los disquetes.

Así como en los discos rígidos, los disquetes son divididos en pistas y sectores. La diferencia es que, mientras que un disco duro posee más de 2.000 pistas, un disquete de 1.44 MB sólo posee 80 pistas. El número de sectores también es menor, sólo 18 sectores por pista en un disquete de 1.44, lejos de los 200 o 300 sectores que encontramos

en cada pista de un disco duro. Como en los disquetes no se utiliza el recurso de *Zoned Bit Recording*, todas las pistas poseen el mismo número de sectores. La densidad en un disquete de 1.44 MB es de 135 pistas por pulgada.

Una unidad de disquetes

La velocidad de rotación en las unidades de disquete también es muchas veces menor que la de los discos duros. Mientras que un disco duro de gama alta llega a sobrepasar las 10.000 rotaciones por minuto, una unidad de 1.44 MB trabaja a sólo unas 300 rotaciones por minuto, es decir, sólo 5 rotaciones por segundo. Uno de los motivos para utilizar una velocidad de rotación tan baja es la fragilidad del soporte físico magnético de los disquetes, que sería dañado fatalmente durante la lectura y grabación de datos a velocidades más altas.

Mientras en los discos duros utilizamos el *actuator* para controlar el movimiento de los cabezales de lectura, lo que permite un movimiento extremadamente rápido, en las unidades de disquetes se utiliza un anticuado motor de paso. Cuando recibe una carga eléctrica, este motor da un giro completo, y a través de un sistema de engranajes hace que el cabezal de lectura recorra la distancia correspondiente a una pista. A pesar de que motor de paso funcione bien, es muy lento, haciendo que el cabezal de lectura tarde cerca de 1/6 de segundo en moverse de un punto a otro del disquete. Sumando 160 milisegundos de tiempo de búsqueda, con más de 100 milisegundos de tiempo de acceso (usando el tiempo medio de acceso, que corresponde a la rotación media), el tiempo de acceso de una unidad de disquetes queda alrededor de los 260 milisegundos, más de 25 veces más lento que un disco duro de gama media.

Al contrario de los demás componentes del ordenador, la unidad de disquetes ha evolucionado poco en las últimas dos décadas, limitándose a ampliar la capacidad de los discos de los 360 KB a los 1.44 MB actuales, y reducir su tamaño de las 5.25 pulgadas a las 3.5 pulgadas actuales. A pesar de los pesares, todavía utilizamos disquetes como medio de almacenamiento y transporte de pequeños archivos o documentos debido a su bajo coste, aunque los unidades de memoria flash (lápices) los están empezando a sustituir.

7. CD-ROM, CD-R y DVD

Originalmente, el *Compact Disk*, o *CD,* fue desarrollado para almacenar música y sustituir a los ya anticuados discos de vinilo. En un *CD*, el sonido es grabado en formato digital con un muestreo de 44.100 Hz y 16 bits de resolución, con lo que obtenemos un sonido completamente libre de ruidos y con una calidad casi perfecta.

No pasó mucho tiempo desde su creación hasta que los fabricantes percibieron que, con un mínimo esfuerzo, un *CD* también podía ser usado para almacenar una gran cantidad de datos. Entonces se creó una distinción: los *CDs* destinados a grabar música pasaron a ser llamados de *CD-DA*, o "*Compact Disk Digital Audio*", mientras que los *CDs* destinados a la grabación de datos pasaron a ser llamados *CD-ROM,* o "*Compact Disk Read Only Remory*".

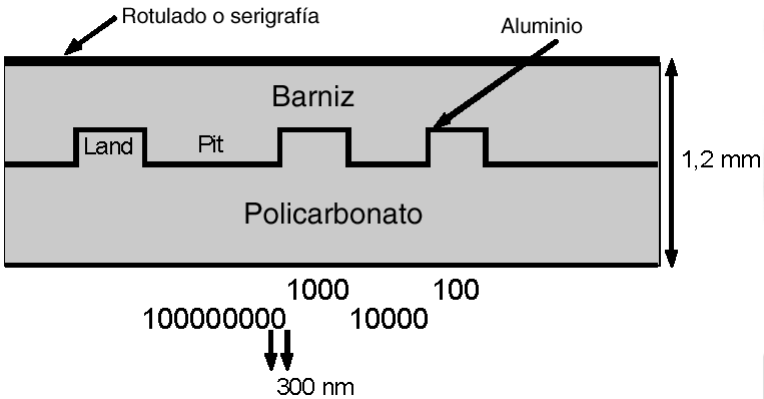
En aquella época, los discos duros costaban más de lo que valen ahora y disponían de una capacidad de almacenamiento mucho menor. También existía un gran problema relacionado con la distribución del software, ya que con sólo 1.44 MB de capacidad, eran necesarios incontables disquetes para almacenar los programas grandes o juegos.

Una unidad típica de CD-ROM

La gran capacidad y el bajo coste de producción de los *CD-ROM* hicieron posible la aparición de las enciclopedias digitales con sonido y vídeo, títulos multimedia y los juegos más avanzados. Imaginemos cuántos disquetes se hubiesen necesitado para almacenar una enciclopedia como la *Encarta* de la casa Microsoft.

7.1 El medio físico

El *CD-ROM* es un disco de aproximadamente 12 cm de diámetro, o 4.75 pulgadas, con poco más de 1.2 milímetros de grosor y está compuesto por tres capas principales. La primera capa y la más densa, es la base plástica del *CD-ROM*, hecha de un plástico especial, llamado policarbonato.



La estructura de las capas de un CD-ROM

La capa central, también llamada capa reflexiva, es una fina capa de aluminio, platino u oro. Esta capa es extremadamente fina, midiendo sólo algunos nanometros. Su función es reflejar el láser del lector, permitiendo la lectura de los datos grabados en la capa de policarbonato. Para la protección de la frágil capa reflexiva, sobre ella se aplica una capa de barniz. En la mayoría de los casos también se aplica un estampado o una etiqueta encima de todo.

La capa reflexiva queda encima de la capa de plástico y justamente por eso se le aplica encima una capa protectora de barniz. Si realizamos un pequeño rasguño sobre la capa inferior del disco, por el lado del plástico, probablemente el *CD-ROM* continuará leyéndose con normalidad, ya que la capa reflexiva, donde se almacenan los datos no es alcanzada, y a pesar del riesgo de dificultar la lectura, la unidad todavía conseguirá leer el *CD* variando el ángulo del láser. Si por otro lado, arañásemos la cara superior del disco, probablemente tendremos el *CD* completamente inutilizado, ya que se dañará la capa de datos.

7.1.1 El proceso de grabación

La técnica de grabación de los *CD* de audio recuerda un poco a la de los antiguos discos de vinilo, consistiendo en el prensado de la capa de policarbonato sobre un molde, donde son realizan los surcos que contienen los datos.

Todo empieza con la creación de un molde de vidrio. Este molde del mismo tamaño que un *CD* normal recibe una capa especial sensible a la luz. Luego se usa un láser especial para hacer los surcos que almacenan los datos. Este molde es examinado con

un microscopio y descartado si contiene el más mínimo error. Muchas veces se descartan varios moldes antes de producir uno perfecto. Después el molde pasa por un proceso de metalización, que produce un negativo de metal. Este negativo es duplicado, formando los moldes metálicos usados para grabar los *CDs* de audio. Después de que la capa de policarbonato sea prensada contra el molde se le aplica la capa reflexiva encima (nuevamente a través de un proceso de prensado) y, finalmente, la capa protectora de barniz. Es justamente por eso que en los *CDs* comerciales es imposible alterar su contenido, ya que es imposible eliminar los surcos hechos en la capa reflexiva.

Además de los *CDs* prensados, también existen los *CDs* grabables (*CD-R*), que son vendidos como discos vírgenes y pueden ser grabados una única vez en una unidad grabadora de *CDs*, y los *CDs* regrabables (*CD-RW*) que, como un disquete, pueden ser grabados y regrabados varias veces en una unidad grabadora que soporte este tipo de medio físico.

Un *CD-ROM* de buena calidad puede durar más de un siglo, siempre que se conserve bien. Además de vigilar los arañazos, debemos evitar exponer el *CD* directamente al sol u otras fuentes de calor, pues el calor puede fundir la capa reflexiva, inutilizando el *CD-ROM*. En este aspecto, los *CDs* grabables son especialmente sensibles.

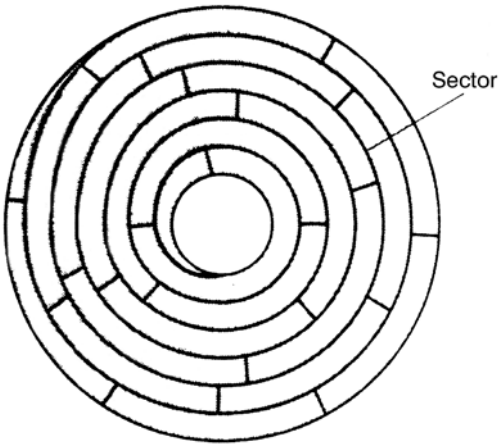
7.1.2 El formato

Originalmente, el *CD* fue concebido para almacenar sólo audio. Al almacenar datos, éstos son grabados en forma de una gran espiral (como en un disco de vinilo) en vez de usar pistas concéntricas como en un disco duro. Esta espiral engloba todo el disco, dando cerca de 20.000 "vueltas" y teniendo una longitud total de casi 5 kilómetros. Esta inmensa espiral se puede dividir en pistas lógicas, cada una abarcando una franja de un *CD* de audio, por ejemplo. Hablamos de pista lógica ya que la división sólo existirá desde el punto de vista del lector. Físicamente, la espiral continuará siendo ininterrumpida.

Cada bit grabado se corresponde con un conjunto de surcos en el *CD*, y no a un único surco como podríamos suponer. De hecho, se usan 17 surcos para formar un bit, ya sea 1 o 0. El uso de esta cantidad de surcos para cada bit es necesario para que, durante el proceso de lectura, el cabezal de lectura tenga el suficiente tiempo para "percibir" el cambio entre los bits 1 y 0. Un CD posee cerca de 106 billones de surcos, o "bits ópticos". Si cada surco se correspondiera con un bit de datos, dispondríamos de una capacidad increíble de almacenamiento en cada CD pero no sería posible, por lo menos con la tecnología actual, desarrollar procesos de grabación ni, principalmente, lectores de *CDs* con los cabezales de lectura lo suficientemente sensibles para conseguir leer los datos de forma precisa.

Al principio, para llegar a un consenso en cuanto a la capacidad ideal de un *CD* se llegó al pacto de que debería ser capaz de almacenar 74 minutos de música, con un muestreo de 44.100 Hz y con 16 bits de resolución, lo que corresponde a cerca de 742

MBytes. A fin de mejorar la organización interna, los datos se agrupan en forma de sectores, como en un disco duro. La diferencia es que mientras en un disco duro cada sector contiene 512 Bytes, en un *CD* cada sector (también llamado "*large frame*") contiene 2.352 Bytes.



La espiral interna de la grabación de un CD-ROM

Como todos los medios físicos, un CD no está libre de errores de lectura. En un disco duro por ejemplo, además de los 512 Bytes de datos, cada sector contiene algunas decenas de Bytes más con los códigos de corrección de errores, que son usados para detectar y corregir cualquier alteración en los datos grabados. Sin embargo, en el caso de un *CD* de música, los errores de lectura no causan inconvenientes. Si algunos bits no puedan ser leídos, el reproductor del *CD* usará una interpolación para evitar las posibles roturas en el sonido.

En un *CD* de música, además de los 2.352 Bytes de datos, tenemos una pequeña área de 98 bits más en cada sector que almacena los datos del *subcanal Q*. Estos 98 bits son reservados para almacenar información adicional sobre la zona del *CD* que está siendo reproducida, el tiempo transcurrido en minutos y segundos, la dirección del sector y 16 bits que almacenan un código de corrección de errores *ECC* rudimentario. Además de ser esenciales para mantener una velocidad constante en la lectura, esta información es usada por los reproductores de *CDs* para exhibir el tiempo transcurrido desde el inicio de la pista.

En un *CD* de datos, el escenario es un poco distinto. Al contrario de lo que pasa en un *CD* de música, el error en la lectura de sólo algunos bits sería suficiente para inutilizar el programa que debe ser leído. Para hacer que un CD sea realmente fiable para el almacenamiento de datos se reservaron 288 Bytes en cada sector para el almacenamiento de los códigos *ECC*, permitiendo la corrección automática de cualquier alteración en los datos, reservando 16 Bytes más para las señales de sincronismo y direccionamiento. Entonces, en cada sector sólo quedan 2.048 Bytes para el almacenamiento de

los datos. Como cada *CD* posee 330.000 sectores, llegamos a los 650 MBytes de capacidad total. En un CD de datos, no se usan los 98 bits reservados al *subcanal Q*.

Como pasa en un disco duro, cada sector de un *CD* de datos posee una única dirección, almacenada en el campo de 4 Bytes reservado al direccionamiento. En el inicio del *CD,* se graba un índice, llamado *TOC* (*Table Of Contents* o tabla de contenidos) que, de forma semejantemente a la *FAT* de un disco duro, informa de los archivos que están grabados en el *CD* y en que sectores se encuentra cada uno.

7.1.3 Los distintos formatos

Además de los formatos para *CDs* de música y CDs de datos, también se crearon, con el pasar de los años, otros formatos de *CD* destinados a otras aplicaciones. A pesar de que el soporte físico usado en todos los casos es el mismo, la manera de organizar los datos es distinta. Cada uno de estos formatos acabó recibiendo un color como nombre; el formato de *CD* de audio original por ejemplo, es llamado *Red Book* o libro rojo, mientras que el formato de los *CDs* de datos es llamado *Yellow Book* o libro amarillo. Parece que nadie sabe exactamente de donde surgió la idea de dar el nombre de un color a cada formato, pero mucha gente dice que los apodos surgieron a causa del color de la tapa usada en los manuales que describían las especificaciones de cada uno de los formatos.

Red Book: fue el formato original para los *CDs* de audio, desarrollado en asociación entre las casas Philips y Sony, publicado en 1980. Conocido como *Compact Disk Digital Audio* o *CD-DA*. Además del formato de grabación de audio, el *Red Book* conllevó las especificaciones físicas del *CD-ROM* (número de pistas y sectores, capacidad máxima, etc.) que han llegado hasta hoy.

Yellow Book: es el formato para *CDs* de datos que hemos visto antes. De la misma manera que el *Red Book*, fue desarrollado a través de la asociación entre las casas Philips y Sony y fue publicado en 1983. El *Yellow Book* original preveía dos modos de grabación, el modo 1 y el modo 2. La diferencia es que en el modo 1 se reservan 288 Bytes en cada sector para los códigos de corrección de errores y en el modo 2 todos los 2.352 bytes de cada sector se usan para almacenar datos (como en los *CDs* de audio). La idea es usar el modo 2 para grabar *CDs* que almacenasen imágenes y vídeo, donde la corrupción de algunos bits no causaría mayores problemas.

En 1985, una asociación entre varios fabricantes creó la *ISO 9660*, una estandarización para los *CDs* de datos, que establecía el uso del modo 1 como formato entre otras especificaciones. Como la *ISO 9660* era compatible con varios sistemas operativos, se convirtió rápidamente en el formato de la industria.

A pesar de su universalidad, la *ISO 9660* tenía la grave limitación de sólo permitir nombres de archivos de 8 caracteres como máximo (como en el *DOS*). Para eliminar esta limitación, otros fabricantes crearon extensiones para la *ISO 9660* original, que permitían los nombres de archivos largos. Sin embargo, al contrario de la *ISO* que es

un formato universal, cada uno de estos formatos sólo puede ser leído en un sistema operativo en particular: la extensión que permite los nombres largos en *Windows* se llama *Joilet*, la destinada al *Unix* se llama *Rock Ridge*.

Green Book: este formato también es conocido por *CD Interactivo*, o *CD-I*, y fue desarrollado en 1986 por la asociación de las casas Philips y Sony, nuevamente. Este formato fue creado para ser usado en los *CDs* multimedia, que llevasen gráficos, texto, imágenes y sonidos y pudiesen visualizarse en un aparato especial conectado a la tele. Este formato no fue muy usado y hoy está fuera de uso.

Orange Book: es el conjunto de especificaciones para los *CD-R* y *CD-RW*, y lleva varios recursos útiles para la grabación de *CDs*, como por ejemplo, la posibilidad de grabar *CDs* multisesión, o sea, en vez de grabar todo el *CD* de golpe y cerrar la *TOC*, es posible grabar algunos archivos cada vez, hasta completar el espacio del *CD-R*. La *TOC* se deja abierta, y sólo es cerrada cuando se agota el espacio libre del *CD*.

White Book: son las especificaciones para los *Vídeo CDs*, que no son más que *CDs* normales que almacenan vídeo en formato *MPEG* en vez de música. Cada *Vídeo CD* puede almacenar aproximadamente 1 hora de película, pero con una calidad inferior a la del *DVD*.

7.2 Rendimiento

Las primeras unidades de *CD-ROM* eran capaces de leer los datos a 150 KB/s. La siguiente generación de unidades ya era capaz de leer los datos a cerca de 300 KB/s, el doble de la velocidad de los primeros. Estas "nuevas" unidades pasaron a ser llamadas *CD-ROM 2x*, ya que eran dos veces más rápidas que las originales. Enseguida comenzaron a aparecer unidades con tasas de lectura de 600 KB/s o 900 KB/s, siendo llamadas respectivamente de 4x y 6x. Una unidad de 32x debe ser capaz de leer los datos a 4800 KB/s.

7.2.1 Modo de lectura

Otro diferencial en cuanto a la velocidad es el modo de lectura. Las primeras unidades de *CD-ROM* operaban en un modo de velocidad lineal constante. En ellas, la velocidad de rotación era variable, siendo menor en las pistas internas del *CD* y mayor en las pistas externas, lo que mantenía constante la velocidad de lectura. Sin embargo, los modelos a partir de una velocidad de 16x pasaron a trabajar en el modo de velocidad angular constante, pues en velocidades de rotación tan elevadas, era muy difícil desarrollar motores capaces de disminuir y aumentar continuamente la velocidad con la precisión necesaria.

El uso de la velocidad angular constante tiene una desventaja: como el *CD* gira a una velocidad constante, el acceso a las pistas externas se realiza a una velocidad mayor que en las pistas internas, ya que éstas son más cortas. Hasta ahí todo bien, el proble-

ma es que los fabricantes venden las unidades de acuerdo con la velocidad de lectura de las pistas más externas, aunque en las pistas internas la velocidad es menor a la mitad. Y como los *CDs* son grabados desde las pistas internas, una unidad de C*D-ROM* de 16x, al funcionar usando la tecnología de velocidad angular constante, acaba siendo más lenta que una unidad 12x que usa la tecnología con una velocidad lineal constante.

Este mismo modo de lectura se usa en los discos duros desde sus primeras versiones, justamente por ser más fácil de implementar y permitir unidades más rápidas y durables, debido a la mayor velocidad de rotación permitida y al menor desgaste del motor.

7.2.2 Tiempo de acceso

La velocidad máxima de la unidad de *CD-ROM* (52x, 40x etc.) sólo se alcanza cuando se están leyendo datos grabados de forma secuencial, un vídeo o música por ejemplo. Si se leen datos distribuidos por el *CD*, el cabezal de lectura necesitará moverse en busca de los datos, haciendo que la tasa de transferencia disminuya proporcionalmente a la velocidad de desplazamiento. El tiempo medio que tarda el cabezal de lectura para desplazarse hasta el punto donde están almacenados los datos, sumado a un giro del *CD*, hasta que es alcanzado el punto exacto donde se iniciará la lectura, es llamado tiempo de acceso.

Mientras más rápido sea el desplazamiento del cabezal de lectura menor será el tiempo de acceso y, consecuentemente, la pérdida de rendimiento. Una buena unidad de *CD-ROM* debe tener un tiempo de acceso en torno a los 80 milisegundos.

7.2.3 Caché

Para acelerar el rendimiento de las unidades, los fabricantes suelen incluir pequeñas cantidades de memoria caché, que almacenan los últimos datos accedidos por el cabezal de lectura. Si el sistema requiere un dato almacenado en la caché, este dato se transferirá automáticamente, dispensando una nueva lectura.

Lo más común es el uso de 512 o 1024 KBytes de memoria caché. Así como en el caso del disco duro, *Windows* permite usar un poco de la memoria *RAM* del sistema como memoria caché del *CD-ROM*. Se pueden reservar desde pocos KBytes hasta 2 MBytes de memoria *RAM*. La caché en la memoria *RAM* es capaz de mejorar sutilmente el rendimiento de la unidad de *CD-ROM*, sacrificando un poco el rendimiento de la memoria *RAM*.

7.2.4 Soporte físico

Muchos *CDs* grabados, y algunos *CDs* prensados de baja calidad, como los que eran vendidos con algunas revistas de informática, no podían leerse por encima de 4x o 6x sin que hubieran muchos errores de lectura. En este caso, no servía de mucho tener

una unidad de 40x, pues la velocidad de lectura quedaba limitada a la velocidad permitida por el soporte físico. Generalmente, sólo los *CDs* de juegos o de programas pueden leerse a cualquier velocidad.

Conociendo este problema, los fabricantes de lectores de *CD* crearon una nueva tecnología, a través de la cual el lector disminuye la velocidad de rotación, probando varias velocidades cada vez menores, hasta conseguir la velocidad de lectura adecuada. Por esto, al intentar leer algunos *CDs*, la luz de nuestra unidad de lectura queda parpadeando durante 2 o 3 segundos, hasta que accede a leer el *CD*.

7.2.5 La importancia del rendimiento

La velocidad del disco duro afecta directamente al rendimiento general del equipo. En muchas aplicaciones, es más importante un disco duro rápido que un procesador rápido, valiendo más la pena comprar un disco duro rápido y un procesador un poco más lento que un procesador de última generación y un disco duro lento. En el caso de una unidad de *CD-ROM*, su rendimiento tiene un impacto menor en el rendimiento global del sistema.

En realidad, sólo en algunas circunstancias, una unidad muy rápida presenta ventajas considerables sobre una unidad un poco más lenta, principalmente en el caso de usuarios domésticos.

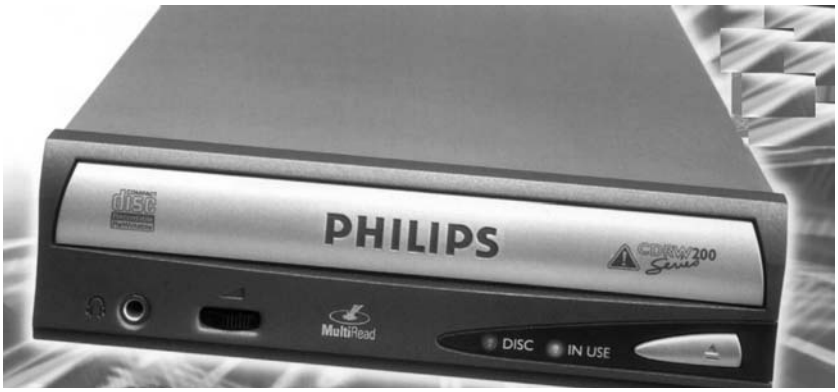
7.2.6 Por encima del límite

Una tecnología interesante que surgió cuando ya se creía que las unidades de *CD-ROM* habían alcanzado el punto máximo de su evolución, es la utilización de varios lásers de lectura en vez de uno. Con varios lásers, leyendo puntos diferentes del *CD*, la velocidad de lectura puede aumentar sorprendentemente. El primer fabricante en lanzar unas unidades compatibles con esta tecnología fue la casa Kenwood, con su *Kenwood's 72X True X CD-ROM Drive*. Su tecnología consiste en un lector que gira a una velocidad menor, semejante a la de los lectores de 12x, pero que usa 7 rayos láser en compensación, alcanzando una velocidad lineal constante de 72x. Notemos que esta velocidad es constante, y no sólo alcanzada en las pistas más externas.

Inicialmente, sólo disponemos de un láser, que al pasar por una lente especial, es dividido en siete rayos más débiles. Con la ayuda de un nuevo sistema de lentes, los lásers son alineados, de forma que pueden acceder a las pistas vecinas de la pista que está siendo leída. Como acontece con todos los lectores de *CD*, los rayos se reflejan a la que rebotan en la superficie reflexiva del *CD* y, a la que vuelven, son reflejados por un minúsculo espejo llegando al *foto-diodo*, que decodifica la señal transformándola en una secuencia de bits que son transmitidos al procesador. El uso de espejos no es ninguna novedad, pero la idea de dividir el láser en 7 rayos más pequeños es realmente ingeniosa. Más ingenioso es aún el *foto-diodo*, que es capaz de leer los 7 lásers al mismo tiempo.

7.3 La grabación de CDs

Durante varios años, los *CDs* fueron un formato físico para la lectura. Podíamos comprar un programa en *CD-ROM*, pero si por algún motivo necesitábamos copiarlo teníamos que usar disquetes, unidades *ZIP* u otro dispositivo. Sin embargo, actualmente, ya hemos vivido la gran popularización de las grabadoras de *CDs*, que ya se han convertido en tan populares como las unidades de disquetes.



Un unidad de grabación de CDs

La posibilidad de grabar *CDs* también es muy útil para el transporte de datos. En este ramo, el *CD* se ha revelado como una opción muy interesante en términos de coste y rendimiento, ya que posee una capacidad equivalente a la de 6.5 discos *ZIP* de 100 MB, o más de 400 disquetes.

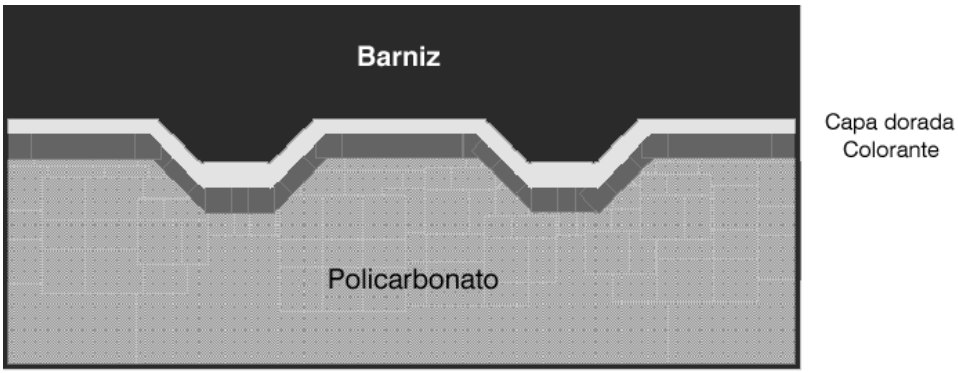
Otro recurso interesante es el recurso de multisesión, que permite dejar "abierto" un *CD* grabado. A través de este recurso es posible grabar una cantidad pequeña de datos y después ir grabando más datos hasta llenar la capacidad total del *CD,* disminuyendo bastante el número de *CDs* físicos necesarios para hacer las distintas copias de seguridad diarias o para transportar pequeñas cantidades de datos

Así como en las unidades de CD-ROM, la velocidad de grabación también se muestra en múltiplos de 150 KB/s. Una grabadora 1x será capaz de grabar los *CDs* a una velocidad de 150 KB/s, una de 2x a 300 KB/s, una de 4x a 600 KB/s y así en adelante. Generalmente, las unidades grabadoras son capaces de leer los *CDs* a una velocidad mayor de la que son capaces de grabar: una unidad 12x 10x, por ejemplo, es capaz de leer a 12x y grabar a 10x.

7.3.1 El soporte físico

Como hemos visto en el inicio de este capítulo, un *CD* prensado común está compuesto por tres capas: una capa de plástico de cerca de 1.2 mm de espesor, una capa de aluminio, oro o platino donde se graban los datos y, sobre ella, una capa protectora de barniz. En un *CD-R* también tenemos estas capas, la diferencia es que existe una cuarta capa, entre la de plástico y la capa reflexiva: la capa donde se graban los datos.

Esta fina capa está compuesta de productos sensibles al calor, que tiene una composición química alterable debido al calor generado por el láser de la grabadora, mucho más potente que el usado en la lectura del *CD*. Las partes de la superficie quemadas por el láser quedan opacas y crean pequeñas ampollas, dejando de reflejar la luz del lector, sustituyendo los surcos de los *CDs* prensados. Por eso, en la jerga de los informáticos es usada la expresión "quemar *CDs*" para referirse al proceso de grabación.



Las distintas capas de un CD-R

Capa de policarbonato

En los *CDs* prensados la capa de policarbonato es prensada contra un molde, grabándose en el proceso los surcos que almacenan los diferentes datos. En un *CD-R*, la capa de policarbonato también es prensada, aunque sin embargo, en vez de grabarse los surcos, se grabada la espiral de grabación. Todo el *CD* virgen ya viene con esta espiral pregrabada, donde se grabarán los datos a posteriori. La grabadora no es capaz de grabar datos fuera de la espiral.

Capa reflexiva

En los *CDs* prensados, la capa reflexiva siempre está compuesta de aluminio, plata, platino u oro. En los *CD-R*, la capa reflexiva puede ser tanto de oro de 24 quilates como de plata. Por otra parte, la capa grabable puede ser transparente, azul o verde, dependiendo del material usado. De acuerdo con la combinación de la capa reflexiva y de la capa de grabación, podemos tener *CD-R* de varios colores diferentes.

7.3.2 Colores

Actualmente existen 5 substancias diferentes que pueden ser usadas para formar la capa de grabación de los *CD-R*. Todas estas substancias son orgánicas, un tipo de plástico o combustible, y justamente por esto pueden ser quemadas por el láser de la unidad grabadora. Cada una de estas substancias fue desarrollada por una compañía diferente, que tienen su patente. Al ser diferentes, cada sustancia tiene un color predominante.

Como la capa de grabación es semitransparente, el color del *CD-R* está compuesto por una combinación del color de la capa reflexiva (que puede ser dorada o plateada) con el color de la capa de grabación (azul, verde o transparente).

Además del color, las sustancias difieren en durabilidad y refracción de la luz. La *Phthalocyanine* es la de mejor calidad de entre todas. Como es casi transparente, los *CD-R* fabricados con ella poseen una refracción de luz casi equivalente a la de los *CDs* prensados, siendo compatibles con todos los lectores de *CD-ROM*. La Phthalocyanine también tiene una durabilidad mayor, cerca de 100 años según el fabricante. Esto no significa que los *CDs* fabricados con esta sustancia durarán todo este tiempo, pues tenemos también la variable de la durabilidad de la capa reflexiva y condiciones de almacenaje y conservación. Sólo los *CDs* dorados usan *Phthalocyanine*.

La casa Mitsui creó un tipo más sofisticado de *Phthalocyanine* llamado *Advanced Phthalocyanine* o de *Phthalocyanine tipo 5*. Esta nueva sustancia posee una refracción de luz semejante a la de la *phthalocyanine* común, pero es más durable, llegando cerca de los 200 años. Su color también es un poco diferente, en vez de transparente es de un verde clarito.

La *Cyanine* fue la primera sustancia usada en los *CD-R*, pero posee algunas desventajas en comparación con la *Phthalocyanine*. Su durabilidad ronda los 50 años y su color azulado causa una disminución en la refracción de luz, haciendo que algunos discos sean incompatibles con algunos lectores más antiguos y aumentando el número de errores de lectura en otros.

Por otra parte, la *Metallized Azo* es una sustancia desarrollada recientemente. Su refracción de la luz es semejante a la de la *Cyanine* (a pesar de su coloración más oscura), pero su durabilidad es mayor, estimándose en cerca de los 90 años. Esta sustancia sólo es usada sólo en los *CD-R* de coloración azul, que pueden distinguirse de los azules hechos de *Cyanine* por una tonalidad más oscura.

Finalmente, tenemos el *Formazan* desarrollado recientemente. El *Formazan* combina algunas de las características de la *Cyanine* y la *Phthalocyanine*, poseyendo una durabilidad estimada cercana a los 50 años y una refracción de la luz un poco inferior a la de la *Phthalocyanine*.

7.3.3 Audio vs datos

Usando una grabadora de *CDs*, podemos grabar *CDs* de datos y *CDs* de audio, que podremos reproducir en cualquier reproductor de CDs. Sin embargo, existen diferencias en la lectura de un *CD* de música dependiendo del lector, ya sea un reproductor de *CDs* o un *Discman*.

La velocidad de lectura de un *CD* de música es de 1x, un reproductor de *CDs* los lee a esta misma velocidad, y va reproduciendo la música conforme la va leyendo en el *CD*. Si ocurriese un error en la lectura de algún sector del *CD* no se intentará una

segunda lectura (no habría tiempo para eso) sino que, simplemente, se ignorará el dato ya que no pudo ser leído y se usará la interpolación para encubrir el "agujero". Muchas veces, el arreglo hecho usando la interpolación ni se nota, pero en otras aparece un ruido, una distorsión en el sonido. Mientras más errores se producen en la lectura, peor será la calidad del sonido.

Para grabar *CDs* de música, los soportes físicos dorados (de *Phthalocyanine*) y verde claro (de *Advanced Phthalocyanine*) son mejores, pues presentan una mayor refracción de la luz que los verdes y los azules. Con una mayor refracción de la luz, para el lector es más "fácil" distinguir entre los 1 y 0 grabados, pues la señal es más fuerte y, consecuentemente, ocurren menos errores de lectura. Los discos dorados y verde claro también presentan incompatibilidad con un número menor de lectores, justamente a causa de la mayor refracción de la luz.

7.3.4 Durabilidad

Un punto positivo es la mayor vida útil de los discos físicos de *Phthalocyanine*, que con una conservación adecuada llega a ser de unos 200 años, contra los sólo 50 de los discos físicos verdes y azules.

Uno de los principales determinantes en la durabilidad de un *CD* es, justamente, la capa protectora que se encuentra encima de la capa reflexiva. Algunos fabricantes sólo aplican una fina capa de barniz sobre ella y sus *CDs* presentan una cara superior más brillante y un frágil aspecto. Este tipo de *CDs* es el más sensible tanto a arañazos, como a la luz solar. Otros *CDs,* generalmente los de buenas marcas y más caros, poseen sobre la capa de barniz una nueva capa protectora, destinada a una mayor protección y a facilitar la impresión del *CD*. En algunos casos, como en los *CDs* con tecnología *Infoguard*, se usan varias capas protectoras. Cuanto mayor que sea la protección sobre la capa reflexiva, más resistente es el *CD* resultante.

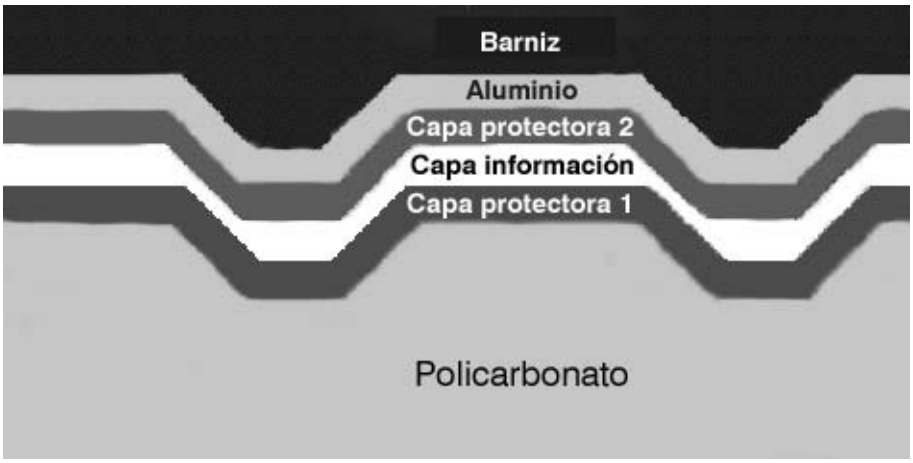
Mi experiencia con la grabación de *CDs* es que, independientemente del color, los *CD-R* de buenas marcas difícilmente presentan problemas, mientras que los *CD-R* sin marca y más baratos, suelen dar más de un dolor de cabeza, independientemente del color.

7.3.5 CD-RW

Las unidades grabadoras de *CDs* son muy prácticas para realizar las copias de seguridad de grandes cantidades de datos, transportar archivos o programas, y duplicar *CDs* de audio o de datos, tareas para las cuales presentan una excelente relación costebeneficio, considerando el bajo coste de los soportes físicos vírgenes.

Sin embargo, los *CD-R* traen la desventaja de que no permiten la regrabación. Si queremos grabar un *CD* hoy y mañana necesitamos cambiar un único archivo, tendremos que grabar otro disco nuevo, sin ser posible cambiar solamente el archivo que fue modificado, como hacemos en los disquetes o en el disco duro. Para solucionar este inconveniente, surgieron los *CDs* regrabables (*CD-RW*), que pueden modificar su conteni-

do de forma fácil, prácticamente con la misma facilidad que tenemos con los soportes físicos magnéticos, como *ZIPs* y disquetes.



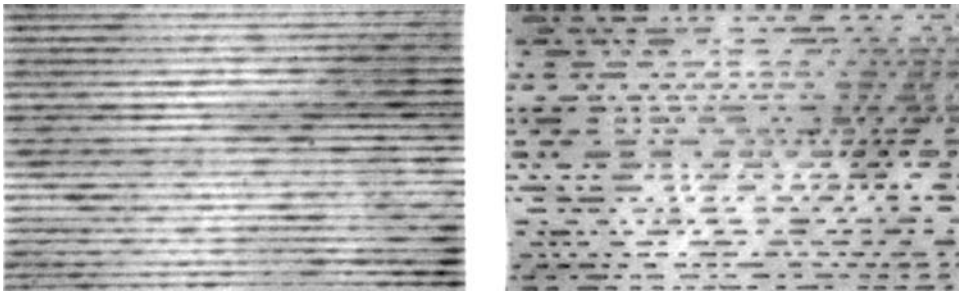
Las distintas capas de un CD-RW

Esto es posible debido a la sustancia usada en la composición de la capa de grabación de los *CD-RW*. Mientras que en un *CD-R* la capa de grabación es quemada por el láser, siendo inalterable después de la grabación, el soporte físico regrabable puede ser alterado entre un estado cristalino y un estado opaco a través de unos lásers con distintas intensidades. Esta técnica es muy interesante, pues con el láser se funde el material, pero según la temperatura de fusión, el material adopta características distintas al enfriarse. Una temperatura de fusión más alta convierte el material en opaco, mientras que un láser un poco más débil lo hace volver al estado original. Según los fabricantes, este tipo de soporte físico puede ser reescrito más de 1.000 veces antes de empezar a presentar cualquier problema, pero esto depende de la calidad del material: algunos *CD-RW* empiezan a presentar errores después de pocas grabaciones.

En principio, cualquier programa actual de grabación de *CDs* es capaz de trabajar con *CD-RW* sin ningún problema. Algunos programas son tan fáciles de usar que permiten grabar archivos en el *CR-RW* con arrastrrlos en las ventanas del *Explorador de Windows* hasta el icono de la unidad grabadora, exactamente como haríamos para grabar los archivos en un disquete.

El mayor problema con los *CD-RW*, sin embargo, es la compatibilidad. Un *CD-R* refleja más del 70% de la luz que es reflejada por un *CD* prensado y, por eso, puede ser leído en cualquier unidad sin mayor dificultad. En el caso de un *CD-RW*, la refracción es menor, cerca de sólo un 20%. Para leer estos soportes físicos, el lector necesita ser equipado con un circuito especial, llamado *AGC* de *Automatic Gain Control* o control automático de ganancia. Este circuito, que está incorporado en el cabezal de lectura es capaz de detectar la baja tasa de reflexión del soporte físico y aumentar la intensidad del láser de lectura. Entonces tenemos un láser más fuerte de lo habitual para com-

pensar la baja reflexión del soporte físico, haciendo que el láser reflejado tenga una intensidad parecida con el normal.



En estas imágenes podemos comprobar la menor intensidad de los surcos en los CDs grabables (derecha) que en los CDs prensados (izquierda)

Sólo los lectores de *CD* más actuales, y aún así tampoco todos, poseen el circuito *AGC*, pues este circuito es relativamente caro. Podemos usar un *CD-RW* para realizar nuestras copias de seguridad, por ejemplo, posteriormente leyéndolos en la misma unidad grabadora, pero a lo mejor tenemos que grabar un *CD-R* para poder transmitírselos a un amigo con una unidad de *CD-ROM* común.

Además de esto, los *CD-RW* tienen algunos inconveniente más. Como la tasa de refracción luminosa es muy baja, la lectura del *CD-RW* es más difícil, haciendo que los soportes físicos regrabables sean mucho más sensibles a los arañazos, suciedad, etc. Trabajando con *CD-RW* deberemos ser especialmente cuidadosos con el cuidado del almacenamiento.

Otro problema reside en la durabilidad, que es menor que la de los *CDs* convencionales, y también en la sensibilidad de las lecturas sucesivas, dado que el material que compone los *CD-RW* es muy sensible a los cambios de temperatura. Por otro lado, debido a la baja refracción luminosa, el lector es forzado a utilizar un láser más fuerte que el normal para realizar la lectura. El resultado es que después de ser accedido varías veces para su lectura, un *CD-RW* empieza a presentar algunos datos corrompidos. Definitivamente, los *CD-RW* no son los soportes físicos más aconsejables para guardar los datos más importantes durante largos periodos de tiempo.

Actualmente, la tasa de grabación de un *CD-RW* es de a lo sumo 12x. La mayoría de las grabadoras fabricadas son capaces de grabar *CD-R* y *CD-RW*, presentando velocidades muy diferenciadas. Hoy en día, lo más común y adecuado, son las unidades de 52x 40x 12x (52x para la lectura, 40x para *CD-R* y 12x para *CD-RW*).

7.3.6 CDs de 80 minutos

El *Red Book*, la especificación original de los *CDs*, estipula una distancia fija entre la primera y la última pista del *CD*, aunque sin embargo, permite una cierta flexibilidad en la distancia entre las otras pistas, previendo soportes físicos de 74 minutos (lo más común) como soportes físicos capaces de almacenar 80 minutos de música.

Los *CDs* de 80 minutos poseen más pistas que los *CDs* normales de 74 minutos, y también permiten almacenar una cantidad mayor de datos: 700 MBytes contra los 650 MBytes de un *CD* normal. Como ya hemos visto, la distancia entre la primera y la última pista del *CD* continúa siendo la misma, lo único que cambia es la distancia entre las demás pistas del *CD*, que pasa a ser un poco menor. Notemos que este recurso ya estaba previsto en el *Red Book* original, y que nos se trata de ningún formato nuevo.

Como los CDs de 80 minutos pueden leerse en prácticamente todas las unidades de *CD* se vienen usando cada vez más. También existen *CD-R* de 80 minutos de distintas marcas. Estos CD-R son aceptados por casi todas las unidades grabadoras, pero es necesario que el programa de grabación soporte el uso de este tipo de soporte físico. Algunos programas compatibles con los *CD-R* de 80 minutos son *Easy CD Creator Deluxe*, *CD Wizard Pro*, *CDRWIN*, *DiscJuggler*, *Nero Burning Rom* y *WinOnCD*.

En caso de incompatibilidades con los *CDs* de 80 minutos es posible hacer que la unidad grabadora sea compatible a través de una actualización del *firmware* de la grabadora. El *firmware* es una especie de "*BIOS*" de la unidad grabadora, una pequeña cantidad de memoria *flash* que almacena las rutinas responsables del funcionamiento de la grabadora. Encontraremos los archivos binarios y el programa de grabación en la página *Web* del fabricante de nuestra grabadora. La actualización del *firmware* conlleva ciertos riesgos y, por eso, no debemos dejar de leer atentamente las instrucciones y las advertencias suministradas por el fabricante.

7.3.7 Oversize

El *Red Book* especifica que además de los *CDs* de 74 o 80 minutos normales, un *CD* debe tener una zona de terminación de por lo menos 90 segundos más. Esta zona de terminación, conocida por *Lead Out* comprende las pistas más externas del *CD* y queda siempre en blanco. A pesar de que el formato exige 90 segundos de *lead out*, generalmente, los fabricantes acostumbran a reservar una zona mayor, de 120, 150 o incluso 180 segundos.

A pesar de que los formatos indican que el *lead out* debe quedarse siempre en blanco, sus pistas son perfectamente utilizables para la grabación de datos. Muchos de las unidades grabadoras más modernas son capaces de grabar datos en la zona de terminación, llegando a los 78 minutos de música o a los 690 MBytes de datos en los CDs normales, o hasta los 83 minutos en un CD de 80 minutos. Este recurso es llamado *oversize*, o por encima de su capacidad en una libre traducción.

No todas las grabadoras de *CDs* soportan el *oversize*, ya que para grabar *CDs* en este modo es preciso desplazar el cabezal de grabación hasta el final del *CD*, tarea para la cual muchas unidades de grabación no están preparadas, ya sea por limitaciones físicas o del mismo *firmware*.

Para grabar *CDs* con *oversize* también es necesario soporte por parte del programa de grabación. Sólo los programas más recientes, como *Nero*, *CDRWIN*, *Disc Juggler*, *CD*

Rep Plus y *CD Wizard Pro* entre otros soportan este recurso. Por ejemplo, en el *Nero* activamos la función de *oversize* a través de la opción de menú *File/Preferences/ Expert Features/Enable Overburn*. El mayor problema es que estos programas son comerciales, al contrario de los programas más simples que vienen con las unidades de grabación.

Todo *CD-R* virgen ya viene con algunas informaciones pregrabadas. Estas informaciones son llamadas *ATIP* e incluyen las informaciones esenciales para que el grabador pueda realizar la grabación: la capacidad del disco, el fabricante del disco, la formulación física del disco, la velocidad máxima de grabación soportada y la última dirección donde pueden ser grabados los datos o sea, la capacidad máxima del disco incluyendo el recurso de *oversize*.

Sólo la unidad grabadora puede leer el *ATIP*, la unidad de *CD-ROM* o el reproductor de *CDs*, normalmente, no pueden leerlas ya que no necesitan conocer esta información para realizar la lectura del *CD*.

Notemos que en el *ATIP* están indicadas tanto la última pista del *CD* donde se pueden grabados datos (la última pista antes del inicio del *Lead Out*), pista que indica la capacidad normal del *CD* (alrededor de los 74 minutos) como la pista donde termina el *Lead Out*. Una unidad grabadora que no soporte el *oversize* sólo grabaría hasta el inicio del *Lead Out*, en la pista indicada en el *ATIP*, mientras que una grabadora compatible con el recurso de *oversize* puede ignorar esta indicación y grabar hasta el final del *Lead Out*, usando la capacidad máxima del *CD*.

Los fabricantes advierten que, en algunos casos, intentar grabar usando el recurso de *oversize* en una unidad grabadora que no sea no compatible puede incluso dañar la unidad. Otro problema con el *oversize* es que muchos lectores de *CDs*, incluidos algunos de los de más recientes, no son capaces de leer *CDs* grabados con este recurso y, además de eso, la calidad del soporte físico en las últimas pistas aprovechadas para grabar más datos suele ser inferior (ya que esta zona no sería usada normalmente), ocasionando algunos errores de lectura y disminuyendo el rendimiento del *CD*.

7.3.8 Buffer Underrun

La grabación de un *CD-R* es un proceso razonablemente lento que no puede ser interrumpido de ninguna manera, ya que de lo contrario, el soporte físico que está siendo grabado quedará inutilizable. El problema que más atormenta a los usuarios de grabadoras de CD es el famoso *Buffer Underrun*, una situación donde la grabación se interrumpe por la falta de datos.

Toda grabadora posee un pequeño búfer, de 2 o 4 MBytes en las unidades más recientes y 1 MBytes o 512 KBytes en las grabadoras más antiguas o de baja calidad. Este búfer funciona guardando datos que se usarán si se produce cualquier interrupción momentánea en el suministro de datos hacia la grabadora, evitando la pérdida del soporte físico. Cuando la grabadora vuelve a recibir datos, el búfer se llena de nuevo.

Puede parecer extraño que cualquier equipo no sea capaz de suministrar un flujo de datos de sólo 600 o 1200 KB/s, cuando un buen disco duro es casi 20 veces más rápido que eso, y hasta un puerto paralelo *ECP* sería capaz de soportar de sobras ese flujo.

En este caso, el problema no tiene mucho que ver con la velocidad en la transmisión de datos y sí con la capacidad de transmitir datos desde el disco duro hacia la grabadora de forma ininterrumpida. En este aspecto, los discos duros y las grabadoras *SCSI* son mucho mejores que las *IDE*, ya que el *SCSI* permite una transferencia de datos mucho más estable y menos dependiente de la disponibilidad del procesador.

Lo ideal sería usar una grabadora *SCSI* y un disco duro *SCSI*, lo que aseguraría unas grabaciones sin problemas de *buffer underrun*. La segunda opción sería una grabadora *SCSI* y un disco duro *IDE* razonablemente rápido, lo que también generaría unos buenos resultados. Las grabadoras *SCSI* son un poco más caras que las *IDE*, pero esta diferencia vale la pena, principalmente porque casi todas las grabadoras *SCSI* vienen acompañadas de una tarjeta *SCSI*. Teniendo una tarjeta *SCSI* instalada, también tendremos la posibilidad de instalar discos duros y otros periféricos *SCSI* en el equipo.

Si ya tenemos la grabadora *IDE*, nuestra tercera opción será instalar la grabadora sola en una *IDE* secundaria, dejando el disco duro en una *IDE* primaria. Si nuestro disco duro es lo razonablemente rápido y la grabadora posee por lo menos 2 MBytes de búfer, esta combinación también va a funcionar sin problemas. Deberíamos intentar evitar instalar la grabadora en el mismo puerto *IDE* que el disco duro, pues esto causaría algunas interrupciones en las transferencias de datos que pondrían la grabación en riesgo.

En cualquiera de las situaciones que acabamos de mencionar, debemos desactivar todos los programas residentes antes de realizar las grabaciones, especialmente el antivirus, pues entorpecen mucho las grabaciones, ayudando a causar vacíos en el búfer. Cuanto mayor sea el búfer de la grabadora, mayor es la seguridad. Justamente por eso, muchos de las grabadoras modernas traen 4 MBytes de búfer, lo que garantiza unas grabaciones más seguras.

También es posible hacer grabaciones a partir del *CD-ROM*, facilitando la tarea de duplicar un *CD*. Siempre se ha dicho que el *CD-ROM* no se debe instalar en el mismo puerto IDE que el disco duro, y realmente eso es correctísimo. Sin embargo, si tuviésemos un *CD-ROM IDE*, un disco duro *IDE* y una grabadora también *IDE*, lo más aconsejable en el caso de querer realizar grabaciones desde el *CD-ROM* sería instalarlo en la *IDE* primaria junto con el disco duro, dejando la grabadora sola en la *IDE* secundaria.

A pesar de ser más escasas, también existen grabadoras externas, que utilizan el puerto paralelo. A pesar de la portabilidad, estas unidades acaban siendo las peores, pues además de ser más lentas, el puerto paralelo consume muchos recursos del procesador durante las transferencias de datos, lo que acaba causando muchos problemas de *buffer underrun*.

Deberíamos evitar al máximo comprar grabadoras paralelas. Si por casualidad ya tenemos una, las pistas para conseguir hacer buenas grabaciones son configurar el puerto paralelo como *ECP* en el *Setup* del *BIOS* y desactivar todos los programas que se estén ejecutando.

Otra opción de grabadora portátil son las grabadoras *USB*. Como todos los demás periféricos *USB*, estos son cada vez más comunes. El puerto *USB* es una interfaz mucho más adecuada que el puerto paralelo para la grabación de *CDs*: es más rápida y, principalmente, consume muchos menos recursos del procesador durante las transferencias de los datos Las grabadoras *USB* suelen funcionar mejor que las paralelas.

La instalación de la unidad también es muy simple, después de conectar el aparato basta encender el equipo para que *Windows* lo detecte automáticamente. Entonces sólo deberemos suministrarle los *drivers* del fabricante. Sólo será preciso suministrar los *drivers* la primera vez que usemos la unidad. Como las placas base vienen con dos salidas *USB* como mínimo, deberíamos reservar una para la grabadora y conectar los demás periféricos *USB* en la otra

Un paso muy interesante antes de realizar la grabación, es desfragmentar el disco duro, pues con los datos almacenados de forma secuencial en el disco duro, el tiempo de acceso a los datos y la tasa de transferencia son mejores, disminuyendo mucho la frecuencia de un *buffer underrun*. Si es posible, también podríamos reservar una partición de poco más de 700 MBytes formateada con *FAT32* para hacer una imagen del *CD* que vamos a grabar.

Debemos evitar usar el equipo mientras estamos grabando *CDs*, pues cualquier acceso al disco duro puede resultar en un más que posible vacío del búfer, especialmente en grabadoras *IDE* con 1 MByte o menos de búfer. Abrir un programa o un archivo puede estrangular la transferencia de datos durante un tiempo razonable, más que suficiente para vaciar el búfer, principalmente si estamos a merced de la multitarea de *Windows 95/98*.

Dependiendo de la configuración de nuestro equipo de la velocidad del disco duro, puede ser posible ejecutar otros programas durante la grabación de los *CDs*. En este caso, sólo lo sabremos después de intentarlo, pero la mayoría de los programas de grabación permiten ejecutar una simulación antes de realizar la grabación. En esta simulación, se reproducen todas las etapas de la grabación, sin que ningún dato sea grabado en el *CD-R*, permitiendo probar cosas sin arriesgarse a perder ningún soporte físico.

Si con todos estos cuidados seguimos teniendo problemas de buffer underrun (eso puede pasar en el caso de que la grabadora posea un búfer pequeño o el disco duro sea muy lento) podemos disminuir la velocidad de la grabación a 2x o 1x. Con una velocidad menor, el búfer tardará más en vaciarse, y el problema podría quedar totalmente resuelto.

7.4 DVD o Digital Versatile Disc

Aparecido originalmente como *Digital Video Disc*, el *DVD* se conoce actualmente como *Digital Versatile Disc* y ya es un medio de almacenamiento universal. Las unidades de *DVD-ROM* están destinadas, y de hecho ya lo han echo, a tomar el relevo de las unidades *CD-ROM* habituales hasta hoy día, y algunos de los fabricantes más conocidos ya han anunciado su próxima dedicación exclusiva a la producción de unidades *DVD-ROM*.

Una de las primeras unidades DVD-ROM en la que el DVD se introducía directamente en la unidad

Los precios de las unidades *DVD-ROM* se sitúan actualmente al mismo nivel que los de una buena unidad de *CD-ROM*. No obstante, la gama de artículos *DVD* todavía es algo confusa, ya que los fabricantes han desarrollado diferentes tipos de *DVD* grabables (*DVD-R*, *DVD-RAM*) y todavía no existe la absoluta certeza de que los medios sean compatibles entre sí.

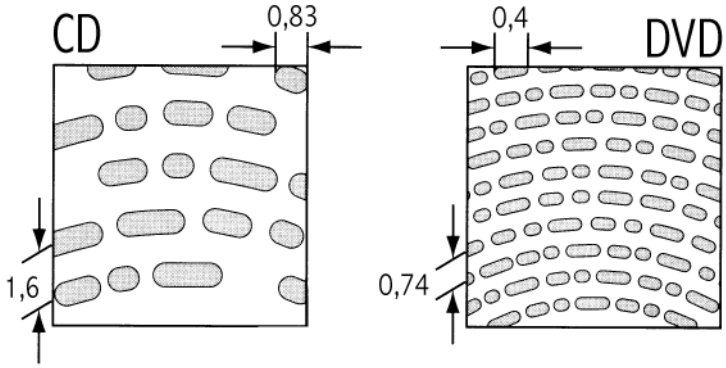
7.4.1 El principio de funcionamiento

Un disco *DVD* tiene las mismas medidas que un *CD-ROM* (12 cm de diámetro), pero su capacidad mínima es de 4.7 GBytes, lo que representa una cantidad de datos casi siete veces superior a la que pueden guardar los *CD-ROM*. Para conseguir esta capacidad se han tenido que introducir, naturalmente, bastantes modificaciones básicas respecto a los *CD-ROM*:

1. Los *pits* y *lands*, así como sus distancias, son más pequeños.
2. A fin de que la focalización sea todavía más precisa, el láser trabaja en el área de rojo (635-650 nm) en lugar de la de infrarrojos (780 nm) como en el *CD-ROM*.
3. Se han modificado los mecanismos de direccionamiento y corrección de errores.
4. Se utiliza un máximo de dos capas de información por cara y también en los *DVD* dobles, con lo que se consigue una capacidad máxima de 17 GBytes (dos caras y dos capas) en total.

Debido a estas modificaciones, el *DVD* no es compatible con el *CD-ROM*, y se necesitan unidades nuevas que, naturalmente, también pueden leer los *CDs* normales. El aumento de densidad de datos producido en el *DVD* respecto al *CD-ROM* tiene como

consecuencia inevitable que la suciedad (huellas de dedos) y los daños ligeros (arañazos) se notan mucho antes en el *DVD*, por lo que la corrección interna de errores de las unidades *DVD* es más estricta.



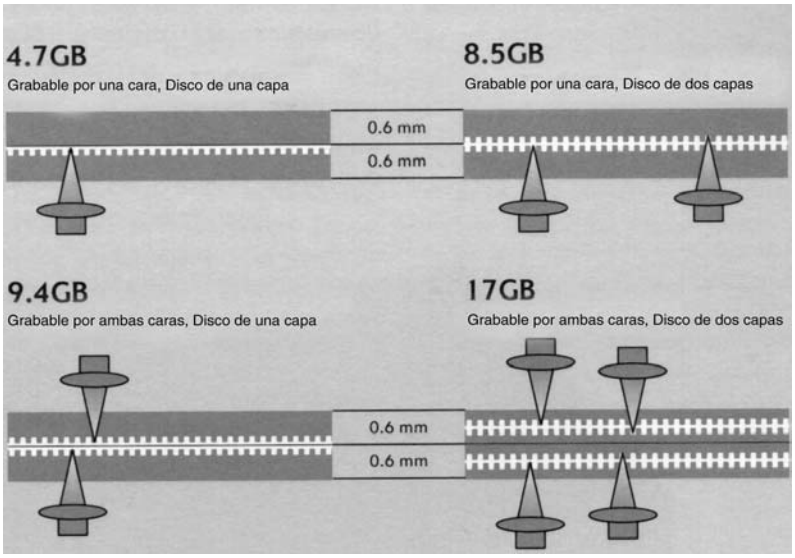
Las distancias y medidas de los pits en los CD-ROM y DVD en micras (micrómetros)

De manera simplificada, la estructura de un *DVD* se puede imaginar igual a la de un *CD-ROM*. No obstante, el *DVD* tiene una característica especial, y es que puede estar formado por dos capas de información (*Dual Layer*), de diferentes materiales cada una de ellas. Primero se grava en el disco la estructura de *pits* y *lands*, y después se recubre con una capa semi-reflectante (de oro). A continuación viene una capa de material plástico en la que se grava la segunda capa de información, y sobre ésta se extiende una capa reflectante de aluminio.

Uno de los pasos más difíciles es la aplicación de la capa semi-reflectante (y semitransparente), pues su grosor ha de ser tal que el dispositivo óptico de lectura pueda registrar la luz reflejada por ambas capas de información y diferenciarlas entre sí. Por eso el láser trabaja con diferentes intensidades en la lectura de un mismo *DVD*.

Un *DVD* de doble capa no tiene exactamente el doble de capacidad que un *DVD* sencillo (4,7 GBytes), pues los *pits* y *lands* son algo más grandes en la segunda capa a fin de facilitar su detección. Lo mismo sucede con los *DVD* de dos caras, que pueden contener una cantidad máxima de datos de 17 GBytes. Para poder leer la segunda cara de estos *DVD,* normalmente debe darse la vuelta al disco, tal como se hacía en los antiguos tocadiscos. En este proceso no es del todo fácil evitar las huellas (de los dedos) en la superficie. En general se distinguen las cuatro variedades siguientes de *DVD*:

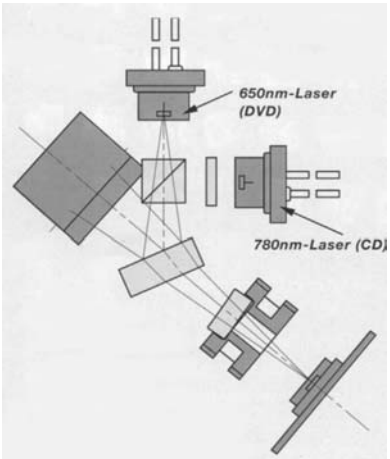
- **DVD-05**: una cara y una capa, con una capacidad máxima de 4.7 GBytes.
- **DVD-9**: una cara y dos capas, con una capacidad máxima de 9.4 GBytes.
- **DVD-10**: dos caras y una capa, con una capacidad máxima de 8.5 GBytes.
- **DVD-18**: dos caras y dos capas, con una capacidad máxima de 17 GBytes.



Los distintos tipos de DVD-ROM

7.4.2 Tipos

Una unidad *DVD-ROM* puede estar dotada con dos láseres de diferente longitud de onda, y garantizar así la compatibilidad hacia atrás con los *CD-ROM* y *CD-R/RW*. Este tipo de unidad *DVD* puede, por lo tanto, leer los formatos estándar habituales de *CD*, pero esto no siempre es así en las antiguas unidades de *DVD*. También aquí se dan las incompatibilidades típicas que ya conocemos de las unidades *CD-ROM*: unos *CDs* se pueden leer, otros no, y en el caso de las unidades de *DVD* estas incompatibilidades son todavía más patentes, lo que hay que achacar en muchos casos a una corrección de errores más bien insuficiente.



Las unidades DVD-ROM trabajan con dos láseres y un solo sistema de lentes

En las unidades *DVD-ROM* se utiliza el formato *Universal Disc Format* (*UDF*) tal como fue introducido con los *CD-RW* y que se puede emplear también con los *DVD-RAM*. Puesto que los *DVD-RAM* representan un sistema de disco removible, en estas unidades también se pueden utilizar los formatos habituales como *FAT16*, *FAT32* o *NTFS*.

Las unidades *DVD-ROM* de segunda generación se distinguieron por el factor 2x, que corresponde a una tasa de transferencia de datos de 2.700 KBytes/s aproximadamente y que equivale, por lo tanto, a 20x o los 24x en el modo *CD*. La tercera generación de unidades *DVD-ROM* vino especificada como 6x y corresponde aproximadamente a una unidad *CD-ROM* de 32x.

7.4.3 Vídeo y audio DVD

Una de las razones principales que promovieron al desarrollo del *DVD* fue cubrir las necesidades de empresas como los estudios cinematográficos de Hollywood y las de grandes medios de comunicación (Time Warner), ya que el *DVD* permite almacenar películas completas, en una calidad inmejorable, en un solo disco. Por eso, las unidades *DVD-ROM* resultan especialmente apropiadas para la reproducción de *Vídeo-DVDs*. Para ello se necesita, adicionalmente, un decodificador, ya que los datos de vídeo y audio están codificados en el formato *MPEG-2/4*, es decir, grabados en un formato que permite ahorrar espacio en memoria: el formato del *Motion Pictures Experts Group*. Mientras que hasta hace no mucho estas tarjetas decodificadoras todavía eran muy caras, en la actualidad muchas unidades *DVD-ROM*, por ejemplo las de la casa CreativeLabs, las incluyen en su paquete.

Teóricamente, estos decodificadores no son necesarios en los PC con una *CPU* de 300 MHz y más (*AMD-K6-2*, *Pentium II*, *Celeron* en adelante), ya que estas CPU pueden descodificar los datos de vídeo. Lo único que se necesita es un programa de reproducción apropiado (decodificador de programas) que suele ir incluido, en la mayoría de los casos, en el paquete de entrega de la tarjeta gráfica.

PowerDVD es un reproductor DVD para vídeo MPEG-2/4 y AC-3-Audio

Si nuestra tarjeta no dispone de ningún reproductor de programas *DVD*, podemos echar una ojeada a la página en *Internet* del fabricante de la tarjeta, ya que es muy posible que en ella encontremos el programa reproductor *DVD* que necesitamos y podamos bajarlo a su sistema. Naturalmente, estos programas sólo funcionan con tarjetas gráficas actuales *AGP* que utilizan un moderno *chip* 3D. Por lo tanto, es la tarjeta gráfica quien determina en última instancia la calidad de la imagen de vídeo, y no la unidad de *DVD*, si exceptuamos el caso de que una unidad muy lenta puede hacer "vibrar" la imagen.

Si la tarjeta gráfica tiene una salida de TV, el vídeo también se puede ver en una televisión conectada al PC. Lo único que faltará entonces será el sonido. En los *DVDs* de vídeo normales el sonido suele estar guardado tanto en *Dolby Surround* como en formato *Dolby AC-3*. El formato *Dolby Surround* es el sistema audio utilizado en el cine y las cintas de vídeo tradicionales.

En los *DVDs*, el formato utilizado para este sistema es el *PCM* (*Pulse Code Modulation*). Con ayuda de un *Dolby Prologic Decoders*, a partir de la información acústica estereofónica se generan seis canales: delante izquierda, delante derecha, centro, bajo profundo (*subwoofer*) y detrás; la información para los dos altavoces posteriores es idéntica. En contrapartida, el sistema *Dolby AC-3* viene preparado "de fábrica" con informaciones separadas para seis canales, a saber, en forma de *bitstream*, es decir, en forma de datos digitales que se decodifican de forma parecida a un *MP3* y que, a pesar de la reducción de datos, ofrecen un sonido mejor que el *Dolby Prologic* analógico.

Además del *AC-3* existen también otros formatos de audio digitales, como *MPEG-2* (8 canales con 640 Kbits/s) o *DTS* (*Digital Theater System*, con 1,4 Mbits/s), pero todavía se está lejos de alcanzar un estándar general para los *DVDs*, por lo cual, en la actualidad, sólo el audio analógico se puede considerar como estándar "compatible" con todos los demás formatos. La "lucha" entre los distintos fabricantes y la industria cinematográfica por los formatos que soporta el *DVD*, así como una protección óptima frente a las copias ilegales, son aspectos que frenan considerablemente el trabajo común para conseguir un estándar.

Un resultado de esta lucha son los códigos regionales, de los cuales existen seis. Con ayuda de estos códigos se pretende impedir, por ejemplo, que un vídeo *DVD* de la Región 1 (USA) se pueda reproducir en un reproductor *DVD* de la Región 2 (Europa y Japón). Este control de código regional, *Regional Playback Control* (*RPC*), se da en dos variantes: las unidades *DVD* según la primera fase del *RPC* no guardan internamente ningún código regional, mientras que las unidades en la segunda fase de *RPC* utilizan internamente un número que es el que guarda el número de cambios del código regional. Después de un máximo de cinco cambios, la unidad queda fijada en el último código configurado, que ya no se puede cambiar.

La mayoría de las unidades *DVD* están configuradas en la fase 1 de *RPC*, aparentemente con menos peligro, o bien se pueden configurar en esta fase con ayuda de puentes. En la fase 1 de *RPC*, sin embargo, el programa del reproductor no guarda el número de los cambios, algo que se puede manipular muy fácilmente, por ejemplo instalando de nuevo el programa, o manipulando directamente el *Registro*.

El procesamiento del código regional no parece, por lo tanto, muy seguro (para la industria cinematográfica) por una parte, y por otra, sólo tiene inconvenientes para el usuario. Cargar nuevos productos (*Firmware*) desde *Internet* para una unidad según la fase 2 de *RPC* no debería presentar ningún problema, y en las páginas *Web* internacionales del fabricante uno puede servirse libremente.

7.4.4 DVD-R, DVD+R, DVD-RW y DVD+RW

Los formatos de grabación de *DVD* son fundamentalmente tres: *DVD-RAM*, *DVD-R/RW* y *DVD+R/RW*. El primero de ellos es el denominado *DVD-RAM* y fue creado por las casas Panasonic, Hitachi y Toshiba. En las dos primeras generaciones de grabadoras DVD-RAM, los discos *DVD* vírgenes tenían que estar encapsulados en un cartucho de plástico, hoy en día esto no es necesario. Inicialmente nació con una capacidad de 2.6 GBytes (una cara) y de 5.2 GBytes (doble cara) pero en 1999 se consiguió perfeccionar hasta alcanzar los 4.7 GB (una cara) y 9.4 GB (doble cara), utilizándose sobre todo como sistema de copia de seguridad en el mundo informático.

Una unidad de DVD-RAM y los discos antiguos

La guerra por el dominio del mercado de grabación casera de *DVD* se concentra en estos momentos en los formatos *DVD-R/RW*, apoyado por el *DVD Forum*, y los fromatos *DVD+R/RW*, con el respaldo de la *DVD Alliance*. Si nos atenemos al nombre no parecen dos sistemas muy diferentes pues su denominación solamente se diferencia en el uso de un "-" o un "+", pero a la hora de realizar la grabación las diferencias son mas evidentes.

En primer lugar vamos a hablar de los discos grabables solamente una vez (*DVD-R* y *DVD+R*). Actualmente el grado de compatibilidad entre lo grabado en estos discos y los reproductores de *DVD* es altísimo, ligeramente mas alto en el caso de los *DVD-R*, aunque las diferencias son mínimas y es muy difícil que lo que se grabe tanto en un *DVD-R* como en un *DVD+R* no se pueda reproducir en un lector de *DVD*.

Lo que dificulta la elección de una grabadora de *DVD* es el sistema de regrabación (*DVD-RW* o *DVD+RW*) que utiliza, aquí es donde empiezan a surgir las grandes diferencias entre unas grabadoras y otras. En el tema de la compatibilidad esta es menor que con el uso de los discos grabables una sola vez. Si el reproductor de *DVD* es antiguo podemos encontrarnos con problemas a la hora de reproducir nuestro flamante *DVD-RW* o *DVD+RW* grabado, aunque este problema es menor con los reproductores modernos. En cuanto al grado de compatibilidad de uno y de otro este es ligeramente mayor en los discos *DVD+RW* que en los discos *DVD-RW*.

El formato *DVD-R/RW* lleva mas tiempo en el mercado que el formato *DVD+R/RW*, lo que significa que encontraremos un mayor número de trucos y consejos en el mercado para realizar todo tipo de grabaciones. Lo que supone un inconveniente a nivel de prestaciones (antigüedad de las unidades que soportan este formato) es una ventaja a la hora de exprimir la regrabadora, pues muchos usuarios lo han hecho ya antes.

8. La tarjeta gráfica

La función de la tarjeta gráfica es preparar las imágenes que se mostrarán en el monitor. Ya se habían fabricado tarjetas gráficas usando todo los tipos de bus existentes, *ISA*, *PCI*, y también *MCA*, *EISA* y *VLB*. Sin embargo, en la actualidad sólo utilizamos tarjetas gráficas *PCI* o *AGP*, con una predominancia general de las tarjetas *AGP* que al utilizar un bus más rápido casi siempre incorporan más recursos y nos ofrecen un mejor rendimiento.

Algunos años atrás, lo más común eran los ordenadores equipados con tarjetas gráficas y monitores *CGA*, que además de generar una imagen de bajísima calidad, no nos permitían trabajar con una buena interfaz gráfica. De la misma manera que los demás componentes del ordenador, las tarjetas gráficas y los monitores también evolucionaron de una forma increíble en estas dos últimas décadas, permitiéndonos tener imágenes casi perfectas.

A continuación trataremos las tecnologías utilizadas en las tarjetas gráficas, estudiando la evolución de los distintos formatos de vídeo.

MDA y CGA: los primeros PCs sólo disponían de dos opciones de vídeo, *MDA* (*Monocrome Display Adapter*) y *CGA* (*Graphics Display Adapter*). Entre los dos formatos, el *MDA* era el más antiguo y estaba limitado a la exhibición de textos con una resolución de 25 líneas por 80 columnas, permitiendo mostrar 2.000 caracteres por pantalla en total. Como el propio nombre sugiere, el *MDA* era un formato de vídeo que no soportaba la representación de más de dos colores.

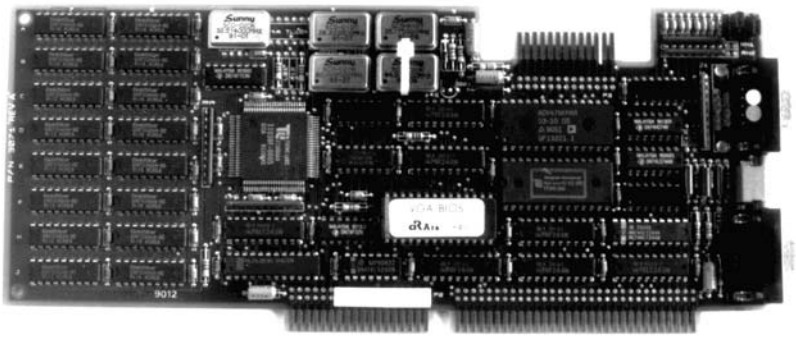
Para quienes necesitaban trabajar con gráficos, existía la opción del formato *CGA*, que podía exhibir gráficos en una resolución de 320 x 200. Pese a que el modo *CGA* poseía una paleta de 16 colores, sólo se podían representar 4 colores a la vez. El formato *CGA* también podía trabajar con una resolución de 640 x 200, pero sólo mostrando textos en modo monocromático, como en el formato *MDA*.

A pesar de ser extremadamente anticuados para los formatos actuales, el *MDA* y el *CGA* eran un buen recurso para los primeros equipos PC, ya que debido a sus limitados recursos de procesamiento se encontraban restringidos básicamente a interfaces de solamente texto.

EGA (Enhanced Graphics Adapter): para equipar el PC *AT*, lanzado en el año 84, la casa IBM desarrolló un nuevo formato de vídeo, bautizado como EGA. Este nuevo formato soportaba la representación de gráficos con una resolución de hasta 640 x 350, con la representación simultánea de hasta 16 colores, que podían elegirse en una paleta de 64 colores. A pesar de los nuevos recursos, el formato *EGA* mantenía una total compatibilidad con el *CGA*.

Una tarjeta gráfica y un monitor *EGA* eran el equipo mínimo a nivel de vídeo para poder ejecutar *Windows 3.11*. Sólo *Windows 3.0* o *Windows 3.11* soportaban la ejecución en sistemas equipados con vídeo *CGA*. Para la ejecución de *Windows 95/98*, el requisito mínimo ya era el formato de vídeo *VGA*.

VGA (Video Graphics Adapter): el *VGA* fue una gran revolución respecto a los patrones de vídeo más antiguos, soportando una resolución de 640 x 480, con 256 colores simultáneos, que podían elegirse de una paleta de 262.000 colores. Un poco más tarde, el patrón *VGA* fue perfeccionado para trabajar con resolución de 800 x 600, con 16 colores. A pesar de los avances, se mantuvo la compatibilidad con los patrones *GCA* y *EGA*, lo que permitía ejecutar las aplicaciones más antiguas sin problemas.



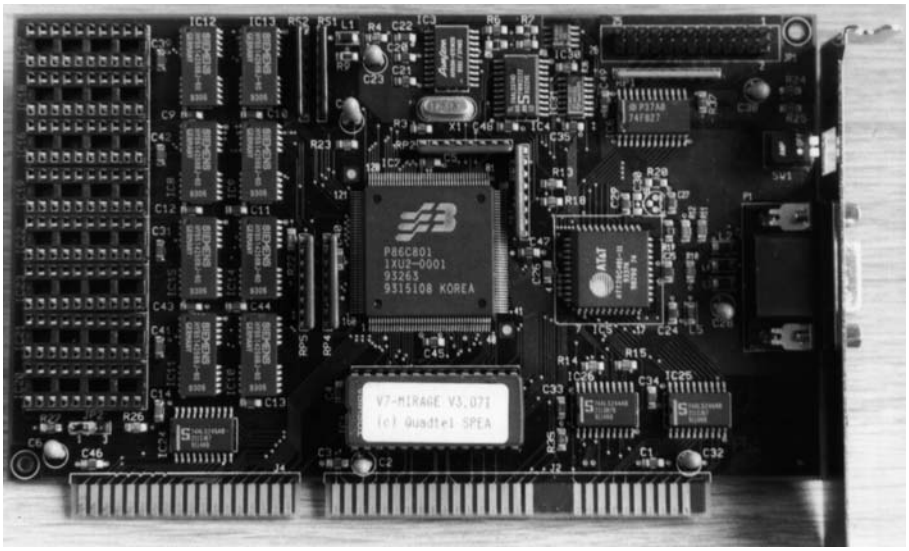
Tarjeta gráfica VGA con 512 KBytes de memoria

Súper VGA: una evolución natural del *VGA*, el *SVGA* es el patrón actual. Una tarjeta gráfica *SVGA* es capaz de utilizar y exhibir hasta 32 bits de memoria, es decir, varios millones de colores. Esto es suficiente para que el ojo humano no consiga percibir diferencias en las colores de una imagen exhibida en el monitor y de una foto coloreada. Justamente por ello, las placas de vídeo *SVGA* son también conocidas como de color verdadero o *True Color*.

8.1 2D vs 3D

Las tarjetas gráficas más antiguas sencillamente recibían las imágenes y las envían al monitor. En ese caso, el procesador era el encargado de realizar todo el trabajo. Este sistema sólo funcionaba bien cuando trabajábamos con gráficos en dos dimensiones, en las aplicaciones de *Office* o accediendo a *Internet* por ejemplo, ya que este tipo de

imagen exige poco procesamiento para generarse. Éstas eran las famosas tarjetas 2D, como por ejemplo las tarjetas *Trident 9440* y *Trident 9680*.



Una tarjeta gráfica con acelerador 2D

Las placas 2D fueron usadas sin mayores reclamaciones durante más de una década. El problema surgió al intentar ejecutar juegos 3D o los programas que utilizan gráficos tridimensionales. Entonces surgió la necesidad de usar las tarjetas gráficas 3D. La función de una tarjeta gráfica 3D es auxiliar al procesador en la creación y exhibición de las imágenes tridimensionales. Como ya sabemos, en una imagen tridimensional existen tres puntos de referencia: anchura, altura y profundidad. Un objeto puede ocupar cualquier posición en el espacio de tres dimensiones, inclusive estar detrás de otro objeto.

Los gráficos tridimensionales son, actualmente, cada vez más utilizados, tanto para aplicaciones profesionales (animaciones, efectos especiales, creación de imágenes, etc.), como para el entretenimiento en forma de juegos. La mayoría de los títulos lanzados actualmente utilizan toda la gama de gráficos tridimensionales. No es difícil entender los motivos de esta fiebre por las imágenes tridimensionales: los juegos en 3D presentan unos gráficos mucho más realistas, movimientos más rápidos y unos efectos imposibles de conseguir usando los gráficos de dos dimensiones.

Una imagen en tres dimensiones está formada por polígonos, formas geométricas como triángulos, rectángulos, círculos, etc. Una imagen en 3D está formada por miles de estos polígonos. Cuántos más polígonos, mayor es el nivel de detalle de la imagen. Cada polígono tiene su posición en la imagen, un tamaño y un color específicos. Para conseguir una imagen más real también se aplican texturas sobre los polígonos. El uso de texturas no se encuentra limitado sólo a superficies planas. Es perfectamente posible moldear una textura sobre una esfera por ejemplo.

El proceso de creación de una imagen tridimensional se encuentra dividido en tres etapas, llamadas de dibujo, geometría y renderización. En la primera etapa, se crea una descripción de los objetos que componen la imagen, es decir: qué polígonos forman parte de la imagen, cuál es su forma y su tamaño, cuál es la posición en la imagen de cada polígono, cuáles serán los colores usados y, por último, qué texturas y efectos 3D se aplicarán. Después se entra en la fase de geometría, donde se crea la imagen y se almacena en la memoria.

Al final de la etapa de geometría ya tenemos la imagen lista. Sin embargo, tenemos un problema: el monitor del equipo, así como otros medios planos (TV, papel, etc.) sólo son capaces de mostrar imágenes bidimensionales. Entramos entonces en la etapa de renderización. Esta última etapa consiste en transformar la imagen 3D en una imagen bidimensional, que es la que será mostrada en el monitor. Esta etapa es más complicada de lo que parece, ya que es necesario determinar (desde el punto de vista del espectador o usuario) qué polígonos son visibles, aplicar los efectos de iluminación adecuados, etc.

A pesar de que el procesador también es capaz de crear imágenes tridimensionales, él solo no es capaz de generar imágenes de calidad a grandes velocidades (como las demandadas por los juegos) pues estas imágenes exigen un número abundante de cálculos y procesamientos. Para empeorar más la situación, el procesador también tiene que ejecutar, simultáneamente, varias tareas más relacionadas con el programa en sí.

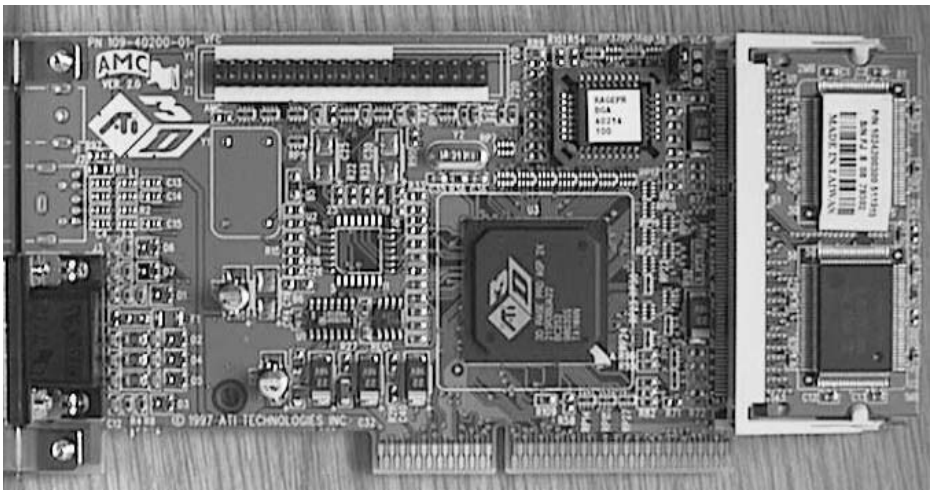
Las tarjetas aceleradoras 3D poseen procesadores dedicados, cuya función es únicamente la de procesar las imágenes, lo que pueden hacer a una velocidad increíble, dejando libre el procesador principal para ejecutar otras tareas. Con ellas, es posible construir imágenes tridimensionales con una velocidad suficiente para crear juegos complejos. Es necesario recordar que una tarjeta gráfica 3D sólo mejora la imagen en las aplicaciones que hacen uso de imágenes tridimensionales. En aplicaciones 2D, sus recursos especiales no se usan.

La conclusión es que si sólo pretendemos trabajar con aplicaciones de despacho, *Internet*, etc. no existe ninguna necesidad de gastar dinero en una tarjeta 3D, ya que no usaríamos su potencial. En este caso, podemos usar sin problemas el vídeo *onboard* de la placa base o una tarjeta gráfica un poco más antigua. Sin embargo, si el equipo va a utilizarse para juegos es fundamental poseer una tarjeta gráfica 3D. Sin una tarjeta 3D, la mayoría de los juegos actuales se volverían lentos incluso en un equipo con un procesador de última generación. Actualmente, todas las tarjetas gráficas que encontramos a la venta, incluso los modelos más simples, poseen recursos 3D, pero existen variaciones en términos de precio y rendimiento.

8.1.1 La memoria

Como el procesador, la tarjeta gráfica también usa memoria *RAM*, memoria que sirve para almacenar las imágenes que se están creando.

En una tarjeta gráfica 2D, la cantidad de memoria no interfiere absolutamente nada en el rendimiento de la tarjeta, sólo determina las resoluciones y la cantidad de colores que serán soportados. Por ejemplo, una tarjeta antigua con sólo 1 MB de memoria, será capaz de mostrar 16 millones de colores (24 bits) con una resolución de 640 x 480 o 65.000 colores (16 bits) a 800 x 600. Una tarjeta con 2 MB ya sería capaz de exhibir 16 millones de colores con una resolución de 800 x 600. Una tarjeta con 4 MB sería capaz de alcanzar los 16 millones de colores con una reolución de 1280 x 1024.



Una tarjeta gráfica que utiliza módulo SO-DIMM de memoria gráfica

Para tener una buena definición de colores, el mínimo necesario es el uso de 16 bits de memoria y lo ideal son 24 bits. Algunas tarjetas también soportan el uso 32 bits de memoria, pero tratándose de imágenes 2D los 32 bits corresponden exactamente a la misma cantidad de colores que con los 24 bits, es decir, 16 millones. Sencillamente, los 8 bits adicionales no se usan.

Ya que estamos tratando este tema, otra configuración importantísima es la tasa de refresco o actualización del monitor. Generalmente, esta opción aparece en el cuadro de diálogo de propiedades del monitor (*Panel de control/Pantalla/Configuración/Opciones avanzadas/Monitor*).

8.1.2 La frecuencia de actualización del monitor

La frecuencia de actualización se refiere al número de veces por segundo que la imagen es actualizada en el monitor. Lo ideal es usar una frecuencia de actualización de aproximadamente 75 Hz, o más. Usando una tasa menor tendremos un fenómeno llamado *flicker*, donde la pantalla queda bastante inestable.

Una cosa que ayuda mucho a disminuir el *flicker* es bajar el brillo del monitor. Lo ideal es usar la pantalla lo más oscura posible, siempre que sea confortable naturalmente. Una indicación es dejar el control de brillo casi al mínimo y ajustar sólo el con-

traste. Cuanto mayor es la tasa de actualización y menor sea la claridad de la imagen, menor será el *flicker* y menor será el cansancio de los ojos.

Las tasas de actualización máximas dependen de la tarjeta gráfica y del monitor. Si elegimos una frecuencia que no sea soportada por el monitor, la imagen aparecerá desenfocada. Cuánto más baja sea la resolución elegida, mayor será la tasa de actualización soportada por el monitor. Las mismas tarjetas gráficas también pueden limitar la resolución máxima soportada. Una tarjeta antigua, una *Trident 9680* por ejemplo, no conseguía trabajar con más de 70 Hz de refresco a 1024 x 768, el vídeo *onboard* que equipa las placas base con el *chipset* i815 es capaz de trabajar con una resolución de 1024 x 768 con una tasa de refresco de 85 Hz, pero de sólo 70 Hz a 1152 x 864.

Pero, cuando hablamos de imágenes en 3D la cosa cambia bastante de panorama. Primero porque al procesar una imagen 3D la tarjeta no usa la memoria de vídeo sólo para almacenar la imagen, sino que sobre todo, la usa para almacenar las texturas que se usan. En los juegos actuales cada vez se usan más texturas cada vez mayores. Es justamente por esto que las tarjetas gráficas actuales son tan poderosas. Para hacernos una idea, en la época del *386* una buena tarjeta gráfica venía con un procesador simple, con 20 o 30 mil transistores, y 256 KB de memoria. La *Voodoo 6*, un monstruo que no acabó saliendo al mercado, a pesar de tener un poder de proceso formidable, traía cuatro procesadores trabajando en paralelo con casi 15 millones de transistores cada uno y 128 MB de memoria.

Volviendo al tema principal, en una tarjeta gráfica 3D la cantidad de memoria no determina la resolución que podrá usarse, pero sí el rendimiento de la tarjeta. El motivo es simple, si las texturas usadas no cupiesen en la memoria de la tarjeta, tendrán que ser almacenadas en la memoria *RAM* y leídas usando el bus *AGP*. En este caso, tenemos el problema de una enorme pérdida de rendimiento, pues se pierde más tiempo al leer una textura almacenada en la memoria *RAM* del equipo que al leerla si estuviese almacenada en la memoria de la tarjeta gráfica, que es mucho más rápida.

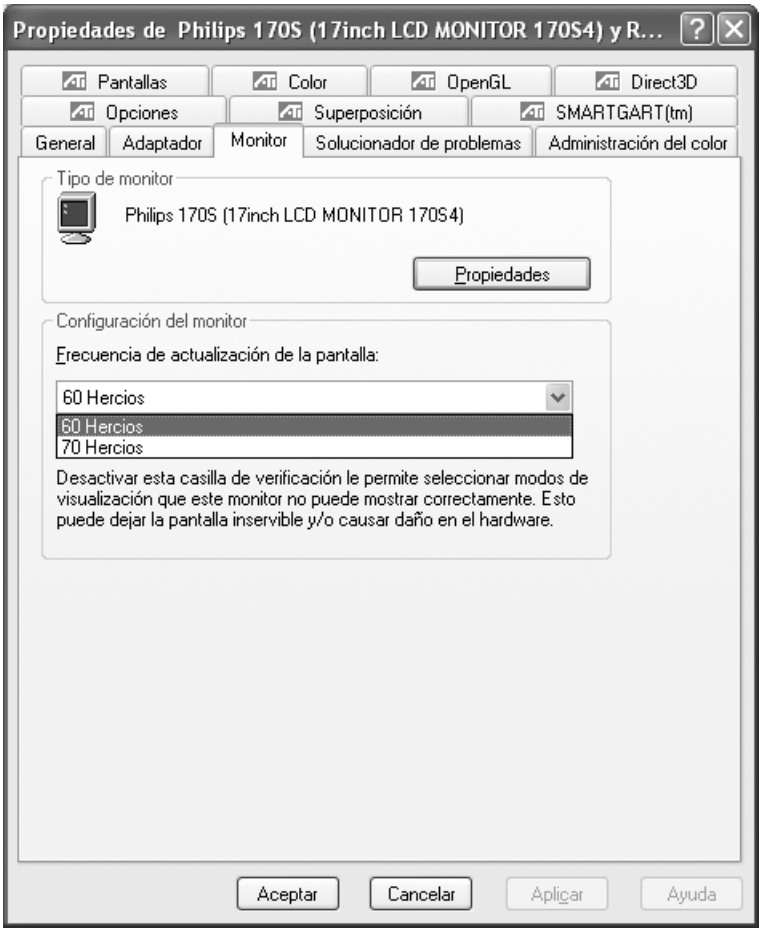
8.1.3 Las ventajas

En los juegos, las dos principales diferencias entre una tarjeta 3D lenta y otra rápida son la calidad de la imagen, que incluye la resolución de pantalla, el número de colores y los efectos 3D que se usarán, y el *frame-rate*, el número de cuadros generados por segundo.

La función de la tarjeta gráfica 3D es básicamente dibujar las imágenes y mostrarlas en el monitor. Cuánto más poderosa es la tarjeta, más polígonos será capaz de dibujar y más texturas será capaz de aplicar en el mismo periodo de tiempo. En un juego hay que renderizar la imagen de cada cuadro. Cuánto más potente es la placa, más cuadros es capaz de generar. Cuántos más cuadros es capaz de generar por segundo, más perfecto es el movimiento de la imagen. Para que no sea posible percibir cualquier error en la fluidez de la imagen, lo ideal serían 30 cuadros por segundo como mínimo. Para poder tener una idea, una TV exhibe 24 cuadros y en los dibujos animados se

varía entre los 16 y 24 cuadros. Es por ello que los 30 cuadros son el valor considerado ideal en el mundo de los juegos. Con menos empezarían a aparecer saltos, sobre todo en las escenas más cargadas.

Cuanto mayor es la resolución usada, mayor el número de colores y mayor el número de efectos, mayor es el trabajo de la tarjeta gráfica para generar cada cuadro y, consecuentemente, más bajo será el *frame-rate*. Existe una relación inversamente proporcional entre las dos cosas. La resolución de las imágenes 3D puede elegirse en el propio juego, en el menú de las opciones de las imágenes. En el cuadro de diálogo de las propiedades de pantalla de *Windows* podemos configurar algunas opciones más de la tarjeta gráfica.



El cuadro de diálogo de las propiedades de pantalla con sus fichas

Usando una tarjeta gráfica moderna podremos jugar a nuestros juegos favoritos con la mejor calidad de imagen posible, usando una resolución de 1024 x 768, 32 bits de memoria, etc.

8.2 Los recursos de las tarjetas 3D

Además de dibujar los polígonos y aplicar texturas y colores sobre ellos, lastarjetas gráficas 3D son capaces de generar varios otros efectos, todos vitales para generar las imágenes de buena calidad. Algunos de estos efectos se pueden realizar vía software, aplicados por el procesador principal por la falta de una tarjeta aceleradora 3D, pero aun así con una velocidad mucho más baja.

Parece que no existan límites para la imaginación de los desarrolladores de la industria 3D. Cada día se desarrollan nuevos efectos y se perfeccionan los ya existentes, buscando desarrollar tarjetas y juegos con imágenes más perfectas.

8.2.1 Efectos básicos

Todas las tarjetas gráficas 3D actuales, incluso las más simples, y la mayoría de los *chipsets* de vídeo *onboard* actuales son capaces de trabajar con todos estos recursos. En algunos juegos existe la posibilidad de inhabilitar algunos de estos recursos para mejorar el *frame-rate*, sacrificando un poco de la calidad de imagen.

Gourad Shading

El recurso *Gourad Shading* intenta corregir un problema generado por la luz. A partir de un único color, se realiza una especie de degradado que va de un vértice a otro de cada polígono, consiguiendo una coloración de la imagen más perfecta. Este efecto es usado sobre todo para simular superficies plásticas o metálicas.

Algunas de las primeras tarjetas gráficas 3D no soportaban este recurso, usando en su lugar otro más simple, llamado *Flat Shading*. Este recurso exigía que cada polígono tuviese un único color sólido, resultando una imagen de baja calidad, donde eran visibles los contornos de los polígonos.

Clipping

En una imagen tridimensional, un objeto puede ocupar cualquier lugar en el espacio, inclusive quedar por delante o detrás de otro objeto. Cuando se realiza la renderización de la imagen, es decir, la conversión hacia 2D para mostrarlo en el monitor, es necesario determinar los objetos que serán visibles (desde el punto de vista del observador) y cuáles deben ocultarse. Este recurso también es llamado *Hidden Surface Removal*.

Z-Sorting

Este recurso es opcional. Tiene la misma función que el recurso *Clipping*, es decir, eliminar las partes encubiertas de la imagen a la hora de hacer la conversión para 2D y enviar la imagen al monitor. La diferencia radica en cómo realizan esta tarea los dos procesos.

Usando el recurso *Clipping*, primero se determinan los polígonos visibles y después sólo se renderizan los que deben mostrarse. Con eso, la tarjeta ahorra procesamiento, ya que se deben renderizar menos objetos en cada cuadro.

El recurso *Z-Sorting* realiza la misma tarea, pero usa un método menos sutil: renderiza todos los polígonos (visibles o no), empezando por los que están más alejados del punto de vista del observador. Conforme la imagen se va renderizando, los objetos que están más al frente van cubriendo los que están detrás.

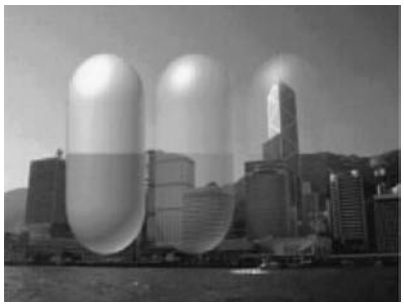
El resultado final es idéntico, sin embargo tenemos un menor uso del procesador, pues es dispensado de determinar las partes visibles de la imagen. Por otro lado, la tarjeta gráfica es más exprimida, pues tiene que renderizar más objetos.

Lighting

Para conseguir una imagen perfecta, también es necesario determinar la intensidad luminosa, es decir, la visibilidad de cada objeto basándose en la distancia y en el ángulo del foco de luz. Uno de los grandes trucos en una imagen 3D es la posibilidad del programador de determinar las fuentes de luz y su intensidad dentro de la imagen: luces, sol, fuego, etc. Al procesarse la imagen, queda a cargo de la tarjeta 3D aplicar el recurso de *Lighting*, calculando los efectos de los focos de luz determinados por el programador.

Transparencia (Transparency)

Muchos objetos, como el agua o el vidrio, son transparentes. Este recurso, también llamado *Alpha Blending*, permite la representación de estos objetos en una imagen 3D, posibilitando ver el fondo de un lago o a través de una puerta de vidrio. El grado de transparencia de cada objeto es definido en un canal de 8 bits, permitiendo 256 niveles diferentes.



Este recurso de transparencia consume mucho procesamiento, pues es necesario mezclar los colores de dos objetos, o dos texturas. Naturalmente, las tarjetas 3D con más poder de procesamiento son las que sacan un mejor partido al aplicar este recurso.

Texture Mapping

Esta es una de las funciones 3D más simple y que en la mayoría de los casos acaba siendo aplicada por el propio procesador. Sobre los polígonos que componen la imagen se aplican las texturas, que tienen determinada su posición exacta en la imagen. Este recurso consiste en estirar las texturas que están más próximas del punto de vista del observador y encoger las más lejanas, manteniendo inalterada la posición de cada textura en la imagen.

Texture Filtering

El recurso *Texture Mapping* deja mucho a desear. Las texturas que muy están próximas aparecen sencillamente como enormes cuadrados, convirtiendo la imagen en poco real. Para contornear este problema, los juegos más actuales usan el efecto *texture filtering* (filtrado de texturas).

Este recurso consiste en interpolar los puntos de las texturas que están más próximas, disminuyendo la distorsión. Al interpolar una imagen, la tarjeta aumenta su resolución, añadiéndolo más puntos de los que ya existen. Debemos notar que este recurso sólo sirve para evitar la granulación de la imagen.

Existen dos tipos diferentes de *texture filtering*, llamados *bilinear filtering* (filtrado bilineal) y *trilinear filtering* (filtrado trilineal). La diferencia es que el filtrado bilineal realiza un cálculo simple, basado en la textura que está siendo exhibida, mientras que en el filtrado trilineal se usa un recurso especial llamado *mip mapping*, que consiste en almacenar varias versiones de diferentes tamaños de la misma textura en la memoria, lo que permite realizar el filtrado a partir de la textura que más se acerca al tamaño de la imagen que debe ser mostrada.

El *trilinear filtering* genera efectos con una calidad un poco mejor y consume menos procesamiento, aunque consume más memoria de vídeo (ya que se almacenan varias texturas). Prácticamente todas las tarjetas gráficas 3D soportan el *bilinear filtering*, pero sólo las más recientes soportan el *trilinear filtering*.

128 x 128

64 x 64

32 x 32

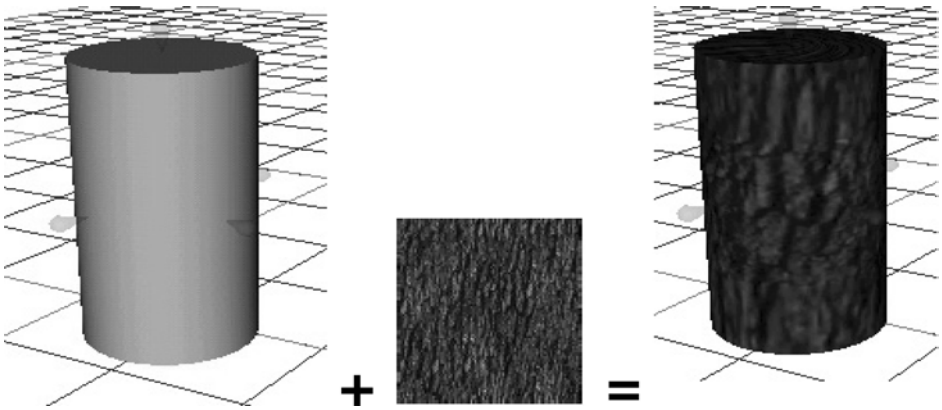
Fogging

Si miramos una montaña que está lejos, percibiremos que parece cubierta por una especie de neblina que surge debido a la humedad del aire, que distorsiona la imagen, volviéndola poco nítida.

El efecto *Fogging* está destinado a proporcionar este mismo recurso a las imágenes 3D. Esto ayuda de dos maneras: primero, aumentando el realismo de la imagen y segundo, disminuyendo el procesamiento necesario, pues las imágenes que estarán cubiertas por la neblina virtual podrán mostrarse a una resolución menor.

Corrección de perspectiva (Perspective Correction)

Este es un recurso muy importante y que encontramos en cualquier tarjeta 3D. A través de él, las texturas son moldeadas sobre los polígonos respetando nuestro ángulo de visión. Este efecto recuerda un poco el efecto del recurso *texture mapping*, pero es mucho más avanzado, pues permite que las texturas sean moldeadas en los objetos de formas irregulares. Este recurso permite conseguir imágenes extremadamente reales, pero es uno de los que demandan más poder de procesamiento.



Z-Buffer

En una imagen tridimensional, además de las informaciones relativas a la anchura y altura (X e Y), también tenemos las informaciones relativas a las profundidades (Z). Estas informaciones son guardadas en un área reservada de la memoria de vídeo y se destinan a determinar, con precisión, la posición de cada polígono en la imagen.

8.2.2 Recursos avanzados

Además de los recursos básicos, muchas de las aceleradoras actuales poseen otros recursos capaces de mejorar todavía más la calidad de las imágenes. Estos nuevos recursos permiten una pequeña mejora en la calidad final de las imágenes pero, en cambio, consumen muchos recursos de procesamiento. Muchos usuarios con equipos len-

tos prefieren inhabilitar estos recursos para mejorar el *frame-rate*, el número de cuadros generados por segundo.

Phone Shading

Este recurso es una evolución del *Gourad Shading*. La función es la misma, permitir aplicar efectos de luz sobre un polígono, simulando superficies plásticas o metálicas.

Los efectos generados usando el recurso *Gourad Shading* generan imágenes bastante buenas, pero no perfectas. En muchos casos, las imágenes parecen sintéticas. Esto pasa dado que el efecto de luz generado por el *Gourad Shading* es bastante simple: sencillamente se determinan la intensidad máxima y mínima de luz dentro del polígono y se realiza un degradado a continuación.

El *Phone Shading*, a su vez, utiliza un algoritmo mucho más complejo, que calcula la intensidad de luz punto por punto, basándose en la posición individual de cada punto en relación al punto de luz. El resultado es bastante mejor, pero necesita mucho más procesamiento.

32 bits de memoria

Los primeros juegos 3D sólo soportaban el uso de 256 colores, lo que limitaba mucho la calidad de la imagen. Cuando los juegos pasaron a utilizar 65 mil colores (16 bits), se percibió un enorme salto en la calidad de las imágenes; por último, los desarrolladores tenían los suficientes colores para aplicar efectos de luz y sombra convincentes y construir texturas más detalladas.

Como todo evoluciona de forma muy rápida, la mayoría de las aceleradoras 3D actuales son capaces de generar imágenes 3D usando 32 bits de memoria. Con más colores disponibles es posible generar transiciones de luz más suaves, mejorando la calidad de la imagen.

La polémica surge justamente por cuánto aumenta la calidad de la imagen. En una imagen estática, grande y que usa pocas tonalidades de memoria, es fácil percibir la diferencia entre el uso de 16 o 32 bits de memoria, pero en un partido de movimientos rápidos, la diferencia de calidad no es tan perceptible. Además, sólo existe una diferencia realmente perceptible en las transiciones de luz. En las texturas pequeñas no tenemos la posibilidad de notar diferencia alguna.

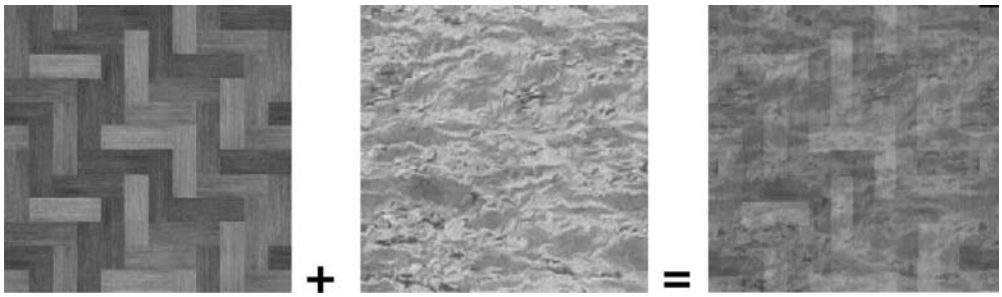
Usando 32 bits de memoria el rendimiento de la tarjeta gráfica siempre es un poco menor que usando sólo 16 bits, ya que la cantidad de datos que se deben procesar será mucho mayor y el bus con la memoria es más transitado. Este recurso puede ser inhabilitado a través de la configuración de la tarjeta gráfica o, en algunos casos, a través de las distintas opciones del propio juego.

Las tarjetas que tienen una pérdida mayor de rendimiento al usar 32 bits de memoria son las que poseen un bus de datos más estrecho con la memoria *RAM*, ya sea porque

usan memorias *SDR* en vez de memorias *DDR*, es decir por acceder a la memoria con sólo 64 bits en vez de 128. Usando 32 bits de memoria, la tarjeta necesita transferir el doble de datos para la memoria de cada cuadro. En la práctica, es como si el bus de datos con la memoria fuese reducido a la mitad.

Single Pass Multitexturing

Un recurso muy utilizado en los juegos actuales es la combinación de dos texturas sobre un mismo objeto. Este recurso es bastante útil al otorgar una mayor libertad a los programadores y disminuir el número total de texturas que van a ser almacenadas, ahorrando memoria de vídeo.



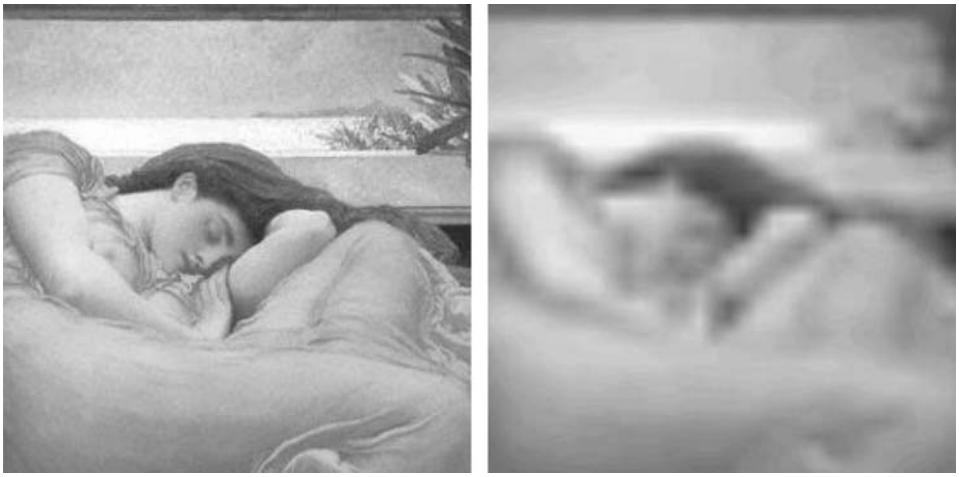
Las aceleradoras compatibles con el recurso de *single pass multitexturing* son capaces de aplicar las dos texturas a la vez, tardando el mismo tiempo que tardarían para aplicar una textura simple. Para conseguirlo, estas tarjetas disponen de dos procesadores de texturas, que trabajan de forma simultánea. Como las texturas son aplicadas línea a línea, para los dos procesadores es fácil mantenerse sincronizados, pues justo después de terminar el primero de aplicar la primera línea de la primera textura y pasar a la segunda línea, el segundo puede empezar a trabajar aplicando la primera línea de la segunda textura (sobre la primera) y así en adelante.

Evidentemente, el uso de dos procesadores de texturas aumenta de forma considerable el coste de producción de la tarjeta, haciendo que algunos *chipsets* y tarjetas más baratas vengan con sólo un procesador de texturas. La ausencia de este recurso convierte la tarjeta en mucho más lenta, sobre todo en los juegos más actuales, que usan intensamente el recurso de superposición de texturas.

Texturas de 2048 x 2048

Otro recurso que no es soportado por todas las tarjetas 3D actuales es el uso de grandes texturas, de hasta 2048 x 2048 píxeles. El uso de estas texturas permite mejorar un poco la calidad visual de algunos juegos, a pesar de disminuir un poco el rendimiento y consumir más espacio en la memoria de vídeo. Algunos de los juegos más actuales utilizan varias texturas grandes, presentando una mejora perceptible en la calidad visual en conjunto con una tarjeta que soporte este recurso.

Uno de los pocos *chipsets* actuales que no soporta este recurso, estando limitadas las texturas de como máximo 256 x 256 píxeles es el *Voodoo 3*. En el caso de que el juego utilice texturas grandes, sencillamente simplificará las texturas hasta que alcancen los 256 x 256 permitidos, sacrificando la calidad.



El uso de texturas más detalladas es capaz de mejorar la visión de los juegos, el problema es su tamaño. Una única textura de 2048 x 2048 y 32 bits de memoria ocupa nada menos que 16 MB de memoria. Esto equivale a la toda la memoria de vídeo de una *Voodoo 3* por ejemplo, mientras que una textura de 256 x 256 con 16 bits de memoria sólo ocupa 128 KB de memoria de vídeo. Eso sin tener en cuenta la cantidad de procesamiento y el ancho de banda necesario para transportar y procesar una textura de este tamaño. Este es uno de los recursos que mejoran la calidad sacrificando mucho el rendimiento. Los desarrolladores de juegos utilizan este recurso con mucha moderación.

Las tarjetas *GeForce*, entre otras tarjetas gráficas actuales, incorporan algoritmos de compresión que permiten compactar las texturas a razón de 8 a 1 y poseen un poder de procesamiento muy superior al de las tarjetas anteriores. Estas sí que son capaces de presentar un rendimiento razonable, incluso procesando pesadas texturas.

FSAA

Este recurso mejora la calidad de las imágenes generadas. Consiste en mejorar el contorno de los objetos a partir de puntos de colores intermedios, un tipo de interpolación, pero realizada en tiempo real por la tarjeta gráfica, que aplica este efecto durante el proceso de renderización de las imágenes.

FSAA viene de *Full Screen Anti-Aliasing*, que destaca la capacidad de la tarjeta gráfica de suavizar los contornos (*anti-aliasing*). Utilizando el recurso de *Anti-Aliasing*, los fallos en el contorno de las imágenes son suavizados, disminuyendo la granulación de

las imágenes. En la práctica, la impresión es que la imagen posee una resolución más elevada que la real. Una imagen de 640 x 480 píxeles a la que se le aplica el recurso de *Anti-Aliasing* pasa a tener una calidad semejante, o incluso superior, a la de una imagen de 800 x 600 píxeles, pero, en realidad, sólo tenemos una transición más suave entre los colores.

El *FSAA* es un recurso soportado sólo por las tarjetas 3D de mayor gama, pues a pesar de la mejora en la calidad de los gráficos resulta en una gran pérdida de rendimiento, ya que para conseguir el efecto de interpolación, la tarjeta 3D necesitará generar una imagen con el doble o el cuádruple de resolución y luego disminuirla a fin de aplicar el recurso.

Este recurso puede ser activado o desactivado a través de la ventana de propiedades de vídeo. Es conveniente examinar las configuraciones, pues en muchas tarjetas, el *FSAA* viene activado por defecto y, es por ello, que mucha gente se queja de los bajos frames por segundo en comparación con otros usuarios de tarjetas semejantes que tenían el recurso desactivado.

V-Sinc

Este es un recurso más interesante, soportado por prácticamente todas las tarjetas 3D. Al ser activado, el *V-Sinc* sincroniza los cuadros generados por la tarjeta con la frecuencia de actualización del monitor. La tasa de actualización del monitor, o *refresh-rate*, puede ser configurada en las propiedades de vídeo, generalmente con opciones entre los 48 y 85 Hz. Este es justamente el número de veces que la imagen será actualizada por segundo en el monitor. Para obtener una imagen libre de *flicker*, es recomendable el uso de 75 Hz de tasa de actualización, como mínimo.

El *V-Sinc* sirve justamente para sincronizar los cuadros de imagen generados por la tarjeta gráfica con la actualización de la imagen del monitor. En cada dos actualizaciones de la imagen se genera un nuevo cuadro. En términos de fluidez de la imagen, esta es la medida ideal, pues de nada sirve que la tarjeta genere más cuadros que los que es capaz de exhibir el monitor. Es recomendable mantener el *V-Sinc* activado, y sólo desactivarlo cuando vayamos a ejecutar algún programa *benchmark*.

8.3 Conceptos generales

Para llegar a entender mejor los distintos recursos de las tarjetas 3D y *chipsets* gráficos disponibles, es muy interesante presentar algunos conceptos generales.

8.3.1 La división de las tareas

Como ya hemos visto, una imagen 3D no es más que un enorme conjunto de polígonos. Una de las diferencias entre una imagen vectorial (compuesta por polígonos) y una imagen de mapa de bits (*bitmap*), es que la primera puede ser modificada libre-

mente, asumiendo cualquier tamaño y manteniendo la misma calidad. Por su parte, si estiramos la segunda imagen, los puntos se romperán y acabaremos con un borrón sin forma en las manos.

Esto es así porqué una imagen poligonal, en realidad, sólo es un conjunto de ecuaciones matemáticas que indican la forma, posición y tamaño relativo de los polígonos. Al ampliar la imagen, el procesador volverá a hacer estos cálculos, reconstruyendo la imagen con el tamaño deseado. Sin embargo, ampliando una imagen de mapa de bits sólo podemos aumentar el tamaño de los puntos, resultando en una imagen de baja calidad.

Este mismo concepto se aplica a una imagen 3D, que repetimos, es sólo un conjunto de polígonos. Una aceleradora 3D sirve para auxiliar al procesador en la construcción de las imágenes 3D usadas en los juegos y aplicaciones. Notemos que la tarjeta 3D sólo es una asistente (a pesar de hacer la parte más pesada del trabajo), esto significa que el procesador también tiene que realizar sus tareas.

El procesador es el encargado de montar el armazón de la imagen, es decir, montar la estructura de polígonos que la compone, calculando el tamaño y posición de cada uno de ellos. Esta tarea exige una cantidad gigantesca de cálculos matemáticos, justamente por ello, es esencial que el procesador disponga de un coprocesador aritmético muy poderoso. Al terminar de dibujar el armazón de la imagen, el procesador lo envía a la tarjeta junto con las texturas que deben ser aplicadas sobre los polígonos, informaciones sobre los colores de cada polígono, la posición de cada textura, etc.

La tarjeta gráfica a su vez, tiene como tarea aplicar las texturas, colorear los polígonos, aplicar los efectos 3D, determinar las partes visibles de la imagen y, por último, generar la imagen que aparecerá en el monitor.

En un juego, este proceso es repetido de forma indefinida, generando el movimiento de las imágenes. Cuánto más poderosos son el procesador y la tarjeta gráfica, más imágenes se podrán generar por segundo, resultando un *frame-rate* más alto. El frame-rate es el número de cuadros presentados por segundo. Un *frame-rate* ideal serían por lo menos 30 cuadros por segundo, mientras el mínimo serían alrededor de 20 cuadros. Por debajo de esto, el juego empezaría a presentar saltos que perjudicarían su manejo.

Es oportuno recordar que el *frame-rate* no tiene nada que ver con el *refresh-rate*, que es la cantidad de veces que la imagen almacenada en la memoria de la tarjeta gráfica es leída por el *RAMDAC* y actualizada en el monitor. Podemos tener, a la vez, un *frame-rate* de 30 cuadros por segundo y uno *refresh-rate* de 75 Hz, por ejemplo.

El *frame-rate* viene determinado básicamente por 4 factores:

1. La potencia de la tarjeta gráfica.
2. El nivel de detalle de la imagen (con 16 o 32 bits de memoria).

3. La resolución utilizada (800 x 600 o 1024 x 768).
4. La potencia del procesador.

Como ya hemos dicho anteriormente, antes de que la imagen sea transferida a la tarjeta de vídeo, tiene que ser "esbozada" por el procesador. Entonces, este esbozo es transferido a la tarjeta gráfica que realiza el resto del trabajo. Cuántas más texturas tuviesen que aplicarse, más efectos de luz, etc. más tiempo tardará la tarjeta gráfica para terminar cada imagen.

Otro factor importante es la resolución utilizada. Usando una resolución de 640 x 480 o 1024 x 768, el trabajo del procesador es exactamente el mismo, pues como ya hemos dicho, los polígonos pueden rediseñarse y la imagen ampliada a cualquier tamaño sin que exista pérdida de calidad alguna. Sin embargo, cuanto mayor es la resolución, mayor será el trabajo de la tarjeta gráfica (que tendrá que renderizar más píxeles) y, consecuentemente, más bajo será el *frame-rate*.

Usando una resolución muy alta, la tarjeta gráfica se queda con la parte más pesada del trabajo, haciendo que el procesador tenga que estar esperando a que la tarjeta de vídeo termine las imágenes para poder enviar los siguientes cuadros. En esta situación, cambiar la tarjeta gráfica 3D por otra más rápida aumentaría inmediatamente el *frame-rate*. Usando una resolución más baja, de 640 x 480 o menos, el escenario que acabamos de ver se invierte. Los cálculos realizados por el procesador para cada cuadro continuarán siendo los mismos, sin embargo, el trabajo de la tarjeta gráfica es mucho más ligero, pues las imágenes generadas serán menores. Consecuentemente, la tarjeta gráfica realizará su trabajo más rápidamente y nos encontraremos con el escenario opuesto, con la tarjeta gráfica ociosa esperando a los envíos del procesador.

En términos de calidad de imagen, todas las tarjetas gráficas fabricadas hace menos de un año están muy próximas unas de las otras. Un especialista gráfico puede ser capaz de diferenciar las imágenes con facilidad, pero un usuario normal siquiera notará grandes diferencias en la imagen de una tarjeta a otra, usando la misma resolución. Usando una tarjeta de gama más alta, como una *GeForce 2 GTS* o una *ATI Radeon DDR* por ejemplo, tendremos un *frame-rate* satisfactorio, incluso usando resoluciones mucho más altas y habilitando todos los efectos visuales.

En cualquier caso, necesitaremos tener un procesador que la acompañe razonablemente. Algunas tarjetas 3D son menos dependientes del procesador que otras tarjetas, presentando un buen *frame-rate*, incluso en procesadores más lentos. Este es el caso de las tarjetas equipadas con el *chipset Voodoo* (en todas sus versiones). Otras tarjetas, como las basadas en los *chipsets Riva 128* y *Riva TNT*, sólo muestran todo su potencial en conjunto con los procesadores más fuertes.

8.3.2 Frame-rate y rendimiento

El factor para la medición del rendimiento de las tarjetas gráficas más aceptado actualmente es, justamente, la cantidad de cuadros por segundo que cada tarjeta es capaz

de generar en un determinado juego. Como hemos visto, el trabajo de la tarjeta gráfica es, básicamente, dibujar las imágenes y mostrarlas en el monitor, y el objetivo de tener una tarjeta rápida es justamente poder ejecutar cualquier juego con altas resoluciones, con todos los efectos 3D activados y con un buen *frame-rate*.

La idea de usar los juegos para medir el rendimiento de las tarjetas en lugar de programas de *benchmark* se está convirtiendo cada vez en más popular entre las *Webs* especializadas, pues muestra el rendimiento de la tarjeta gráfica en aplicaciones reales, donde realmente se usarán. Para que tengan validez, los tests *deben* ser realizados con equipos con una configuración exactamente igual, usando la misma resolución de pantalla y el mismo juego, cambiando sólo la tarjeta gráfica usada entre una medición y otra. Después, los resultados son comparados y la tarjeta gráfica que es capaz de generar más cuadros por segundo es la más rápida.

Los fabricantes dicen que el ideal serían unos 60 *fps*, pues así difícilmente el *frame-rate* bajaría por debajo de los 30 fps en las escenas más pesadas de algunos juegos. Personalmente, considero que la cifra media de 30 *fps* es un valor más que aceptable, pues las bajadas momentáneas a 15 o 13 fps no llegan a molestar tanto como para justificar gastar mucho más para tener un tarjeta gráfica de alto rendimiento.

8.3.3 Los drivers

En la actualidad, un punto fundamental cuando hablamos de tarjetas 3D son justamente los *drivers*. Simplificando, un *driver* (controlador) es un pequeño programa que le permite al sistema operativo utilizar todos los recursos de la tarjeta de vídeo. Los fabricantes mantienen los *drivers* de sus tarjetas en constante desarrollo, y en cada nueva versión tenemos una mejora, tanto en el rendimiento como en la compatibilidad. Antes de instalar una tarjeta gráfica 3D, no debemos dejar de visitar la Web del respectivo fabricante y bajar los *drivers* más recientes, que tendrán más recursos y serán más rápidos que los *drivers* que vienen con la tarjeta.

> Nota: navegando por *Internet* también encontraremos drivers beta, *drivers* que todavía están en fase de *test* y que, por ello, todavía no han sido oficialmente liberados por el fabricantes. Como siempre, un *driver beta* permite que tengamos nuevos recursos a mano, pero no totalmente estables.

8.3.4 Las APIs

Así como todos los programas son construidos usando algún lenguaje de programación, como *C++*, *Visual Basic*, *Delphi*, etc. que al programador le permiten construir su programa y acceder a los recursos del sistema. Las aplicaciones 3D, en especial los juegos, son construidos a través de una interfaz de programación, es decir, una *API* (*Application Programming Interface*).

Simplificando, una *API* es, más o menos, como un lenguaje de programación para generar gráficos 3D, compuesta de varios comandos que le permiten al programador

construir las imágenes, aplicar los efectos 3D, etc. Todos los juegos son construidos en base a los recursos permitidos por una de las *APIs* disponibles. Actualmente, sólo se usan estas tres *APIs*: *Direct3D* (o *D3D*), *OpenGL* y *Glide*.

Direct3D: desarrollada por Microsoft, *D3D* es la *API* más utilizada en la actualidad. Esta no es la *API* con más recursos, ni la más rápida, pero es la más fácil de utilizar, motivo principal de su amplia aceptación. Los recursos permitidos por la *D3D* no son nada modestos y permiten crear juegos con gráficos muy interesantes.

Esta *API* se puede utilizar en cualquier tarjeta 3D, sólo es necesario que el fabricante desarrolle el *driver* adecuado. Felizmente, existen drivers *D3D* para prácticamente todas las tarjetas 3D actuales. Esta *API* está en constante desarrollo. Los nuevos recursos de la *D3D* se van incorporando a las tarjetas gráficas ya existentes a través de los nuevos *drivers*, resultando en un aumento de la calidad de imagen y rendimiento.

En realidad, *Direct3D* forma parte del *DirectX* de Microsoft y, por ello, todos los juegos que se ejecutan sobre esta *API* necesitan que *DirectX* esté instalado en el equipo. Con la falta de una aceleradora 3D, los juegos hechos en *D3D* (a menos que el desarrollador determine lo contrario) se pueden ejecutar en modo software, donde es el propio procesador el encargado de hacer todo el trabajo. Evidentemente, ejecutándose en modo software el rendimiento será bastante malo y en bajas resoluciones.

OpenGL: si la *D3D* es la *API* más fácil de utilizar, *OpenGL* es la *API* que posee más recursos. Originalmente, *OpenGL* fue desarrollado para ser utilizado en aplicaciones profesionales y, de hecho, es prácticamente la única *API* 3D utilizada en las aplicaciones como *3D Studio MAX*, programas de ingeniería y aplicaciones profesionales. Los fabricantes de juegos percibieron muy rápidamente que también podían usar esta poderosa herramienta en sus productos. Uno de los primeros títulos fue el *GL Quake*, una versión del *Quake 1* modificada para utilizar efectos los 3D de *OpenGL*, que abrió el camino para el posterior lanzamiento de varios otros títulos.

Uno de los principales problemas de esta *API* es el hecho de ser incompatible con un número considerable de tarjetas gráficas, no por las limitaciones de hardware sino, sencillamente, por la falta de *drivers*. En otros casos, la tarjeta de vídeo es compatible, pero los *drivers* son malos, aprovechando sólo algunos de los recursos de la *API*, resultando en un bajo rendimiento y en una baja calidad de imagen.

Los *drivers OpenGL* son llamados *drivers OpenGL ICD*, o *Installable Client Driver*. El término *ICD* se usa en relación a un *driver* completo y que soporta todos los recursos *OpenGL*. Muchas veces, los mismos fabricantes lanzan *mini-drivers*, llamados *mini-GL* o *mini-ICD*, optimizados para los juegos. Estos *drivers* sólo poseen algunas de las instrucciones *OpenGL*, las utilizadas por los juegos. Al poseer menos instrucciones, es más fácil que los fabricantes mejoren el rendimiento del *driver* e incluyan soporte a las instrucciones *3D-Now!* y *SSE*, mejorando el rendimiento de los *drivers* y, consecuentemente, de la tarjeta gráfica.

Evidentemente, estos *mini-drivers* sólo sirven para juegos; para la ejecución de aplicaciones profesionales necesitaremos tener instalado el *driver ICD* completo.

Glide: es la *API* más antigua y, a la vez, la más simple. Fue desarrollada por la casa 3dfx para ser usada conjuntamente con sus *chipsets Voodoo*. El problema es que la *Glide* siempre fue una *API* propia y, por ello, sólo compatible con las tarjetas equipadas con los *chipsets* de la casa 3dfx. Durante mucho tiempo, esta fue la *API* más utilizada, pues las tarjetas equipadas con el c*hipset Voodoo* eran de lejos las más vendidas.

A medida que se sacaron al mercado tarjetas 3D de otros fabricantes (que sólo eran compatibles con *D3D* y *OpenGL*), poco a poco, los fabricantes de juegos dejaron de usar la *Glide*, para tener una mejor compatibilidad con el mayor número de tarjetas.

8.3.5 AGP: ser o no ser

Desde el punto de vista de una tarjeta gráfica, el bus *AGP* conlleva varias ventajas sobre el bus *PCI*: es hasta 8 veces más rápido (*AGP* 4x), es de uso exclusivo para la tarjeta gráfica (a diferencia del bus *PCI*, donde los 133 MB/s son compartidos por todos los dispositivos *PCI* instalados) y permite que la tarjeta gráfica utilice la memoria *RAM* del equipo para almacenar las texturas sin que exista una pérdida de rendimiento tan grande como pasaba con el bus PCI. Sin embargo, esto no implica que la tarjeta gráfica utilice todos estos recursos.

Muchas tarjetas gráficas sólo utilizan el bus *AGP* por cuestiones de marketing, pues existe la idea generalizada de que las tarjetas *AGP* son mejores y, consecuentemente, una tarjeta gráfica lanzada en su versión *AGP* se vende mucho más que si fuese lanzada en su versión *PCI*. Pero, en el caso que la placa no sea lo suficientemente rápida para utilizar la mayor velocidad de transferencia permitida por el bus *AGP* y no utilice la memoria del sistema para almacenar las texturas, entonces la única ventaja es dejar libre un *slot PCI*. Este es el caso de todas las tarjetas gráficas 2D que utilizan el bus *AGP* y el de muchas aceleradoras 3D.

De modo general, las tarjetas gráficas lanzadas simultáneamente en versiones *PCI* y *AGP* son las que no utilizan todos los recursos permitidos por el bus *AGP* y, por ello, son fácilmente adaptables al bus *PCI*. En estos casos, la diferencia de rendimiento entre la versión *PCI* y la versión *AGP* es casi imperceptible.

Finalmente, los recursos permitidos por el bus *AGP* mejoran el rendimiento de la tarjeta gráfica, pero estos recursos sólo son realmente necesarios para las tarjetas 3D más rápidas o para las tarjetas 3D que poseen poca memoria de vídeo y que por ello dependen de la velocidad del bus *AGP* para almacenar los datos en la memoria *RAM* del sistema. Otra categoría muy dependiente de la velocidad del bus *AGP* son los *chipsets* de vídeo *onboard*, que sólo disponen de la memoria del sistema.

El bus *AGP* será todavía más necesario para la nueva generación de tarjetas y juegos 3D, donde ya empiezan a utilizarse texturas muy grandes, de hasta 2048 x 2048 píxe-

les y demandarán unas transferencias de datos mucho más amplias que las permitidas por el bus *PCI*.

8.3.6 Uso de la memoria

Mientras que en las tarjetas gráficas 2D la memoria de vídeo sólo determina las resoluciones permitidas y el número de colores soportados, en las tarjetas 3D, la cantidad de memoria de vídeo está directamente relacionada con el rendimiento de la tarjeta. Si se utilizan más texturas en un juego, y estás son muy grandes, más memoria deberá tener la tarjeta para poder ejecutar el juego de la forma adecuada. A pesar de que las tarjetas *AGP* puedan utilizar la memoria del sistema para almacenar las texturas, siempre existe una pérdida considerable de rendimiento.

Para las nuevas generaciones de juegos, 32 MB de memoria de vídeo ya empiezan a ser insuficientes. Por ello, ya debemos empezar a considerar la posibilidad de adquirir una tarjeta gráfica con 64 MBytes de memoria de vídeo como mínimo.

8.3.7 Chipsets

Como en el caso de las placas base, el componente principal de una tarjeta gráfica es el *chipset*, en este caso el *chipset* de vídeo. Este *chipset* es el encargado de comandar toda la puesta en funcionamiento de la tarjeta y determina sus recursos y rendimiento. Es muy común que un mismo *chipset* de vídeo sea usado en tarjetas de distintos fabricantes. Al usar el mismo procesador gráfico central, todas estas tarjetas poseen básicamente los mismos recursos y un rendimiento similar (siempre considerando modelos con la misma cantidad de memoria de vídeo).

Normalmente, las únicas diferencias son la cantidad de memoria y la presencia o no de otros accesorios, como la salida de vídeo por ejemplo. En cambio, existen muchas diferencias entre dos tarjetas equipadas con *chipsets* diferentes, pero estas son mínimas entre tarjetas equipadas con el mismo *chipset*.

8.4 Rendimiento

Los factores que determinan el rendimiento de una tarjeta gráfica son muy parecidos a los que determinan el rendimiento de un procesador: la frecuencia de trabajo, el número de operaciones por ciclo, la anchura del bus de acceso a la memoria de vídeo, la cantidad de memoria de vídeo y el bus utilizado (*PCI*, *AGP*, *AGP* 2x, etc.). Teniendo presentes todos estos factores obtenemos la potencia "bruta" de la tarjeta gráfica, ya que el rendimiento efectivo también va a depender de los recursos 3D utilizados y de los *drivers* utilizados.

Por ejemplo, una tarjeta gráfica que ejecute más funciones tendrá un rendimiento inferior, en términos de cuadros por segundo, al de otra semejante pero que ejecute un número menor de funciones 3D. Al contrario, tendrá una calidad de imagen superior.

A continuación, detallaremos de forma más profunda los distintos factores que determinan el rendimiento de las tarjetas gráficas:

Frecuencia de trabajo: cada *chipset* de vídeo tiene su propia frecuencia de trabajo, medida en millones de ciclos por segundo (MHz). Esta frecuencia no tiene nada que ver con la frecuencia de trabajo del procesador, de la placa base, o del mismo bus *PCI* o *AGP* al que está conectado la tarjeta gráfica. Como en el caso de los procesadores, cuantos más ciclos por segundo, mayor es el poder de procesamiento del *chipset* de vídeo.

Píxeles por ciclo de reloj: de la misma manera que los procesadores son capaces de ejecutar más de una instrucción en cada ciclo de reloj, también existen casos de *chipsets* de vídeo capaces de procesar más de un píxel en cada ciclo de reloj. Algunos *chipsets* más antiguos sólo procesaban 1 píxel por ciclo, los *chipsets* más recientes ya procesaban 2 píxeles por ciclo. También existen bastantes casos de *chipsets* que procesan 4 píxeles por ciclo.

Fill rate: multiplicando el número de píxeles procesados en cada ciclo de reloj por el número de ciclos por segundo obtenemos el *fill rate*, que es el número total de píxeles que la tarjeta gráfica puede generar en cada segundo. Este valor se mide en "megapíxeles", o millones de píxeles por segundo. Por ejemplo, en una tarjeta gráfica en la que el *chipset* procesa 2 píxeles por ciclo y trabaja a 150 MHz, tenemos un *fill rate* de 300 megapíxeles.

Otro dato relacionado con el *fill rate* que encontraremos en las especificaciones de las tarjetas es la cantidad de "*texels*", es decir, la cantidad de texturas que la tarjeta es capaz de aplicar. "*Texel*" es un término semejante a píxel, uno de los puntos que forma una imagen, sin embargo, el término píxel se usa para referirse a la imagen mostrada en el monitor, mientras que el término "*texel*" se usa para referirse a los puntos que componen las texturas que se aplicarán en los polígonos.

Poder de procesamiento (polígonos por segundo): otro factor determinante en el rendimiento final de la tarjeta es la cantidad de polígonos que se pueden dibujar por segundo. Cuanto mayor es el número de polígonos que la tarjeta es capaz de generar por segundo, mayor será el rendimiento de la tarjeta. A diferencia del *fill rate*, este recurso no tiene una relación directa con la frecuencia de trabajo.

Bus de la memoria: otro factor importante es la amplitud del bus de comunicación con la memoria. Cuánto más ancho es el bus, más rápidas serán las transferencias de datos entre el *chipset* y la memoria de vídeo y, consecuentemente, mayor será el rendimiento de la tarjeta. En las tarjetas gráficas 3D no se utiliza memoria caché pero, en cambio, la memoria de vídeo trabaja a frecuencias mucho más altas que la memoria principal del sistema.

Resolución: el f*ill rate* es una medida del rendimiento bruto de la tarjeta, la cantidad de operaciones que la placa es capaz de ejecutar por segundo. Como ya hemos visto

anteriormente, cuanto mayor sea la resolución utilizada se necesitará un mayor procesamiento para generar cada imagen.

Efectos 3D utilizados: calidad de la imagen y rendimiento son dos términos incompatibles. Para tener una imagen de mayor calidad se deben utilizar más efectos 3D que consumen preciosos ciclos de procesamiento. Efectos como el uso de 32 bits de memoria, texturas de 2048 x 2048 y *FSAA*, mejoran la calidad de las imágenes pero, en cambio, consumen más procesamiento y disminuyen el *fps*.

9. El sonido

Originalmente, el PC posee una capacidad sonora bastante limitada, pues el procesador sólo puede trabajar con unos y ceros. Todo PC viene con un pequeño altavoz incorporado, que está conectado directamente a la placa base. Los sonidos emitidos por el altavoz son generados por el propio procesador, que sólo puede combinar secuencias de 1 y 0 para generar algunos sonidos rudimentarios, que generalmente se usan para llamar la atención del usuario cuando ocurre algún error.

Por otro lado, en un sonido analógico tenemos una gran onda, que puede asumir un número ilimitado de frecuencias. Los aparatos analógicos, como las grabadoras de cintas *K-7*, se limitan a captar estas ondas y transformarlas en una señal magnética, con tramos más fuertes o más débiles, dependiendo de la intensidad del sonido. Para la reproducción del sonido grabado, cuando se toca la cinta, esta señal magnética se convierte en una señal eléctrica que mueve el cono de un altavoz, reproduciendo así el sonido original; se trata de un esquema relativamente simple.

Sin embargo, un PC no puede trabajar con señales analógicas e intenta representar los sonidos reales en forma de secuencias de unos y ceros. Surgió entonces la idea de convertir la señal analógica a un formato digital a través de muestras. Imaginemos que la onda sonora fuese colocada en un gráfico cartesiano, y cada punto de la onda pasase a recibir los valores de las coordenadas X e Y.

Entonces podríamos coger muestras de la onda y atribuir un valor numérico a cada una, que representaría su posición en el gráfico. Cuanto mayor sea la cantidad de muestras por segundo, mejor será la calidad del sonido. También es importante la amplitud de la señal o sea, la cantidad de valores diferentes que podrá asumir. Si, por ejemplo usásemos 8 bits para representar la amplitud de cada muestra, sería posible reproducir sólo 256 variedades distintas de sonidos. Si se usasen 16 bits, ya serían posibles 65.000 valores distintos.

Podemos hacer una analogía entre un sonido digital y una imagen digitalizada, donde tenemos una cierta cantidad de puntos y una determinada cantidad de colores. Cuanto mayor sea la resolución de la imagen y mayor sea la cantidad de colores, más perfecta será la representación de la imagen. Por ejemplo, una señal telefónica es transportada entre los nodos digitales de la red de telefonía con un formato de 8.000

muestras por segundo y una amplitud de 8 bits, resultando un sonido nítido pero de baja calidad. En una música grabada en un CD nos encontramos con 44.100 muestras por segundo y una amplitud de 16 bits que permite 65.000 valores distintos. En un CD ya tenemos un sonido casi perfecto, hasta el punto que una persona normal no consigue distinguir entre un sonido real y el mismo sonido grabado.

Sin embargo, un músico experto o una persona con una capacidad auditiva más desarrollada sería capaz de percibir una pérdida de calidad en el sonido, ya que estamos usando sólo 65.000 tonalidades, mientras que en un sonido analógico encontramos una cantidad infinita de frecuencias. Un sonido digital nunca será perfecto, pues es imposible grabar un número infinito de muestras. La cuestión es grabar el sonido con una cantidad de frecuencias suficientes para que el oído humano no sea capaz de notar ninguna diferencia entre el sonido digital y el sonido analógico original. En estudios profesionales de grabación ya se trabaja con sonidos de 94 KHz de frecuencia y 24 bits de amplitud, una calidad suficiente para engañar incluso a los oídos más sensibles.

9.1 La generación del sonido digital

El encargado de realizar la conversión de la señal analógica en sonidos digitalizados es un circuito llamado *ADC* (*Analog Digital Converter*, o conversor analógico/digital). Básicamente, un *ADC* se encarga de extraer las muestras de la onda eléctrica generada por un aparato analógico y transformarlas en señales digitales. Estas señales pueden ser manipuladas por el procesador y transmitidas como cualquiera otro tipo de dato.

Cuando es necesario reproducir un sonido digital previamente grabado, necesitamos realizar el camino inverso, o sea, transformar las muestras digitales en una onda analógica, que pueda ser reproducida por los altavoces conectados a la tarjeta de sonido. Esta conversión es realizada por el *DAC* (*Digital Analog Converter*, o conversor digital/analógico). El *ADC* y el *DAC* son los dos componentes más importantes de una tarjeta de sonido.

Grabar los sonidos en formato digital permite reproducir con la calidad suficiente cualquier sonido analógico. El problema es que los archivos generados acaban siendo muy grandes, lo que dificulta su manejo. Sólo un minuto de audio grabado con una calidad de CD equivale a un archivo *WAV* de 10.5 MBytes. Para solucionar este inconveniente se usan dos técnicas: la compresión y la síntesis de audio.

9.1.1 La compresión

Así como podemos comprimir documentos de *Word* o imágenes *BMP*, también podemos también comprimir archivos de audio, eliminando la información redundante. En música, un largo periodo con muestras de sonido del mismo valor podría ser sustituido por un pequeño código que informase que la misma frecuencia debe repetirse

X veces. También podemos eliminar informaciones que ejerzan poca influencia sobre la calidad del sonido, eliminando pequeñas variaciones.

Hasta cierto punto, es posible comprimir el sonido sin ninguna pérdida de calidad (sustituyendo secuencias de sonidos iguales por códigos que representan que el sonido debe ser repetido). Pero llega un punto en que es preciso prescindir de un poco de calidad para generar archivos más pequeños. Ejemplos de algoritmos de compresión de audio son el *ADPCM*, *True Speech* y el *MP3*, un formato de compresión extremadamente eficiente, popular y muy utilizado actualmente para transmitir música vía *Internet*.

El formato *MP3* permite una compresión de archivos *WAV* de casi un 90%, o sea, un archivo de música de 4 minutos que se correspondería con un archivo *WAV* de 42 MB puede ser convertido a un archivo *MP3* de cerca 4 MB, sin ninguna pérdida significativa en la calidad del sonido.

El formato *MP3* consigue estos resultados a través de la eliminación de las frecuencias sonoras que no son captadas por el oído humano, pero que sirven para aumentar el tamaño de los archivos de audio. En la conversión de un archivo *WAV* a *MP3*, la degradación del sonido es muy pequeña, sólo se produce una pequeña distorsión en los sonidos graves, pero que generalmente no se percibe.

Debido a la popularización del formato *MP3*, varios fabricantes lanzaron reproductores portátiles que permiten cargar la música desde el ordenador y oírla en cualquier lugar. Algunos modelos pueden ser conectados al puerto serie, la mayoría al puerto *USB* y otros ofrecen soporte para ambos. El problema de usar el puerto serie para transferir la música es la lentitud, cada archivo de 4 MB tarda casi 5 minutos en ser transferido. Usando un puerto *USB* la misma música se transfiere en pocos segundos.

Otro formato que va adquiriendo bastante popularidad es el *VQF*, que usa un algoritmo de compresión más eficiente que el usado por el *MP3*, generando archivos hasta un 30% más pequeños. El problema del formato *VQF* es que, debido a la complejidad del algoritmo, la descompresión de los archivos es extremadamente dificultosa, exigiendo la intervención de un procesador muy potente.

Otro formato relativamente nuevo es el *WDM* de Microsoft. Además de eliminar las frecuencias que no son audibles, este formato destaca los sonidos que son más perceptibles, recurso llamado "*ringing*". El objetivo principal de este formato es competir con *Real Audio* en las transmisiones de audio vía *Internet*, donde debido al poco ancho de banda, el sonido necesita ser muy comprimido. Usando tasas de compresión tan elevadas no es posible mantener una calidad muy buena, pero según la casa Microsoft los recursos del formato *WDN* permiten un audio con una calidad superior a la del formato de *Real Audio* y otros formatos semejantes.

El formato *WDM* permite varios niveles de compresión y por eso puede ser usado para generar archivos con calidad semejante a la del CD, compitiendo en este caso con

el formato *MP3*. Sin embargo, en una comparación directa, a pesar de que los archivos *WDM* sean cerca de un 35% más pequeños que los equivalentes en *MP3*, tenemos una calidad bastante inferior. En la actualidad, el único formato de audio capaz de generar archivos menores que el formato *MP3* manteniendo la misma calidad es el *VQF*.

9.1.2 La extracción de audio

Un recurso muy utilizado actualmente es la extracción digital de audio, *Digital Audio Extraction* o simplemente *DAE*, que consiste en extraer los datos grabados en un CD de música y grabarlos en el disco duro en forma de archivo *WAV* o *MP3*. En la jerga de los fanáticos de la informática este proceso es conocido por "*ripear*" CDs.

> Nota: una vez en el disco duro, estos archivos de música pueden ser editados con total libertad. Es posible, por ejemplo, extraer pistas de varios CDs y después usar un grabador de CDs para grabar una colección variada. Desde el momento en el que hemos comprado los CDs originales de forma legal, y los CDs grabados sean para uso personal, no existe nada de ilegal en esto.

Cuando se reproduce un CD, el lector mantiene una velocidad de lectura constante y los datos obtenidos son convertidos a una señal analógica que será enviada a los altavoces. Cuando realizamos una extracción de audio, los datos son grabados en forma de archivo, con lo que se mantiene el formato digital. Ahí reside el mayor problema: el sonido es grabado en el CD para ser leído de forma secuencial, en tiempo real, y no tenemos los 304 bytes de códigos *ECC* y el direccionamiento en cada sector que tenemos en un CD de datos, sólo los datos del *subcanal Q* que sólo nos informan de los minutos y segundos transcurridos desde el inicio de la música, con lo que no existe un posicionamiento exacto. Un segundo de audio es una eternidad, pues se corresponde con 75 sectores del CD.

Mantener el sincronismo es una tarea complicada en estas situaciones, ya que la extracción de audio se realiza a altas velocidades, 8x, 16x, o 32x dependiendo de la unidad y debido al uso de la velocidad angular constante la velocidad de lectura varía en función de la parte del disco que se está leyendo. No todas las unidades de *CD-ROM* realizan la extracción de audio, y la velocidad de la extracción también varía mucho de un modelo a otro, sin tener una relación directa con la velocidad de lectura. Algunas unidades son capaces de realizar la extracción a 16x o más, mientras que otras no pasan de 4x o 6x. También podemos usar una grabadora de CDs para realizar la extracción, el único problema es que la grabadora casi siempre será más lenta que el lector.

Teniendo un lector compatible, sólo necesitaremos el software adecuado. Existen excelentes "*rippers*", como *MusicMatch Jukebox*, *Fast CD ripper*, *RealJukebox*, etc. Pero todos estos programas son comerciales. Si queremos "*ripear*" nuestros CDs sin la necesidad de pagar, una buena opción es el programa *CDex*, que tiene todas las funciones

básicas, inclusive el recurso de extraer audio directo para *MP3*, además de realizar la conversión de *WAV* a *MP3* y viceversa, permitiendo también configurar la calidad (y el tamaño) de los *MP3* que se generarán. El programa *CDex* puede encontrarse en la dirección http://www.cdex.n3.net. También podemos encontrar otros programas gratuitos, así como programas comerciales, en la dirección http://www.mp3.com.

Existen algunas funciones presentes en casi todos los "*rippers*" que pueden ser configuradas para obtener una mejor calidad de sonido. Veamos algunas de ellas:

Normalizar Volumen: normalmente, los CDs distintos poseen volúmenes diferentes. Este recurso permite dejar todas las pistas de audio "*ripeadas*" con el mismo volumen, lo que es bastante útil en el momento de realizar colecciones.

Activar corrección Jitter: los códigos *ECC* contenidos en el *subcanal Q* no permiten corregir grandes errores. Un error de lectura que no puede corregirse con el uso de los códigos del *subcanal Q* es llamad *jitter*. Los *jitters* se pueden detectar con facilidad, pues generan unos chasquidos oíbles, algunas veces hasta fuertes en la música "*ripeada*". Para corregir este tipo de errores la única solución es intentar releer el sector varias veces. Habilitando esta opción, la extracción se realizará de forma más lenta, sin embargo, tendremos menos *jitters*.

Número de reintentos: el número de veces que debe releerse un sector cada vez que se produzca un *jitter*. Mientras más alto es el número, más baja será la velocidad de la extracción, pero menor será la cantidad de errores.

Lectura simultánea: esta opción determina el número de sectores que pueden leerse cada vez. Algunas unidades pueden leer, a lo sumo, 26 sectores (en general este es el valor por defecto de esta opción) pero otras unidades pueden leer un número más alto de sectores. En este caso, aumentar el valor de esta opción aumentará la velocidad de la extracción, aunque sin embargo, escoger un valor mayor que el de la capacidad de la unidad causaría errores.

Intercambiar canales izquierdo y derecho: algunas unidades de CDs intercambian los canales de audio durante el proceso de extracción. Los sonidos que deberían salir por el auricular de la izquierda pasan a salir por el de la derecha y viceversa. Activar esta opción corrige este problema.

9.1.3 Generando archivos menores

El formato *MP3* permite grabar una canción de 4 minutos con calidad de CD en poco más de 4 MBytes. Sin embargo, es posible generar archivos menores siempre que disminuyamos un poco la calidad del sonido. En las configuraciones del programa que convierte los archivos, podemos modificar el valor del *Min Bit-rate*, ya que esta opción determina el número de muestras extraídas del sonido, representada en forma de bits por segundo. Cuanto menor sea el *Bit-rate*, menor será la cantidad de muestras extraída, peor será la calidad del sonido, pero obtendremos un archivo menor .

El valor por defecto es de 124 Kbps, que equivale a la calidad de CD. Sin embargo, también podemos escoger frecuencias menores, que van hasta los 8 Kbps dependiendo del programa. Usando un valor de 112 Kbps obtenemos unos archivos *MP3* sin una degradación considerable del sonido y cerca de un 10% menores. Un valor de 64 Kbps equivale a la calidad de una radio FM, usando el valor de 32 Kbps la calidad queda próxima a la del teléfono, el mínimo recomendable para obtener un sonido oíble.

En las configuraciones del programa también podemos elegir la opción de generar archivos mono o estéreo. La diferencia entre los dos modos es que en mono sólo tenemos un único canal de sonido, mientras que en estéreo tenemos dos. El *Bit-rate*, por su parte, es fijo y esto significa que usando el modo mono tendremos todas las muestras usadas en un único canal, mientras que usando el modo estéreo tendremos la mitad de las muestras en cada canal de sonido.

9.1.4 Síntesis de audio

En vez de grabar música en formato digital (lo que aún usando un formato de compresión de audio generaría un archivo razonablemente grande) podemos sintetizarla, usando el sintetizador *MIDI* de la tarjeta de sonido. En este caso, sólo tendremos que grabar la secuencia de notas que debe ser reproducida, generando un archivo pequeño. Mientras 5 minutos de música con calidad de CD ocupan unos 52 MBytes, un archivo *MIDI* de 10 minutos no ocupa más de 100 o 150 KBytes. Existen dos tipos de síntesis de audio: la síntesis por *FM* y la síntesis por *Wave Table* (tabla de ondas).

9.1.5 Síntesis por FM

En 1971, un estudiante de la universidad de Stanford desarrolló una tesis que demostraba que cualquier señal que varía en amplitud puede representarse como una suma de varias frecuencias. Según esta tesis, cualquier sonido es un conjunto de frecuencias armónicas.

Si cualquier sonido se puede dividir en varias ondas, también podemos hacer lo contrario, o sea, sintetizar diferentes sonidos simplemente calculando y sumando las ondas sonoras que lo componen, en base a una tabla con sólo algunas ondas de diferentes frecuencias. Sería más o menos lo mismo que mezclar varios colores de tinta para obtener nuevos colores. A través de esta técnica, es posible producir desde sonidos de instrumentos musicales hasta voces humanas, pasando por casi todo tipo de ruido.

Tomando como base esta tesis, la casa Yamaha creó esta tecnología de generación de sonido a través de frecuencias moduladas, y hasta hoy fabrica la mayoría de los sintetizadores *FM* usados en las tarjetas de sonido y otros equipos.

Algunas aplicaciones, especialmente los juegos, utilizan el sintetizador *FM* (frecuencias moduladas) para generar ruidos de tiros, explosiones, y otros efectos sonoros evitando el uso de sonidos digitalizados que ocupen mucho espacio. Otra aplicación del

sintetizador *FM* es la generación de sonidos de instrumentos musicales. Estos sonidos son usados para componer las bandas sonoras de prácticamente todos los juegos antiguos, ya que actualmente los juegos usan pistas de música grabadas en el propio CD del juego, y que se van reproduciendo conforme se está juegando.

9.1.6 MIDI

El *MIDI* (*Musical Instrument Digital Interface*) es un patrón usado para garantizar que el sonido generado, por diferentes sintetizadores, se corresponda exactamente con las mismas notas de los instrumentos. Usando el formato *MIDI*, el do de un piano será siempre un do, y no un re, independientemente de la tarjeta de sonido instalada en el equipo, permitiendo que la misma música codificada en un archivo *MIDI* sea reproducida con total perfección en cualquier tarjeta de sonido.

Los archivos *MIDI*, por su parte, representan las notas que deben ser reproducidas, así como el orden y la sincronía. También es posible reproducir varias notas al mismo tiempo, siendo posible simular una orquesta entera. El número de voces, o sea, el número de instrumentos que pueden tocarse simultáneamente depende de la tarjeta de sonido. Por ejemplo, una *Sound Blaster AWE 32* es capaz de reproducir hasta 32 notas de forma simultánea, mientras que una *Sound Blaster AWE 64* es capaz de reproducir hasta 64 notas.

Como en el archivo sólo es necesario informar de las notas que ser van a reproducir, los archivos *MIDI* acaban siendo extremadamente pequeños. Como ya hemos visto en el inicio de este capítulo, un archivo de música *MIDI* de 10 minutos difícilmente ocupará más de 100 o 150 KBytes, incluso con varios instrumentos reproduciéndose de forma simultánea.

Otro aspecto interesante del *MIDI* es la total compatibilidad con la mayoría de los teclados musicales y otros instrumentos digitales que pueden conectarse al equipo a través de las salidas *Midi In/Out* de la tarjeta de sonido. En este caso, podríamos usar algún software específico para componer música, tocando las notas directamente en el teclado musical acoplado al equipo.

9.1.7 Síntesis por Wave Table

A pesar de los sintetizadores de *FM*, como el *Yamaha OPL 3* usado en las tarjetas de sonido *Sound Blaster 16*, que son simples y baratos, el sonido instrumental generado deja mucho que desear en términos de calidad. A pesar de que el sonido es parecido al de los instrumentos reales, es fácil percibir que se trata de un sonido sintético.

Para corregir esta deficiencia, además del sintetizador de *FM* se usa un sintetizador *Wave Table* en las tarjetas de sonido más actuales. En lugar de sintetizar el sonido a través de la combinación de varias frecuencias, como en los sintetizadores *FM*, en los sintetizadores *Wave Table* se usan muestras de sonidos generados por instrumentos reales, lo que garantiza una calidad del sonido muy superior.

Inicialmente se contrata a un músico para tocar individualmente cada nota de varios instrumentos. Estas notas son digitalizadas y grabadas en *chips* de memoria *ROM*, generalmente de 2 o 4 MBytes, que son incorporados en la tarjeta de sonido. Por su parte, el sintetizador se limita a leer estas muestras de sonido y a reproducirlas en el orden correcto.

Todas las placas *Sound Blaster AWE 32* y *AWE 64*, así como las tarjetas compatibles, poseen tanto el sintetizador *FM* como el sintetizador *Wave Table*. Dado que los dos sintetizadores pueden usarse de forma simultánea, es posible mezclar los sonidos generados, haciendo que la música de un juego sea sintetizada por *Wave Table*, mientras que los efectos sonoros son sintetizados vía *FM*. Podemos experimentarlo haciendo una pequeña prueba: intentamos escuchar una música en *MIDI* y otra en *WAV* o *MP3* al mismo tiempo y veremos que los sonidos se reproducen de forma simultánea a través de la tarjeta de sonido.

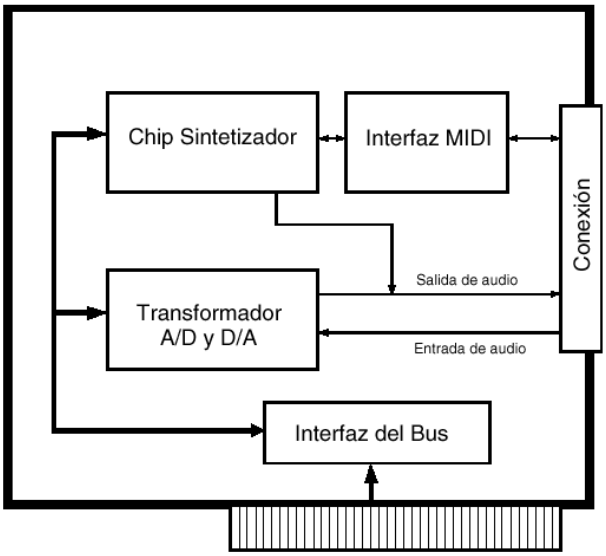


Diagrama interno de una tarjeta de sonido estándar

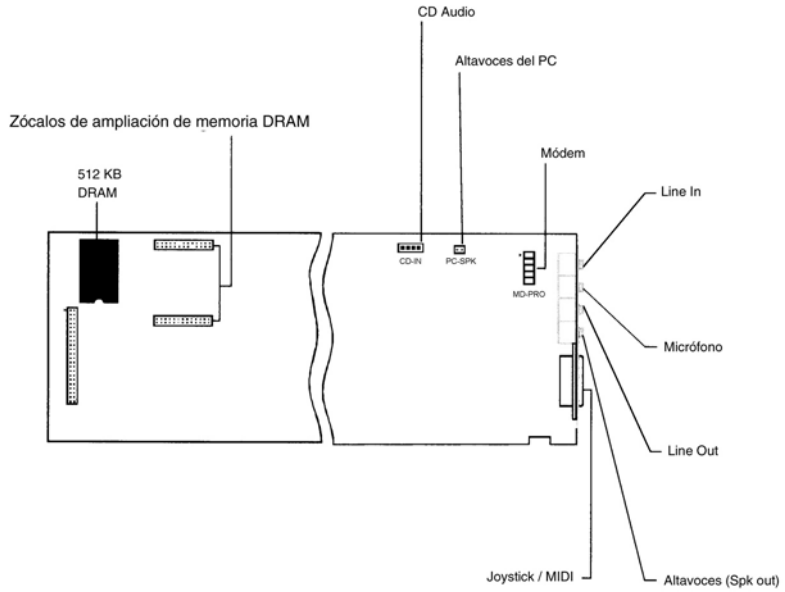
Sólo las tarjetas *AWE 32* y *AWE 64*, así como otras tarjetas recientes, llevan los sintetizadores por *Wave Table*. Las tarjetas de sonido *Sound Blaster 16*, *Sound Blaster Pro* y otras tarjetas de sonido más antiguas sólo llevan el sintetizador por *FM*.

9.1.8 Los conectores externos

En los diversos modelos de tarjetas de sonido encontramos las conexiones en su parte posterior, y además son muy parecidas entre los diferentes modelos:

Line Out y **Speaker Ou**t: cualquier sonido generado por la tarjeta de sonido es enviado a estas dos salidas. La diferencia es que la salida *Line Out* no posee ningún tipo de

amplificación, estando indicada para la conexión de auriculares o altavoces con amplificación, mientras que la señal de la salida *Speaker Out* sí que es amplificada, estando indicada para el uso de altavoces de sonido comunes. La mayoría de las tarjetas de sonido actuales sólo vienen con la salida *Line Out* y, por eso, es necesario el uso de altavoces con amplificación, ya que de lo contrario el sonido queda muy bajo.



La tarjeta de sonido con sus conectores

Line In: podemos conectar varios aparatos sonoros a esta entrada, como un walkman, una televisión o un videocassete con salida *audio-out*, posibilitando la digitalización de cualquier sonido proveniente de estos aparatos.

MIC: esta entrada está destinada a la conexión de un micrófono, que conjuntamente con un programa de grabación, como la *Grabadora de sonidos* de *Windows*, puede usadarse para grabar pistas de voz. La aplicación más interesante, sin embargo, es la posibilidad de conversar por *Internet* usando programas como *Skype* o *Messsenger*.

Conector para el joystick: todas las placas de sonido traen incorporado un puerto de juegos que permite la conexión de un joystick al equipo, conectándolo a la salida de 15 pins de la parte posterior de la tarjeta. Existen tanto puertos de juegos analógicos como digitales. La diferencia entre los dos puertos es que en el puerto digital la cobertura del joystick es realizada por el procesador que controla el puerto, mientras que en el puerto analógico es realizada por el procesador principal. El uso de un puerto de juegos digital disminuye la utilización del procesador principal del equipo mientras se está usando el joystick, haciendo que los juegos se ejecuten hasta un 10% más rápido. Sólo las tarjetas de sonido más nuevas y caras llevan puertos digitales, pero su uso se está convirtiendo en un patrón.

Puerto MIDI: para economizar espacio, los pins desocupados del puerto de juegos son utilizados para transmitir las señales del puerto *MIDI*. Como ya hemos visto anteriormente, podemos conectar teclados musicales al equipo a través de esta salida, pero para ello es necesario un cable especial, que transforma la salida de 15 pins en los conectores *MIDI In/Out* que se acoplarán al teclado musical.

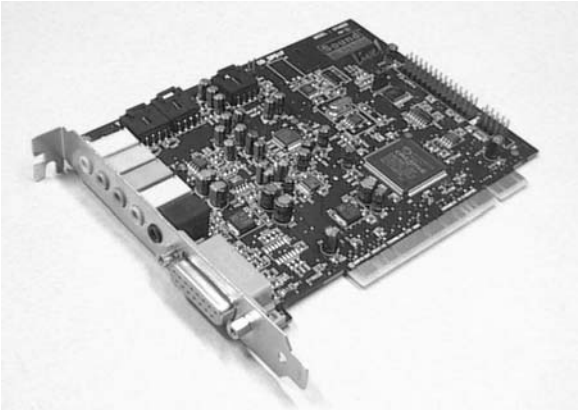
Interfaz para el CD-ROM: las primeras unidades de *CD-ROM* utilizaban interfaces propias y, por eso, eran vendidas junto con una tarjeta de sonido que traía la interfaz adecuada. Más tarde, las unidades pasaron a utilizar las interfaces *IDE* o *SCSI*. El problema era que, en esa época, la mayoría de los equipos aún eran *486* que utilizaban placas *super-IDE*, que sólo traían un puerto *IDE*. En esos equipos sólo era posible conectar 2 periféricos *IDE*. Temiendo que muchos usuarios ya tuvieran las dos plazas ocupadas, los fabricantes de tarjetas de sonido pasaron a incorporar una interfaz *IDE*, aunque algunas tarjetas traían interfaces *SCSI*.

Actualmente, estas interfaces no son utilizadas, pues todas las unidades de *CD-ROM* son *IDE*, y como podemos conectar hasta 4 dispositivos *IDE* al equipo es raro que no tengamos una plaza libre para el *CD-ROM*.

Entrada de audio: para poder oír CDs de música en el equipo, debemos conectar el *CD-ROM* a esta entrada localizada en el interior de la tarjeta de sonido. Sin esta conexión, el CD y la tarjeta de sonido continuarán funcionando normalmente, pero no podremos oír los CDs de música.

9.2 La familia Sound Blaster

Fabricadas por la casa Creative, las tarjetas de sonido *Sound Blaster* rápidamente se convirtieron en un estándar para las tarjetas de sonido. A partir de ahí, prácticamente todas las tarjetas lanzadas al mercado, independientemente del fabricante o del *chipset* que usaban, fueron compatibles con las tarjetas *Sound Blaster*.



Una tarjeta de sonido Sound Blaster

Comparadas con las tarjetas que existían hasta entonces, realmente fue un gran salto en términos de recursos. Las tarjetas de sonido *Sound Blaster* fueron las primeras capaces de reproducir sonidos con calidad CD, por ejemplo. Hasta entonces teníamos varios formatos de tarjetas de sonido diferentes.

Dentro de *Windows*, la tarjeta de sonido es accedida a través del *driver* del dispositivo y, por eso, cualquier juego o programa de *Windows* puede usar cualquier tarjeta de sonido compatible, o no, con el estándar *Sound Blaster*, siempre que existan los *drivers* adecuados y estén correctamente instalados. Sin embargo, en el caso de los antiguos juegos para *MS-DOS* no existían los controladores del dispositivo y la tarjeta era accedida usando rutinas incluidas en el propio juego. Justamente por esto, la mayoría de los juegos sólo eran compatibles con las tarjetas *Sound Blaster*, daba menos trabajo incluir las rutinas para una sola tarjeta.

Desde la década de los años 1990 hasta ahora las cosas han cambiado bastante, las tarjetas de sonido más actuales poseen una capacidad de procesamiento entre 30 y 50 veces mayor que la *Sound Blaster* original, sin embargo, al contrario de lo que pasa con los procesadores y tarjetas gráficas 3D, el aumento de la capacidad de procesamiento no resulta en una diferencia muy apreciable para el usuario.

Cualquier *Sound Blaster 16* con más de 10 años puede reproducir archivos *WAV* o *MP3* con calidad CD, reproducir archivos *MIDI* con un mínimo de calidad y ofrecer entradas para un micrófono y joystick. O sea, cualquier tarjeta de sonido que podamos encontrar posee los recursos básicos y necesarios para funcionar correctamente en la actualidad.

El principal recurso aportado por las tarjetas de sonido modernas, como la *Sound Blaster Live*, *Monster Sound*, *Turtle Beach Montego*, etc. es la capacidad de generar sonido en tres dimensiones. La tarjeta utiliza cálculos especiales para que los sonidos generados por los altavoces lleguen a nuestros oídos como si viniesen de varios puntos diferentes: de los lados, de encima e incluso de detrás.

Como venimos haciendo a lo largo del libro, empezaremos explicando las tarjetas más antiguas hasta llegar a las más actuales, explicando la evolución de los recursos, comenzando con las antiguas *Sound Blaster*.

Sound Blaster y **Sound Blaster Pro**: estas dos placas eran de 8 bits y, por eso, el sonido que generaban no tenía una calidad tan buena. La diferencia entre las dos es que la *Sound Blaster* original era mono y la *Sound Blaster Pro* era estéreo.

Ambas tarjetas sólo poseían un sintetizador *FM* y, por eso, la calidad de los sonidos tampoco es de las mejores. Estas tarjetas fueron vendidas principalmente en kits multimedia de la casa Creative y, por eso, traían una interfaz para *CD-ROM* usada por las unidades Panasonic/Matsushita 1x que equipaban esos kits. En aquella época, los *CD-ROM* aún utilizaban interfaces propias y no existían muchos fabricantes.

Ambas fueron inmediatamente sustituidas por la *Sound Blaster 16*, más avanzada en recursos. Por eso, no llegaron a venderse unas cantidades muy grandes. Actualmente, es casi imposible encontrar una de estas tarjetas.

Sound Blaster 16: el 16 en el nombre indica justamente que esta tarjeta era capaz de producir sonidos con 16 bits de resolución y un muestreo de 44 KHz, la misma calidad de sonido que los CDs de audio. Este ya fue un gran avance sobre las versiones anteriores, que sólo eran capaces de producir sonidos de 8 bits con un muestreo de 22 KHz, una calidad de sonido un poco mejor que la del teléfono. En términos *MIDI*, continuábamos con un sólo sintetizador *FM*, por lo que no era adecuada para la edición musical, dado que la calidad del trabajo era muy baja.

Existieron algunas variaciones de la *Sound Blaster 16*. Las primeras versiones traían interfaces propias para unidades de *CD-ROM* Panasonic, Sony o Mitsumi, una interfaz *SCSI* o, incluso, para un puerto *IDE*, y no eran *Plug & Play*. En estas tarjetas de sonido encontrábamos un zócalo negro vacío, destinado a la conexión de un *chip* procesador de señales, que mejoraba una poco la calidad sonora de la tarjeta y permitía algunos efectos nuevos. El *chip* no hacía milagros y, como era caro, poca gente se interesó en comprarlo.

Las series más nuevas ya sólo traían una interfaz *IDE* y eran *Plug & Play*. Como casi nadie usaba el procesador de señales, la casa Creative resolvió eliminar el zócalo para su conexión, a fin de disminuir los costes de producción de la tarjeta.

Sound Blaster AWE 32: la *Sound Blaster 16* ya había alcanzado una calidad de reproducción sonora equivalente a la de un CD, aunque sin embargo, su talón de Aquiles era la calidad en la reproducción de sonidos *MIDI*, muy baja debido al sintetizador *FM*. La *Sound Blaster AWE 32* continúo utilizando el mismo procesador de señales de la *Sound Blaster 16*, con lo que la en reproducción de músicas *WAV* o *MP3* la calidad era la misma que en las dos tarjetas anteriores.

Sin embargo, además del sintetizador *FM*, la *AWE 32* traía un sintetizador de sonido por *Wave Table*, capaz de reproducir hasta 32 instrumentos de forma simultánea. Esto significó que, finalmente, se podían reproducir archivos *MIDI* de buena calidad y, consecuentemente, que se trataba de una tarjeta de sonido útil para la composición musical.

El *AWE 32* se refiere justamente al sintetizador *MIDI*, ya que *AWE* significa *Advanced Wave Effects*, indicando que la tarjeta es capaz de reproducir archivos *MIDI* por *Wave Table*, y el 32 se refiere al número de instrumentos simultáneos, en este caso 32. Así como la *Sound Blaster 16*, esta también era una tarjeta de 16 bits y no podía ser diferente ya que, igualmente, utilizaba el bus *ISA*.

Ya que se trataba de una tarjeta pensada para quien le gustaba la composición musical, también disponía de dos soquetes para la conexión de memoria adicional. Eran soportados 2 módulos de 30 vías comunes de 1 MB cada uno. Esta memoria adicional

servía para almacenar nuevos instrumentos y efectos especiales creados con la ayuda de un programa de edición musical, aunque también era posible incluir sonidos provenientes de un teclado *MIDI*.

Esta tarjeta de sonido fue lanzada poco tiempo después de la *Sound Blaster 16*. Las primeras versiones traían varias interfaces propias para *CD-ROM* y no eran *Plug & Play*, aunque las más comunes eran las que traían una interfaz *IDE* y sí que eran compatibles con la tecnología *Plug & Play*.

Sound Blaster AWE 64: esta tarjeta no traía grandes avances sobre la *AWE 32* que acabamos de ver. En vez de traer los soquetes para la conexión de módulos de memoria *SIMM*, la *AWE 64* traía 512 KB de memoria *onboard* para almacenar nuevos instrumentos. Nativamente, esta tarjeta sólo era capaz de reproducir 32 instrumentos de forma simultánea (como la *AWE 32*). Sin embargo, junto con los controladores de la tarjeta era posible instalar un programa de polifonía, capaz de crear 32 instrumentos simultáneos más vía software, alcanzando los 64 anunciados en el nombre.

Para el segmento profesional fue lanzada una segunda versión llamada *AWE 64 Gold*. Esta tarjeta venía con 4 MB de memoria para instrumentos y traía conectores *RCA*, los conectores usados por la mayoría de aparatos de sonido. El *Gold* venía justamente del hecho de que los conectores externos estaban bañados en oro, lo que mejoraba un poco el contacto.

9.3 Las tarjetas de sonido PCI

Si teníamos tarjetas gráficas *PCI* y tarjetas *SCSI PCI*, por qué no teníamos también tarjetas de sonido *PCI*? La primera respuesta que nos viene a la cabeza es que por la lentitud del periférico, el bus *ISA* ya era más que suficiente. Hasta cierto punto, este argumento es verdadero, ya que realmente, las primeras tarjetas de sonido no tenían mucho poder de procesamiento y, consecuentemente, no necesitaban usar un bus de datos muy ancho.

Sin embargo, existen varias razones más fuertes para que las tarjetas de sonido actuales sólo sean producidas en su versión *PCI*: la primera es que el bus ISA cada vez era más raro en las placas base y la tendencia general llegó a que dejase de formar parte de las placas base nuevas a finales del año 2002.

La segunda es que el bus *PCI* permite unas transferencias de datos con una utilización del procesador mucho menor que las mismas transferencias realizadas a través del bus *ISA*. Esto significa que una tarjeta de sonido *PCI* carga mucho menos al procesador durante la reproducción de los sonidos, ayudando a mejorar el rendimiento del equipo en general.

Finalmente, la tercera razón es que las tarjetas de sonido actuales poseen un poder de procesamiento incomparablemente superior al de las tarjetas de inicios de la década de los años 90, necesitando mucho más ancho de banda que los 16 MB/s permitidos

por el bus *ISA*. Por ejemplo, una *Sound Blaster Live* poseía un poder de procesamiento estimado de 1 Gigaflop, más de 30 veces el poder de procesamiento de una *Sound Blaster 16*.

9.4 Tarjetas de sonido 3D

Si podemos tener tarjetas gráficas 3D capaces de realizar las imágenes de los juegos y aplicaciones 3D más reales, por qué no podemos tener tarjetas de sonido 3D? Los sonidos del mundo real vienen de todos los lados. El sonido en tres dimensiones da una nueva perspectiva a los juegos, haciéndolos más reales.

La primera empresa en desarrollar una *API* de sonido tridimensional fue la casa Aureal, con su *Aureal 3D*, o simplemente *A3D*. Las primeras tarjetas de sonido compatibles con esta *API*, como la *Monster Sound*, fueron lanzadas al mercado a inicios del año 1997. La *A3D 1.0* permitía simular 3 ejes: delante y detrás, derecha e izquierda y arriba y abajo, aplicando unos filtros especiales para que el sonido realmente pareciese venir de todas las direcciones posibles. Estos filtros eran capaces de distorsionar sutilmente las ondas, consiguiendo engañar a nuestros oídos y haciéndonos pensar que venían de diferentes direcciones. Estos filtros consumían una enorme cantidad de procesamiento y su utilización fue el principal motivo de que los *chipsets* de sonido actuales sean tan poderosos. La ventaja es que cómo todo es procesado en la propia tarjeta de sonido, casi no existe una utilización del procesador principal. En la mayoría de los casos, sustituir una tarjeta de sonido *ISA* antigua por una tarjeta de sonido 3D mejorará el rendimiento general del equipo.

Normalmente, las tarjetas de sonido 3D pueden trabajar tanto con un par de altavoces como con auriculares de oído o sistemas de cuatro altavoces. Una de las mayores dificultades para conseguir aplicar los efectos 3D es mantener un posicionamiento exacto del espectador en relación a las fuentes de sonido. Usando tanto un par de altavoces como los conjuntos de 4 altavoces es necesario posicionar los altavoces de manera que formen un ángulo de aproximadamente 45 grados con nuestros oídos, y estén más o menos a la misma altura de los mismos.

Las tarjetas de sonido que soportan 4 altavoces poseen dos salidas *line-out*. Deberemos acoplar dos altavoces a cada salida, totalizando los 4 altavoces. Para oír perfectamente el audio 3D, usar auriculares es la mejor opción. En segundo lugar viene el uso de sólo dos altavoces. El uso de 4 altavoces sólo debe ser considerado si realmente vamos a tener la paciencia suficiente para buscar el posicionamiento más adecuado de los mismos.

La *A3D 1.0* usada en la *Monster Sound Original* y en otras tarjetas más antiguas, o más simples, quedó sobrepasada con el lanzamiento del *Aureal 2.0* que aportó varios recursos 3D nuevos, resultando una simulación muy real. La *A3D 2.0* sólo fue soportada por las tarjetas más modernas, como la *Monster Sound MX300*.

Estas *APIs* eran implementadas vía hardware y, por eso, para soportar los recursos de una nueva *API* era necesario un nuevo chipset de sonido.

El cuadro de diálogo para la configuración de los 4 altavoces

En el mundo real, el ambiente y los obstáculos naturales causan una distorsión en el sonido. Sin embargo, usando la *A3D 2.0* no existe este tipo de consideración, sólo se calcula el origen y la distancia del sonido. Además de permitir simular la dirección, la *A3D 2.0* posee varios filtros más que permiten simular ecos, sonidos distorsionados por el agua o cualquiera otro obstáculo e, incluso, determinar la intensidad del eco basándose en el material que compone la sala. Esto significa, por ejemplo, un eco más fuerte en una sala con paredes de piedra que en otra con moqueta. Entre los filtros permitidos por la *A3D 2.0* encontramos:

Reverb: determina la intensidad del eco basándose en la distancia de los objetos o el tamaño de la sala. El eco será diferente en una sala grande o en una sala menor.

Reflections: el eco será diferente de acuerdo con el material que rebate el sonido, será más intenso en una sala con paredes de piedra que en una sala con moqueta.

Occlusions: en el caso que entre el espectador y la fuente de sonido exista algún obstáculo, se calculará la trayectoria del sonido siendo rebatido entre varias paredes hasta llegar al espectador (más bajo o distorsionado como en la realidad). Si no existe ninguna apertura, se calculará la distorsión que el sonido sufrirá al atravesar la pared.

Estos efectos se consiguen usando un recurso llamado *Wavetracing*, o trayecto de onda, que consiste en analizar la geometría del escenario 3D para determinar como se van a comportar las ondas sonoras. Sin embargo, estos efectos no son automáticos y es necesario que los desarrolladores los utilicen en sus juegos. Los primeros juegos que utilizaron los recursos de la *A3D 2.0* fueron *Half-Life*, *Quake 3*, *Motorhead* y *Recoil*. La mayoría de los juegos posteriores ya incluyeron soporte para ellos. Sin embargo, hasta hace poco, la mayoría de los títulos del mercado sólo ofrecían soporte a la *A3D 1.0*.

Sin embargo, los efectos avanzados utilizados por la *A3D 2.0* consumen un cierto poder de procesamiento del procesador principal, que bajo las órdenes del software es el encargado de analizar la posición de los obstáculos, generando las informaciones que permiten a la tarjeta de sonido generar los efectos de forma correcta. La mayoría de los juegos con soporte a la *A3D 2.0* ofrecen la opción de deshabilitar este recurso, con lo que podemos aumentar un poco el *frame-rate* de los juegos más pesados.

La *A3D* es la *API* utilizada en la mayoría de las tarjetas anteriores, pero no es la única; otra es la *EAX*, o *Enviromental Audio Extensions*, utilizada por las tarjetas de sonido que utilizaban el *chipset EMU10K1*, como la *Sound Blaster Live*. Así como la *A3D*, la *EAX* también tenía sus versiones 1.0 y 2.0. A pesar de que los efectos sonoros eran muy parecidos, la manera de crearlos es muy diferente. En la *EAX*, los efectos son aplicados por el programador del juego. Es el programador el encargado de determinar los efectos qué se usarán y en qué zonas, etc.

Por su parte, la *A3D* no depende tanto del trabajo del programador, ya que los efectos son calculados en base a la geometría de las escenas y, justamente por eso, tenemos una mayor utilización del procesador. Mientras se diseñan los *frames*, el procesador realiza los cálculos sonoros, en base a la posición de los objetos en el escenario 3D.

En la práctica, los sonidos generados por la *A3D* son más reales y, sin embargo, más imprevisibles. Es más difícil percibir la localización del enemigo en *Quake 3* usando la *A3D* que usando la *EAX*, justamente porque en la *EAX* los efectos son más previsibles. Debido a eso, muchos jugadores prefieren la *EAX*, diciendo que así tienen un mejor dominio del juego, siendo capaces de detectar las posiciones de los enemigos con más facilidad. Otros jugadores prefieren la *A3D*, argumentando que los efectos son más reales.

Un lanzamiento posterior fue la *EAX 3.0*, desarrollada por la casa Creative. Esta nueva *API* utilizaba los recursos del *EMU10K1* de manera diferente a la *EAX 2.0*, permitiendo algunas mejoras sobre los efectos generados por la *EAX 2.0*:

Dynamic morphing between environments: permite calcular la forma en la que el sonido será rebatido por los objetos en movimiento.

Localized reflection clusters: permite definir partes de las superficies que rebatirán el sonido con más intensidad.

Individual reflections for distant echoes: calcula el sonido rebatido por los objetos situados a distintas distancias del emisor. El resultado es que en algunas situaciones el eco es oído varias veces, como en la vida real.

Statistical reverberation model: mejora el cálculo de la distancia entre los objetos, permitiendo que los efectos sean mucho más precisos.

Finalmente, tenemos la *Direct Sound 3D*, implementada a través de *DirectX*. Comparado con la *A3D* y la *EAX*, la *Direct Sound* posee efectos limitados, pero suficientes para generar sonidos convincentes. Esta *API* es soportada por la mayoría de las tarjetas de sonido *PCI* más simples o por las tarjetas genéricas sin marca, que no tienen el suficiente poder de procesamiento para soportar las *APIs* más avanzadas.

9.4.1 Otros recursos

Además de los recursos 3D, las tarjetas de sonido modernas utilizan algo más de poder de procesamiento para ofrecer algunos recursos más interesantes. Vamos a verlos a continuación:

Interpolación de sonido: de la misma forma que podemos interpolar una imagen para aumentar su resolución y mejorar un poco su calidad, también es posible interpolar sonidos. Esto significa aumentar la cantidad de muestras por segundo, incluyendo frecuencias intermedias entre las reales. Con más muestras por segundo, la calidad del sonido mejora de forma perceptible, principalmente en músicas en archivos *WAV* o *MP3* con una baja tasa de muestreo.

Las tarjetas basadas en el *Aureal Vortex 2*, como la *Monster Sound MX300* y la *Turtle Beach Montego 2*, trabajaban con interpolación de 26 puntos, mientras que las tarjetas basadas en el *EMU10K1*, como la *Sound Blaster Live*, usaban un algoritmo más simple de sólo 8 puntos. Por los números, la diferencia parece ser muy grande pero, en la práctica, la diferencia es imperceptible.

Este recurso se aplica automáticamente vía hardware a través del *chipset* de sonido mientras se van reproduciendo los sonidos y, por eso, no causa ninguna pérdida de rendimiento y no depende de *drivers* ni de ningún soporte especial para funcionar.

Reproducción simultánea: otro recurso interesante es la capacidad de reproducir varios sonidos, o músicas, al mismo tiempo. Las primeras tarjetas de sonido, como la *Sound Blaster 16*, sólo eran capaces de reproducir un flujo de audio a la vez, o sea, mientras se oía un *MP3* o cualquier otro sonido digital, los otros sonidos digitales, como los sonidos de *Windows*, etc. no eran reproducidos. Como máximo, era posible reproducir un archivo *WAV* y un *MIDI* al mismo tiempo.

Las tarjetas más modernas son capaces de reproducir decenas de flujos de sonido digital al mismo tiempo. Esto significa que podemos oír los avisos de *Windows* mientras estamos oyendo un *MP3* o, incluso, oír varios archivos de música al mismo tiempo. Los sonidos son procesados de forma simultánea por la tarjeta, y reproducidos al

mismo tiempo por los altavoces. La calidad final es semejante a la de dos aparatos de sonido conectados al mismo tiempo, sin ninguna distorsión. La tarjeta *Monster Sound MX300*, por ejemplo, era capaz de reproducir hasta 24 flujos de sonido.

Windows 2000/XP es capaz de tratar simultáneamente varios flujos de audio. Esto permite que incluso las tarjetas de sonido *ISA* más antiguas sean capaces de reproducir varios sonidos al mismo tiempo, siempre que se usen los *drivers* adecuados. Sin embargo, en este caso el procesamiento no es realizado por la tarjeta de sonido, y sí por el procesador principal, haciendo que el equipo se vuelva más lento al reproducir varios flujos de audio de forma simultánea. Al usar una tarjeta *PCI* la utilización del procesador es mínima, incluso reproduciendo muchos flujos de audio al mismo tiempo, ya que la mayoría del trabajo es realizado por la propia tarjeta de sonido.

Decodificación vía hardware: reproducir un simple *MP3* puede consumir buena parte de los recursos del procesador. Para disminuir este efecto, algunas tarjetas incorporan el recurso de decodificación de músicas *MP3* vía hardware, disminuyendo la carga del procesador mientras se reproduce la música.

Puerto de juegos digital: en las tarjetas de sonido más antiguas, el procesador era el responsable de leer los movimientos del joystick, haciendo que varias veces tuviese que parar lo que estuviese haciendo para realizar esta tarea. La gran mayoría de las tarjetas *PCI* incluyeron un puerto de juegos digital, que se encarga de leer los movimientos del joystick dejando libre al procesador.

9.4.2 Monster Sound

Así como la casa Diamond fue una de las primeras a entrar en el mercado de las tarjetas gráficas 3D con su tarjeta *Monster 3D*, también fue una de las pioneras en el ramo de las tarjetas de sonido 3D, usando el mismo nombre que utilizó en sus primeras tarjetas gráficas, la *Monster Sound*.

Existen nada menos que 5 versiones de la *Monster Sound*: en orden cronológico tenemos la *Monster Sound* original, la *Monster MX200*, *MX80*, *MX300* y *MX400*. Las tres primeras tarjetas poseían los mismos recursos básicos y estaban basadas en un *chipset* producido por la propia casa Diamond, que estaba basado en la arquitectura del *chipset Aureal Vortex*. La *Monster MX300* ya formó parte de la segunda generación de tarjetas de sonido 3D, estando basada en el *chipset Aureal Vortex 2*, que ya soportaba el recurso de *Wavetracing* y la *A3D 2.0*.

Para infortunio de la casa Diamond, la casa Aureal decidió producir sus propias tarjetas de sonido basadas en sus *chipsets*, con lo que dejó de venderlos a terceros. Como ya no podía contar con los *chips* de la casa Aureal, acabó optando por utilizar un *chip* alternativo en su *Monster Sound MX 400*, el *ESS Canyon 3DTM*. Este *chip* poseía unos recursos muy diferentes a los soportados por el *Aureal Vortex 2* y existió una gran polémica en torno a si era superior o inferior al *Vortex 2*.

MX200

Poco tiempo después de lanzar la *Monster Sound* original, la casa Dimond lanzó una segunda versión basada en el mismo *chipset* pero con algunas mejoras, abandonando así la producción de la *Monster Sound* original.

La *MX200* era compatible con la *A3D 1.0* y con *Direct Sound 3D*. Veamos que al tratarse de una tarjeta más antigua, no teníamos soporte a la *A3D 2.0* ni a los recursos 3D más avanzados. Otros recursos eran el soporte al uso de conjuntos de 4 altavoces, un sintetizador con capacidad para reproducir hasta 64 instrumentos *MIDI* de forma simultánea vía hardware (al contrario de la *Sound Blaster AWE 64*, donde teníamos 32 vía hardware y 32 vía software) y capacidad para reproducir hasta 23 sonidos *WAV* de forma simultánea.

Como toda tarjeta *PCI* teníamos algunos problemas de compatibilidad con los juegos antiguos, pero junto con los *drivers* de la tarjeta era posible instalar un emulador que hacía que la tarjeta fuese reconocida como una *Sound Blaster Pro* para las aplicaciones de *MS-DOS*. Esto funcionaba con la mayoría de los juegos, pero no con todos. La *MX200* traía un recurso curioso para asegurar la compatibilidad con todos los juegos antiguos, simplemente mantener instalada simultáneamente una tarjeta de sonido *ISA* y conectarlas usando un cable que acompañaba a la placa, llamado *Monster cable*. En *Windows* se usaba la *MX200* de forma normal, pero a la hora de ejecutar algún juego antiguo, pasaba el poder a la tarjeta de sonido *ISA*.

MX80

Los recursos eran idénticos a los de la *MX200*, sin embargo, el sintetizador *MIDI* sólo poseía 32 instrumentos simultáneos y fue retirado el soporte a los conjuntos de 4 altavoces: en la *MX80* sólo era posible usar 2 altavoces o auriculares. En compensación, el precio fue reducido de forma drástica.

MX300

La principal innovación de la *MX300* fue el uso del chipset Vortex 2 de la casa Aureal y, consecuentemente, el soporte a la *A3D 2.0,* que resultaba en unos efectos 3D muy superiores a los obtenidos en las tarjetas *MX200* y *MX80*. Continuábamos con los 64 instrumentos *MIDI* vía hardware, pero la *MX300* incluía un software especial que permitía hasta 256 instrumentos vía software. El número de archivos *WAV* simultáneos pasó de los 23 a los 26 y la tarjeta soportaba el uso de 4 altavoces.

Una de las únicas deficiencias de la *MX300*, con sus antecesoras, era la ausencia de una salida de sonido digital. Es posible suplir esta deficiencia comprando una segunda tarjeta llamada *MX25*, que se conecta al conector de 34 pins de la *MX300* e incluye una salida de sonido digital y el soporte a los conjuntos de 6 altavoces *Douby Digital*, que a pesar de ser bastante caros pueden generar efectos 3D muy interesantes (en el caso de estar bien situadas) y proporcionan una calidad de sonido semejante a la de

los cines. Usando un conjunto de 6 altavoces y una pantalla conectada a la salida de vídeo de la tarjeta de vídeo tendremos nada menos que un *home theater*. Para la instalación, basta conectar la *MX25* a la tarjeta principal y ensamblarla a la torre. Después de inicializar *Windows*, basta instalar el software que acompaña a la tarjeta de sonido para activar su funcionamiento.

MX400

Como ya hemos visto anteriormente, la casa Aureal abandonó la venta de *chipsets* de audio para concentrarse en la fabricación y venta de sus propias tarjetas de sonido. Eso dejó a la casa Diamond teniendo que salir a buscar otro *chipset* de audio para poder equipar su próxima generación de tarjetas de sonido. El escogido acabó siendo el *chipset Canyon 3D* de la casa ESS.

Este *chipset* poseía una arquitectura muy diferente a la del *Vortex 2* usado en la tarjeta *MX300*, empezando por las mismas *APIs* soportadas. Ambos *chipsets* soportaban el *Direct Sound 3D* y la *A3D 1.0*, pero las semejanzas se acababan ahí: mientras el *Vortex 2* soportaba la *A3D 2.0* conjuntamente con sus recursos de *Wavetracing*, el *Canyon 3D* no soportaba el uso de la *A3D 2.0*, soportando en compensación la *AX 1.0* y la *EAX 2.0*, las mismas *APIs* soportadas por el *EMU10K1*, el *chipset* que equipaba la *Sound Blaster Live*.

Otros recursos como el número de instrumentos *MIDI*, número de archivos *WAV* simultáneos, soporte para 4 altavoces, etc. eran muy parecidos a los de la tarjeta *MX300*. El *chipset Canyon* traía algunas ventajas sobre el Vortex 2 en términos de recursos, pero, en compensación, consumía un poco más de procesador durante la reproducción de los sonidos.

La *MX400* también incluía una salida de sonido digital y soporte a los conjuntos de 6 altavoces, los mismos recursos que se añadían a la *MX300* a través de la *MX25*. Para no perder el filón de las tarjetas de expansión, la casa Diamond también lanzó la *Río Upgrade*, una tarjeta muy parecida a la *MX25* y que también se conectaba a la tarjeta principal a través de un cable. Esta pequeña tarjeta hacía la codificación *MP3* vía hardware, esto significa transformar archivos *WAV* en *MP3*, o *ripear* CDs de música a *MP3* en 1/5 del tiempo, y con mucha menos utilización del procesador.

9.4.3 Creative Sound Blaster Live

La empresa de las tarjetas *Sound Blaster* no podía quedarse fuera del ramo de las tarjetas de sonido 3D. Heredando el nombre de sus antecesoras, la *Sound Blaster Live* estaba equipada con el chipset *EMU10K1* y soportaba las *APIs EXA 1.0* y *EAX 2.0* y era capaz de reproducir hasta 256 instrumentos *MIDI* vía hardware .

La *Sound Blaster Live* fue vendida en 4 versiones diferentes: Value, *MP3+*, *X-Gamer* y *Platinium*. A pesar de los distintos nombres, las tarjetas poseían las mismas características básicas, lo que cambiaba sólo era la cantidad de salidas, los programas que

venían juntos con la tarjeta y, claro, el precio. Todas poseían dos salidas de sonido y, consecuentemente, soportaban el uso de 4 altavoces. Todas las versiones, también soportaban el uso de los conjuntos *Douby Digital* de 6 altavoces.

La Live Value es la versión más simple, ya que sólo incluye las salidas y entradas de sonido analógicas y no lleva muchos programas adicionales. Tanto la *MP3+* como la *X-Gamer* llevan un puerto que permite la conexión de la unidad *Live drive* (comprada por separado). La unidad *Live drive* es un accesorio que lleva entradas y salidas digitales, además de extensiones de entradas y salidas analógicas para altavoces, micrófono, etc., además de controles de volumen. La posición de la unidad *Live drive*, instalada en la parte frontal de la torre, permite que los conectores sean mucho más accesibles. También están disponibles los conectores *MIDI*: usando la unidad *Live drive* no es necesario comprar el cable *MIDI* por separado.

La *Sound Blaster Live Platinium* es igual a las dos anteriores, aunque ya viene con la unidad *Live drive* y lleva un mayor conjunto mayor de programas, principalmente para la edición musical.

9.4.4 Turtle Beach Montego

La casa Turtle Beach fue una de las pioneras en el ramo de tarjetas de sonido para el mercado del PC. Muchos recursos, como el audio de 16 bits y el *MIDI* por *Wave table* fueron utilizados por esta compañía en sus tarjetas de sonido. El problema es que el mercado de la Turtle Beach siempre fue el ramo profesional y, por eso, sus tarjetas nunca tuvieron un precio accesible para el gran público y, consecuentemente, nunca fueron muy conocidas.

Sin embargo, con la evolución de las tarjetas de sonido, las tarjetas más baratas pasaron a incorporar los recursos que antes sólo encontrábamos en las tarjetas del ramo profesional, nivelando cada vez más las dos líneas de mercado. Para sobrevivir, la casa Turtle Beach no tuvo otra salida que entrar en el ramo de las tarjetas de sonido domésticas, justamente con la *Turtle Beach Montego*. Existen dos versiones y ambas están basadas en los *chipsets* de la casa Aureal. La *Montego* original está basada en el *chipset Vortex 1*, mientras que la *Montego 2* está basada en el *Aureal Vortex 2*, el *chipset* que equipaba la *Monster Sound MX300*.

Al estar basada en el *chipset Aureal Vortex*, la *Montego* incorpora todos los recursos de la A3D 1.0 y, además de eso, disponemos de 64 instrumentos *MIDI* simultáneos y un puerto de juegos digital. La tasa de *signal to noise*, o señal para ruido, también es muy buena, de 92 dB (cuánto mayor es el valor más puro es el sonido), lo que garantiza una calidad excepcional en las grabaciones hechas a partir de la entrada *line-in* de la tarjeta, siendo especialmente recomendable para grabaciones de sonido en general, pero en especial para grabar música de discos de vinilo o cintas *K7* a CD.

La *Montego 2*, por su parte, ya estaba basada en el *chipset Aureal Vortex 2*, incorporando los recursos de *Wavetracing* aportados por la *A3D 2.0*. Continuábamos con 64

instrumentos *MIDI* simultáneos, pero con la posibilidad de utilizar hasta 320 instrumentos más vía software. El puerto de juegos digital se mantuvo y la tasa de *signal to noise* es de 97 dB, una mejora significativa.

También se lanzó una segunda versión de la *Turtle Beach Montego 2*, llamada *Home Studio*. Dedicada principalmente al segmento profesional, nos encontramos con una *Montego 2* equipada con salidas digitales, tanto coaxiales como ópticas y un segundo sintetizador *MIDI*, que combinado con el primero, permitió 128 instrumentos vía hardware y 256 instrumentos más vía software. También tenemos 4 MB de memoria *RAM* para almacenar instrumentos *MIDI* adicionales.

La salida digital óptica se usa, principalmente, para grabadores de *mini disks*. Estos aparatos razonablemente comunes en muchos sitios del mundo permiten grabar y regrabar músicas en *mini disks*. Las músicas se pueden obtener a partir de otros aparatos de sonido, o de la salida *line-out* de la tarjeta, utilizando el cable adecuado. Sin embargo, como estas salidas son analógicas y el *mini disk* almacena el sonido en formato digital, tenemos una pérdida de calidad en el momento de la conversión. Los modelos más avanzados ya incorporan una entrada óptica que permite obtener el sonido digitalmente. Usando un cable óptico, en realidad un cable de fibra óptica con dos terminaciones especiales, es posible conectar el grabador en la salida óptica de la tarjeta de sonido y grabar la música a partir de un CD, o archivos *MP3*, sin ninguna pérdida de calidad.

9.4.5 Aureal SQ1500 y SQ2500

Después de trabajar durante un buen tiempo produciendo y vendiendo *chipsets* de sonido para terceros, la casa Aureal decidió cambiar de ramo y pasar a producir sus propias tarjetas de sonido usando sus propios *chipsets*.

Inicialmente fueron lanzados dos modelos, la *Aureal SQ1500* y la *Aureal SQ2500*, basadas respectivamente en el los *chipsets Aureal Vortex* y *Aureal Vortex 2*. La tarjeta *SQ1500* presenta una calidad sonora excelente en la reproducción de CDs o en archivos *WAV* o *MP3*, un sintetizador *MIDI* de 64 instrumentos simultáneos vía hardware y 512 más vía software, una tasa de *signal to noise* de 92 dB, el soporte para los conjuntos de 4 altavoces, salida de sonido digital (coaxial), soporte a la *A3D 1.0* y al *Direct Sound 3D*.

La *SQ2500,* el modelo más avanzado trajo varias mejoras respecto la *SQ1500*, siendo la principal el uso del *chipset Vortex 2*. El sintetizador *MIDI* es el mismo de la *SQ1500*, pero con una pequeña mejora en la calidad de las muestras, que resulta en una reproducción un poco más fiel. Tenemos los mismos 64 instrumentos vía hardware y 512 más vía software, totalizando 576 instrumentos.

La reproducción de música se reraliza con el uso de un algoritmo de interpolación de 27 puntos, se soportan conjuntos de 4 altavoces, el puerto de juegos es digital y tenemos la misma salida de sonido digital coaxial de la *SQ1500*.

9.4.6 La sucesora de la SB Live

La *Sound Blaster Live* de la casa Creative tuvo muchos días de gloria en el mercado de las tarjetas de sonido. Al ser lanzada, superaba en calidad y traía varios recursos de las tarjetas profesionales, que eran mucho más caras. La *Sound Blaster Live* aún puede considerarse una buena tarjeta de sonido, pero seguramente ya no es la mejor. A pesar del gran atraso, finalmente, la casa Creative lanzó una sucesora de la *SB Live*, la *Sound Blaster Audigy*.

La idea era crear una tarjeta de sonido que pudiera superar a las concurrentes, como las tarjetas con los *chips Vortex*. En relación con la *SB Live*, la *SB Audigy* evolucionó en casi todos los sentidos, empezando por la tasa de *signal to noise* (que mide la pureza del sonido) que pasó a ser de 100 dB, en lugar de los 96 dB de la *SB Live*. Es una mejora notable, considerando que las tarjetas *Sound Blaster Live* ya poseían una óptima calidad de audio.

La *SB Audigy* soporta la grabación y reproducción de audio con 24 bits y 96 KHz, en lugar de los 16 bits y 48 KHz del CD y de la mayoría de las tarjetas de sonido. Eso garantiza una pequeña mejora en la calidad del audio, pero nada espectacular, ya que internamente la tarjeta todavía procesa el audio con un muestreo de 48 KHz.

Los recursos 3D también evolucionaron. La *EAX*, la *API* 3D soportada por la *SB Live* ganó nuevos recursos y pasó a llamarse *EAX Advanced HD*, y eso mejora la fidelidad de los efectos de audio 3D en los juegos, que era uno de los puntos débiles de la *SB Live*. Finalmente, la tarjeta incorporó una interfaz *Fireware*, que puede ser usada por algunos grabadores de CD, cámaras digitales, etc.

Existen 4 versiones de la tarjeta, la *MP3+*, la *X-Gamer*, la *Platinium* y la *Platinum EX*. En todas las versiones, la tarjeta de sonido es la misma, lo único que varía son los programas y accesorios que la acompañan.

10. Los monitores

10.1 Una visión global

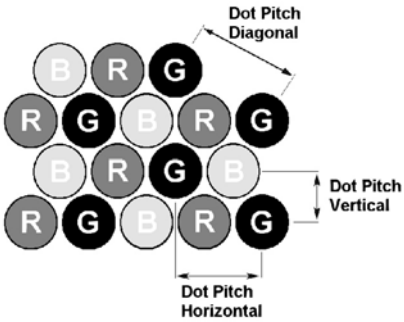
El monitor tiene una importancia vital pues, en conjunto con la tarjeta gráfica, forman el principal medio de comunicación entre el equipo y el usuario. Los factores que diferencian los incontables modelos de monitores son, básicamente, el tamaño, el *Dot Pitch*, o el tamaño de los puntos que lo componen, las resoluciones admitidas y la frecuencia máxima de actualización de la imagen.

Un monitor típico de PC

El tamaño lo da la medida en pulgadas entre las diagonales de la pantalla. Actualmente, los más usados aún son los monitores de 14 y 15 pulgadas, pero en caso de querer trabajar con aplicaciones gráficas es mucho más recomendable un monitor de 17 o 20 pulgadas. Además del tamaño físico, la ventaja de los monitores de mayor tamaño, es que soportan mayores resoluciones, así como unas mayores frecuencias de actualización.

Otra cosa importante con relación a los monitores es el tamaño de los puntos que componen la pantalla, conocido por *Dot Pitch*. Si cogemos una buena lupa y examinamos la pantalla del monitor, veremos que la imagen está formada por puntos verdes, azu-

les y rojos. Cada conjunto de tres puntos es llamado tríada, y la distancia diagonal entre dos puntos del mismo color, lo que compone la medida de una tríada es llamada *Dot Pitch*. Lo más común es que encontremos monitores con un *Dot Pitch* de 0.29 milímetros cuadrados. Algunos monitores más recientes, sin embargo, utilizan unos *Dot Pitch* menores, de 0.22 o 0.19 mm, lo que garantiza una imagen de mejor calidad.



Las medidas del Dot Pitch

Un buen monitor de 15 pulgadas debe soportar resoluciones de hasta, como mínimo, 1024 x 768. Los monitores de un mayor tamaño también deben ser capaces de mostrar resoluciones de 1280 x 1024, o 1600 x 1200 en el caso de los monitores de 20 pulgadas. Lo más común por parte de los usuarios que usan monitores de 14/15 pulgadas es el uso de una resolución de 800 x 600. En el caso de monitores más grandes, sin embargo, el uso de resoluciones mayores es muy recomendable.

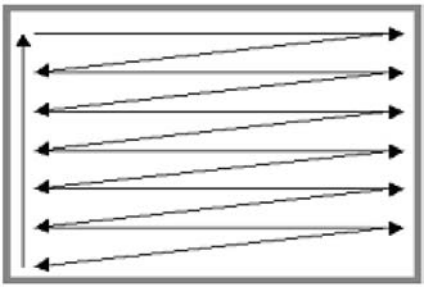
La última característica, y tal vez de más importante en los monitores, es la frecuencia de actualización de la imagen, o *refresh rate*. En un monitor, un grupo de electrones bombardea continuamente la pantalla formando la imagen. La cantidad de veces por segundo que este grupo actualiza la imagen es llamada la frecuencia de actualización. Un buen monitor debe ser capaz de actualizar la imagen por lo menos 75 veces por segundo (75 Hz). Sin embargo, monitores de menor calidad sólo son capaces de mantener una frecuencia de refresco de 60 Hz, lo que causa un centelleo en las imágenes, también conocido por *flicker*.

El *flicker* ocurre debido a la pérdida de luminosidad de las células de fósforo del monitor. Usando una tasa de refresco de menos de 75 Hz el tiempo que tarda el grupo de electrones para pasar es muy largo, haciendo que las células pierdan parte de su brillo, siendo reencendidas bruscamente en el próximo paso del grupo de electrones. Esto hace que las células centelleen, volviendo la imagen inestable.

La tasa de actualización del monitor también depende de la resolución utilizada. En el monitor, la imagen es actualizada línea a línea, de arriba hacia abajo. La cantidad de líneas que el monitor es capaz de barrer por segundo es llamada frecuencia horizontal, que es medida en KHz. Los monitores de 14/15 pulgadas tienen una frecuencia horizontal de 49 KHz generalmente, o sea, son capaces de actualizar 49.000 líneas por

segundo. Es decir, suficiente cuando usamos una resolución de 640 x 480 o de 800 x 600, pues 49 KHz son suficientes para una frecuencia de actualización de 75 Hz, lo que ya es un buen valor.

Podríamos preguntarnos por qué 75 Hz es un buen valor si 49.000/600 resulta 81.6. La respuesta es el retraso vertical y horizontal, que corresponde al tiempo que tarda el grupo de electrones, cuando llega al final de una línea o a la última línea de la pantalla, para volver al inicio y reiniciar el barrido. El tiempo perdido con el retraso varía de un monitor a otro, pero consume un 5 o 6% del tiempo total en general. Los monitores mayores, sin embargo, poseen frecuencias horizontales que pueden llegar a ser de más de 135 KHz, lo que nos proporciona unas buenas frecuencias de actualización, incluso en las resoluciones más elevadas.



El sentido de la frecuencia de actualización

Una curiosidad sobre los monitores es que los utilizados actualmente son, en su mayoría, monitores analógicos, mientras que los antiguos monitores *CGA* y *EGA* eran digitales. En realidad, los antiguos monitores *CGA* y *EGA* trabajaban con una cantidad muy limitada de colores, haciendo que fuese muy fácil usar señales digitales para formar la imagen. A partir de los monitores *VGA* se pasó a usar señales analógicas para formar el color de los puntos del monitor, permitiendo generar una cantidad, en teoría, ilimitada de colores, así como existen ilimitadas frecuencias posibles para una señal analógica.

En la práctica, sin embargo, teníamos 256 colores en los monitores *VGA* y 16 millones en los monitores *Super VGA*, valores nada modestos comparados con los de los monitores más antiguos.

Evidentemente, la imagen que va a mostrarse es grabada en la memoria de la tarjeta gráfica en formato digital. Sin embargo, en la tarjeta gráfica, existe un circuito especial llamado *DAC* (*Digital Analog Converter*) que convierte las señales digitales en señales analógicas que pueden ser entendidas por el monitor. La función del circuito es la de sólo leer los datos de la memoria de vídeo, convertirlos en señales analógicas y enviarlos al monitor. Al decir que los monitores actuales son analógicos, nos referimos al hecho de que las señales que van desde la tarjeta gráfica al monitor son señales analógicos en lugar de señales digitales.

10.2 Los monitores LCD

Los monitores *LCD* (*Liquid Cristal Display* o monitores de cristal líquido) ya hace tiempo que vienen siendo usados en los ordenadores portátiles. Actualmente, también existe una popularización de esta tecnología en el mercado de los ordenadores de sobremesa.

Un equipo de sobremesa con un monitor LCD

10.2.1 Las ventajas

Los monitores *LCD* ofrecen distintas ventajas sobre los tradicionales monitores *CRT* (*Catodic Ray Tube*, o tubo de rayos catódicos) usados hasta ahora. La primera diferencia que salta a la vista entre un monitor *LCD* y un monitor *CRT* es el tamaño. Los monitores de cristal líquido son más finos que los tradicionales monitores *CRT*, lo que explica su uso en los ordenadores portátiles. En el caso de un equipo de sobremesa la ventaja en este caso no es tan evidente, pero de cualquier modo disponemos de alguna mejora en el espacio encima de la mesa.

Otra ventaja de los monitores *LCD* es el hecho de que poseen una pantalla realmente plana, lo que elimina las distorsiones de imagen causadas por las pantallas curvas de los monitores *CRT*, y aumenta el área útil del monitor ya que no tenemos espacios desperdiciados en las esquinas de la imagen. Por ejemplo, un monitor *LCD* de 12.1 pulgadas dispone de un área de exhibición de la imagen casi equivalente a la de un monitor *CRT* de 14 pulgadas. Un monitor *LCD* de 14 pulgadas posee un área de exhibición de la imagen mayor a la de un monitor *CRT* de 15 pulgadas, mientras que en un monitor *LCD* de 15 pulgadas el área es casi equivalente a la de un monitor *CRT* de 17 pulgadas.

Los monitores de cristal líquido también gastan menos electricidad. Mientras un monitor tradicional de 14 pulgadas consume alrededor de 90 vatios y uno de 17 pulgadas alrededor de los 110 vatios, un *LCD* difícilmente sobrepasa la marca de los 40

vatios. Otra de las ventajas es que estos monitores emiten una cantidad mucho menor de radiación nociva (prácticamente ninguna en algunos modelos), lo que los hace especialmente atractivos para quienes están mucho tiempo frente al monitor.

Finalmente, en los monitores de cristal líquido no existe el *flicker*, pues en vez de que la imagen sea formada por la acción de un flujo de electrones, como en los monitores *CRT*, cada punto de la pantalla actúa como una pequeña bombilla, que cambia su tonalidad para formar la imagen. El término *refresh rate* no se aplica a los monitores de cristal líquido pues, en ellos, la imagen es siempre perfecta.

10.2.2 Las desventajas

Seguramente, la apariencia de un monitor *LCD* es mucho más elegante y moderna que la de un monitor *CRT* tradicional, aunque sin embargo, como nada es del todo perfecto, los monitores de cristal líquido también tienen sus desventajas: el área de visión es más limitada, el contraste es más bajo y las resoluciones permitidas se encuentran más limitadas.

Mientras que en los monitores tradicionales podemos ver la imagen mostrada prácticamente desde cualquier ángulo, en los monitores *LCD* el ángulo de visión es mucho más limitado, sólo a 90 (45 hacia la izquierda y 45 hacia la derecha), y por encima de eso la imagen aparece con los colores distorsionados o no puede verse. Esto puede ser hasta deseable en algunas situaciones, en el cajero de un banco por ejemplo, pero normalmente es un inconveniente para trabajar.

El contraste de la imagen también es muy bajo. Mientras que en un monitor convencional tenemos normalmente un contraste de 500:1, o sea, una variación de 500 veces en la emisión de la luz del blanco para la luz del negro. En los monitores de cristal líquido el contraste varía entre 250:1 y 300:1, lo que perjudica un poco a la calidad de la imagen, principalmente a la fidelidad de los colores.

También tenemos las limitaciones en cuanto a las resoluciones admitidas. En los monitores *CRT* disponemos de varias resoluciones de pantalla diferentes, que van desde los 320 x 200 puntos usados en el *MS-DOS* hasta 1024 x 768, 1200 x 1024 o incluso los 1600 x 1200, pasando por varias resoluciones intermediarias como 400 x 300, 320 x 400, 320 x 480, 512 x 384, 1152 x 864 entre otras, y en todas las resoluciones tenemos una imagen sin distorsiones.

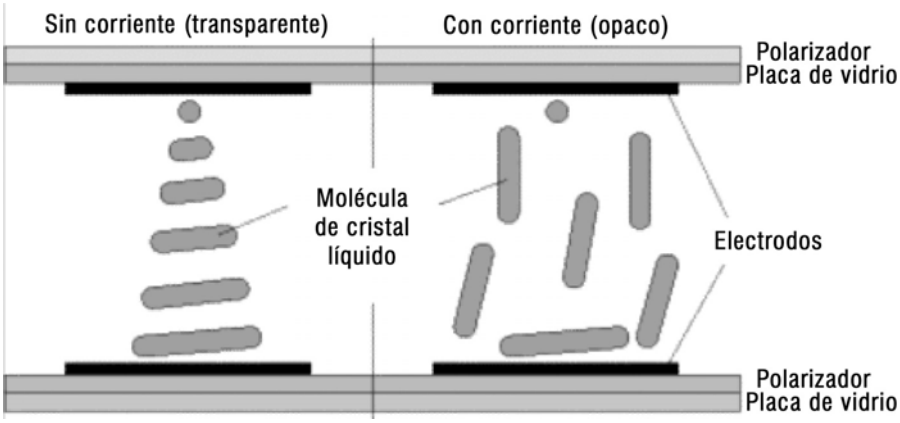
Por su parte, los monitores de cristal líquido están más limitados en este aspecto, pues cada punto de la imagen es físicamente representado por un conjunto de 3 puntos (verde, rojo y azul). Por ejemplo, en un monitor *LCD* con una resolución de 1024 x 768 tenemos 3072 puntos horizontales y 768 verticales, donde cada conjunto de 3 puntos forma un punto de la imagen Como no es posible alterar la disposición física de los puntos, la resolución máxima queda limitada al número de puntos que componen la pantalla. Podemos usar resoluciones menores, usando más de un punto de la pantalla para representar cada punto de la imagen.

A pesar de que no dejan tanto que desear en términos de calidad de la imagen y que poseen algunas ventajas interesantes, los monitores *LCD* aún son caros en comparación con los monitores *CRT*.

10.2.3 Cómo funciona el LCD

En los monitores convencionales tenemos un tubo de rayos catódicos que bombardea constantemente las células luminosas de la pantalla para formar la imagen. En el caso de un monitor *LCD* se usa una tecnología totalmente diferente, que consiste en el uso de cristales líquidos para formar la imagen.

Los cristales líquidos son sustancias que tienen una estructura molecular que se altera al recibir corriente eléctrica. En su estado normal son transparentes, pero al recibir una carga eléctrica se convierten en opacas, impidiendo el paso de la luz. En los visores de cristal líquido más primitivos, como los de los relojes de pulsera, sólo teníamos estos dos estados, transparente y opaco, o sea, o el punto está encendido o está apagado. En los visores más sofisticados, como los usados en los portátiles, también tenemos estados intermedios, que forman las distintas tonalidades de gris o de los colores. Estos tonos intermedios se obtienen con el uso de distintas tensiones.

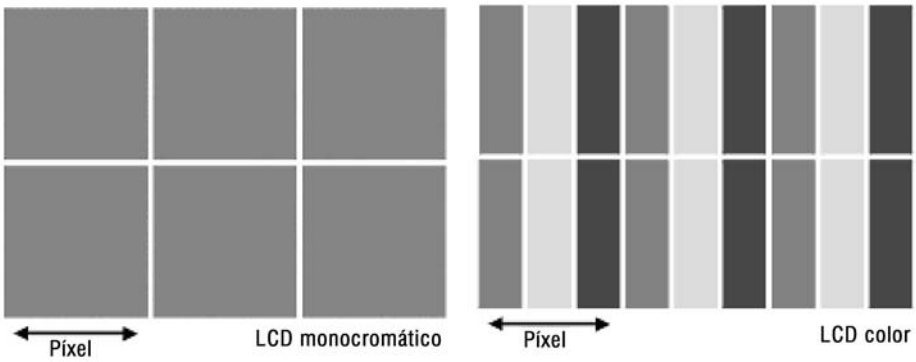


El funcionamiento de las moléculas de cristal líquido

Para formar la pantalla de un monitor se coloca una fina capa de cristal líquido entre dos capas de vidrio. Estas finas placas poseen pequeños surcos, aislados entre sí, con un electrodo conectado a un transistor en cada uno de ellos. Cada uno de estos surcos representa uno de los puntos de la imagen. Por su parte, el conjunto de estas capas es colocado entre otras dos capas de un elemento polarizador. Detrás se instala una fuente de luz, generalmente compuesta por bombillas fluorescentes (usadas debido a que generen poco calor) o *LEDs*, responsables de la iluminación de la pantalla.

En el caso de monitores *LCD* monocromáticos, cada punto de la pantalla corresponde a uno de los puntos de la imagen. En el caso de los monitores *LCD* de color, cada píxel de la imagen está formado por un grupo de 3 puntos, uno verde, uno rojo y otro azul.

En este caso, los colores son obtenidos a través de diferentes combinaciones de tonalidades de los tres puntos, como en los monitores *CRT*.



La composición de puntos para cada píxel

Actualmente, existen dos tecnologías en la fabricación de pantallas LCD, conocidas como de matriz pasiva (*DSTN*) y de matriz activa (*TFT*). Las pantallas de matriz pasiva presentan un ángulo de visión más restrictivo y es necesario un mayor tiempo para que la imagen sea actualizada. Mientras que en un monitor *CRT* un punto tarda cerca de 15 a 20 milisegundos para cambiar de color, en un monitor *LCD* de matriz pasiva son necesarios entre 150 y 250 milisegundos. Por eso es tan difícil encontrar el cursor del ratón en la pantalla de un ordenador portátil. En estos monitores, la imagen presenta una calidad inferior, debido al bajo contraste. Felizmente, los monitores de matriz pasiva ya sólo se encuentran en equipos antiguos y ya no se fabrican en la actualidad.

Los monitores *LCD* de matriz activa, usados actualmente, presentan una calidad muy superior, con un tiempo de actualización de imagen más próximo al de los monitores *CRT*, entre 40 y 50 milisegundos. Los monitores *LCD* de matriz activa también poseen un mayor ángulo de visión y un contraste mayor, además de ser más finos y menos pesados.

Al contrario que los monitores *CRT* actuales, todos los monitores de cristal líquido son digitales. Como todas las tarjetas gráficas actuales le envían señales analógicas al monitor, se usa un nuevo circuito que convierte las señales analógicas recibidas de la tarjeta gráfica al formato digital representado por el monitor.

El cambio digital-analógico-digital en este caso es totalmente innecesario y sólo sirve para degradar la calidad de la imagen y aumentar la cantidad de circuitos usados en el monitor, encareciéndolo.

10.2.4 Los monitores Touch Screen

Los monitores sensibles al tacto son muy usados en cajas, quioscos multimedia, ordenadores de mano y varios equipamientos más. Estos monitores están compuestos de

un monitor *CRT* o *LCD* común y de una película sensible al tacto. Además de estar conectados a la tarjeta gráfica, estos monitores también están conectados en uno de los puertos serie del equipo, bastando instalar el software adecuado para que los toques en la pantalla sustituyan a las pulsaciones del ratón.

10.3 Nuevas tecnologías

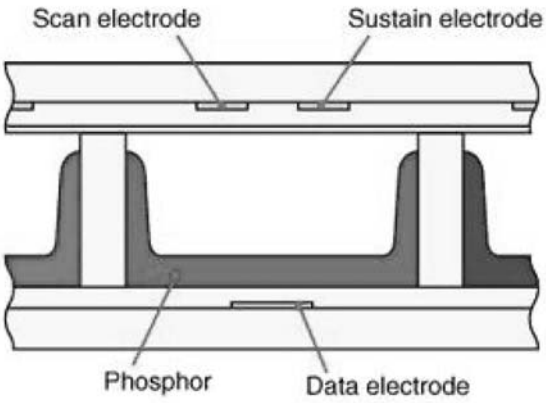
Los monitores *CRT* todavía son los más populares. Por su parte, el uso de monitores *LCD* conlleva varias ventajas, pero también son más caros. Pero no hay nada decidido en el mercado de los monitores, ya que además de estas dos tecnologías también existen otras que prometen mucho para un futuro próximo.

10.3.1 Los monitores de plasma

Además de los monitores *CRT* y *LCD* que ya conocemos bien, existe una tercera categoría de monitores que ya son producidos a gran escala y van ganando popularidad, los monitores de plasma, o *PDP*.

En los monitores *CRT* se usa un tubo de imagen, donde un cañón de electrones bombardea las células de fósforo de la parte frontal formando la imagen. En los monitores *LCD* la imagen es formada por pequeñas células de cristal líquido entre dos placas de vidrio, que son activadas a través de pequeños pulsos eléctricos.

Los monitores de plasma tienen una estructura un poco parecida a la de los monitores *LCD*. También tenemos dos placas de vidrio y electrodos que aplican tensiones eléctricas a las células que generan la imagen. La gran diferencia está justamente en las células. En vez de cristales líquidos, los monitores de plasma están formados por estructuras con el mismo principio de funcionamiento que las bombillas fluorescentes. *Grosso modo*, podemos decir que la imagen está formada por algunos millones de minúsculas bombillas fluorescentes verdes, azules y rojas.



El funcionamiento en los monitores de plasma

La imagen anterior muestra bien esta estructura. En las pequeñas cavidades hechas en la placa de vidrio inferior se deposita una capa de fósforo colorido. Además del fósforo, la cavidad contiene gases como néon, xenón y helio que al recibir una descarga eléctrica liberan una radiación ultravioleta que enciende la capa de fósforo, generando así la imagen.

Estas cavidades no pueden ser muy pequeñas y por eso los monitores de plasma suelen tener un *Dot Pitch* cercano a 1 milímetro, casi 30 veces mayor que en un buen monitor *CRT*. En principio, esto es una gran desventaja que no permite a los monitores de plasma soportar altas resoluciones pero, por otro lado, permite que el tamaño de las pantallas sea muy grande. Ya existen monitores de plasma de 30, 35, o incluso 42 pulgadas.

A pesar de todo, la pantalla sólo tiene una resolución de 1385 x 768. Como ya hemos dicho, la resolución no es el punto fuerte de los monitores de plasma.

10.3.2 Los monitores FED

En una comparación directa entre los monitores *CRT* y *LCD* actuales, los monitores *CRT* pierden. Prácticamente, el único motivo para que aún se usen los monitores *CRT*, que mantienen el mismo principio de funcionamiento descubierto a principios del siglo XX, es el precio. Poca gente compraría un monitor *CRT* si los monitores *LCD* costaran lo mismo o muy poco más.

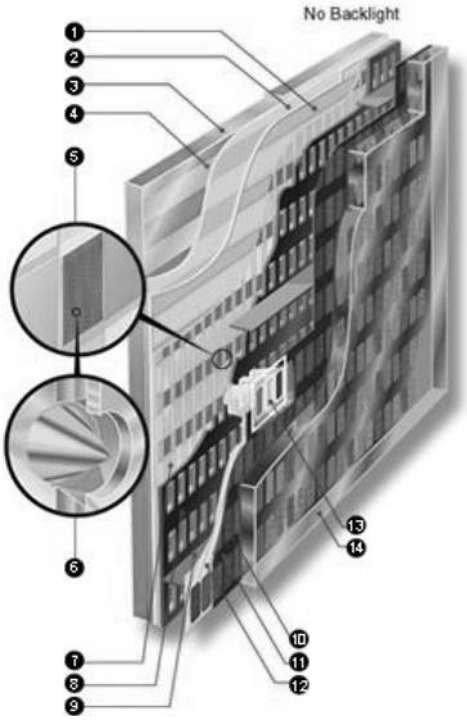
Existe una tecnología prometedora, que puede servir de mucho en un término medio, o incluso como sustituta de ambas tecnologías. Se trata de los monitores *FED*, o *Field Emission Displays* (monitores de emisión de campos). El principio de funcionamiento es el mismo que el de los monitores *CRT*. Despejar electrones sobre las células de fósforo del monitor que transforman la carga en luz, generando la imagen. La diferencia es que en vez de un único y poderoso cañón de electrones que se va desplazando para actualizar, una a una, todas las células de fósforo, como hemos visto en los monitores convencionales, cada célula de fósforo pasa a ser alimentada por un conjunto de mini-cañones, en forma de unas minúsculas puntas metálicas .

Existen dos grandes ventajas en esta técnica. La primera es la posibilidad de producir monitores *CRT* casi tan finos como los monitores *LCD* actuales, y la segunda es una disminución en el consumo eléctrico, pues no existe el desperdicio llevado a cabo por el cañón de electrones.

En el esquema siguiente podemos ver como funciona un monitor FED:

1. Dieléctrico
2. Capa resistiva
3. Cátodo (emite cargas eléctricas negativas)
4. Líneas de metal
5. Conjunto de micro puntas

6. Una de las micro puntas
7. Columnas de metal
8. Reja de metal (impide que las emisiones destinadas a una célula de fósforo alca cen a las demás, mejorando la calidad de la imagen)
9. Pared de vidrio
10. Célula de fósforo
11. Matriz negra
12. Capa de aluminio con cargas positivas (ánodo)
13. Píxel
14. Capa de vidrio



Un esquema del funcionamiento de los monitores FED

Los monitores *FED* presentan ventajas incluso sobre los monitores *LCD*, pues no existen limitaciones en el ángulo de visión y el consumo eléctrico es muy bajo, menos de 5 vatios para un monitor *FED* de 14 pulgadas.

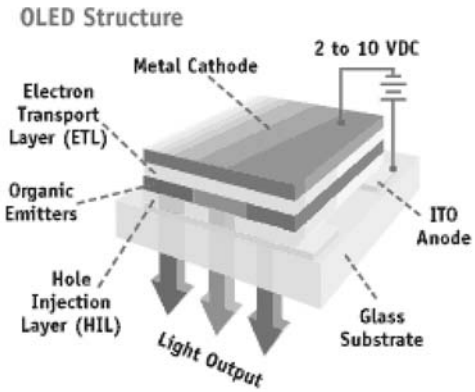
Ya existen algunas empresas produciendo monitores *FED*, entre ellas la casa Candescent, que fabrica monitores de entre 4.4 y 13.2 pulgadas, destinadas a varias aplicaciones. De momento los monitores *FED* aún son muy caros, incluso más que los monitores *LCD*.

10.3.3 Los monitores OLED

OLED viene de *Organic Light Emitting Diode* o diodo orgánico emisor de luz. Esta tecnología promete pantallas planas mucho más finas y menos pesadas que las actuales pantallas *LCD*. La idea es usar diodos orgánicos, compuestos por moléculas de carbono que emiten luz en el momento que reciben una carga eléctrica. La ventaja es que al contrario de los diodos tradicionales, estas moléculas pueden ser directamente aplicadas sobre la superficie de la pantalla, usando algún método de impresión. Una vez añadidos los filamentos metálicos que conducen los impulsos eléctricos a cada célula, la pantalla está lista.

Como estas pantallas son muy simples, esta tecnología presenta un coste inferior respecto al cristal líquido. Otro posible uso sería el tan aguardado *e-paper*, un tipo de pantalla flexible, lo suficiente para ser usada en libros digitales o lectores para periódicos y otras publicaciones vendidas en formato digital.

Otra posibilidad es la construcción de pantallas más elaboradas, capaces de competir con los monitores *LCD*. Los monitores *OLED* pueden ser utilizados tanto en pantallas de matriz pasiva como de matriz activa; lo que cambia en las dos técnicas es básicamente el uso o no de un transistor por célula, y no el material en sí. La casa Kodak, junto con otras compañías trabaja en pantallas que tienen *OLED* de tres colores, prensados entre un cátodo, un emisor de cargas negativas y un ánodo, cargado con cargas positivas. La combinación de las dos capas hace que las cargas eléctricas pasen por los *OLEDs*, produciendo la luz.



El funcionamiento de la tecnología OLED

Esta tecnología, muy revolucionaria, puede ser usada en pantallas de dos o tres pulgadas de diámetro, pero con una alta resolución, que pueden ser usadas en varios campos, como proyectores o en aparatos de más alta tecnología como gafas donde se muestran imágenes generadas por un ordenador portátil. Eso todo sin hablar de la próxima generación de móviles, que serán capaces de reproducir vídeos.

No está descartado el uso de esta tecnología en los monitores del futuro, pero el gran problema es la fabricación de monitores de matriz activa con una calidad de imagen semejante a la de los monitores *LCD*. Lo más probable es que en el futuro los monitores *OLED* sean utilizados en pantallas de ordenadores de mano, móviles y otros tipos de aparatos portátiles, con pantallas de baja densidad.

10.4 El uso de dos monitores

Deberíamos haber oído hablar del soporte de hasta nueve monitores del *Windows 98*. Este recurso que también es soportado por *Windows 2000* puede ser bastante útil, principalmente, para quien utiliza monitores de 14 o 15 pulgadas que no soportan resoluciones muy elevadas.

Lo más común y práctico es el uso de dos monitores. Para eso, sólo necesitamos sólo comprar una tarjeta gráfica más. El segundo monitor puede ser cualquier monitor *VGA* o *SVGA*. Incluso podemos utilizar el monitor viejo que sobró de la actualización de un viejo *486* o *Pentium*, por ejemplo. Es posible qué tanto la configuración de la resolución de vídeo como la cantidad de colores usados sean diferentes para cada monitor, ya que cada uno posee su propia tarjeta gráfica.

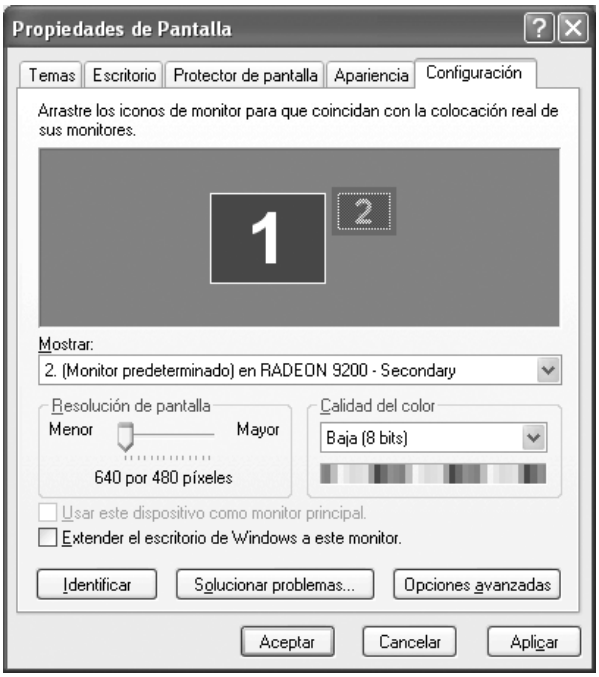
El uso de más de un monitor permiten ampliar el área de trabajo

El segundo monitor puede utilizarse para expandir el área de trabajo. Las posibilidades son muchas: mientras estamos navegando por *Internet*, por ejemplo, podemos dejar el navegador abierto en el primer monitor y al mismo tiempo mantener abierto el correo electrónico y el *Messenger* en el segundo monitor, en vez de tenerlo todo en un único monitor y que haya que minimizar uno y maximizar el otro todo el rato.

El recurso de usar múltiples monitores sólo es soportado por las tarjetas gráficas PCI o *AGP*. Las tarjetas gráficas *ISA*, *VLB*, *EISA*, etc. no se pueden usar. Para ello, pode-

mos utilizar tanto dos tarjetas *PCI* como una *AGP* y una *PCI*. Una de las tarjetas corresponderá al vídeo primario y la otra al vídeo secundario. Casi todas las tarjetas gráficas actuales soportan su utilización como vídeo secundario. El único problema puede venir porque no todas las tarjetas gráficas soportan ser usadas como vídeo primario. De cualquier modo, son muy pocas las tarjetas gráficas incompatibles con este interesante recurso.

Después de instalar físicamente la segunda tarjeta gráfica, basta cargar *Windows* para que sea encontrado el nuevo hardware. Si *Windows* posee el *driver* adecuado, la tarjeta gráfica será instalada de forma automática, de lo contrario será necesario utilizar los *drivers* del fabricante. Después de reiniciar el sistema, el primer monitor exhibirá el escritorio normal, pero el segundo sólo mostrará un aviso de texto, avisando que *Windows* detectó el uso de dos monitores. Abrimos el icono *Pantalla* del *Panel de control* y en la ficha *Configuración* aparecerán ahora dos monitores, en vez de uno. Pulsamos en el icono del segundo monitor y el sistema nos preguntará si deseamos activarlo, respondemos que sí. Ahora sólo queda configurar la resolución y la cantidad de colores de cada monitor.



Las opciones de configuración de un segundo monitor

El segundo monitor funciona como una extensión del área de trabajo del primero. Esto significa que basta mover el ratón en dirección al segundo monitor para que el cursor pase a él. En la misma ventana de configuración deberemos arrastrar los monitores a modo de representar su posición física. Esta información es usada para controlar la acción del cursor del ratón.

10.4.1 Vídeo primario y secundario

Al usar más de un monitor, una de las tarjetas gráficas será configurada como vídeo primario y las demás como secundarias, terciarias, etc. El vídeo primario será su monitor principal, donde surgirán los cuadros de diálogo y el monitor que la mayoría de los programas usará por defecto, etc. El estatus de la tarjeta no es definido por *Windows*, pero sí por el *BIOS*, que elige qué tarjeta será la primaria conforme al *slot PCI* al que esté conectada.

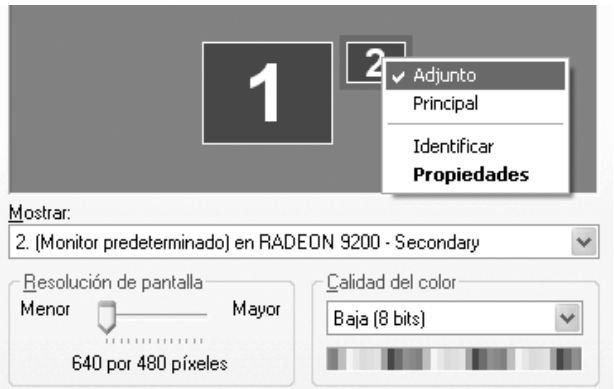
Si estamos utilizando una placa *AGP* y otra *PCI* tendremos un poco más de trabajo, pues por defecto, la tarjeta gráfica *PCI* será detectada como primaria. En la mayoría de los casos podremos alterar eso a través de la opción correspondiente del *BIOS*.

Windows 2000/XP permite escoger que tarjeta será la primaria a través de la propia ventana de *Propiedades* de *Pantalla*.

10.4.2 Limitaciones

Trabajar con dos o más monitores conlleva algunas limitaciones. Lo más grave es el hecho de que *Windows 98* deshabilita el soporte a *Open GL* de la tarjeta gráfica 3D al ser activado el segundo monitor. En este caso, deberemos desactivar el segundo monitor en la ventana *Propiedades* de *Pantalla* siempre que queramos jugar con algún juego que dependa del soporte *Open GL*. Los juegos que utilizan *Direct 3D* sólo recibirán la aceleración 3D en el caso de ser abiertos en el monitor primario.

Para deshabilitar temporalmente el segundo monitor, basta pulsar con el botón derecho del ratón sobre su icono en la ventana *Propiedades* de *Pantalla* y desactivar la opción de activación.



Deshabilitando el segundo monitor

Muchos protectores de pantalla no soportan el uso de múltiples monitores y por eso sólo serán abiertos en el monitor primario, dejando a los demás sin protección. También existen algunos problemas menores en algunas aplicaciones.

10.4.3 Interferencias

Gran parte del funcionamiento de los monitores *CRT* actuales está basado en el magnetismo. Sin embargo, los monitores no están blindados. Si colocamos dos monitores muy juntos surgirán pequeñas interferencias en muchos casos, generalmente en forma de una línea horizontal subiendo o desciendo constantemente. Para minimizar este efecto, basta usar la misma frecuencia de actualización en ambos monitorices, 75 Hz en el primero y 75 Hz en el segundo, o intentar trabajar con los dos monitores un poco más separados.

En general, este problema es casi imperceptible, pero en el caso de que no incomode, también podemos intentar colocar alguna cosa de metal entre los monitores. La barrera de metal ofrecerá un mejor blindaje si está conectada a una toma de tierra.

11. El teclado

Existen toda una serie de teclados diferentes para un PC. La figura siguiente muestra el teclado IBM original, con el que se suministraba el *PC IBM*. Desde el punto de vista actual es relativamente pesado y las teclas disponen de un comportamiento agradable ante la pulsación, lo cual no puede decirse siempre de los teclados habituales hoy en día.

El teclado original del PC IBM

Los teclados muy económicos, con frecuencia, tienen una estructura relativamente mala desde el punto de vista mecánico, y las teclas suelen tener un punto de presión diferente, lo cual se manifiesta en aparentes errores de escritura, aunque son debidos a la no detección de pulsaciones de teclas.

Para realizar una prueba mecánica del teclado debería pulsar con cierta fuerza y durante unos momentos una tecla del centro del teclado (por ejemplo, la letra U) para ejercer presión en la placa que se encuentra debajo. Si las otras teclas también se doblan o se mueven, este teclado no le dará muchas alegrías durante mucho tiempo, ya que la mecánica sencillamente es demasiado floja como para ser apropiada para el trabajo diario.

Un teclado ergonómico de la casa Cherry, conocida por la buena calidad de sus teclados.

El comportamiento ante la pulsación de los teclados en general suele variar mucho, y aquí más bien depende del gusto personal y de la velocidad de escritura cuál de los comportamientos (suave, presión fuerte, pulsación ligera) es el apropiado para cada uno.

Para *Windows 95* se introdujo un teclado especial que tenía tres teclas nuevas. Estas teclas servían para facilitar el manejo de *Windows 9x* y se denominan *Left Windows*, *Right Windows* y *Application Menu Function Key* (tecla <Win> con la bandera de Microsoft).

El teclado especial para Windows 95 con las tres teclas adicionales

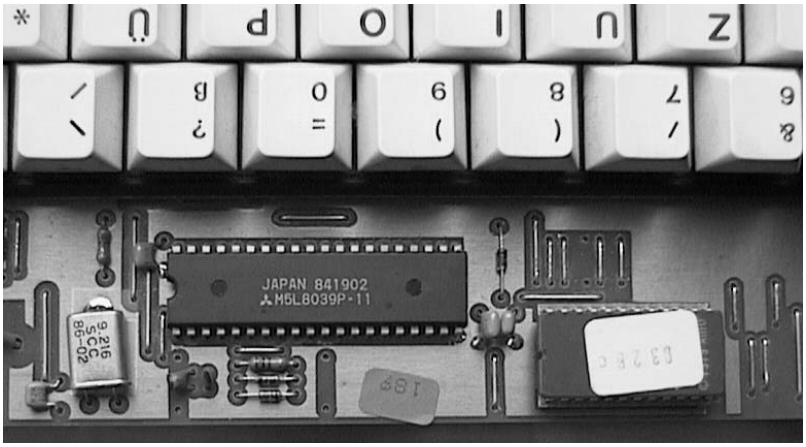
Las nuevas teclas se encuentran en la fila inferior, y para que ni el formato ni las dimensiones del teclado no difieran mucho del diseño habitual, la barra espaciadora es algo más corta. Con ayuda de la tecla <Win> se accede muy rápidamente a algunas

funciones especiales, como podemos observar en las distintas funciones de la tabla siguiente.

Combinación de teclas	Función
<Win>	Activar menú Inicio, mostrar barra de tareas
<Win> + <E>	Inicio del Explorador de Windows, abrir ventana
<Win> + <F>	Buscar carpetas y archivos
<Win> + <Ctrl.> + <F>	Buscar ordenadores en la red
<Win> + <F1>	Ayuda de Windows con función de búsqueda
<Win> + <M>	Minimizar la representación de aplicaciones abiertas
<Win> + <Mayús> + <M>	Abrir las ventanas minimizadas
<Win> + <R>	Inicio del cuadro de diálogo para ejecutar programas
<Win> + <Tab>	Cambiar entre las aplicaciones de la barra de tareas
<Win> + <Pausa>	Mostrar las propiedades de sistema

11.1 Principio de funcionamiento

Si se pulsa una tecla del teclado se cierra un contacto que se encuentra debajo de las teclas. Las teclas están dispuestas en filas y columnas (en una matriz), y cuando se cierra un contacto, se une una columna con una fila. El análisis de la información de la tecla en el teclado la lleva a cabo un *microcontrolador* que genera un código determinado llamado código *Scan* del teclado.



Este teclado tiene incorporado un minicontrolador que utiliza una EPROM como memoria

Un *microcontrolador* en principio no es otra cosa que un *microprocesador* ampliado con algunos elementos de hardware que tienen que conectarse externamente en los

procesadores "normales". Se trata, por ejemplo, de la memoria, un temporizador y un puerto de entrada/salida. El programa del *microprocesador* utilizado en los PCs y en los periféricos a menudo se encuentra dentro del *chip* en una *ROM* programada. Así pues, el programa se ha "grabado" en la *ROM* para un determinado objetivo y no va a poder modificarse. Como puede verse en la figura anterior, el *microcontrolador* utiliza en el teclado en cuestión una memoria de programa externa (una *EPROM*) que está provista de una pegatina. En este caso posiblemente sea posible una actualización *firmware* del teclado.

Para ser exactos, cada tecla tiene dos códigos distintos: uno cuando se pulsa y otro cuando se suelta. El código *Scan* se guarda en una memoria temporal, es procesada posteriormente por el *microcontrolador* y a continuación se envía al PC. El procesamiento posterior del código *Scan* en un PC tradicional (*CPU 8088/86*) se efectúa de for-a distinta que en un *AT* (procesador a partir del *286*), por lo que los códigos también difieren entre sí.

El *microcontrolador* del teclado ejecuta diferentes funciones. Así, no sólo se encarga del análisis de las teclas, sino que también efectúa por su cuenta un autotest del teclado después de que lo ordene el PC. Esta orden tiene lugar durante el proceso de inicialización (*POST, Power On Self Test*) del PC y puede reconocerse porque se iluminan los diodos *NUM LOCK, CAPS LOCK* y *SCROLL LOCK* del teclado. Durante el test también se comprueba si las teclas se atascan. Resumiendo, el controlador del teclado se encarga de las siguientes tareas:

- El autotest.
- La selección de teclas.
- El almacenamiento temporal de los códigos de selección de teclado.
- La creación de datos y frecuencias para la conexión en serie con el PC.
- La ejecución del protocolo de transferencia.
- La configuración de la velocidad y el tiempo de retraso de la repetición para las distintas pulsaciones de las teclas.

En la mayoría de los casos, el teclado está conectado al PC con un cable de 5 polos. La clavija entonces corresponde mecánicamente a la conocida conexión enchufable *DIN* (clavija de diodo). Por lo tanto, pueden realizarse prolongaciones sin problemas.

La conexión de 6 polos se introdujo con los modelos *IBM-PS/2* y también se utiliza a menudo en ordenadores portátiles debido a las menores dimensiones mecánicas. Esta conexión de teclado *PS/2* representa igualmente el estándar en todas las placas base en formato *ATX*.

Dado que las dos variantes de conexión son idénticas desde el punto de vista eléctrico, un teclado PS/2 también puede utilizarse con ayuda de un adaptador adecuado en una conexión *DIN* de 5 polos, y también es posible al revés.

11.2 Configuración y solución de errores

Para el funcionamiento del teclado en general se requiere un programa controlador que convierta los datos que se envían del teclado al PC en los caracteres apropiados (letras, números, caracteres especiales). En el controlador de teclado de la placa base se encuentra un programa controlador de este tipo, aunque este interpreta los caracteres para un teclado inglés americano, de modo que su asignación no se corresponde con la española debido a los caracteres especiales que son habituales aquí (ñ, á, ü).

Por esta razón, con el sistema operativo se suministra un programa controlador especial para cada país, el cual se copia automáticamente en el archivo *AUTOEXEC.BAT* durante la instalación del *DOS*. Esta entrada puede tener, por ejemplo, este aspecto:

```
KEYB SP,,C:\DOS\KEYBOARD.SYS
```

Si falta esta entrada o la misma se ha indicado mal y, por lo tanto, el sistema operativo del PC no la detecta correctamente, se dará por sentada la existencia de un teclado inglés americano. En tal caso, por ejemplo, los dos puntos (:) de un teclado español se encuentra en el carácter Ñ y no allí donde uno esperaba.

Si el teclado del PC no funciona, no está bien configurado o no está correctamente conectado, después de conectarlo el *BIOS* del PC emitirá un mensaje de error como *Keyboard Error* y/o una señal acústica (*Beep-Code*). Antes de que se solucione un error comunicado por el *BIOS* no tiene sentido manipular las entradas de los controladores en los sistemas operativos, ya que primero el *BIOS* tiene que haber aceptado e inicializado pertinentemente el teclado.

En relación con los teclados *PS/2* ocurre con frecuencia que el teclado se conecta incorrectamente en la conexión del ratón *PS/2*, que mecánicamente es idéntica a la del teclado *PS/2*. En algunas placas base no tiene ninguna importancia en cuál de las dos conexiones se enchufa el teclado o el ratón, ya que el *BIOS* busca la conexión apropiada casi por sí solo, como ocurre, por ejemplo, en algunas placas base de la casa Intel. Sin embargo, esto no se puede generalizar porque otras placas base sí que hacen una distinción aquí, y en ese caso el *BIOS* no encontraría ninguno de los dos dispositivos. Es una buena idea rotular claramente las conexiones para que no puedan producirse confusiones.

La conexión de teclado PS/2 (abajo) tiene la misma forma que la conexión de ratón PS/2, lo cual puede provocar confusiones. Un teclado con interfaz USB puede conectarse en un conector USB

Un error frecuente en el teclado es que una tecla esté "atascada", lo que por regla general puede comprobarse fácilmente mediante una revisión visual así como ir probando cada una de las teclas. Naturalmente, un teclado también se ensuciará con el paso del tiempo, y también ocurrirá que un *clip* u otro material de escritorio caerá dentro del teclado y provocará un cortocircuito o el bloqueo de una tecla. Cuando se tenga una sospecha de este tipo deberá sacudirse el teclado (con cuidado) con la parte más fina sobre el borde de una mesa, y posiblemente entonces caiga la suciedad que molesta, con lo cual ya se habrá solucionado el supuesto error.

Siempre y cuando el teclado esté correctamente conectado, existen algunos otros motivos para que el *BIOS* no detecte el teclado. En primer lugar debería comprobarse si no se está ante un mensaje falso, ya que puede ocurrir perfectamente que no cualquier teclado de PC pueda trabajar con cualquier placa base (*Keyboard-BIOS*). El punto *Halt On* del *Setup* del *BIOS* ofrece las siguientes opciones, pudiendo evitar el test del teclado, con lo que el PC puede iniciarse aunque el teclado no funcione:

- All Erros.
- No Errors.
- All, But Keyboard.
- All, But Diskette.
- All, But Disk/Key.

Si desactivando el test del teclado en el *Setup* del *BIOS* no ha sido posible utilizar este dispositivo, deberá darle la vuelta para comprobar si dispone de un pequeño interruptor que permita activar diferentes modos. Generalmente este interruptor sólo se encuentra en modelos antiguos o en aquellos teclados que pueden utilizarse también para otras arquitecturas de ordenador. Estos modos se denominan *PC-Mode*, *AT-Mode* y *MF2-Mode* (modo multifunción), siendo este último el modo de funcionamiento habitual de PC.

Antes de adentrarnos más en la búsqueda de errores, seguramente lo más sencillo es conectar en modo de prueba otro teclado para poder localizar más fácilmente el error. No obstante, raramente ocurre que sea defectuoso uno de los controladores. A pesar de todo, si el controlador del teclado realmente no está en condiciones, lo mejor es comprarse un teclado nuevo. Un controlador de teclado defectuoso en la placa base también es sumamente molesto, ya que no puede comprarse suelto, así que hay que comprar una placa base nueva.

Mucho más frecuente que un defecto eléctrico, es una incompatibilidad entre el controlador de la placa base y el del teclado utilizado. Sin embargo, eso sólo es posible si no se utiliza el teclado previsto originalmente para el PC. Algunos fabricantes (por ejemplo, Compaq) van por su propio camino. En algunos casos puede ser útil una actualización del *BIOS*, ya que habitualmente éste está adaptado al controlador de teclado de la placa base.

Como es natural, en la búsqueda de errores no hay que olvidar el cable de conexión del teclado, que tal vez presente una hendidura debido a un manejo inadecuado, con lo cual queda dañada la conexión. Eventualmente esto puede constatarse si uno mueve y tuerce un poco el cable mientras se inicia el PC. Los pasos siguientes serían una revisión de las conexiones del enchufe, siempre y cuando pueda desatornillarse, y del propio teclado.

Si el *BIOS* ha aceptado el teclado, en realidad sólo puede considerarse el *driver* del teclado del sistema operativo como causa del comportamiento incorrecto del teclado. En este caso lo mejor es reinstalar el *driver* que acompañaba al teclado.

Los enchufes para el teclado, el ratón y el resto de conexiones nunca deben extraerse ni insertarse con el PC conectado porque podría dañarse la electrónica. En muchas placas base existe un fusible para la conexión del teclado que podría fundirse.

12. El ratón

Después del teclado, el ratón es el segundo medio de entrada por orden de importancia, y resulta imprescindible en los entornos gráficos de usuario. Como periférico del PC fue descubierto bastante tarde, en cambio, el sistema Apple Macintosh lo utilizaba ya de manera estándar en el año 1984, ya que este sistema ya disponía entonces de un entorno gráfico de usuario. El primer ratón que se utilizó ampliamente en el mundo del PC fue el ratón de Microsoft. Actualmente existe una gran variedad de fabricantes que lo producen en diferentes modelos y colores.

Los "ratones" más corrientes utilizan como interfaz un puerto serie (*RS232*) o bien un puerto especial para ratón, que suele ser el conector *PS/2*, el mismo que el teclado, por lo que es fácil confundirlos. Los nuevos y más recientes modelos pueden tener, además, un conector *USB*.

Además de la interfaz correspondiente, los ratones se distinguen entre sí por su forma (lo que los hace más o menos adaptables a la mano, aunque esto es en el fondo una cuestión muy personal), el número de botones y, naturalmente, su calidad. Existen asimismo ratones "sin cable" (por ejemplo *Cordless MouseMan* de la casa Logitech) que se comunican con el PC mediante una señal de radio (con una distancia máxima de 2 metros) que el PC recibe por el receptor correspondiente, conectado a uno de sus puertos *PS/2*. Normalmente, tanto en el ratón como en el PC se utilizan baterías, lo que puede verse como una pequeña desventaja frente a la ventaja de no tener ningún cable a la vista.

Uno de los modelos más apreciados en los últimos tiempos es el ratón de rueda, que puede tener una o dos ruedecitas (botón de rueda), las cuales permiten un rápido movimiento por la pantalla (*scroll*) tanto en sentido horizontal como vertical (con dos ruedas), así como el efecto de zoom. Estas son funciones de gran utilidad, por ejemplo, en las aplicaciones de *Microsoft Office* y en *Internet Explorer*.

Una variante especial de ratón la constituye el *Trackball*, o ratón de bola, en el que la bola no está situada en la parte inferior sino en la parte superior, por lo que el usuario la puede accionar directamente. Estos ratones de bola también se incorporan a veces directamente en el teclado, para combinar así ambos medios de entrada (ratón y teclado). En el fondo, un ratón de bola no es más que un ratón "boca arriba", con la

diferencia de que no se ensucia tanto como el ratón habitual, que se "arrastra" continuamente por una superficie.

Un ratón típico de rueda, con dos ruedas giratorias y un conector PS/2

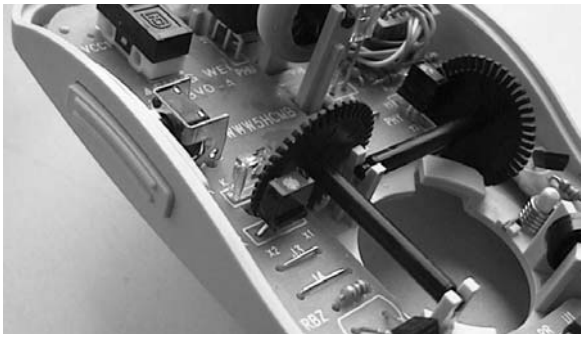
En tiempos algo pretéritos, todavía era habitual el ratón llamado de bus, que no utilizaba el puerto serie del PC, sino que se conectaba mediante una tarjeta especial de conexión al PC. En algunas tarjetas gráficas (antiguas) de la casa ATI, y en algunos otros modelos, todavía existe una conexión para el ratón de bus, aunque los contactos suelen tener asignaciones muy distintas en cada caso y, por lo tanto, no es posible hablar de un estándar general.

Puesto que en un ratón es el movimiento y no la posición lo que se evalúa, se puede utilizar perfectamente en una superficie relativamente pequeña. No sucede lo mismo, por ejemplo, con las tabletas digitales, en los que la posición del periférico de entrada sobre la tableta corresponde a su vez a una posición determinada en pantalla.

12.1 Principio de funcionamiento

El principio de funcionamiento del ratón mecánico, el más habitual, descansa en una bola de acero recubierta de una capa de goma, que transmite su movimiento giratorio a través de dos rodillos, uno para la dirección X y otra para la dirección Y, a dos ruedas dentadas. Estas ruedas se mueven entre dos barreras de luz, interrumpiendo así un haz de luz. Los dos impulsos emitidos son transformados por el sistema electrónico interno en datos serie que se envían hasta un puerto serie del PC. Lo mismo sucede con los impulsos de las ruedas giratorias que pueda haber, y los impulsos de los botones también se integran en el flujo de datos que va al PC, después de lo cual se ejecuta la función seleccionada del programa. Puesto que en el ratón se utilizan barreras de luz, hacia el exterior deberá estar protegido de la luz solar, algo que no resulta tan obvio, pues sigue habiendo modelos en los que el puntero del ratón "se vuelve loco" cuando al ratón le da el sol directamente.

La corriente de alimentación para el ratón se obtiene del puerto, pues consume realmente muy poca. Existen, sin embargo, algunos modelos antiguos que consumen más corriente, por lo que entre la parte macho y la hembra del conector del teclado se ha de utilizar un adaptador activado que permita desviar 5V de esa conexión.



Un ratón abierto (sin la bola), en el que pueden reconocerse las dos ruedas que se mueven, respectivamente, dentro de una barrera de luz

Por estas razones, en general no puede esperarse que un ratón *RS232* vaya a funcionar realmente en un puerto *PS/2* con ayuda de un adaptador, a pesar de que en los comercios se pueden encontrar adaptadores de este tipo (aunque sin indicaciones de ningún tipo). El ratón y el controlador correspondiente deben estar hechos expresamente para funcionar tanto en el puerto *RS232* como en el *PS/2*, como es el caso, por ejemplo, de algunos modelos de la casa Logitech, que incluyen un adaptador en el paquete de entrega.

12.2 Configuración y solución de errores

A diferencia del controlador del teclado, para un ratón conectado en el puerto *RS232* o el *PS/2*, el *BIOS* del PC no ofrece automáticamente soporte al programa. Por eso, en general, hay que cargar un programa controlador que convierta los datos series del ratón, que dependen a su vez del movimiento del ratón y el uso del teclado, en los movimientos correspondientes en pantalla o las acciones que corresponda. Eso impide que se pueda hacer un sencillo *test* con el ratón, ya que primero hay que cargar el controlador y, a continuación, un programa que se pueda manejar con el ratón.

El ratón sólo puede funcionar si en el *Setup* del *BIOS* se ha activado el puerto correspondiente. Para un ratón conectado en el primer puerto *RS232*, eso significa que el punto denominado *On Board Serial Port 1*, o algo similar, y que se encuentra en el menú *Integrated Peripherals Setup*, no debe estar configurado con la opción Disabled, sino (por lo general) en *COM1/3F8*. El *Onboard Serial Port 2* ocupa la dirección *2F8*.

En muchas versiones de *BIOS*, el puerto *PS/2* del ratón no se puede activar y desactivar; siempre está activado y ocupa de manera estándar la *IRQ* 12. En cualquier caso, si hubiera algún problema con un ratón *PS/2*, se debería comprobar exhaustivamente el *Setup* del *BIOS* y asegurarse de que no hay ninguna opción de menú parecida a *PS 2 Mouse Port: Disabled*. En algunos casos , en la placa base existe un *jumper*, denominado *PS/2 Mouse Enable/Disable Header*, que es con el que se activa y desactiva el puerto del ratón.

Las conexiones de ratón *PS/2* presentan problemas muy a menudo cuando se encuentran en las placas *BAT*. La conexión de ratón *PS/2* en las placas *ATX* se encuentra directamente en la placa base, por lo que no hay que realizar ninguna operación con los cables.

Algo distinto sucede con las placas *BAT*, en las que el conector de ratón *PS/2* se encuentra en una tarjeta de expansión o directamente en la parte posterior de la caja, y desde la conexión sale un cable que va a un pequeño conector (normalmente de 8 polos), que se ha de conectar en la fila de pins correspondiente de la placa base (*PS/2 Mouse Header*). Lamentablemente, este conector y los *pins* son tan "maleables" que se puede enchufar del revés o incluso sin enganchar bien los contactos. Este conector *PS/2-Mouse Header* suele encontrarse, generalmente, cerca del enchufe *DIN* (hembra) del teclado. En caso de que un PC que utilice una placa base *BAT* no reconozca un ratón *PS/2*, siempre hay que comprobar este conector. Para ello tal vez sea necesario echar un vistazo al manual, pues no siempre es fácil ver dónde se encuentra el *pin* 1 en la placa base.

El controlador de programa citado anteriormente suele encontrarse en el disquete o *CD-ROM* que acompaña al ratón, bajo el nombre de *MOUSE.COM* (versión residente en memoria) y *MOUSE.SYS*, o un nombre parecido. En cualquier caso, sólo se necesita uno de los dos controladores, que se cargará en el archivo *CONFIG.SYS* o en el archivo *AUTOEXEC.BAT*. Cual de los dos se utilice es algo que no afecta para nada al funcionamiento del ratón.

El controlador correspondiente se suele cargar automáticamente en el archivo que se indique, mediante un programa de instalación previsto para ello. Durante la instalación se especifica, además, si se ha de utilizar un puerto serie para el ratón, y cuál de ellos (*COM1* o *COM2*), también se ejecuta una prueba de funcionamiento del ratón. Los programas de instalación buenos detectan automáticamente (sin intervención del usuario), el puerto en que está conectado el ratón (*COM1*, *COM2*, *PS/2*).

El programa controlador debe ser adecuado, además, para la tarjeta gráfica que se utiliza, pues la forma del cursor, y con ello la información que proporciona al programa, que el ratón representa en la pantalla, depende del modo gráfico de la tarjeta. Si se carga un controlador de ratón incompatible, puede suceder que la pantalla se quede completamente a oscuras o que sólo muestre caracteres incomprensibles. Este fenómeno se producía frecuentemente con las tarjetas Hércules, en el modo gráfico, pero en la actualidad es muy improbable que ocurra algo parecido. Si, de todos modos, ocurre, sólo puede ayudar el fabricante del controlador del ratón con una versión actualizada del mismo. La resolución normal de un ratón suele ser de 400-7500 dpi (*dots per inch*, puntos por pulgada) respecto a la representación gráfica en pantalla.

Para la utilización en *Windows* no es necesaria ninguna adaptación especial del controlador del ratón en los archivos de configuración de *DOS*, sino en el *Panel de control* de *Windows*, en la opción *Mouse* (ventana *Propiedades de Mouse*). La entrada del controlador del ratón tampoco aparece en estos archivos de configuración de

manera estándar, pues sólo se necesita para el viejo modo de *DOS* y, por eso mismo, su localización se reduce exclusivamente a los archivos *CONFIG.SYS* o *AUTOEXEC.BAT* de *DOS*.

Según el tipo del ratón y del controlador correspondiente, en la ventana Propiedades de Mouse podrá encontrar diferentes opciones

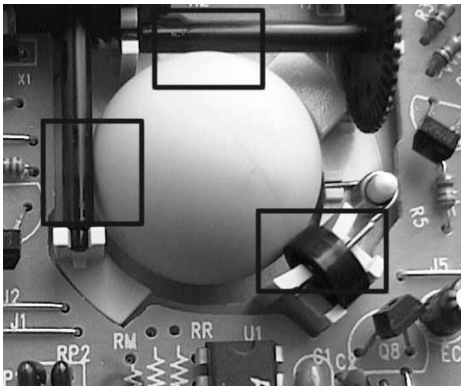
Si se presenta algún problema con *Windows*, en primer lugar debería investigar en el *Panel de Control/Mouse*, o en la ventana *Propiedades de Mouse/General*, si está instalado el controlador (supuestamente) correcto, y hacer alguna modificación si es necesario. Por lo demás, debería comprobar también, en el *Administrador de dispositivos* si realmente hay configurado un sólo ratón y no varios. Si hay varios, es suficiente con borrar las entradas sobrantes.

Al buscar la causa de los problemas, se debería prescindir de las configuraciones especiales (por ejemplo, cursor animado) del ratón, ya que éstas se pueden combinar de tantas maneras que a veces parece que el ratón no funciona. Sí se debería, en cambio, probar con la opción *Mostrar rastro del puntero del mouse*, para comprobar si así puede verse un puntero del ratón en la pantalla.

Comprobar y ajustar alguna configuración del ratón es algo que se dice muy fácilmente, pero que no es tan fácil de hacer cuando el ratón no funciona y hay que manejar *Windows* exclusivamente a través del teclado. En ese caso, la tecla <Tab> y las teclas del cursor cobran especial importancia ya que sirven para seleccionar las diferentes funciones y configuraciones.

Si el ratón se vuelve lento o la bola no se mueve bien, se debe limpiar el interior del ratón. En los lugares indicados en la figura siguiente suele acumularse con el tiempo

polvo y suciedad, los cuales hay que limpiar periódicamente para garantizar el perfecto funcionamiento del ratón



Los lugares a limpiar de suciedad y polvo

12.3 Los ratones ópticos

Un ratón óptico está construido en base a un sensor que rastrea la superficie a una velocidad determinada por el dispositivo. La precisión del ratón depende de la calidad de la imagen registrada electrónicamente y de la velocidad. Los factores clave a tener en cuenta son el tamaño y la calidad del sensor, la velocidad a la que se realiza el muestreo y se envían las imágenes, y la resolución, medida en número de análisis por pulgada cuadrada. El rendimiento viene dado, por tanto, por tres variables, las cuales rara vez son especificadas por los fabricantes.

El tamaño, por ejemplo, es muy difícil de determinar, a menos que el fabricante lo indique expresamente. Este valor está comprendido entre los 15 x 15 y 30 x 30 píxeles. Una vez conocido este valor, los cálculos son mucho más fáciles: a igual sensibilidad, un sensor mayor será capaz de ofrecer valores de coordenadas más precisos.

La tasa a la que el ratón captura imágenes afecta a la velocidad de respuesta y, por tanto, a la precisión, pero este valor no representa mucho por sí mismo. La tasa de captura de los sensores probados varía entre las 2.500 y 6.000 imágenes por segundo. En concreto, los ratones de la casa Microsoft fueron disminuidos hasta las 2.500 imágenes por segundo sin presentar diferencias apreciables de precisión o respuesta en comparación con las originales 6.000 imágenes por segundo.

12.3.1 La resolución

La resolución es una cuestión muy compleja. Se mide en cpi (counts per inch, unidades por pulgada). A mayor número de unidades en cada pulgada, menor será la distancia que el ratón ha de moverse para detectar el cambio. Siendo más explícitos, con una resolución de 800 cpi, el ratón no se tiene que desplazar tanto como con una resolución de 400 cpi para que el cursor se desplace de un lado a otro de la pantalla. Esto

crea una ilusión de velocidad porque si la distancia es relativamente corta, el usuario tiene la sensación de que la velocidad es mayor.

Sin embargo, esto se vuelve más complicado cuando tenemos en cuenta que Windows y los programadores de juegos ofrecen la posibilidad de cambiar el valor. Por defecto, el valor medio es el mismo que la resolución del ratón, a menos que el fabricante del periférico permita también cambiar la resolución del ratón, lo que modifica el valor por defecto de Windows y de los juegos.

En la práctica, no se trata tanto de encontrar la máxima velocidad, sino la mejor, porque si se acelera demasiado, la precisión se verá considerablemente mermada. Esto no se debe al ratón en sí sino al movimiento, que es demasiado rápido como para poder apuntar con precisión. Obviamente, la resolución de la pantalla tiene una influencia directa porque, a 1600 x 1200, la distancia a recorrer es mayor que, por ejemplo, a 800 x 600. Con resoluciones de hasta 1280, un ratón de 400 cpi es adecuado para usos normales, aunque más allá, especialmente en los juegos, conviene acelerarlo con las opciones que nos ofrece el mismo Windows. El ratón pierde entonces precisión porque ha sido llevado más allá de sus límites. Sin embargo, dicha pérdida sólo es apreciable con resoluciones de 1600 x 1200 e, incluso en ese caso, pocos lo notarían. Con 800 cpi no será necesario acelerar el ratón, pues incluso con una resolución de 1600 x 1200, la velocidad por defecto es lo suficientemente rápida. Si se aumenta, no será posible apuntar a un píxel específico. Por tanto, 800 cpi tienen una ligera ventaja sobre 400 cpi cuando se usan resoluciones superiores a 1280.

12.3.2 Los fabricantes y los sensores

Los protagonistas, las casas Microsoft y Logitech, emplean actualmente tres dispositivos distintos en sus periféricos: uno para Microsoft y dos para Logitech. El más avanzado tecnológicamente y también el más preciso es el Logitech MX. Gracias a su enorme sensor CMOS de 30 x 30 píxeles, disfruta de una excelente superficie sensible. Además, la resolución de 800 cpi garantiza una gran precisión con altas resoluciones

Ratón óptico

de 1280 o más. Finalmente, la tasa de 5.250 imágenes por segundo llegará a satisfacer las necesidades de los usuarios más exigentes. Como cabe suponer, es el sensor ideal para jugadores y diseñadores gráficos (retoque fotográfico, CAD, etc).

Logitech también dispone de un sensor de menor calidad, con una resolución de 20 x 20 píxeles y 2.000 imágenes por segundo, aunque también con 800 cpi. Esta razonable combinación ofrece buenos resultados.

Microsoft, por su parte, emplea su sensor clásico de 6.000 imágenes por segundo, el cual ofrece una resolución de sólo 400 cpi. Sin embargo, no hay diferencia de precisión al compararlo con el sensor de la casa Logitech. Sólo con grandes resoluciones se nota que el ratón Logitech es algo más rápido y preciso.

12.3.3 Sin cables ni retraso

Los jugadores comenzaron a aceptar los ratones inalámbricos cuando Logitech presentó su tecnología FastRF. Esta es una tecnología de radio convencional, a 27 MHz, pero cuya tasa de envío ratón-ordenador ha sido elevada hasta alcanzar el límite de 125 muestras por segundo, el límite del USB. Por tanto, ya no hay retraso comparado con un ratón con cable. Y dado que todos los ratones inalámbricos de Logitech emplean la tecnología FastRF, los jugadores ya no pueden escudarse en la velocidad de respuesta para no cambiar a un ratón sin cable.

Por su parte, Microsoft nunca ha hecho comentarios acerca de acelerar su sistema inalámbrico, aunque la realidad es que desde hace tiempo sus ratones han alcanzado los 100 Hz, casi el límite del USB. Esto es algo menor que el FastRF de Logitech, pero la diferencia es casi inapreciable. Por tanto, en la práctica ambos sistemas son igual de rápidos que los ratones con cable.

Microsoft ahorra energía

La migración de Microsoft hacia el sistema inalámbrico ha implicado también un aspecto profesional: el número de canales disponibles para el ratón ha crecido desde 1.000 hasta 64.000. Esto significa que ya no hay posibilidades de que se produzcan interferencias. Hay gente que se ha encontrado con el problema de que un mismo ratón maneja a dos PCs al haber una cantidad de 20, o más, en la misma habitación. La solución consiste en obligar a los ratones a reconectarse en canales diferentes, mediante el botón que suelen incorporar. Disponer de más de 64.000 canales es un buen argumento para una zona con más de 100 ordenadores, pero no es algo estrictamente necesario para el usuario doméstico.

Una innovación mucho más relevante, sin embargo, es que Microsoft ha conseguido reducir significativamente el consumo de energía, por lo que dos baterías AA son suficientes para alimentar al ratón durante seis meses en condiciones de uso normales. Logitech, por su parte, ofrece la alternativa en sus ratones de gama alta a través de una batería y un cargador; en los modelos más baratos, por otra parte, basta indicar que

dos baterías AA duran unos tres meses aproximadamente. En concreto, con un uso de cuatro horas diarias, la vida útil se sitúa alrededor de los dos meses.

Otras novedades

Todos estamos de acuerdo en que la rueda de *scroll* ha sido un invento realmente innovador, pero los ingenieros de Microsoft han ido más allá. El desplazamiento vertical es bueno, pero el desplazamiento en todas direcciones es mejor. Por ello, los nuevos ratones Microsoft pueden desplazar la ventana vertical y horizontalmente. Para ello, se han tenido que hacer algunas concesiones: el desplazamiento vertical se sigue haciendo mediante el movimiento rodante, pero el horizontal se realiza presionando los lados derecho o izquierdo, como con un botón normal. Este sistema está pensado principalmente para las aplicaciones de sobremesa, en concreto para Excel y la navegación por Internet.

13. Las impresoras

Una impresora es un componente habitual de un PC de trabajo, y si bien las impresoras casi siempre se han conectado al puerto paralelo antes de pasar al *USB*, lo cierto es que presentan grandes diferencias en cuanto a su estructura. Estas diferencias dependen del proceso de impresión utilizado, que es el que determina, a su vez, la calidad de la impresión final. Los tipos de impresora más conocidos son los siguientes:

- Impresoras de margarita.
- Impresoras de agujas.
- Impresoras de inyección de tinta.
- Impresoras láser.
- Impresoras térmicas.

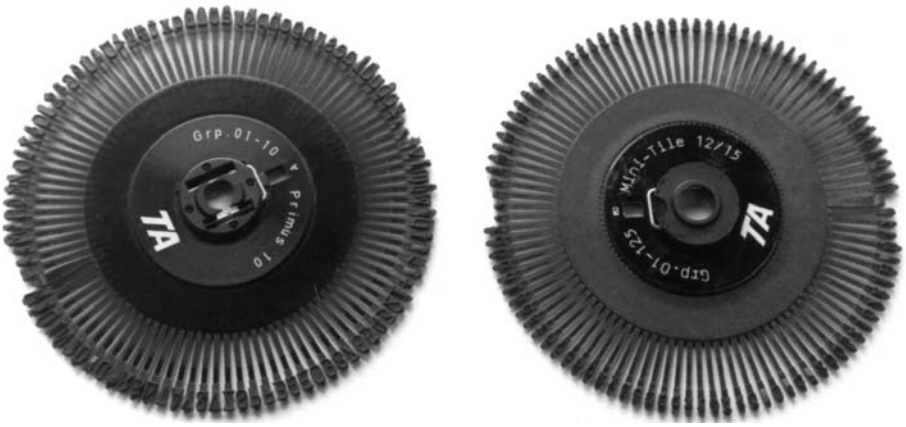
13.1 Impresoras de margarita

Este tipo de impresora, en realidad muy pasado de moda, fue un desarrollo de la máquina de escribir con margarita. En ella, una margarita contiene los caracteres imprimibles, representados en negativo, y un "martillo de impresión" los golpea y estampa sobre la cinta de color y el papel colocado detrás de ella.

Naturalmente, sólo se pueden imprimir los caracteres contenidos en la margarita. Si se necesitan otros caracteres, o un juego de caracteres distinto, se puede cambiar la margarita de la impresora. La cantidad de caracteres disponibles es, por lo tanto, reducida, y cambiar los caracteres es un proceso complicado e incómodo.

Por otra parte, las impresoras de margarita no pueden imprimir gráficos. En lugar de la margarita, en este tipo de impresora se puede incorporar también un cabezal esférico con los caracteres.

La impresora de margarita, sin embargo, ofrece una escritura muy clara y de calidad, por lo que se utiliza sobre todo en los lugares donde es necesario sacar copias de los documentos (con papel carbón). Por eso todavía tienen un pequeño uso en los lugares oficiales.



Dos margaritas con diferentes juegos de caracteres

En lugar de transmitirse por el teclado de una máquina de escribir, los caracteres del PC se transmiten hasta la interfaz de la impresora a través de un puerto (que suele ser *Centronics* o *RS232*). El sistema electrónico de la impresora crea los impulsos a partir de los datos recibidos, y estos impulsos son los encargados de mover la margarita hasta que el carácter queda situado debajo del martillo de impresión, que luego golpea la cinta de color y el carácter se imprime en el papel. Este proceso de impresión es relativamente largo y va asociado siempre a un ruido bastante molesto, exactamente igual que sucedía con las antiguas máquinas de escribir. Ya apenas existen programas (controladores) para las impresoras de margarita.

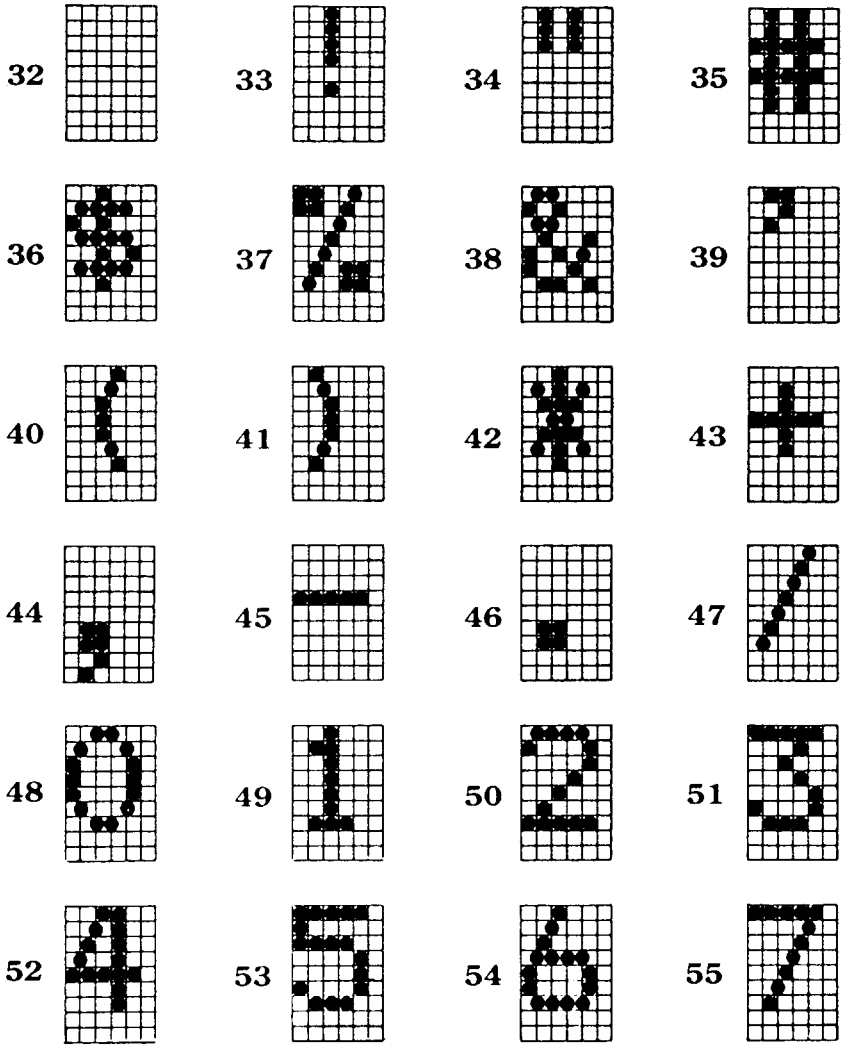
13.2 Impresoras de agujas

Un desarrollo posterior a las impresoras de margarita fueron las impresoras de agujas. Con este nuevo tipo se consigue una mayor flexibilidad en la representación de los caracteres, ya que los caracteres no vienen definidos ahora por un tipo (carácter fijo) sino que se crean con la ayuda de unas agujas en el cabezal de la impresora.

Puesto que las agujas están ordenadas en forma de matriz (en filas y columnas), estas impresoras se llaman también impresoras matriciales. Este tipo de impresora existe en versiones de 9 o de 24 agujas. En la figura siguiente podemos ver cómo se forman los caracteres con la ayuda de nueve agujas.

El mecanismo desarrollado para las impresoras matriciales es, en el fondo, el mismo que el de las impresoras de margarita, con la diferencia de que aquí no se golpean tipos contra la cinta de color, sino agujas. Eso permite crear los caracteres a través de un programa. A diferencia de las impresoras de margarita, el papel no tiene por qué moverse siempre verticalmente en la distancia completa de una línea, sino que puede moverse también en distancias libremente escalables gracias al motor de fase que se utiliza para ello.

De ese modo, la impresora, en colaboración con el cabezal, que también es movido por un motor de fase y que sólo puede moverse horizontalmente, adquiere capacidades gráficas. Aquí, capacidades gráficas significa que puede imprimir fuentes libremente escalables, dibujos e imágenes que no están formados por caracteres de texto sino por otras combinaciones desarrolladas por las agujas.

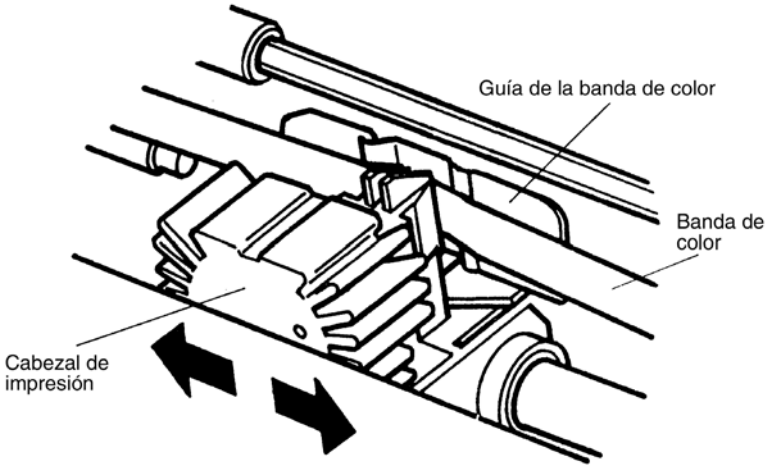


Caracteres impresos con una impresora matricial de 9 agujas

En una impresora de 9 agujas, la matriz consta de nueve líneas, y en una impresora de 24 agujas, la matriz consta de 24 agujas. Sin embargo, el número de columnas no se corresponde al de filas. En este sentido, la matriz no siempre tiene una estructura

1:1, y las columnas de una impresora de 9 agujas están formadas, en muchas ocasiones, por cinco o seis puntos, mientras que una impresora con una matriz de 24 agujas puede presentar formaciones muy diversas. Por eso, los resultados obtenidos con las diferentes impresoras también presentan diferencias grandes en cuanto a calidad, y una buena impresora de 9 agujas puede perfectamente conseguir mejor calidad de impresión (sólo texto) que una de 24 agujas.

Las impresoras matriciales se utilizan frecuentemente para la impresión de listados (por ejemplo, listados de programas y bases de datos) y, en general, allí donde se necesiten copias. Al igual que las impresoras de margarita, también son bastante ruidosas.



El cabezal de impresión con la guía de la cinta en una impresora matricial

Con frecuencia es posible utilizar tanto papel continuo (para listados) como hojas sueltas de papel (para documentos o cartas). El papel continuo tiene una línea de puntos de perforación a ambos lados que facilitan su corte más adelante y ayuda, durante la impresión, a que la parte mecánica de la impresora (el tractor) vaya empujando el papel hacia adelante. Con las hojas sueltas, el papel se empuja hacia delante mediante un rodillo aparte.

Para algunas impresoras matriciales se puede adquirir una bandeja de entrada de papel en la que se pueden depositar varias hojas a la vez; de ese modo no es necesario colocar el papel de forma continua, ni acertar la posición exacta de la hoja, cosa bastante difícil en este tipo de impresoras. En la práctica, estas bandejas para hojas de papel sueltas no siempre cumplen lo que prometen, ya que muchas veces el papel se introduce girado en la impresora, o se queda enganchado dentro y llega a romperse.

La imagen de impresión de una impresora matricial puede manipularse mediante un programa o a través de los botones de la impresora. Se pueden configurar diferentes fuentes y juegos de caracteres, y la mejor calidad de impresión se obtiene en el modo *Near Letter Quality* (*NLQ*), que significa "calidad casi de carta".

En este modo operativo, el cabezal de impresión pasa dos veces por cada línea, es decir, cada línea se imprime dos veces, con una diferencia mínima en la posición de las agujas, lo que equivale en la práctica a un número doble de agujas.

13.2.1 Tipos y mantenimiento

Las impresoras de agujas todavía se fabrican en la actualidad. Si bien no existe casi diferencia de precio entre esta impresora y la de chorro de tinta (cada vez más baratas), que es más rápida, bastante más silenciosa, y ofrece una mejor calidad de impresión, las impresoras de aguja siguen siendo apreciadas porque con ellas el tipo de papel no tiene mayor importancia (a diferencia de la impresora de chorro de tinta), y los costes de mantenimiento son bastante bajos. Por otra parte, con ellas también se pueden realizar copias, como con las impresoras de margarita, algo importante en lugares como consultas médicas, talleres de reparación y otros tipos de talleres manuales.

La DFX-8500 es una impresora de 9 agujas con una velocidad de impresión de 22.4 ppm

A parte del cambio regular de la cinta de tinta, mucho más económica que los cartuchos de las impresoras de inyección, esta impresora no presenta más costes de mantenimiento. El caso de que algunas agujas ya no impriman, o que en el papel no aparezca nada (señal evidente de que hay que cambiar el cabezal de impresión), sucede muy pocas veces. Todo el mecanismo es bastante estable, por lo que la impresora de agujas resulta "casi" imperecedera.

Las impresoras matriciales también existen en color. La cinta de tinta ya no es entonces negra sino de tres colores: rojo, azul y amarillo, y las agujas se reparten por los tres colores. Entonces, la desventaja es que el negro se forma a partir de la combinación de los tres colores: la cinta es golpeada por tres agujas en tres lugares muy próximos, pero ligeramente separados, por lo que si se observa con mucha atención se pueden reconocer los tres colores utilizados en lugar de ver un negro puro. Los colores de mezcla se crean por el mismo sistema. Para que la representación de los colores sea siempre la misma, es necesario que las tres bandas de color de la cinta se gasten al mismo ritmo, algo prácticamente imposible, por lo que la única solución es cambiar la cinta con mayor frecuencia de lo habitual.

Existen impresoras de color que además de los tres colores mencionados utilizan el negro como color adicional, y aunque los problemas a la hora de representar los colores son prácticamente los mismos, no resultan tan evidentes. De todos modos, no hay que esperar demasiado de las impresoras matriciales en color y, en definitiva, es altamente recomendable utilizar una impresora de chorro de tinta.

13.3 Impresoras de inyección o chorro de tinta

Las impresoras de inyección de tinta son, desde hace algunos años, el tipo de impresora preferido por los usuarios. Las razones son evidentes: la calidad de impresión, la velocidad, y el precio cada vez más bajo. La calidad de impresión es comparable a la de las impresoras láser, y además tienen la ventaja de que no emiten ozono (gas perjudicial) como siguen haciendo todavía algunas impresoras láser.

Una de las primeras impresoras de chorro de tinta HP DeskJet. Usando un cartucho especial, esta impresora puede utilizarse también para la impresión de fotos digitales

La casa Hewlett Packard, con sus impresoras *ThinkJet* y, especialmente, con los modelos posteriores, las impresoras *DeskJet*, ha contribuido de forma especial a la gran aceptación y popularidad de las impresoras de inyección de tinta. La mayoría de las impresoras de inyección de tinta utilizan exclusivamente papel en hojas sueltas (y no papel continuo), de un tamaño hasta *DIN-A4*, y que se depositan en bloques en su bandeja correspondiente. La impresión de etiquetas, sobres de cartas, papel grueso (casi cartón), también es posible en la mayoría de los casos, pudiendo imprimirse incluso folios.

Las impresoras de inyección de tinta se pueden utilizar con muchas finalidades tanto en oficinas como en los hogares, y más ahora con la popularización de la fotografía digital. Además de las impresoras A4, también existen también impresoras de inyec-

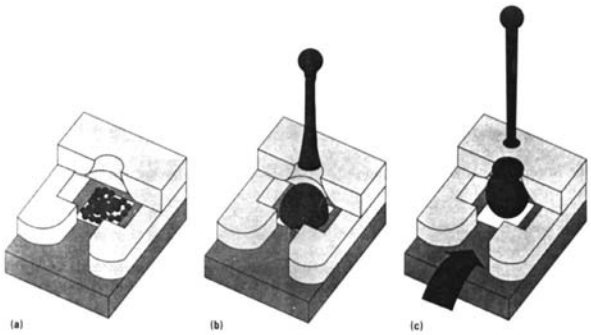
ción de tinta que pueden trabajar con tamaño de papel hasta A0, como por ejemplo los modelos de la serie *DesignJet* de la casa Hewlett Packard. El espectro de impresoras de inyección de tinta existente en el mercado es prácticamente inabarcable, y muchas veces sucede que, cuando por fin uno se ha decidido por un modelo determinado, al irlo a comprar ya ha sido substituido por el modelo siguiente.

La técnica de impresión de las impresoras de inyección de tinta también está basada, esencialmente, en una correlación de matrices que representan los caracteres. Pero los caracteres no se imprimen en el papel con la ayuda de agujas que golpean la cinta, sino a través de pequeñas válvulas que inyectan la tinta directamente sobre el papel. El cabezal de impresión de una impresora de inyección de tinta es casi una maravilla de precisión, pues dispone, según el fabricante y el modelo, de más de 60 válvulas, cada una de ellas con un diámetro aproximadamente de la mitad del de un cabello humano.

Existen diferentes procedimientos para llevar la tinta al papel. Los más conocidos y extendidos son el *bubble jet*, desarrollado por la casa Canon, y el procedimiento piezoeléctrico, que es el que utiliza, por ejemplo, la casa HP en sus impresoras *DeskJet* y la casa Epson.

13.3.1 Principio de funcionamiento: bubble jet

El procedimiento *bubble jet* es, en principio, un procedimiento térmico (por eso algunos fabricantes también lo llaman procedimiento térmico, a pesar de que no tiene nada que ver con los procedimientos térmicos habituales), puesto que cada válvula posee un elemento térmico. Estos elementos se calientan en milésimas de segundo al recibir un impulso eléctrico y la tinta se evapora de forma parcial. De ese modo, en la válvula se forma una pequeña burbuja (*bubble*) que es la tinta que se inyecta desde la válvula al papel. Esta inyección de tinta queda sometida por un breve intervalo de tiempo a una presión que es la que ejerce un efecto de succión sobre la tinta que hay en la válvula, retrayéndola nuevamente. El tamaño de la gota inyectada de tinta se puede controlar por la intensidad de la señal de impulso.



El principio de funcionamiento de una impresora de chorro de tinta bubble jet: a) estado de reposo, b) creación de la burbuja e inyección de la tinta, c) retroacción de la tinta

13.3.2 Funcionamiento impresoras piezoeléctricas

Este procedimiento de impresión utiliza un elemento piezoeléctrico en cada válvula en lugar del elemento térmico. Este elemento piezoeléctrico es un cristal que emite una vibración al ser sometido a una tensión eléctrica. Se trata, por lo tanto, de un procedimiento mecánico. El principio de funcionamiento es el mismo que se utiliza en los "vibradores" utilizados en diferentes electrodomésticos para emitir una señal. En algunos PCs se integran estos emisores de señales en lugar de los altavoces.

Al llegar el impulso eléctrico al elemento piezoeléctrico de la válvula, éste se mueve e inyecta una gota de tinta en el papel. De ese modo, al igual que en el procedimiento *bubble jet*, se crea una presión momentánea que es la que provoca a retroacción de la tinta. La tinta se encuentra en un cartucho de tinta y llega hasta el cabezal piezoeléctrico a través de un conducto. En el procedimiento *bubble jet*, el cartucho de tinta y el cabezal de impresión suelen formar una misma unidad, y cuando se acaba la tinta hay que cambiar el cabezal de impresión completo, a menos que se pueda rellenar.

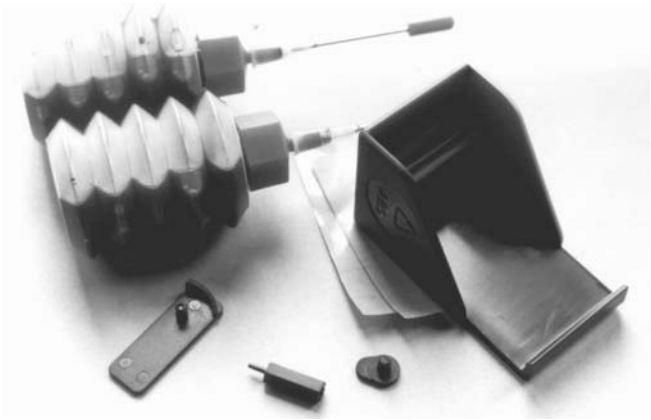
Por lo tanto, no existen grandes diferencias en la calidad de impresión en cuanto al procedimiento utilizado, y éstas se deben más bien a la calidad del papel que se utilice. Por más que los fabricantes no dejen de repetir que con el papel de copia que se adquiere en los comercios ya se consiguen buenos resultados, la práctica demuestra que en la mayoría de los casos esto no es así. Para obtener una calidad óptima hay que utilizar el tipo de papel recomendado por el fabricante de la impresora (papel que no suele ser barato) y que suelen comercializar casi siempre los mismos fabricantes.

13.3.3 Tipos y cartuchos

El número de impresoras de chorro de tinta que se puede encontrar en el mercado es inmenso, y uno de los criterios que se debería tener siempre en cuenta a la hora de decidirse por un modelo son los costes de mantenimiento. Entre estos costes se encuentran, principalmente, los cartuchos de tinta. Los fabricantes de impresoras recomiendan utilizar siempre los cartuchos de la misma marca que la impresora, y en la mayoría de los casos no existen fabricantes que ofrezcan otra alternativa (más económica). Todavía mayor, y más confusa, que la variedad de modelos de impresoras, es la oferta de cartuchos, lo cual no facilita precisamente la elección a la hora de cambiarlos. Determinados cartuchos se pueden utilizar para varios modelos (también para modelos de otros fabricantes), siendo los más utilizados los cartuchos e impresoras de los dos líderes del mercado.

En los comercios existen kits de recambio rellenables para un gran número de impresoras de chorro de tinta, por ejemplo de la casa Pelikan, e incluso para tipos de impresoras en las que el fabricante no ha previsto en un principio la posibilidad de cartuchos rellenables. De acuerdo con la experiencia, un cartucho *bubble Jet* se puede rellenar hasta tres veces antes de que los cabezales de impresión empiecen a tener un efecto negativo sobre la calidad de impresión.

Hay que decir que el relleno de cartuchos no funciona demasiado bien con los cartuchos de color, los cuales suelen ser algo problemáticos en general. Naturalmente, la proporción de color es óptima en un cartucho nuevo, pero los colores nunca se utilizan todos por igual en la práctica, por lo que al final siempre sucede, inevitablemente, que un color se gasta antes que otro. Entonces hay que comprar un cartucho nuevo, a pesar de que el único color que ya no se imprime sea, por ejemplo, el amarillo. El amarillo es, por cierto, un ejemplo típico. Cuando se imprimen imágenes de un escáner o de una cámara de fotos digital, y éstas ofrecen una cierta "tonalidad amarillenta", el amarillo es el color que se gasta primero.



Utilizando los kits de relleno especiales, que se pueden utilizar para diferentes impresoras de inyección de tinta, no es necesario un cartucho nuevo cada vez que se acaba la tinta

Existen modelos en que el cabezal de impresión está integrado ya en el cartucho, lo que significa que, al cambiar el cartucho, se incorporan también nuevos cabezales en la impresora. La ventaja es que, de ese modo, se mantiene siempre el nivel de calidad prometido por el fabricante; la desventaja es, posiblemente, que hay pagar un precio bastante más elevado que si se rellenan los colores o se substituye un cartucho de color sencillo (sin cabezales).

Naturalmente, también hay impresoras (más caras) que tienen cartuchos separados para cada uno de los tres colores básicos: *cyan* (azul claro), *magenta* (violeta claro, casi rosa), *yellow* (amarillo) y *black* (negro). Pero eso no cambia en nada el hecho fundamental de que cuando se gasta un color hay que cambiar también los demás cartuchos, pues de otro modo es imposible conseguir la proporción óptima de mezcla. Si sólo se cambia el amarillo, por ejemplo, éste se verá demasiado fuerte en la impresión en comparación con los otros colores.

13.3.4 Impresión y mantenimiento

Al imprimir imágenes en color, muy a menudo los colores no coinciden con los de la vista previa en pantalla, o con los del documento que acabamos de escanear. Un buen

programa de edición de imágenes debería disponer de una función de calibrado del color, con la que podemos hacer coincidir los colores, más o menos, de los tres dispositivos. El proceso, no obstante, puede ser bastante largo, pues seguramente habrá que hacer varios escaneados y varias impresiones de prueba.

Pero una mala impresión del color también puede ser causada por otro motivo. La representación del color en un monitor se realiza mediante los colores básicos rojo, verde y azul (*RGB*). Un punto de pantalla (*píxel*) está compuesto, a su vez, por estos tres colores que, con diferentes intensidades, crean todos los demás colores. La mezcla de los tres colores con su máxima intensidad da el color blanco, y con su mínima intensidad da el negro. El modelo de color *RGB* funciona, por lo tanto, de forma aditiva, es decir, cuantos más colores se suman, más clara resulta la imagen (que no es la mejor forma cuando se imprime en papel blanco). Por eso, para las impresoras a color se utiliza el modelo *CYMK*: *Cyan*, *Yellow*, *Magenta* y *Black* (que se indica con una K y no una B, para no confundirlo con el color *Blue*, azul). El color blanco (el color del papel) se obtiene no utilizando tinta, y para obtener el negro se mezclan los tres colores base (*cyan*, *yellow* y *magenta*). Este negro se conoce como negro de mezcla, para diferenciarlo del negro puro de un cartucho de tinta negra.

Los cabezales de impresión se deberían controlar con frecuencia, ya que pueden taponarse con facilidad, algo que sucede si la impresora (o un determinado cartucho) no se utiliza demasiado. A menudo, las impresoras tienen un botón especial que sirve para limpiar los cabezales. Si pulsando este botón no se aprecia ninguna mejoría, deberíamos limpiar la superficie del cabezal con un paño húmedo a fin de eliminar las posibles gotas secas de tinta.

Algo que resulta muy ventajoso es que la impresora tenga cartuchos separados para la impresión en blanco y negro y para la impresión en color. De ese modo se consigue un contraste mucho más rico en las gamas de grises, por ejemplo, que si sólo utilizásemos un cartucho de color ya que el color negro se obtiene mezclando otros colores, pero con este sistema casi nunca se consigue la calidad obtenida con un cartucho extra para la impresión en blanco y negro.

Las decepciones con las impresiones a color son a menudo bastante fuertes, ya que el papel se "ondula" considerablemente debido a la gran cantidad de tinta que se absorbe. Este problema puede presentarse con todas las impresoras de inyección de tinta. Por lo tanto, las impresiones a color sólo permiten una calidad aceptable cuando se imprimen letras sueltas, márgenes, o pequeños dibujos formados por superficies de no más de unos centímetros.

Junto a las impresoras de inyección de tinta más económicas, hay otros modelos más caros que disponen de un termocalentador que seca inmediatamente las áreas impresas para evitar que el papel se ondule y para impedir, asimismo, que se extiendan unos colores sobre los otros.

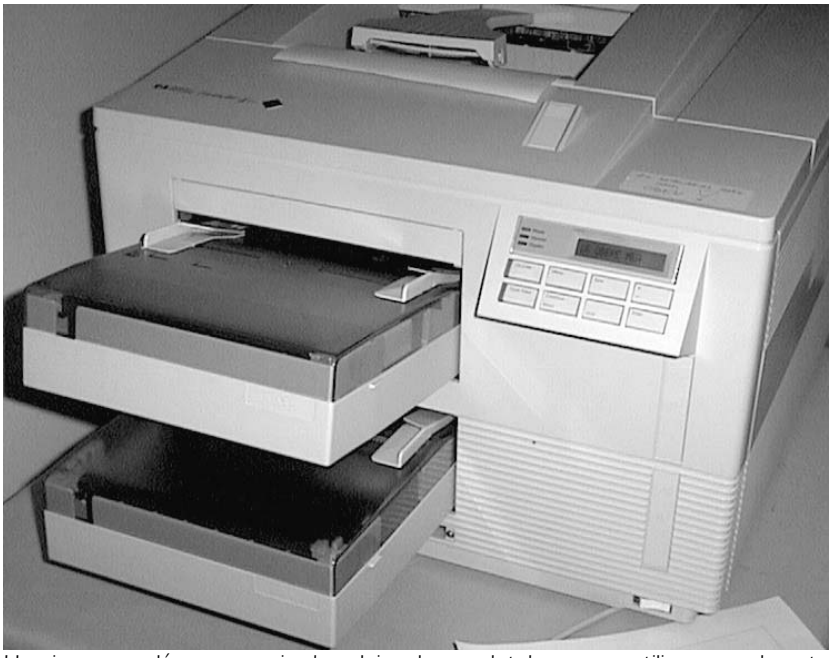
La impresora PaintJet XL300 fue de las primeras impresoras que se encontraban en la categoría de las impresoras de inyección de tinta con mayores prestaciones y que aceptaban un tamaño de papel hasta DIN-A3 y funcionaban con cuatro cartuchos de tinta y una lámpara secadora

Con las impresoras de inyección de tinta más económicas esto sólo se puede conseguir con papel especial de la misma casa fabricante de la impresora (*Glossy Paper*, *Photo Paper*, etc.), hecho exactamente para el tipo de tinta que se utiliza. Este detalle es muy importante, sobre todo, si queremos imprimir fotografías, impresión para la cual también existen cartuchos especiales que suelen trabajar casi siempre con seis colores y gotas de tinta más pequeñas que los cartuchos de tinta estándar. El precio de estos medios es más elevado y esto es algo que deberíamos tener presente al comprar una impresora de inyección de tinta siempre que pretendamos conseguir impresiones a todo color.

13.4 Impresoras láser

Las impresoras láser han alcanzado en los últimos años unas franjas de precios que las hacen interesantes incluso como impresoras de uso privado. En oficinas ya hace tiempo que son las impresoras estándar, y también se utilizan mucho en redes, pues los modelos grandes suelen disponer en la mayoría de los casos, además de un conector *Centronic*, de un puerto de red.

La mayor ventaja de una impresora láser es, ante todo, su alta calidad de impresión. Y si bien su precio sigue estando en la mayoría de los casos por encima de los de las impresoras de inyección de tinta, con un volumen de impresión grande los costes de mantenimiento son mucho menores.



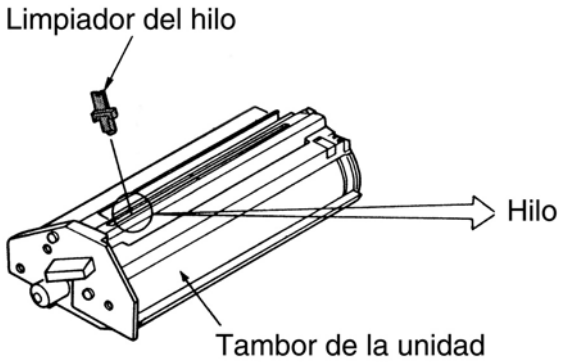
Una impresora láser con varias bandejas de papel, tal como se utiliza normalmente en los grupos de trabajo de red

Principio de funcionamiento

Las impresoras láser funcionan con el mismo principio que las máquinas fotocopiadoras, siendo una de sus partes más importantes el llamado cilindro de impresión. Básicamente, este tambor no es más que un cilindro de cristal con un recubrimiento especial (selenio).

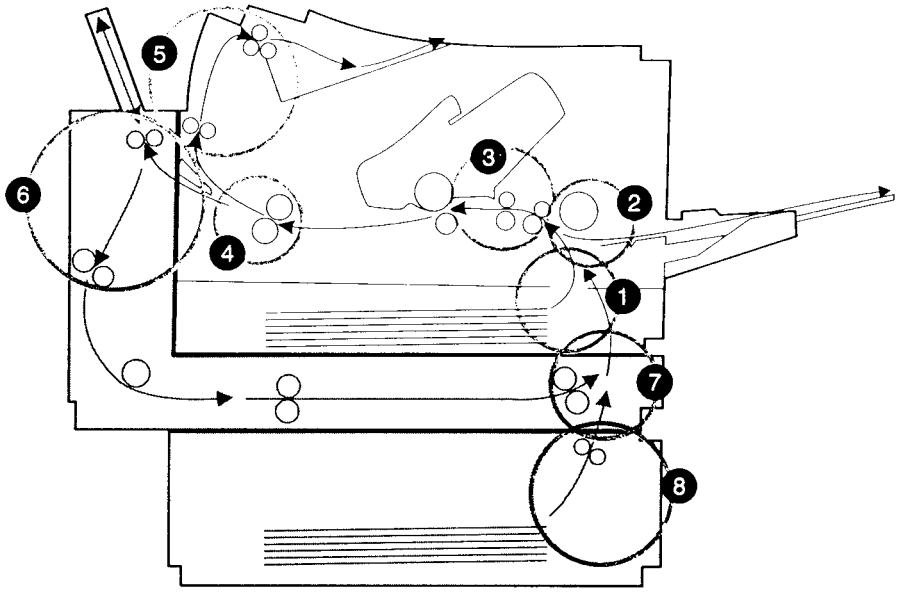
A través de un filamento (llamado muchas veces wolframio o filamento de corona) se carga eléctricamente el cilindro de impresión. Cuando en determinados lugares del tambor incide la luz, y esto es algo que sucede dependiendo del documento que se va a imprimir, se descargan eléctricamente esas partes del tambor. Estas cargas y descargas eléctricas son las responsables de que se produzca gas ozono. Para reducir estas emisiones de ozono, muchas impresoras láser tienen un filtro de ozono formado por carbono activo, que se ha de cambiar aproximadamente cada 40.000 páginas impresas para que mantenga su efecto y un correcto funcionamiento.

Las nuevas impresoras láser no utilizan filamento de corona para cargar y descargar eléctricamente el tambor, sino que trabajan con rodillos de carga incorporados en el cartucho del tóner y el mecanismo de impresión. De ese modo, fuera de la impresora no se produce ninguna emisión de ozono y tampoco es necesario preocuparse por el mantenimiento del filamento de corona. La desventaja es que junto con el cartucho vacío del tóner se debe cambiar también el cilindro de impresión, lo que supone un considerable gasto adicional.



El cilindro de impresión de una impresora láser con el filamento de wolframio, que se debería limpiar, a ser posible, con bastante frecuencia

El tóner, un polvo de impresión que suele ser bastante tóxico, también está cargado eléctricamente y se descarga, o expulsa, allí donde incide la luz en el cilindro de impresión. En las partes en que no incide la luz, el tóner permanece adherido a la superficie del tambor y, al girar éste, se "imprime" en el papel, formando así los caracteres sobre él. El papel pasa después por una unidad de fijación que fija el tóner sobre el papel mediante calor. Si la unidad de fijación es defectuosa, algo que en las fotocopiadoras pasa con frecuencia, los caracteres se pueden borrar fácilmente del papel.



El recorrido del papel en una impresora láser. Explicación de los números: 1, 6, 7, 8 = tracción de las distintas bandejas de papel, 2 = cilindro de impresión, 3 = cartucho de tóner, 4 = unidad de fijación, 5 = salida del papel

Los puntos por los que pasa el papel a través de los diferentes rodillos guía son los más conflictivos, pues es en ellos donde pueden producirse los famosos "atascos" de papel. Durante la impresión, el cilindro de impresión se descarga eléctricamente, quedando libre de tóner, el cual se recoge en un contenedor especial o bien en un cajetín de recogida dentro del tambor del tóner.

13.4.1 La fotoimpresión

La incidencia de la luz en el cilindro de impresión se produce en una fotocopiadora a través de una lámpara. Toda la página queda expuesta, por fases, a la luz. Una impresora láser, por el contrario, recibe primero del PC toda la información necesaria para la impresión e, internamente, con ayuda de un microprocesador propio forma la imagen completa de la página en su propia memoria *RAM*. Las impresoras láser poseen de manera estándar una memoria *RAM* mínima de entre 1 y 4 MB. Las impresoras láser a color, en las que la exposición a la luz se realiza en varias fases (negro, cyan, magenta y amarillo), cuentan frecuentemente con una memoria de 20 MB o más.

El rayo láser "impresiona" el cilindro de impresión a través de un sistema de espejos controlados por un circuito eléctrico. Desde hace algún tiempo, en lugar de este sistema mecánico tan complejo, el rayo láser fue sustituido a veces por diodos lumínicos (*Light Emitting Diodes*, *LED*), de los cuales existen más de 2.000 por línea, y son los que realizan la fotoimpresión del cilindro de impresión línea a línea. Independientemente de que se utilice un rayo láser o diodos lumínicos, se habla de impresora láser siempre que ésta sigue el principio de funcionamiento de una fotocopiadora (con un cilindro de impresión).

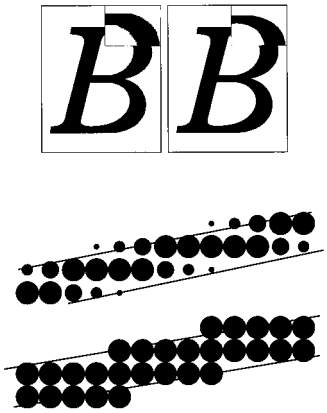
13.4.2 Resolution Enhancement Technology

Suavizando las discontinuidades en líneas inclinadas y curvas de la imagen de impresión, se consigue una mejora importante en la calidad de impresión de caracteres y gráficos. Esto se podría alcanzar aumentando la resolución de la impresora (en dpi, *dots per inch*, puntos por pulgada), pero resulta más sencillo y menos costoso utilizar puntos de impresión de diferentes tamaños.

La casa HP fue la primera en utilizar este procedimiento en sus impresoras láser, que se denomina *Resolution Enhancement Technology* (*RET*). En la actualidad, también se utiliza en las impresoras de inyección de tinta, en las que se necesitan cabezales de impresión diferentes de los de las antiguas impresoras de inyección de tinta (*DeskJet* 500 y similares).

La calidad de la imagen se puede mejorar considerablemente con el nuevo procedimiento *RET*. Una impresora con una resolución de 600 dpi, por ejemplo, y *RET*, ofrece aproximadamente los mismos resultados que una impresora con una resolución de 1000 dpi que trabaja con un tamaño constante de punto. El efecto *RET* se puede configurar en la mayoría de las impresoras (por ejemplo, a partir de la *LaserJet* 4 Plus) desde el PC, a través de un programa de configuración de la impresora, llevando así

a la impresora el contraste que se necesita. En la figura siguiente vemos impreso el carácter "B" sin *RET* (izquierda) y con una configuración media de *RET* (derecha). El dibujo de puntos sirve para ilustrar el principio de funcionamiento del método *RET*.



El sistema RET trabaja con un tamaño variable de puntos, lo que implica menos "escalonamientos" en la imagen impresa y, por lo tanto, contornos más suaves y más continuos

En las impresoras de inyección de tinta a color también se utiliza el procedimiento *Photo-RET*, que funciona como el *RET* estándar, pero puede colocar hasta 16 gotas de tinta en un punto de impresión, proceso en el que también juega un papel importante el *dithering*. Este último hace que los puntos de la impresión de los colores base queden tan juntos que el ojo humano no pueda distinguirlos por separado, percibiéndolos como un solo punto de color (de mezcla).

13.4.3 Tipos y mantenimiento

De las explicaciones anteriores se deriva que una impresora láser es un aparato relativamente complejo, con unidades muy diferenciadas. Independientemente del modelo y del fabricante, en general se puede decir que una impresora láser requiere un mantenimiento más frecuente, o al menos más costoso, que todas las impresoras descritas anteriormente. En todos los modelos nuevos el tóner ya no debe colocarse en la impresora con una botella de plástico, y el cajetín de recogida del tóner ya no debe vaciarse. Cuando falta tóner, simplemente se coloca un cartucho nuevo de tóner, que recogerá igualmente el tóner gastado, y que, en la mayoría de los casos, no manchará las manos de nadie. Esto es sin duda una ventaja, pues el tóner es muy tóxico y de ese modo no entramos en contacto directo con él.

Como ya hemos mencionado, al cambiar el cartucho del tóner, en la mayoría de los casos se cambia también, automáticamente, el cilindro de impresión integrado en él. Por eso, las impresoras antiguas tienen un mantenimiento menos costoso que los nuevos modelos. El tóner pertenece a la categoría de polvo extrafino, por lo que no sólo es peligroso para el hombre sino que podría serlo incluso para un aspirador normal

con el que se intentara limpiar la impresora láser. Por lo tanto, no nunca debemos intentar hacer esto último, ya que el tóner podría penetrar hasta el motor del aspirador y estropearlo. En general, los restos de tóner se deberían eliminar (si es necesario) con un paño ligeramente húmedo, que tiraremos después de haberlo usado, preferentemente en un contenedor de residuos especiales. Si bien los fabricantes afirman que el tóner ya no es tan peligroso como en tiempos pasados, es mejor no arriesgarse a demostrar lo contrario.

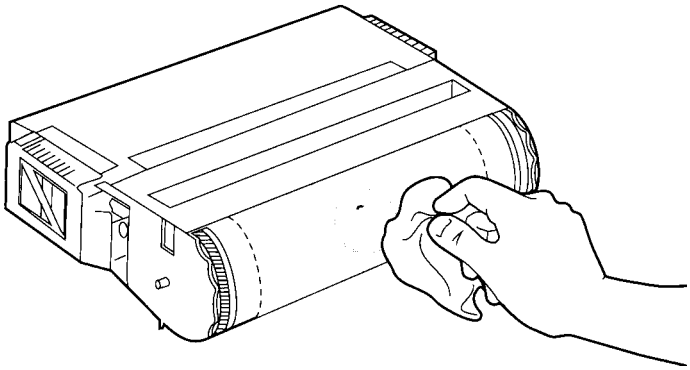
Un cartucho de tóner normal permite la impresión de unas 3.000 a 6.000 páginas, aunque las impresoras más modernas pueden imprimir hasta 20.000 páginas con el mismo cartucho de tóner. La duración del cartucho de tóner no sólo depende del tipo de impresora sino también, y sobre todo, de que ésta se utilice para imprimir básicamente gráficos o sólo texto.

Normalmente, los fabricantes de impresora indican un tiempo de duración del tóner con un grado de variación del 5%, pero en la práctica nunca suele ser así y en la mayoría de los casos tampoco se alcanzan los números indicados de páginas impresas. La unidad de fijación (unidad de revelado) debe substituirse aproximadamente al cambiar el tóner por décima vez (y el cilindro de impresión). Cumpliendo esta norma, se alcanzan valores máximos de entre 30.000 y 80.000 impresiones. Estos datos sólo son valores a modo de orientación, ya que las impresoras láser difieren entre sí en cuanto a su mecanismo, y el número de piezas recambiables también varía de unos modelos a otros.

Los fabricantes deberían retirar gratuitamente los cartuchos de tóner gastados. El cartucho que se ve más abajo, de una impresora HP Laser Jet 6L, tiene un defecto en el cilindro (tóner seco y enganchado)

Con el cuidado y mantenimiento de una impresora láser tampoco deberíamos exagerar, y sólo deberíamos "echar mano al trapo" o cambiar una pieza cuando la impresión presente algunas irregularidades. En la actualidad, las impresoras láser son tan inteligentes*ƒ* que cuando falta tóner o necesitan una limpieza lo indican en pantalla a través de un mensaje de error, y entonces todavía se está más que a tiempo de tomar las medidas oportunas. Muchas veces, antes de cambiar definitivamente el tóner, todavía se puede alargar la vida del cartucho existente retirándolo de la impresora y sacudiéndolo un poco tal como viene descrito en el manual de la impresora láser, porque eso es lo que hay que hacer también al montar la unidad por primera vez.

Uno error que se produce con bastante frecuencia es que en todas las páginas impresas aparece un fallo de impresión en el mismo lugar. A menudo esto es debido a que el cilindro, que puede hallarse en el cartucho del tóner o, según el modelo, en cualquier otro punto del camino que recorre el papel en el interior de la impresora, está sucio o ha sufrido algún daño en su superficie. En ese caso, deberíamos cambiar el rodillo y, eventualmente, el cartucho completo del tóner. Puesto que en un caso así no podemos perder nada, antes de cambiar ninguna pieza deberíamos al menos, intentar limpiar la suciedad, que es una operación que se desarrolla normalmente sin demasiados problemas.



La suciedad en el cilindro se puede eliminar fácilmente en la mayoría de los casos, aunque siempre se debe actuar con mucho cuidado para no dañarlo

El rodillo se puede girar manualmente y la causa del error suele ser casi siempre un poco de tóner seco o un trozo de papel enganchado, que se pueden retirar fácilmente con un paño mojado con isopropanol o, en casos extremos, rascándolo de forma cuidadosa.

13.5 Impresoras láser en color

Los precios de las impresoras láser en color también han bajado mucho en los últimos tiempos, por lo que en la actualidad son una buena alternativa, a veces incluso económica, a las impresoras de inyección de tinta en color de mejor calidad.

Con las impresoras láser no necesitamos papel especial para conseguir una buena calidad de impresión, y se pueden imprimir superficies completas sin perder calidad, los colores no se secan y la proporción de mezcla de los colores se mantiene siempre constante. Si tenemos que imprimir un gran volumen de páginas en color, rápidamente amortizaríamos la compra de una impresora láser en color, ya que, en la mayoría de los casos, podremos utilizar los mismos folios (resistentes al calor) que se utilizan para las máquinas fotocopiadoras, y que son bastante más económicos que los papaeles especiales para las impresoras de inyección de tinta.

La Color LaserJet 5/5M fue una impresora láser en color que, gracias a sus costes de mantenimiento relativamente bajos, podía amortizarse muy rápidamente

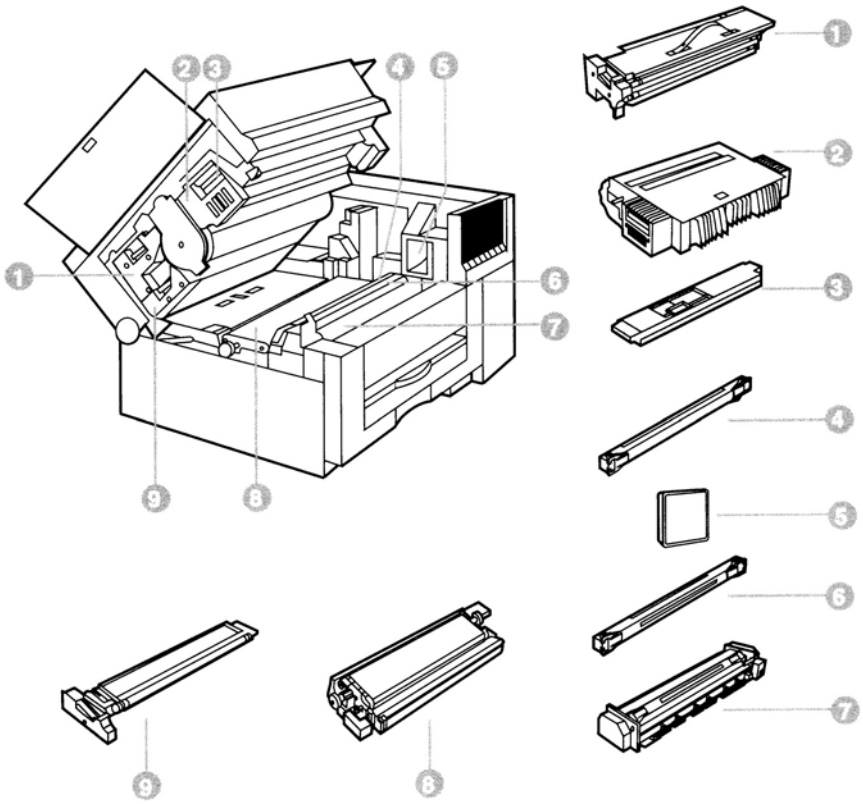
Básicamente, una impresora láser en color funciona igual que una impresora láser en blanco y negro, pero con la diferencia de que hace un recorrido por cada color base (*CMYK*), esto es, cuatro en total, lo que significa que es mucho más lenta que una impresora láser convencional. Además, necesita una memoria interna mayor para construir la imagen previa.

Según el fabricante, se pasa cuatro veces el papel o bien una cinta especial. Esta cinta dibuja la imagen que debe imprimirse y la transfiere de una vez al cilindro de impresión, desde el que pasa finalmente al papel. Normalmente, todas las nuevas impresoras láser a color funcionan con una cinta de este tipo, llamada cinta de transferencia.

Es evidente que los costes de mantenimiento de una impresora láser en color son mayores que los de una impresora láser en blanco y negro. De todos modos, para imprimir con calidad fotográfica, la impresora láser en color no es la solución óptima, siendo las impresoras de inyección de tinta las que ofrecen una mejor calidad.

El número de piezas que hay que ir cambiando en una impresora láser durante su tiempo de duración es bastante mayor que en una impresora de inyección de tinta. La imagen muestra las piezas de una *LaserJet* 5M. Éstas son:

1. Revelador de color
2. Rodillo
3. Cajetín de recogida del tóner
4. Rodillo de limpieza
5. Filtro de ozono
6. Rodillo de recubrimiento
7. Unidad de fijación
8. Cinta de transferencia
9. Revelador de negro



Las unidades recambiables de una impresora láser en color

13.6 Impresoras térmicas

El concepto de impresora térmica tal vez nos lleva a pensar en aparatos de fax, o sistemas de caja, que funcionan con un tipo de papel sensible al calor, que es calentado por un cabezal de presión térmica. Sin embargo, nada tienen que ver con estos aparatos de resolución tan baja y una calidad de impresión regular con tendencia a mala, ya que se trata de impresoras de alto rendimiento, especialmente adecuadas para las impresiones con calidad fotográfica, y que se utilizan sobre todo en las áreas profesionales (como publicidad y estudios fotográficos). Según los dos procedimientos más utilizados, podemos distinguir entre dos tipos de impresoras térmicas:

- Impresora de termotransferencia.
- Impresora de termosublimación.

El principio de funcionamiento de una impresora de termotransferencia está basado en un folio de poliester, recubierto por una fina capa de cera de color. A través de un gran número de elementos térmicos, colocados horizontalmente en línea, se funden los puntos del folio y se transfieren así al papel. En la mayoría de los casos, estas impresoras pueden utilizar folios de termotransferencia monocromos (negros), de tres colores (cian, magenta, amarillo), o incluso de cuatro (cián, magenta, amarillo, negro). Cada folio consta de un segmento para cada color, del tamaño de una página.

Durante la impresión, el mecanismo de la impresora lleva el primer color (por ejemplo cian) hacia delante y se imprimen todos los puntos de impresión que contienen cian. A continuación, se trae el color siguiente (magenta) y el proceso se repite nuevamente, hasta que se han impreso los cuatro colores.

La técnica de termotransferencia proporciona colores totalmente brillantes, con un gran poder de cobertura, y que reflejan la luz y el agua con total realismo. El único inconveniente es que no se pueden gastar los folios en la medida en que se necesitan, pues siempre se aplican todos los colores del folio, incluso cuando no se necesitan todos. Los costes de impresión no dependen del número de colores ni del grado de cobertura, por lo que en definitiva, la impresión resulta tan cara como la de una impresora de inyección de tinta en color. Además del papel con recubrimiento especial, algunas impresoras de termotransferencia pueden utilizar papel normal para impresora láser, y ellas mismas lo recubren automáticamente con una capa especial como la que lleva el papel de termotransferencia.

Las impresoras de **termosublimación** se utilizan principalmente para impresiones de calidad fotográfica (16.7 millones de colores, con realismo fotográfico), y en esta técnica también se utiliza, como en la técnica de termotransferencia, un folio de soporte especial, de los colores cian, magenta, amarillo y negro.

Otro detalle idéntico es que también se disuelven los colores con la ayuda de elementos térmicos. En este caso, sin embargo, la cera se calienta a temperaturas tan altas que pasa a un estado de gas, y es en este estado como se transfiere al papel especial. Esto

es precisamente lo que significa la palabra sublimación: una substancia pasa del estado sólido al de gas, saltándose el estado intermedio de líquido. Y eso es lo que hace la cera: pasar de sólido a gas, dándole así el nombre a este proceso.

Una impresora de sublimación en color que posee un cabezal de impresión formado por 2.550 elementos, y la impresora es capaz de imprimir hojas de papel de hasta un tamaño DIN-A4 plus (222 x 322 mm). La conexión de la impresora se puede realizar a través del puerto SCSI, consiguiéndose así una transferencia de datos muy rápida

La intensidad del color se puede regular controlando la temperatura, y normalmente se pueden conseguir hasta 256 grados de intensidad. Cuanta más alta es la temperatura, con mayor intensidad se imprime el color. Las impresoras de termosublimación alcanzan una resolución real en color de como mínimo 300x300 dpi, ya que los tonos intermedios se consiguen fundiendo los colores en el papel, y no uniendo diferentes puntos de color.

La calidad incomparable de estas impresoras va unida, lamentablemente, a unos precios comparativamente muy altos. Al igual que en el proceso de termotransferencia, los colores de los folios no se va gastando, más o menos, en función de las imágenes impresas, sino que siempre se aplican todos los colores por igual. Además de estos dos procesos térmicos, también existen otros que funcionan siguiendo el principio de las impresoras de inyección de tinta, como por ejemplo las impresoras *Lexmark*. Estos procesos no tienen nada que ver con los dos que acabamos de describir, y son más bien variantes de los procesos *bubble jet* y piezoeléctrico.

13.7 Configuración

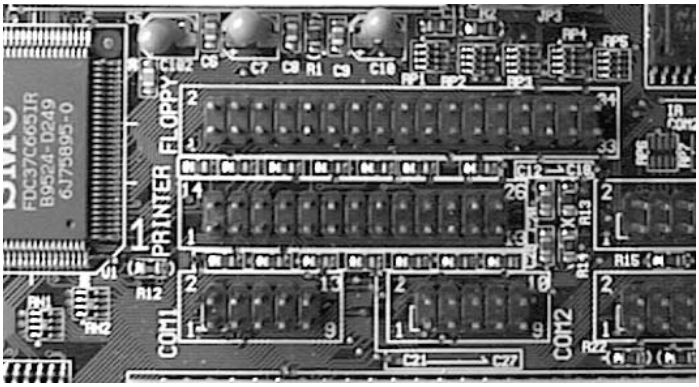
Una condición previa para poder utilizar la impresora es que funcione el puerto en el que la vamos a conectar (en *Windows* esto se comprueba en el *Panel de control*). Al instalar los programas correspondientes a la impresora, a veces también debemos instalar un controlador (actualizado) para la unidad correspondiente de *Windows* (por ejemplo *IrDA*, USB) que vaya a utilizar la impresora.

Lamentablemente, con esta operación puede ser que se substituya un controlador ya instalado y que funcione perfectamente, por otro "no tan bueno". Esta circunstancia, y el hecho de que los controladores de impresora a veces no están realmente muy bien programados (por lo que la instalación y configuración de las impresoras de red, sobre todo, puede convertirse en una tortura), no los tendremos ahora en cuenta, ya que la cantidad de detalles que pueden provocar la confusión e impaciencia del usuario durante la instalación es tan amplia como el número de tipos de impresoras existentes. Por lo tanto, en este punto nos limitaremos a revisar a fondo el funcionamiento de la impresora conectada al puerto paralelo (*LPT*).

El puerto paralelo se nos muestra durante el arranque del PC. En caso de no ser así, es recomendable hacer una revisión del programa de instalación del *BIOS*. El cable de la impresora merece especial atención (longitud, buen funcionamiento), ya que los cables muy económicos, con conectores de plástico, sin que los hilos internos estén completamente enrollados y sin la protección suficiente, han creado más de un problema a la hora de realizar las impresiones.

En el conector situado en la impresora hay dos pinzas de sujeción que sirven para fijar el enchufe del cable de la impresora y evitar el peligro de un falso contacto. Pero las pinzas no son una garantía suficiente de que el enchufe no quede suelto. Por eso, lo mejor es apretar también los tornillos que llevan los conectores de 25 polos del PC. En caso de producirse problemas de comunicación, estos dos conectores son los primeros que debemos comprobar.

Si surge algún problema, también deberíamos comprobar el cable de conexión que va del PC a la placa base, y que la tarjeta del puerto esté correctamente fijada en su lugar. En la mayoría de los casos, estos cables no tienen una forma que impida invertir los polos, por lo que es posible conectarlos con los extremos cambiados. En las placas base *ATX,* los puertos y los conectores se encuentran directamente en la placa base, por lo que la causa de los posibles problemas no puede ser aquí una inversión de cables interna.



En muchas placas base, los contactos de conexión de los puertos no tienen una forma que impida invertir los polos

Una prueba muy sencilla y que no precisa instalar ningún controlador especial para comprobar si funciona la impresora conectada, se puede realizar desde *DOS* con los comandos siguientes:

```
Copy AUTOEXEC.BAT PRN.
Copy AUTOEXEC.BAT LPT1.
Pulsar la tecla <IMPR>.
```

Si la impresora no reacciona ante ninguno de estos comandos, deberemos realizar un *autotest* de la impresora, para lo cual casi todas las impresoras disponen de un botón especial que activa el *autotest*. El autotest se puede activar también mediante una opción de menú en la impresora.

14. Módems

Los módems tuvieron un papel esencial en el desarrollo y popularización de *Internet*, ya que son aparatos relativamente baratos y que permiten navegar por Internet a los que tengan un equipo y una línea telefónica sólo pagando una conexión local. Si no hubiese sido por ellos, *Internet* jamás se hubiese hecho lo popular que es hoy. Sin embargo, actualmente vemos que los módems ya han llegado al tope de lo que tenían que ofrecer.

El módem nos permite la conexión a Internet

Los módems son lentos comparados con otras formas de acceso a la red de redes, y no permiten quedarse conectado mucho tiempo, debido al precio de las llamadas telefónicas y al hecho de que la línea queda ocupada. Eso sin tener en cuenta la velocidad. Descargar archivos grandes a través de un módem puede llegar a ser una lenta tortura. Para alguien que utilice *Internet* de forma esporádica puede ser suficiente, pero para los navegadores más expertos es realmente muy lento. Actualmente han surgido varias opciones de acceso, algunas ofreciendo sólo descargas más rápidas, mientras que otras también ofrecen la realización del sueño de poder quedarse conectado las 24 horas del día sólo pagando una cuota mensual. Actualmente tenemos disponibles las líneas *RDSI* (*ISDN*) y *ADSL*, acceso vía cable y acceso vía satélite. Claro que estas tecnologías aún no están disponibles para todo el mundo, pero dentro de poco tiempo es muy probable que estemos navegando por *Internet* a través de uno de estos nuevos medios de acceso a la red de redes.

14.1 Los módems de 56K

Las líneas telefónicas fueron desarrolladas para servir de medio de comunicación a través de la voz. Siendo así, no es preciso siquiera una buena calidad de sonido, sólo que sea nítido. La baja calidad de las transmisiones a través de la línea telefónica limita la velocidad de transmisión de datos de los módems a sólo 33 o 56 Kbps.

En el sistema telefónico actual se utilizan cables de cobre sin blindaje para la transmisión de las señales de voz, que son analógicas. Esto no llega a ser una gran limitación, pues utilizando cables de cobre comunes es posible transmitir datos a hasta 6 Mbps entre distancias de varios kilómetros a través de tecnologías como la *ADSL*. Por increíble que pueda parecer, la mayor limitación de la velocidad de transmisión de datos a través de la línea telefónica no son los cables de cobre, sino la comunicación digital entre las centrales telefónicas.

En el sistema telefónico actual, todas las líneas de una determinada zona están conectadas con una central telefónica. En una gran ciudad existen varias centrales, cada una de ellas englobando las líneas de una determinada zona. Por ejemplo, si queremos llamar a un amigo que vive al otro lado de la misma ciudad, la señal de entrada será transmitida de nuestro aparato telefónico a la central de nuestro barrio, la central entrará en contacto con la central del barrio de nuestro amigo, y ésta encaminará la señal hasta su línea telefónica.

Entre nuestra casa y nuestra central, la señal se transmite a través del hilo de cobre en forma de una señal analógica que utiliza frecuencias de entre los 400 y 3400 Hz. Al llegar a la central, la señal se transforma en una señal digital con un muestreo de 8 bits a 8 KHz, que ocupa 64 Kbps del ancho de banda. La señal permanece en su formato digital hasta llegar a la otra central, donde es nuevamente transformada a una señal analógica y enviada a nuestro amigo. Cuando él habla por el teléfono ocurre el camino inverso, la señal es transmitida desde su aparato hasta su central, convertida a señal digital, transmitida a nuestra central, convertida en analógica y enviada hasta nuestro teléfono.

Una conclusión lógica es que la tasa máxima de transferencia de datos posible usando una línea telefónica común sería de 64 Kbps, ya que se trad°ta de la velocidad de comunicación entre las centrales. El problema es que cada vez que una señal analógica es convertida en una señal digital tenemos una pérdida de calidad, generada por el "ruido". La teoría más aceptada sobre este tema es la ley de Shannon, según la cual, la conversión analógico/digital hecha en la central A limita la transferencia de datos a través de la línea telefónica a sólo 35 Kbps. El ruido sólo surge cuando una señal analógica es convertida a una señal digital, y no a la inversa.

Durante mucho tiempo se creyó que era realmente imposible transmitir datos a más de 35 Kbps a través de las líneas telefónicas y que, por eso, los módems de 33.6 eran el tope. Entonces, alguien tuvo la idea de eliminar la conversión analógica a digital en el sentido proveedor/usuario, acabando con el ruido y permitiendo tasas de transfe-

rencia más altas en este sentido. Fue posible implementar esta idea estableciendo una conexión digital entre el proveedor y su central telefónica.

En este caso, cuando son enviados datos del proveedor al usuario no es necesaria la conversión analógico/digital, ya que las señales ya llegan a la central en formato digital. No tenemos entonces la aparición del ruido, permitiendo tasas de transferencias más altas.

Como el usuario continúa teniendo una conexión analógica con la central telefónica, continúa siendo necesaria la conversión analógico/digital en el sentido usuario/proveedor, generando el famoso ruido y limitando las transferencias a 33.6 Kbps en este sentido de la comunicación. Tenemos entonces una tasa de descarga (al bajar archivos) de 56 Kbps, pero con una tasa de subida (al transmitir archivos) de sólo 33.6 Kbps.

En la práctica, la velocidad máxima de los módems de 56K se queda en 53 Kbps, pues las centrales telefónicas no consiguen trabajar con tasas más altas. Pero, incluso para conseguir conexiones a 53 Kbps es preciso que su central telefónica sea digital (para saber si la central es digital basta ver si acepta la marcación por tonos) y que la calidad de la línea sea buena. De lo contrario, las conexiones van a quedarse en sólo 44 o 48 Kbps, o no van a pasar incluso de 33.6.

14.1.1 Compresión de datos

A fin de aumentar la cantidad de datos transmitidos, manteniendo la misma velocidad de la conexión, los módems utilizan unos algoritmos de compresión de datos. Esta compresión tiene un funcionamiento idéntico al de la compresión de datos que se realiza en los discos duros, con la única diferencia de que cada paquete de datos se comprime de forma individual antes de ser enviado y se descomprime al ser recibido.

Comprimiendo los datos conseguimos aumentar bastante la velocidad de transmisión de datos. Por ejemplo, en la transmisión de un archivo de texto obtenemos una compresión de 2/1, lo que permite doblar la velocidad de la transmisión.

> Nota: debemos tener en cuenta que la compresión hecha por los módems no funciona en los archivos que ya están comprimidos, como las imágenes *JPG*, *GIF* o los archivos en formato *ZIP*.

14.1.2 Actualización del módem

A pesar de que los protocolos de comunicación *V.34* y *V.90* demandan equipos un poco diferentes, muchos módems de 28.800 y 33.600 Kbps poseen todo el hardware necesario para utilizar el protocolo *V.90* y transmitir datos a 56 Kbps. En estos casos, todo lo que necesitamos hacer para convertirlos en auténticos módems *V.90* es alterar su *firmware*, incluyendo el soporte para el nuevo protocolo.

Muchos de estos módems utilizan memoria *flash* para almacenar su *firmware*, permitiendo su actualización vía software. Algunos fabricantes como las casas US Robotics

y Motorola ofrecen actualizaciones gratuitas para sus módems bajando el programa de actualización vía *Internet*. Otros módems tienen el *firmware* grabado en memorias *EPROM*. En este caso, para hacer la actualización es preciso cambiar el *chip* que almacena el *firmware* del módem por otro. Generalmente, los fabricantes también ofrecen la posibilidad de actualizar estos módems, aunque sin embargo, cobrándose una tasa de servicio.

Puede parecer extraño que solamente ejecutando un programa ofrecido por el fabricante podamos aumentar la velocidad de un módem, pero en realidad es algo muy simple. El *V.90* es un protocolo más eficiente que el antiguo *V.34*, permitiendo transmitir más datos a través del mismo medio de comunicación (la línea telefónica). Así, si nuestro módem posee la capacidad para entender este nuevo protocolo, basta comunicárselo a través de una actualización para que se convierta en un auténtico módem de 56K.

Durante el proceso de actualización, los datos de la memoria *flash* son borrados en un primer paso, para después empezar la grabación de los nuevos datos. Este proceso tarda algunos segundos. Si se produjese cualquier interrupción, ya sea por uno pico de tensión que reinicie el equipo, falta de electricidad, bloqueos en general, etc. la regrabación se interrumpiría y sin su software de bajo nivel el módem dejaría de funcionar, impidiendo una nueva tentativa. Por esto, es muy importante no dejar de leer atentamente las instrucciones del programa de grabación y asegurarse de haber bajado los archivos correctos.

14.1.3 La corrección de errores

Al tratarse de un medio analógico, las líneas telefónicas son muy sensibles a las interferencias, que podemos notar incluso cuando conversamos con un amigo. Esto, aparte de otros problemas, como las líneas cruzadas, los chasquidos en la línea, etc. que son generados por problemas con la línea o con la propia central telefónica.

Conversando por teléfono, los pequeños ruidos no llegan a entorpecer, pero incluso los menores ruidos son capaces de distorsionar los datos enviados por el módem, ya que una pequeña alteración en el tono de una portadora puede alterar completamente su contenido. De la misma manera que usamos códigos de corrección de errores en los discos duros, disquetes, CDs y hasta en la memoria *RAM* para aumentar su rendimiento, es imprescindible el uso de la corrección de errores en las comunicaciones que realizamos a través del módem.

Los primeros protocolos de corrección de error eran implementados vía software, ya que en aquella época los módems no ofrecían ningún recurso de corrección vía hardware. Actualmente, el protocolo usado es el *V.90* y el *V.92*, que es implementado vía hardware, dejando libre al procesador para que pueda ejecutar otras tareas.

La corrección de errores vía módem funciona agrupando los datos enviados en pequeños bloques, añadiéndoles pequeños bloques de código de detección de errores, lla-

mados *CRC* (*Cyclical Redundancy Check*). En el caso del protocolo *V.90*, los códigos usados también permiten corregir algunos errores simples. El módem receptor, por su parte, verifica los bloques recibidos usando el bloque *CRC*. Si percibe que los datos llegan alterados, se envía una señal al módem transmisor solicitando el reenvío del bloque de datos

14.1.4 Los protocolos

Para enviar los datos a través de la línea telefónica, el módem los transforma en señales analógicas. Los primeros módems, que eran capaces de transmitir sólo 300 bits por segundo, utilizaban una técnica muy simple de modulación, trabajando con sólo dos señales distintas. Así, un tono alto representaba un bit 1 y un tono más bajo representaba un bit 0. Para poder conseguir las mayores velocidades de comunicación, los módems actuales trabajan con un número mayor de señales, cada uno representando un conjunto de bits, a los que son sumados códigos de corrección de errores y compresión de datos. A este conjunto de códigos le damos el nombre de protocolo.

Actualmente los protocolos más usados son el *V.34*, usado por los módems de 33.6K y el *V.90* utilizado por los módems de 56K. El *V.90* es justamente una unificación de los antiguos protocolos *X2* y *56KFlex*, incompatibles entre sí y que tanto dolor de cabeza dieron a los usuarios y proveedores de acceso. Los principales protocolos utilizados han sido los siguientes:

V.22: este fue el primer protocolo desarrollado para módems. El *V.22* utilizaba una modulación bastante primitiva, permitiendo transmitir a sólo 1.200 bps.

V.22bis: un poco más avanzado que el *V.22* original, el protocolo *V.22bis* permitía transmitir a 2.400 bps. No es preciso decir que fue considerado una gran revolución en la época.

V.29: este fue el primer protocolo para los módems de 9.600 bps, pero fue inmediatamente sustituido por *el V.32*, considerado más seguro.

V.32: este es el protocolo patrón para los módems de 9.600 bps.

V.32bis: una evolución del *V.32*, este es el protocolo usado por los módems de 14.400 bps.

V.34: este protocolo dobló la tasa de transmisión de los módems, alcanzando los 28.800 bps.

V.34+: este es el protocolo para los módems de 33.600 bps. Fue originalmente desarrollado por la casa US Robotics y después licenciado para otros fabricantes. Durante mucho tiempo se creyó que el protocolo *V.34+* sería el máximo en la evolución de los módems, pues una teoría, llamada ley de Shannon, decía que era imposible que los módems operaran a más de 35 Kbps.

Los módems de 56K surgieron a partir de la idea de eliminar la conversión analógico/digital en el sentido proveedor a usuario. Con esto, las bajadas pasaron a ser de hasta 56K, pero la subida continuó limitada a 33.6 Kbps.

V.90: en la época de la definición del patrón para los módems de 56K surgieron dos patrones incompatibles, el *X2* de la casa US Robotics y el *56Kflex* desarrollado por las casas Motorola y Lucent. Ambos permitían conexiones a 56 Kbps y tenían un rendimiento muy parecido, pero dieron muchos dolores de cabeza a los proveedores de acceso (que necesitaban mantener módems de los dos patrones) y a los usuarios que necesitaban escoger entre uno u otro. Felizmente, los dos patrones convergieron en el protocolo *V.90*, que es el patrón actual para los módems de 56K.

El protocolo *V.90* ya tuvo su actualización, que la mayoría cree ser la última, el *V.92*, que mantiene la misma tasa de bajada de 56 Kbps, pero amplía la tasa de subida de 33.6 hasta los 44 Kbps. Además de eso, el tiempo necesario para establecer la conexión es menor en los módems *V.92*, cerca de 10 segundos, contra los 20 segundos de los módems *V.90*. Otro avance es la compatibilidad con el recurso de espera ofrecido por algunas operadoras de servicios telefónicos. Es posible dejar la conexión vía módem "en espera" al recibir una llamada y restablecerla al colgar el teléfono.

14.1.5 Softmodems vs hardmodems

Actualmente, se pueden encontrar dos tipos bien diferentes de módems en el mercado: *hardmodems* y *softmodems*. Los *softmodems* también son conocidos como *Winmodems*, módems *HCF*, *HSP* o *HSF*.

Los *hardmodems* son los módems completos que ejecutan todas las funciones de envío y aceptación de datos, corrección de errores, control del flujo, etc. Por otra parte, los *softmodems* funcionan sólo como una interfaz de conexión con la línea telefónica, todas las demás tareas son ejecutadas por el procesador principal, lo que disminuye el rendimiento global del equipo. Al estar compuestos por menos componentes, los *softmodems* son más baratos que los *hardmodems*.

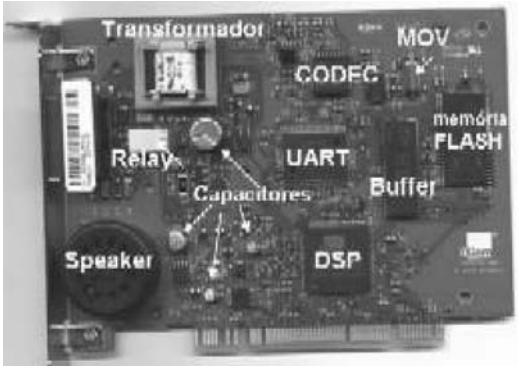
14.1.6 Los componentes

Un *hardmodem* posee todos los componentes necesarios para su funcionamiento. Por eso puede funcionar (salvo limitaciones relacionadas con el *Plug & Play*) en cualquier PC y en cualquier sistema operativo.

> Nota: si el módem utiliza un *slot ISA* de 8 bits y puede configurarse a través de *jumpers*, podremos utilizarlo hasta en un *XT*.

Por otra parte, un *softmodem* depende enteramente del trabajo del procesador. Para que el módem funcione es necesario instalar el programa que lo acompaña, que coordinará todas sus funciones. Como el programa necesita ser reescrito para que pueda ser usado en varios sistemas operativos distintos, en general sólo funcionará en *Windows 95/98/2000/XP*.

En la ilustración siguiente, tenemos un diagrama que muestra los componentes que componen un *hardmodem*, en nuestro caso un US Robotics de 56K modelo *PCI*.



Un hardmodem PCI

UART: se trata del circuito que coordina el envío y recibimiento de datos a través del puerto serie.

Buffer: almacena los datos recibidos, permitiendo la transmisión sólo cuando el procesador esta libre (evitando cualquier pérdida de rendimiento).

DSP: un procesador relativamente poderoso, de 92 MHz, que coordina el funcionamiento del módem y ejecuta las funciones de corrección de errores.

CODEC: transforma las señales digitales en las señales analógicas que se van a transmitir a través de la línea telefónica y hace la decodificación de las señales recibidas.

Memoria Flash: almacena el *firmware* del módem.

Relay: conecta físicamente el módem a la línea telefónica.

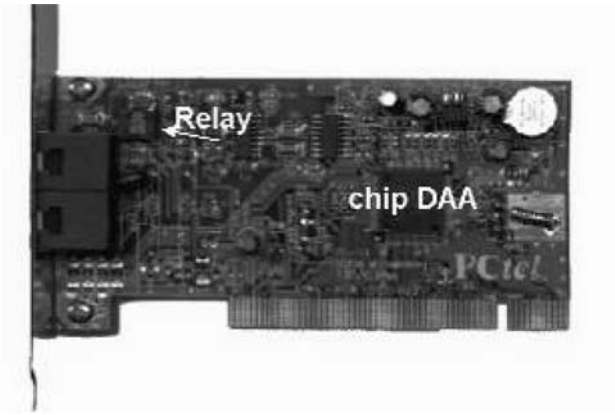
Transformador: aísla el ordenador de la línea telefónica, impidiendo que cualquier cambio de voltaje de la línea pueda estropear el módem.

MOV: actúa como un fusible, sirviendo como protección adicional contra las posibles variaciones del voltaje.

Speaker: emite los sonidos que el módem hace al conectar.

Capacitores: evitan que los fallos momentáneos en el suministro de electricidad entorpezcan el funcionamiento del módem.

En un *softmodem*, el *UART*, el *buffer*, el *DSP*, el *CODEC* y la memoria *flash* son sustituidos por un único *chip*, llamado *DAA*, que actúa como una interfaz entre la línea telefónica y el procesador principal, que será el encargado de realizar el trabajo de los demás componentes. Veamos la imagen de un softmodem, en este caso, un US Robotics *Winmodem*, de 56K.



El softmodem con sus componentes

Entre los *softmodems* existen buenos y malos productos. Existen módems que sólo poseen el *DAA* y el *Relay*, los dos componentes necesarios para que el módem funcione. Podemos observar que no tiene ni transformador, ni *MOV*, lo que significa que cualquier variación en la tensión de la línea telefónica alcanzará al resto del equipo. Esto es bastante peligroso, porque si cae un rayo próximo a los hilos telefónicos y la corriente pasa a través de los hilos hasta el módem, el *capacitor* junto con el *MOV* bloquean la corriente; en bastantes ocasiones el módem se quema, pero el resto del PC queda protegido. Sin esta pareja, la corriente pasaría directamente por el módem, pudiendo dañar la placa base, los módulos de memoria, el procesador o la tarjeta gráfica, por ejemplo.

14.1.7 Rendimiento

Es ineludible que el equipo trabaje más lentamente si se utiliza un *softmodem*. Si disponemos de un equipo a partir de un *Pentium II 266* y sólo usamos *Internet* para navegar y el correo electrónico, no notaremos mucha diferencia entre un *softmodem* (de buena calidad) y un *hardmodem*. Sin embargo, si nos gustan los juegos *online* o disponemos de un procesador más simple, probablemente un *hardmodem* sería la mejor elección.

Entre los *softmodems* también existen grandes diferencias de calidad, que se reflejan en el nivel de utilización del procesador y en los tiempos de respuesta (*pings*). Existen dos motivos para eso, diferencias de calidad en los c*hips DAA* y del *Relay*, y en la calidad de los *drivers* emuladores.

Desarrollar el proyecto de un *softmodem* es relativamente simple, prácticamente sólo en necesario definir qué componentes se usaran, con base al precio y a la calidad que se pretenden conseguir con el producto. Sin embargo, desarrollar los *drivers* demanda mucho más tiempo y dinero. Como el *driver* es el encargado de controlar todo el funcionamiento del *softmodem*, se tratan de programas complejos y caros de desarro-

llar. Cuanto mejores sean los *drivers* mejor será el rendimiento del módem, pero más caro le costará al fabricante.

En cuanto a la configuración del equipo, recomendaríamos un *Pentium 166* como mínimo y un *Pentium II 266* como el ideal mínimo. Si tuviéramos un procesador muy débil, el *softmodem* no tendría como obtener el procesamiento que necesita mientras se estuvieran ejecutados otros programas, haciendo que las transferencias fuesen más lentas y frecuentasen las caídas de línea, sin hablar de la lentitud general del sistema.

14.1.8 Instalación

El proceso de instalación de los *softmodems* no es muy diferente al de una tarjeta gráfica o de sonido. El nuevo *hardware* será detectado de forma automática y bastará con suministrar los *drivers* adecuados.

En muchos casos se exige un *Pentium* como requisito mínimo. Se trata de un hecho muy natural, ya que de cualquier forma el módem no funcionaría en un *486*. Sin embargo, existen casos de errores de instalación en equipos con procesadores *K6* o *Cyrix* ejecutando *Windows 95*. Esto es porque *Windows 95* posee un pequeño *bug* (que puede ser corregido a través de un *patch* disponible en *Internet*) que hace que el sistema reconozca a estos procesadores como si fuesen *486*. Eso no afecta al rendimiento, es sólo un error de identificación. Sin embargo, como el programa se basa en las informaciones suministradas por *Windows*, en algunos casos acaba por abortar la instalación diciendo que el procesador no cumple los requisitos mínimos.

14.1.9 A simple vista

Sólo observando el módem y viendo si existen o no el *UART*, el *DSP* y los otros componentes que hemos visto antes, es posible ver fácilmente si nos encontramos con un hardmodem o con un *softmodem*.

Otro indicador es la necesidad de instalar algún programa para que el módem funcione. Un módem tradicional sólo necesita que sea instalado un archivo *INF* (un archivo de texto con las configuraciones y especificaciones del módem) para funcionar. Por su parte, un *softmodem* necesitará que se instale un programa emulador.

14.1.10 Mantenimiento

Cualquiera que utilice *Internet* desde hace el suficiente tiempo ya debe haber perdido por lo menos uno o dos módems durante alguna tormenta. Estos accidentes son extremadamente comunes pues las líneas telefónicas, al ser metálicas, acaban sirviendo como verdaderos pararrayos. Los módems son componentes relativamente frágiles, que se queman con facilidad al recibir una de estas descargas.

Existen dos posibilidades para este caso. Si la descarga es débil, sólo se quemará el *Relay*, el componente del módem responsable de ocupar la línea. En este caso, el módem continuará siendo reconocido e instalado normalmente por *Windows*, aunque

sin embargo, al marcar nos dirá que no existe tono de marcado, justamente debido a la quema del *Relay*. En este caso, la reparación es relativamente simple y sólo necesitaremos sustituir el componente quemado por otro. Para quien tiene experiencia con las soldaduras es bien fácil.

Si el módem recibe una descarga más fuerte, además del *Relay* se quemarán (en este orden) el transformador (si el módem tiene uno) y el *MOV*. Si, por casualidad, la descarga llegar a alcanzar el *chip DAA* o el *DSP*, el módem no será ni reconocido por *Windows*, y quedará completamente muerto e inservible. En este caso no existe reparación posible, pues no hay forma de soldar estos circuitos de forma manual.

14.1.11 Enviar un fax

Además de transmitir datos, prácticamente todos los módems son capaces de transmitir y recibir faxes, haciendo que el equipo pueda sustituir el aparato de fax, inclusive con algunas ventajas. Justamente por esta doble función, llamamos a estos módems como *Fax-Modems*.

Para transmitir y recibir los faxes a través del equipo sólo necesitaremos tener el *fax módem* correctamente instalado y un programa que permita enviar y recibir faxes. *Windows 95* ya incluía el programa *Microsoft Fax*, pero podemos encontrar incontables programas mejores en *Internet*. La mayoría de los *fax-modems* ya llevan buenos programas de fax.

14.2 ADSL

La *ADSL* o "*Assimetric Digital Subscriber Line*" ya es una tecnología muy adoptada en todo el mundo como medio de acceso rápido a *Internet*. En España existe una gran competencia entre los distintos proveedores de acceso a *Internet*: Telefónica, Jazztel, Wanadoo, etc.

La gran ventaja de la *ADSL* es que permite navegar por *Internet* al mismo tiempo que la línea de teléfono queda libre para las conversaciones de voz, aunque usa el mismo hilo telefónico para las dos funciones.

Acontece que las llamadas de voz sólo utilizan frecuencias bajas, entre los 300 y 3400 Hz, desperdiciando todas las demás frecuencias que podrían ser transportadas a través del cable telefónico. La *ADSL* consiste entonces en instalar dos módems *ADSL*, uno en casa del usuario y otro en la central telefónica, que se comunican entre sí utilizando las frecuenciasque se encuentran por encima de los 5000 Hz, sin interferir en las llamadas normales de voz.

El cable telefónico sólo es usado como medio de comunicación entre los dos módems *ADSL*, sólo para permitir la comunicación de nuestro módem con el módem de la central. Justamente por esto que sólo se paga una cuota mensual y no se paga por pulsos telefónicos. La señal va entonces a un *router*, para el proveedor de acceso y ensegui-

da para *Internet*. Por eso, es por lo que continúa siendo necesario pagar por un proveedor de acceso. Como la comunicación entre los dos módems es continua, basta encender el equipo y el módem para estar conectado a la red de redes.

La *ADSL* permite velocidades de hasta 8 Megabits. Sin embargo, la velocidad de acceso queda limitada a la del plan de precios que escojamos: mientras más rápido, más caro. Por ejemplo, Telefónica cobra casi 40 euros por el servicio de hasta 2 Megabits.

Como el mismo nombre ya sugiere, una de las desventajas de la *ADSL* es el hecho de que el servicio es asimétrico. Los 512 Kbps, o más, son sólo de bajada. La subida de datos queda limitada a sólo 128 Kbps y 512 Kbps o 256 Kbps en el plan de los 2 Megabits.

14.2.1 Las limitaciones de la ADSL

El acceso *ADSL*, ofrecido por Telefónica y los demás proveedores, tiene varias ventajas sobre el ultrapasado acceso de marcado vía módem: es más rápido, ofrece una conexión ininterrumpida con el pago de sólo una cuota mensual fija, etc., pero como nada es perfecto, el servicio también tiene sus limitaciones y el objetivo de este apartado es hablar un poco de ellas, explicar el por qué de cada una y ver cómo podemos saltarnos algunas de ellas.

Inestabilidad

La mayor reclamación de los usuarios de *ADSL* es la inestabilidad de la conexión, que en muchos casos cae constantemente, muchas veces obligando al usuario a reiniciar el módem para que vuelva la conexión. En algunos casos, las caídas son esporádicas y no llegan a incomodar, pero existen casos en los que la línea se cae frecuentemente.

En este caso, el problema es el tramo de hilo entre la casa del usuario y la central telefónica. En el acceso vía *ADSL* tenemos dos módems *ADSL*, uno en la central y otro en la casa del usuario, que mantiene una conexión continúa usando el cable telefónico. El gran problema es que en muchos casos el cable está deteriorado y a pesar de no ofrecer problemas en las conexiones de voz, compromete la velocidad de la comunicación entre los módems y puede hacer que la conexión se caiga a menudo, debido a la inestabilidad de la señal.

Cómo resolver este problema? Normalmente existen varios cables disponibles en cada central, cables sobrantes, que pueden ser usados en sustitución del cable defectuoso. Existen también derivaciones en los cables que se pueden eliminar para poder conseguir una mejora en la señal.

Puertos bloqueados

Otra limitación son los puertos *TCP* bloqueados por el proveedor de la línea telefónica que impiden "conexiones entrantes", o sea, intentan impedir que montemos un servidor de *FTP* o transformemos nuestro equipo en un servidor *Web*.

Los puertos bloqueados acostumbran a ser los siguientes:

Puerto 21: usado por los servidores de *FTP*.

Puerto 23: usado para el acceso vía Telnet.

Puerto 25: usado por los servidores *SMTP* (servidor de correo electrónico).

Puerto 80: usado para el acceso vía *HTTP*, para servidores de páginas *Web*.

Puertos 137 y 138: usados para compartir archivos e impresoras (sirve más como una medida de seguridad para los usuarios, ya que mucha gente acaba olvidando estar compartiendo zonas de su disco duro, exponiéndose a invasiones de extraños).

Puerto 1352: usado para la administración remota del *Lotus*.

Puertos 1503 y 1720: estos puertos son usados por *NetMeeting* para recibir las llamadas. Con los puertos bloqueados, podemos llamar a alguien para conversar, pero nadie conseguirá llamarnos.

Puerto 5631: usado para la administración remota usando el programa *PC Anywhere*.

En el caso de los servidores *FTP* y *HTTP* es fácil esquivar la limitación. Al configurar nuestro servidor *FTP*, cambiamos el puerto por defecto (el 21 bloqueado) por otro puerto cualquiera, el 22 por ejemplo. En el caso del servidor *HTTP*, nuevamente configuramos el programa para usar otro puerto y no el 80 seleccionado por defecto, podemos usar el 1080 por ejemplo.

> **Nota**: en este caso, para visitar la página, en vez de teclear http://suIP, deberemos introducir http://suIP:1080, ya que eso orientará al navegador a visitar el servidor usando el puerto 1080 en vez de el puerto por defecto, esquivando la limitación del puerto bloqueado.

14.2.2 Compartir la conexión ADSL

Usando una conexión *ADSL* sólo tenemos una dirección *IP* asignada, pero claro, esto no significa que no podamos compartir la conexión a través del programa *Internet Conection Sharing*, o un servidor *proxy* cualquiera, así como los haríamos para compartir una conexión vía módem.

En este caso, el equipo que tiene el módem *ADSL* instalado debe tener dos tarjetas de red, una para el módem y otra para la red interna, a través de la que nos comunicaremos con el resto de equipos de la red local. En este caso, la desventaja radica en que el equipo principal tendrá que quedarse conectado las 24 horas, de lo contrario los demás perderían el acceso a *Internet*.

Otra opción es comprar un *hub*, conectar los equipos en red usando el *hub*, y conectar el módem a él, usando el puerto *Up-Link*, el puerto que normalmente es usado

para conectar un *hub* a otro de la misma red. No se olvide de usar un cable de finalización de circuito. La velocidad de conexión va a limitarse a cerca de los 2 Mbps, que es la velocidad media de comunicación del módem *ADSL* en la actualidad. De cualquier forma, ya es suficiente para 8 conexiones de 256 Kbps.

14.3 RDSI (ISDN)

La *ISDN* (o *RDSI*, que es el término nacional) es otra tecnología de acceso rápido, que surgió antes de la *ADSL*, que ya está disponible en la mayoría de las ciudades, pero que ofrece varias desventajas, al ser más lento y caro.

En la *RDSI* se continúa usando el sistema telefónico común. La ventaja es que en este sistema tenemos a nuestra disposición dos líneas digitales de 64 K cada una, que pueden ser usadas tanto para la conexión a *Internet* como para las llamadas de voz. Al tiempo de conectarnos tenemos la opción de usar las dos líneas, conectándonos a 128 Kbps, o dejar la segunda línea libre para las llamadas de voz, pero en compensación navegando a sólo 64 Kbps.

14.4 Otras opciones de acceso

El acceso vía cable y vía radio, que fuera de la *ADSL* y la RDSI, son actualmente las únicas opciones disponibles. A pesar de sus correspondientes desventajas, todos los servicios presentan una calidad muy superior a la que presenta el acceso vía módem común, la única cuestión es considerar si el coste de la cuota mensual vale la pena.

En el caso del acceso vía cable existe el problema de que el mismo cable es compartido por todos los usuarios de la región. Muchas veces, el mismo cable pasa por casi un barrio entero. A pesar de la velocidad contratada varía de los 512 kbps a los 4 Mbps, dependiendo del plan de precios escogido, se nos dividirán los 30 Mbps permitidos por el cable con todo el mundo que esté conectado al mismo tiempo. La queja más común entre los usuarios del cable es que el acceso es rápido al comienzo, pero que empieza a disminuir de velocidad a medida que más usuarios se van conectando. Además de la capacidad máxima del cable, también existe la posibilidad de que el propio bucle del proveedor quede congestionado. Sin embargo, no existe el problema de que la conexión pueda caerse algunas veces al día, como sucede con algunos usuarios del acceso vía *ADSL*.

En el caso del acceso vía radio, que se viene ofreciendo en más ciudades, se instala una antena en los patios de los edificios. Esta antena se comunica con la central a través de microondas y se usa por todos los usuarios próximos a la antena, los habitantes del propio edificio en general, que son conectados a la antena vía cable o fibra óptica.

La ventaja del acceso vía radio es el hecho de ser un poco más barato que la *ADSL* o el cable, pues la empresa que instala la antena y los cables también actúa como prove-

edora del servicio, haciendo que sólo exista una mensualidad. La desventaja es el hecho de que el servicio sólo está disponible para edificios o habitantes próximos, dependiendo de un acuerdo con los habitantes.

Una cuarta opción que debe estar disponible en breve es el acceso vía satélite, usando las mismas antenas parabólicas usadas para recepción de las señales de distintas televisiones. La gran limitación del acceso veía satélite es que a pesar de que la tasa de descarga sea de hasta 386 Kbps, dependiendo del plan de precios, el servicio es unidireccional, o sea, continúa siendo necesaria una conexión con un módem para poder realizar las subidas, manteniendo el problema de la línea telefónica.

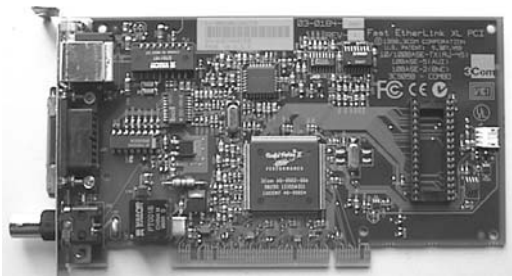
15. Redes

Si bien en tiempos pasados las redes se utilizaban de forma casi exclusiva en el terreno profesional, en la actualidad no es raro encontrar pequeñas redes de uso privado. La palabra "red", sin más, ha dejado de tener la suficiente fuerza explicativa, pues actualmente existe una gran variedad de redes de todo tipo, especialmente para Windows 98/2000/NT/XP. Por ejemplo:

- Redes *TDD*, transmisión de datos a distancia, con una conexión vía módem.
- Conexión directa de un PC a otro PC a través de un puerto paralelo o un puerto *RS232*.
- Red punto a punto (*Peer-to-Peer*) (*Microsoft Network*).
- Red de cliente-servidor (por ejemplo, *Novell Network*).

En realidad, sólo las redes punto a punto y cliente-servidor ofrecen las funciones de red en el sentido de intercambiar datos, entre el número de ordenadores que deseemos. Las redes *TDD* y las de conexión directa sólo permiten el intercambio de datos entre dos ordenadores, pero sin una transmisión simultánea en ambas direcciones.

Las redes se soportan de manera estándar desde la introducción de *Windows* para el trabajo en grupo (Windows 3.11). Esto significa, ante todo, que no es necesario ningún programa especial de red, aparte del sistema operativo *Windows*, para montar una red. Por otra parte, en la actualidad se pueden adquirir las tarjetas de red *PCI* desde unos 15 o 20 euros y si le añadimos los costes de los demás accesorios, como los cables y el material de montaje, podemos unir dos PCs en red por unos 60 euros.



Una tarjeta de red PCI (Fast Ethernet) con conectores de par trenzado, coaxial y AUI

En este capítulo trataremos directamente las redes que permiten la conexión de más de dos ordenadores. Para ello, será imprescindible entrar primero en la teoría necesaria, como mínimo, en el momento en que la red se vaya haciendo más y más grande con el paso del tiempo.

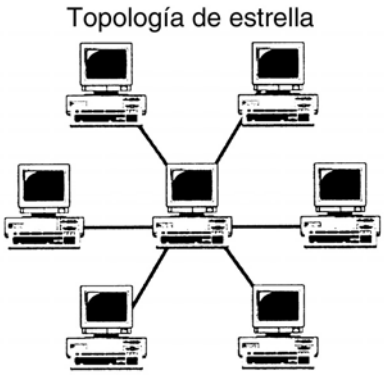
Con unos pocos conocimientos teóricos, la ampliación se podrá hacer desde el principio de la manera más económica y sencilla posible y sacándole el máximo partido y rendimiento posible.

15.1 Estructuras y estándares de red

Una red local (*Local Area Network, LAN*) puede presentar básicamente tres topologías (estructuras básicas) distintas: la topología de bus, la de estrella y la de anillo, pudiéndose dar casos de topologías combinadas, como la de árbol por ejemplo.

15.1.1 La topología de estrella

La estructura clásica de red, y la más antigua, es la de estrella. El centro de la estrella lo forma el servidor, o una gran computadora central. Cada estación de trabajo, que en tiempos pasados estaba formada por una terminal sin inteligencia propia, se conecta al ordenador central a través del cable.



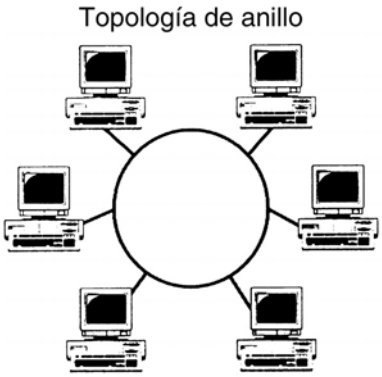
Topología de estrella

El trabajo de cableado en este tipo de topología puede ser considerable, y puesto que todas las comunicaciones, incluso las de estación a estación, pasan siempre por el servidor, la carga que éste tiene que soportar es a veces muy alta.

Actualmente, el cableado de las topologías de red se realiza mediante conexiones "punto a punto" entre los diferentes nodos de la red (por ejemplo, desde un *Hub*, o distribuidor de estrella, a los diferentes PCs), y como medio de transmisión se utilizan cables de cobre de par trenzado o cables de fibra óptica (*LWL*).

15.1.2 La topología de anillo

En esta topología, las estaciones de trabajo individuales se ordenan en forma de anillo y la información pasa de una estación a la siguiente hasta que llega al destinatario al que va dirigida.



Topología de anillo

Si este anillo cerrado queda interrumpido por el fallo de una estación de trabajo, toda la red deja de funcionar. Para solucionar este problema, lo que se hace es aplicar un cableado redundante (anillo activo más anillo *Stand-By*), o colocar relés *By Pass* que permiten "saltarse" las estaciones que se encuentran fuera de servicio. Esto supone, en ambos casos, un incremento de los costes de instalación de la red.

El ejemplo más famoso de esta topología de red es la *Token-Ring* de la casa IBM, cuyo procedimiento de acceso se conoce como *Token-Passing*, y las redes *FDDI* (*Fiber Distributed Data Interface*).

15.1.3 La topología de bus

La topología de bus es la estructura que más se utiliza en las redes pequeñas, y también la utiliza, por ejemplo, la red *Ethernet*. Todas las estaciones de trabajo, con el servidor incluido, están conectadas a un segmento de red a través de un cable propio que lleva una resistencia terminal en cada extremo.

Las estaciones se pueden comunicar con el bus a través de un *transceiver*, por lo que se puede retirar perfectamente una de ellas durante el funcionamiento de la red sin que eso signifique el paro total de toda la red.

En las instalaciones sencillas para oficina, las más habituales, no se utilizan los transceiver externos, pues éstos ya se encuentran en la tarjeta de red corriente. El cable coaxial que se utiliza para la conexión se va llevando de PC a PC y, con piezas T, se va conectando al bus de la red. Después se cierra con una resistencia terminal (terminador) en cada extremo.

Topología de bus

15.1.4 La topología de árbol

Las redes no están limitadas a una determinada topología. En realidad, cuanto más grande sea la red, mayor es la probabilidad de que se utilicen varias topologías a la vez, resultando así una topología combinada: la de árbol. En esta topología tiene lugar un escalamiento de la red global mediante concentradores de cables (*hubs*) y (*switches*), tal y como veremos en el apartado dedicado a los *Hubs* y *Switches*.

El protocolo *CSMA/CD*, explicado en el apartado *Ethernet*, es el procedimiento de acceso más utilizado en las redes locales (*LAN*). Naturalmente, existen otros protocolos, entre los que uno de los más importantes es el *Token-Passing*, que vamos a ver a continuación.

15.1.5 Token Passing

El protocolo *Token-Passing*, utilizado por ejemplo en la topología de anillo *Token-Ring*, utiliza un modelo especial de bit para el control de los datos (*Token*), que va pasando, dentro del anillo, de una estación de trabajo a la siguiente.

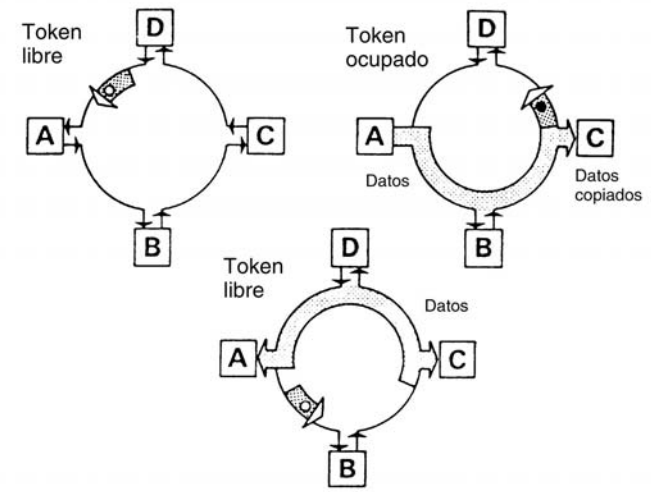
La estación que quiere enviar los datos, selecciona el *Token* como ocupado y añade los datos que quiere enviar y la dirección de la estación de destino. Las otras estaciones miran la dirección para saber si la información es para ellas, y en caso negativo, el *Token* va pasando de una a otra hasta que llega a la estación de destino (C).

La estación copia el mensaje y confirma en el *Token* que lo ha recibido, el *Token* sigue su camino hasta llegar a la estación que lo envió (A). Esta estación acepta el mensaje, lo borra y marca de nuevo el *Token* como libre.

En caso de que nadie recoja el mensaje, lo cual se reconoce porque no existirá la confirmación de recibo, el *Token* recorre el anillo nuevamente. Puesto que sólo se envía un *Token* cada vez, nunca se producen colisiones de distintos mensajes, y en la práctica se puede alcanzar (casi) la velocidad máxima de transmisión (teórica) de 16 Mbits/s, algo imposible de conseguir con el protocolo *CSMA/CD*, pues cuantas más estaciones existen en la estructura, más colisiones se producen y menor es la velocidad de transmisión de datos.

Las estaciones se ordenan en forma de anillo, como en las topologías de anillo. Los distribuidores de líneas del anillo pueden encontrarse a una distancia de hasta 200

metros entre sí, y con distribuidores activos la distancia puede ser de hasta 750 metros. La distancia permitida entre el distribuidor y una estación del anillo (pueden haber hasta 260) es de 100 metros. Alternativamente, la transmisión de mensajes también se puede realizar en ambas direcciones, en cuyo caso la velocidad de transmisión puede ser más baja, lógicamente, pues habrá que transmitir una mayor cantidad de datos.



El principio de funcionamiento del protocolo Token-Passing

El protocolo *Token-Passing* también se utiliza en las redes de bus, como por ejemplo en las grandes instalaciones de red tipo *Arcnet*. Sin embargo, puesto que en una red de bus no existen estaciones próximas entre sí, éstas se representan mediante direcciones que conforman un "anillo lógico" en el que el *Token* se puede ir enviando de una estación a otra estación (*Token-Bus*). Cuando una de estas estaciones no está operativa, el protocolo se da cuenta y reordena las restantes direcciones para mantener la red en funcionamiento.

Un funcionamiento similar al del protocolo *Token-Passing*, porque también está basado en un *Token*, lo encontramos en la *Fiber Distributed Data Interface* (*FDDI*), la cual puede llegar a alcanzar los 100 Mbits/s, que es la velocidad que se consigue también en uno de los últimos desarrollos de *Token-Ring*, el *High Speed Token Ring* (*HSTR*).

15.1.6 100BaseVG-Any-LAN

Otro desarrollo es el *Priority Medium Access* (*DPMA*), que en cierta manera se puede considerar como una combinación de los protocolos *Ethernet* y *Token-Ring*. Con este protocolo no se produce ningún tipo de colisiones como sucede con el protocolo *CSMA/CD*, ya que se sigue un control de prioridad de demanda (*Demand Priority*). El mecanismo de acceso se concentra en un *Hub* central que se puede entender como un conmutador de puerto libre del *Hub* al PC.

Este protocolo se utiliza en la *100BaseVG-Any-LAN*, de la cual hizo mucha propaganda la casa Hewlett Packard, sobre todo como alternativa a la *Fast Ethernet*. En contra de lo esperado, sin embargo, no ha conseguido imponerse en el mercado, pues la *Fast-Ethernet* (estándar *100Base*) ofrece, con un rendimiento similar, una ruta de migración más económica desde el estándar *Ethernet* (estándar *10Base*).

15.1.7 Asynchron Transfer Mode

El *Asynchron Transfer Mode* (*ATM*) fue desarrollado para redes de alta velocidad y su funcionamiento se basa en el principio de las conexiones virtuales, aplicado en general en las redes de transmisión de paquetes. En este modo no se activan canales fijos, sino células/celdas (siempre de 53 Bytes de longitud) de una conexión previamente definida, que se transportan a media que se van creando.

ATM permite el uso de anchos de banda definidos (escalables), lo que lo convierte en el único protocolo de red capaz de transmisión en tiempo real en la actualidad. Por eso resulta tan apropiado para las redes multimedia (audio, vídeo).

La técnica actual permite un ancho de banda global de 2,4 Gbits/s, que se pueden dividir, por ejemplo, en 16 puertos de 155 Mbits/s cada uno. *ATM* se utiliza cada vez más como tecnología *Backbone* (*Backbone* es el segmento principal de una red, al que están conectados los demás segmentos) y puede convertir directamente protocolos *WAN* (*Wide Area Networks* o red de área global) en protocolos *LAN*, y viceversa, que es lo que puede significar, por ejemplo *RDSI* o *ADSL* en *Ethernet*.

15.1.8 Los estándares LAN más importantes

El I*nstitute of Electrical and Electronic Engineers* (*IEEE*) ha definido diversos estándares obligatorios para redes, cuyos estándares más importantes aparecen en la tabla siguiente.

Estándar	Significado
IEEE 802.1	Definición, arquitectura, administración y funcionamiento en Internet.
IEEE 802.2	Definición del protocolo y de paquetes de formatos de datos (frames) entre los controladores de conexión lógica (Logical Link Controls).
IEEE 802.3	CSMA/CD para redes de bus, Ethernet.
IEEE 802.4	Token-Passing para redes de bus, estándar MAP.
IEEE 802.5	Token-Passing para redes de anillo, estándar Token-Ring.
IEEE 802.6	Red de área metropolitana (Metropolitan Area Network: MAN).
IEEE 802.7	Broadband Media, redes de banda ancha.
IEEE 802.8	Fiber Optic Media, medios de fibra óptica.
IEEE 802.9	Integrated Voice and Data Line, comunicación por voz.
IEEE 802.10	Secure Data Interchange, mecanismos de seguridad y confidencialidad.

Estándar	Significado
IEEE 802.11	Wireless LAN, LAN inalámbrica.
IEEE 802.12	100VG-AnyLAN, Fast-Ethernet.
IEEE 802.14	Redes CAT-TV, televisión por cable en la red.
IEEE 802.30	100Base-X, 100Base-T, Fast-Ethernet.

La arquitectura de red más conocida, *Ethernet*, funciona con el estándar *10Base* y su desarrollo posterior, de mayores prestaciones, el estándar *100Base*. De ambos existen dos versiones, aunque como ya hemos visto, la versión de Hewlett Packard y AT&T (*100Base-VG Any-LAN*) no se ha impuesto en la medida en que lo ha hecho la versión 100Base-T de las casas 3Com, Intel y Sun, basada en el viejo protocolo de colisión (*CSMA/CD*), y lo mismo puede decirse del último desarrollo de *Gigabit Ethernet* (*1000Base T*).

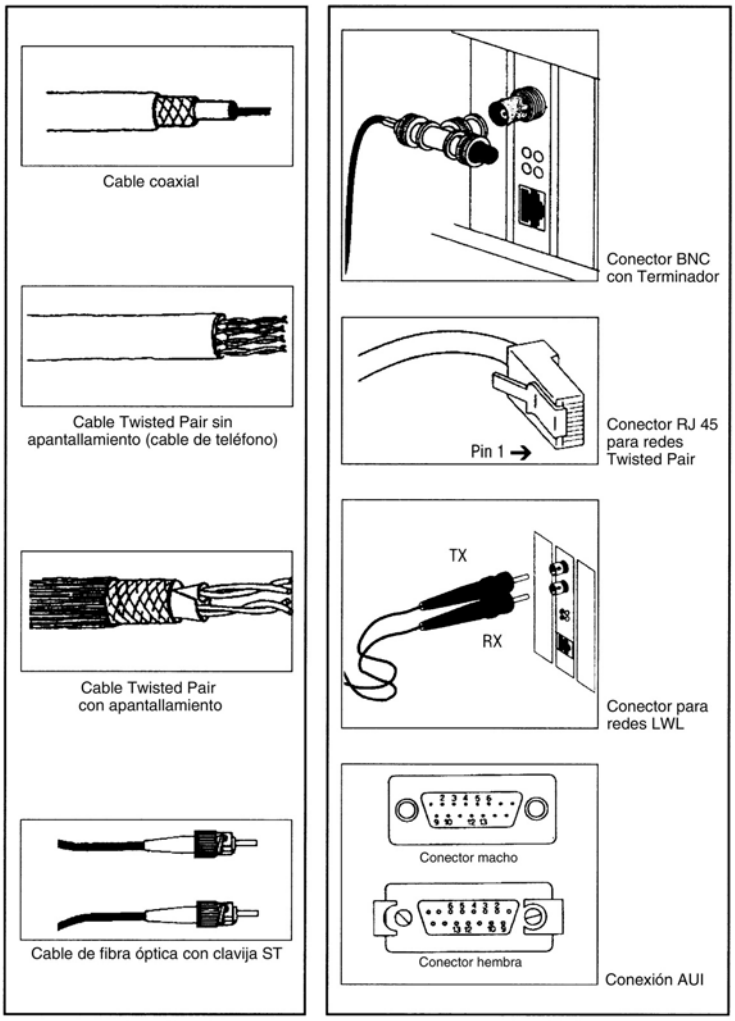
El fundador de la empresa 3Com, por cierto, fue el creador del *Ethernet* original, que todavía dará mucho más de sí, pues cuenta con un amplio abanico de posibilidades, definidas en los estándares *Base* que indicamos en la tabla siguiente.

Estándar	Significado/Utilización
10Base 2	Cheapernet o Thin-Wire (cable coaxial, RG58) con una longitud máxima de 185 metros por segmento para un máximo de 30 estaciones.
10Base 3	Ethernet de banda ancha, con cable de 3.600 metros de longitud máxima por segmento.
10Base 5	Thick-Ethernet con cable coaxial (Yellow Cable) de 500 metros de longitud máxima por segmento para un máximo de 100 estaciones.
10Base F	Ethernet con conexiones de fibra óptica. 10Base-FP contiene directrices para ejes pasivos LWL.10Base-FB para el uso de LWL como Backbone.10Base-FL se corresponde prácticamente al estándar FORIL (Fiber Optic Repeater Inter Link), con segmentos de cable de 1 km como máximo.
100Base FX	Conexiones de fibra óptica de 100 Mbps (Fast Ethernet, CSMA/CD), máxi mo 410 metros con cable multi-modo.
10Base T	Ethernet con cable de par trenzado y segmentos de una longitud máxima de 100 metros.
100Base T	Fast-Ethernet con protocolo CSMA/CD y cable de par trenzado.
100Base TX	Utiliza el cable UTP/STP de la categoría 5, es capaz de soportar dúplex completo y está considerado como el estándar Fast-Ethernet. 100Base-T4 ya se puede utilizar con cable de la categoría 3, para lo cual se toman tres líneas de datos y otra para la detección de colisiones, pero entonces no es posible el modo de dúplex completo. Para salvar esta dificultad, se puede utilizar alternativamente un cable de par trenzado de 8 hilos.

Estándar	Significado/Utilización
100Base VG	Fast-Ethernet con protocolo DPMA (Demand Priority Medium Access), con cable de par trenzado (de 8 hilos) o cable LWL, y una topología en forma de estrella. Existe una versión conjunta de Ethernet con Token-Ring.
1000Base T	Gigabit Ethernet (IEEE 802.3z), 1000 Mbits/s con cable de par trenzado a partir de la categoría 5, o con cable de fibra óptica.

15.2 Instalación de redes

La imagen siguiente ofrece una visión de conjunto de los diferentes tipos de cable, así como de los conectores, utilizados normalmente en la instalación de redes.

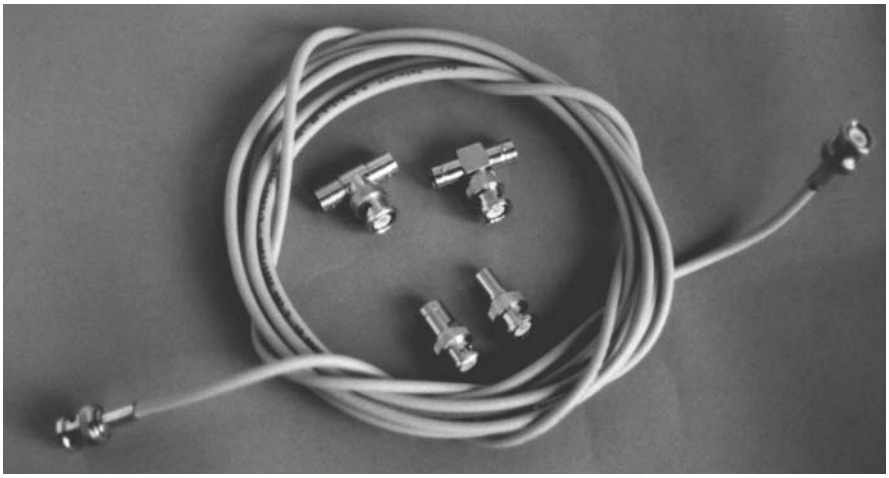


Las conexiones de red más utilizadas

15.2.1 Conexiones de cable coaxial

La manera más sencilla (y rápida) de montar el cableado de una red de PCs es utilizando cable coaxial (*RG58, Cheapernet*). Este cable está formado por un conductor interno que es el que transmite la señal de datos, un conductor externo y una protección. Gracias a ésta, que protege la señal de datos del exterior, la posibilidad de fallo de estas conexiones es mínima. El cable es barato, flexible y no presenta problemas en su manejo.

En realidad, lo único que hay que vigilar siempre es que cada PC tenga su pieza en T, que se respeten las longitudes máximas de cable, tal como se indican en los estándares *Ethernet*, y que los segmentos de cable vayan realmente de PC a PC. Además, son absolutamente imprescindibles las dos resistencias terminales (terminadores) en cada extremo del bus (en los extremos de cada segmento), pues sin ellas no funcionaría nada. Por eso la búsqueda de errores en una red de este tipo empieza, normalmente y siempre que hayan quedado excluidos posibles errores de programa, con el control de las diferentes conexiones, lo cual puede ser un trabajo de bastante tiempo, según cómo estén puestos los cables y el número de PCs conectados.



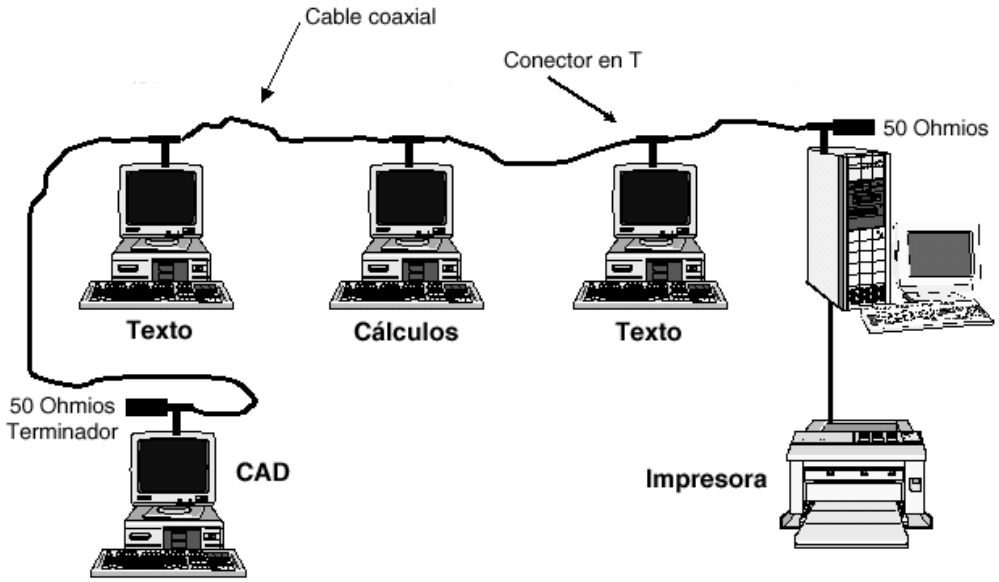
Un cable coaxial con dos piezas en T y dos resistencias terminales son los elementos básicos para la conexión por cable de dos PCs

El bus siempre debe llevar terminales en sus extremos y nunca debería haber cables sueltos. Para comprobar que tanto la conexión como la terminación del cable es la correcta se puede utilizar un sencillo ohmímetro. Para ello, se extraen todas las piezas en T (con los cables conectados a ellas) de los conectores *BNC* de la tarjeta de red y, a continuación, se saca una de las resistencias terminales para realizar la medición por ese extremo.

Con un ohmímetro se pueden medir resistencias de hasta unos 50 Ohms, y cuando se retira la resistencia del otro extremo, el ohmímetro debería señalar una "resistencia infinita". En caso contrario, deberíamos controlar todas las conexiones, una por una.

Si se trata de una conexión de red con cable coaxial relativamente larga, la masa que está conectada con cada uno de los PC a través del cable puede dar problemas y hacer que la red deje de funcionar, o incluso impedir que funcione desde el principio.

Naturalmente, cada PC va enchufado a una caja de conexión y está conectado a la toma de tierra (en contacto con el suelo), que no tiene porqué presentar el mismo potencial en todos sus puntos, produciéndose así una masa "fluctuante" del cable coaxial.



Una instalación típica de oficina con cable coaxial

Es por eso por lo que se utilizan las resistencias terminales. Éstas llevan una pequeña cadena que es la que se ha de conectar directamente a la toma de tierra, impidiendo así que el potencial de masa fluctúe y haciendo que permanezca fijo. Esta conexión, sin embargo, se debe realizar sólo en un extremo de la red, y nunca en ambos, pues de lo contrario volveríamos a tener el problema.

15.2.2 Conexiones de par trenzado

Muchas tarjetas de red (tarjetas *Combo*) poseen, además de un conector *BNC* y otro *AUI*, un conector *Western* (conector *RJ45*) adicional como los que se emplean también en los módems. Este conector es el que se utiliza para el cable de par trenzado, formado por varios hilos enrollados entre sí. Los cables de par trenzado existen en dos versiones: con 4 u 8 hilos, siendo el más común el de 4 hilos.

Pin Nr.	Cable de 4 hilos	Cable de 8 hilos
1	TXD+	TX_D1+
2	TXD-	TX_D1-
3	RXD+	RX_D2+
4	-	BI_D3+
5	-	BI_D3-
6	RXD-	RX_D2-
7	-	BI_D4+
8	-	BI_D4-

A su vez, el cable puede llevar protección (*STP*, *Shielded Twisted Pair*) o no llevarla (*UTP*, *Unshielded Twisted Pair*), siendo naturalmente la primera versión la que ofrece una mayor resistencia al medio externo y es, por lo tanto, menos vulnerable a las interferencias. El cable *STP* existe a su vez en dos versiones, una con una protección general (*S-UTP*) en forma de camisa alrededor de todos los hilos, y otra donde cada par de hilos lleva una protección individual.

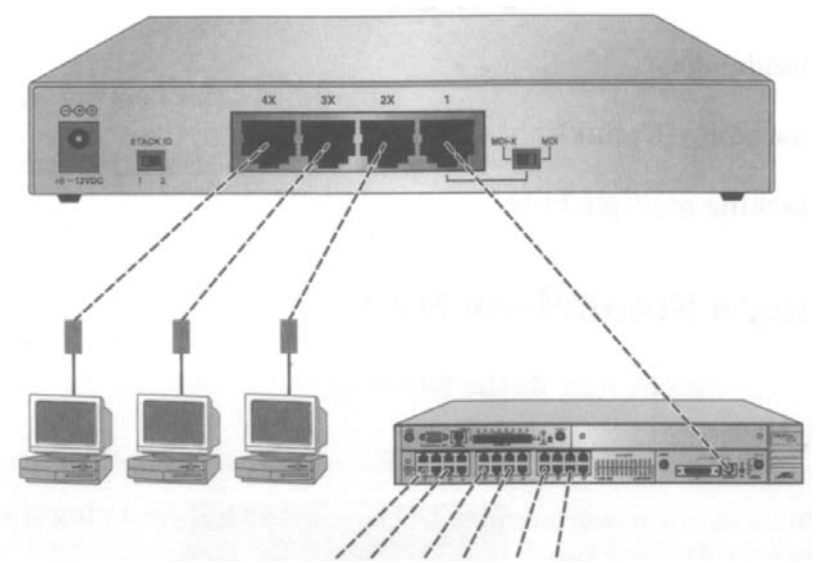
El cable más utilizado es, sin embargo, el *S-UTP* (categoría 5), que es el que da menos problemas. Mezclar diferentes tipos de cable dentro de un mismo ramal de la red puede provocar, casi siempre, una relación "irregular" de las masas, aumentando así la vulnerabilidad del cable.

Como puede deducirse de la tabla anterior, los cables de par trenzado transfieren los datos de forma diferencial, es decir, cada cable tiene dos líneas de envío de datos (*TXD+*, *TXD-*) y dos líneas de recepción (*RXD+*, *RXD-*). Consecuentemente, las señales de datos no se refieren a la masa, como en un cable coaxial, sino que la señal de uso se encuentra entre *TXD-* y *TXD+*, y lo mismo sucede con la señal de recibo en las líneas *RXD+* y *RXD-*.

Esta transmisión de datos diferencial es más segura que la transmisión normal frente a los fallos externos, pues si se produjese una interferencia externa, la consecuencia sería que esa señal tendría el mismo efecto en ambas líneas, por lo que la señal diferencial -la información de datos- no se vería afectada en lo más mínimo.

Los cables de par trenzado se instalan normalmente de periférico a periférico, lo que tiene como resultado un cableado en forma de estrella en el que el cable no pasa simplemente de un PC a otro, como en el cable coaxial. El trabajo de cableado es mayor, pero también permite retirar un PC de la red sin ningún problema, mientras ésta está en funcionamiento, y en caso de que un conector o un cable tengan algún defecto, la red sigue funcionando.

Para realizar un cableado en forma de estrella se necesita un distribuidor (*Hub*) al que van conectados todos los PCs mediante los cables de par trenzado individuales.



Un Hub de cuatro puertos con conectores para par trenzado, en el que el puerto 1 también se puede utilizar como puerto uplink para la conexión con otro Hub o Switch. El pequeño interruptor se utilizará entonces en la posición X (hacia la izquierda) para cruzar las líneas de las señales, tal como se hace también con los cables cruzados

Si sólo hay que unir dos PC con cable *TP*, también podemos utilizar un cable *TP* especial en el que los pares de líneas *TXD* y *RXD* van cruzadas, con lo que las líneas de envío de un PC corresponden a las líneas de recepción del otro y, naturalmente, a la inversa.

Para la conexión de dos dispositivos de datos del mismo tipo, como por ejemplo dos *Hubs*, se utilizan básicamente líneas "cruzadas". El principio es el mismo que el que se aplica en el puerto *RS232*. En la siguiente tabla se muestran los contactos que se han de unir en una sencilla conexión *UTP* de dos PC.

Denominación	Conector	Conector	Denominación
TD+	1	3	RD+
TD-	2	6	RD-
RD+	3	1	TD+
RD-	6	2	TD-

El *Fast Ethernet* (100 Mbits/s) funciona con cables de par trenzado y también con cables de fibra óptica. Al realizar un cableado con estos medios debería dejarse abierta, desde el principio, la posibilidad de integrar en el futuro una ruta de migración, también para *Gigabit Ethernet*, sin que para ello sea necesario realizar de nuevo todo el cableado.

Por regla general resulta más económico cambiar las tarjetas de red y los elementos necesarios para el funcionamiento en Internet, que hacer de nuevo toda la instalación de cables de la red. Con los cables de par trenzado, la velocidad de transmisión de los datos viene determinada, en cada caso, por la categoría del cable que se utilice.

Categorías de cables de par trenzado

Para clasificar los cables de par trenzado (*Twisted-Pair*) se han definido un total de siete categorías. Como conector macho se utiliza en cada caso un conector *Western* (*RJ45*) que puede ir con un cable de cuatro o de ocho hilos. El la siguiente tabla podemos ver las distintas categorías de cables de par trenzado.

Categoría	Significado/ Datos
1	Rendimiento de un cable de teléfono convencional con una transferencia de datos máxima de 1 Mbit/s.
2	Cable de reemplazo para los cables de la categoría 1. Se pueden alcanzar velocidades de transferencia de hasta 4 Mbits/s en distancias medias. Se uti liza también para RDSI.
3	Cable UTP o STP, Unshielded Twisted Pair (sin protección), o Shielded Twisted Pair (con protección). Se pueden alcanzar velocidades de transmisión de hasta 10 Mbits/s con una longitud de cable de hasta 100 metros. Se utiliza, por ejemplo, para Ethernet (10Base-T) y también para Token Ring.
4	Cable UTP/STP para distancias mayores a las permitidas en los cables de la categoría 3, permite velocidades de transmisión de hasta 20 Mbits/s pero apenas se utiliza.
5	Intervalo de frecuencia ampliado (100 MHz). Está considerado como cable estándar y también se utiliza, por ejemplo, para FDDI y Fast-Ethernet.
6	Intervalo de frecuencia de hasta 200 MHz, recomendado para ATM. El ATM (155 Mbits/s) también funciona con los cables de la categoría 5.
7	Intervalo de frecuencia hasta 600 MHz. Menos ruidos adicionales y amorti guación que en los cables de la categoría 6.

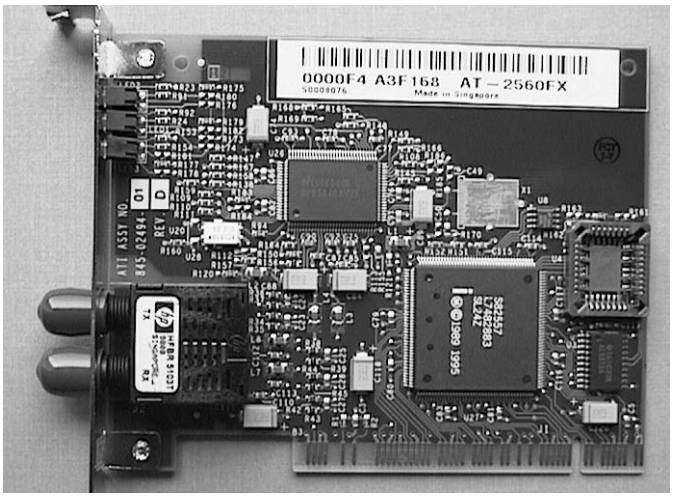
15.2.3 Conexiones de fibra óptica

Las redes basadas en las conexiones de fibra óptica (*CFO*) son menos vulnerables a las averías que las conexiones de hilo de cobre, a prueba de intercepción y muestran, en general, una tasa de errores más baja. Las tarjetas de red para fibra óptica, así como los *Hubs* y *Switches*, son bastante más caras que las utilizadas para las redes de conexión de cobre.

De todos modos, son mucho mayores las ventajas que presenta la fibra óptica, especialmente en el campo industrial, donde los intensos campos eléctricos y magnéticos podrían impedir el funcionamiento normal de una red convencional.

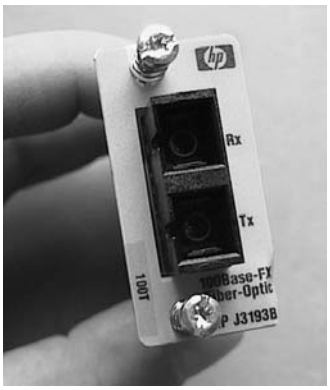
Por otra parte, las *CFO* facilitan considerablemente la transmisión de datos a altas velocidades, pues el medio de transmisión se pude conservar cuando hay que cambiar, por ejemplo, de *Standard* a *Fast-Ethernet* (*100Base-FX*).

Los cables de fibra óptica pueden ser, en general, cables monomodo (o *Single-Mode*) y cables multimodo. Los cables multimodo utilizan varios modos (varias formas de onda de luz) para la transmisión de las señales, mientras que los cables monomodo transmiten la luz en un sólo modo (paralela al eje), lo que se traduce en principio, en valores de amortiguación más bajos y anchos de banda mayores.



Una tarjeta de red con conector de cable de fibra óptica para 100Base-FX

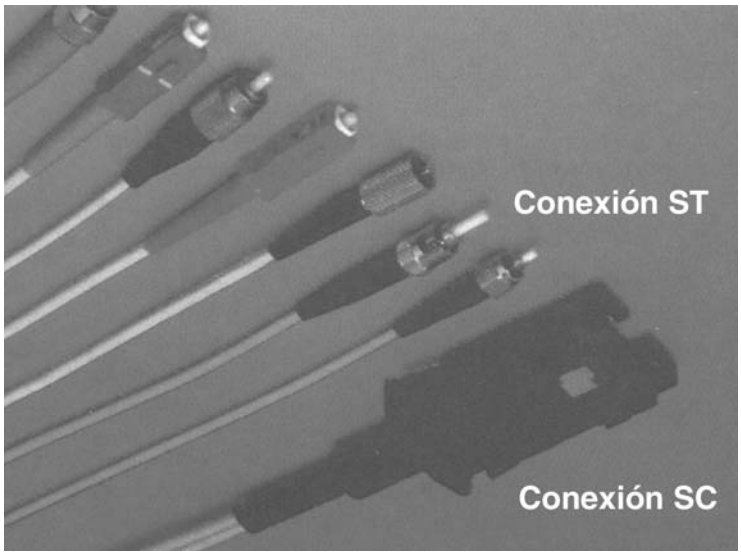
De todos modos, los diferentes acabados de los cables de fibra óptica, como pueden ser los empalmes de las fibras en las cajas de conexión y la calidad misma de las fibras, juegan un papel muy importante en la amortiguación y el ancho de banda de la conexión. Otra diferencia la marca el diámetro de las fibras ópticas, que en los cables multimodo es de 62.5 micras y en los cables monomodo de 8 a 10 micras.



Un módulo 100 Base-FX de la casa Hewlett Packard

Si bien *10Base-FX* tampoco ofrece una velocidad máxima de transmisión de datos superior a 10 Mbits/s, lo cierto es que este valor se consigue en la práctica mucho más rápidamente con este estándar que con las soluciones de conexiones de cobre, pues la menor vulnerabilidad de las conexiones y los mecanismos de colisión que tienen lugar en los *CFO* hacen que los envíos de datos no se tengan que repetir tantas veces como en otros sistemas (y lo mismo sucede con *100Base-FX*).

Las conexiones de fibra óptica se hacen en forma de estrella. Para ello se utilizan dos líneas, provistas a menudo con conectores macho *ST* y denominadas *RX* y *TX* (dúplex). La conexión de dos PC se realiza, al igual que con el cable de par trenzado, conectando una de las líneas *RX* a la línea *TX* de la otra tarjeta, y la línea *TX* a la *RX* del otro PC. Para conectar varios periféricos con cables de fibra óptica, se necesitarán los *hubs* o *switches* correspondientes. Además de los *ST* existen también otros conectores para fibra óptica, como por ejemplo los *SC* y *MIC*, que sólo se diferencian entre sí desde el punto de vista mecánico, en el modo de sujeción.



Diferentes conexiones CFO

Por lo demás, en una red de fibra óptica también es muy sencilla la búsqueda de errores, pues en base a la luz de las fibras (casi siempre roja) se puede reconocer de forma inmediata si están llegando o no los datos. Este método no funciona, sin embargo, cuando la línea correspondiente "pasa" por un *switch*, pues los cambios se producen entonces tan rápidamente que el ojo humano no puede percibir ninguna luz.

Actualmente, tanto en las tarjetas de red para fibra óptica como en las de cables de par trenzado, es habitual añadir varios diodos lumínicos que indican algún tipo de actividad (enviar, recibir, colisión, modo), método que resulta de ayuda en la búsqueda de posibles errores.

15.3 Ethernet

Ethernet es una tecnología de red que ya tiene más de 30 años de vida. Originalmente, la empresa Xerox diseñó *Ethernet* para conectar varios ordenadores a una impresora láser. Esta es una de las aplicaciones de red típica, la posibilidad de que varios ordenadores puedan aprovechar los servicios de un dispositivo de calidad.

El nombre de "Ether" obedece a la verdadera unión en una red y viene de un tiempo en el que las redes se instalaron por todos los Estados Unidos para poder mantener en pié vías de comunicación ante el temido ataque soviético. Esta idea ha constituido la base sobre la que se ha construido la red universal que es hoy *Internet*.

15.3.1 Principio de funcionamiento

La red *Ethernet* original se basaba en una topología de bus y utilizaba el método *CSMA/CD* (*Carrier Sense Multiple Access/Collision Detection*) en la comunicación. Cuando una estación quería enviar un mensaje a través de la red, primero comprobaba si en el bus ya había un mensaje de otra estación (*Carrier Sense*). Si el bus estaba libre podía acceder a él y enviaba los datos (*Access*).

Si varias estaciones exploraban el bus al mismo tiempo y empezaban a acceder a él a la vez (*Multiple Access*) surgía una colisión que debía ser detectada por cada una de esas estaciones (*Collision Detection*). A continuación se retiraban las señales y las estaciones intentaban colocar sus mensajes en el bus con diferencias de tiempo.

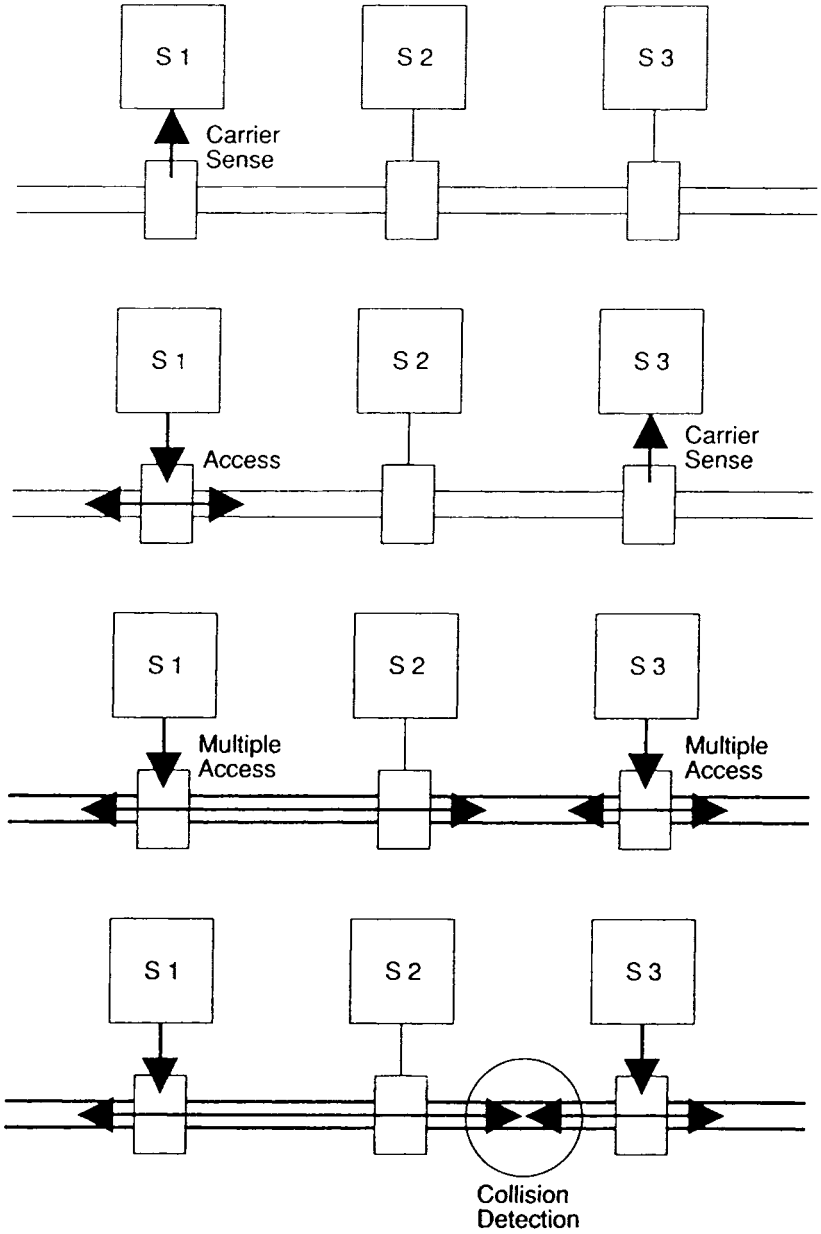
No era posible predecir qué estación obtenía el acceso y cuánto tiempo debían esperar el resto de estaciones, por tanto este procedimiento no era apropiado para sistemas de tiempo real o aplicaciones en las que el tiempo jugaba un papel relevante. A partir de un determinado número de estaciones la velocidad de procesamiento se veía ralentizada considerablemente debido a las colisiones que se producían.

El ratio de transferencia logrado con *Ethernet*, o más exactamente el ancho de banda, era de 10 Mbits/s, que se debían repartir de forma razonable entre las diversas estaciones conectadas. Por esta razón, también se considera *Ethernet* como un medio *Shared*.

Ethernet puede utilizar, aunque no obligatoriamente, la tipología de bus. Todas las estaciones (PCs) se conectan a un segmento del bus a través de un cable. Como es habitual en un bus, los dos extremos del segmento deben estar provistos de resistencias de terminación o terminadores.

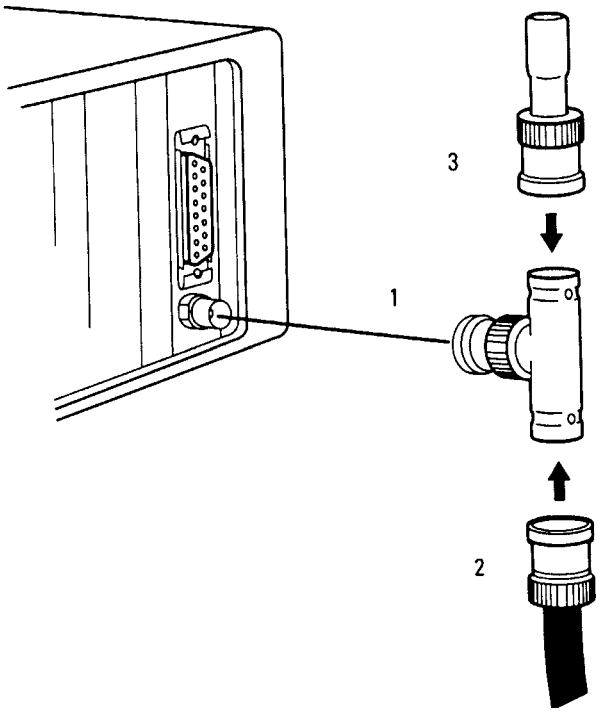
En principio existían dos posibilidades de conexión. En la primera posibilidad un ordenador debía poseer un controlador (por ejemplo una tarjeta Ethernet con conexión *AUI*) que se conectaba al bus con un cable *Drop-Out* de 50 metros como máximo y un *transcept*. La separación mínima entre dos *transcepts* era de 2.5 metros, lo que en ocasiones obligaba a colocar líneas adicionales.

Por esta razón existían transcepts *Multiplexer* a los que se podían conectar varias estaciones, cada una con su cable *Drop-Out*. La ventaja que aportaba esta posibilidad es que permitía quitar del sistema cualquiera de las estaciones mientras la red estaba en funcionamiento sin ocasionar, con ello, una interrupción de todo el sistema.



Principio del procedimiento de acceso CSMA/CD

La segunda posibilidad de conexión se extendió considerablemente gracias a las tarjetas *Ethernet* para PC. Además de la conexión *AUI* para el *transcept* externo, también disponía de un *transcept* con conexión coaxial. La conexión eléctrica se establecía con un cable coaxial y un adaptador en T, y en los dos extremos de la red, o del segmento de red, se conectaban directamente las resistencias de terminación obligadas.



Conexión a la red de un PC con un cable coaxial. Si el equipo se encuentra en uno de los extremos del segmento de bus se necesita una resistencia de terminación (3) (1 = adaptador T, 2 = cable coaxial)

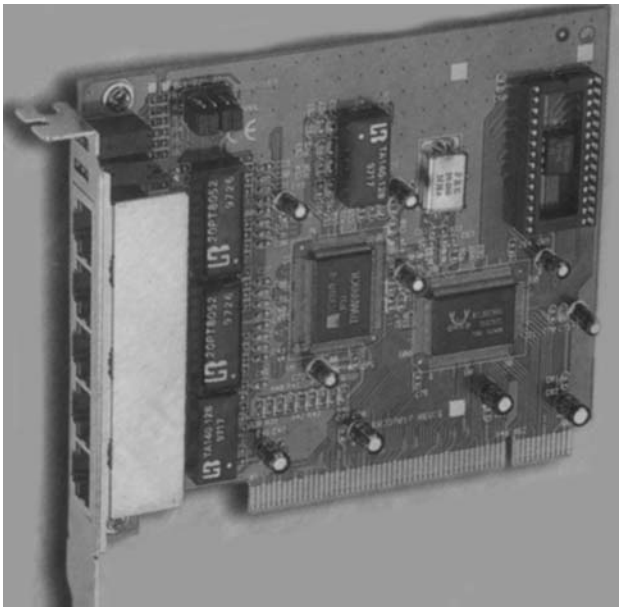
La longitud de los segmentos de cable en una red *Ethernet* dependía del tipo de cable usado. Con un cable coaxial con cuádruple revestimiento no debe exceder de 500 metros, y la longitud total permitida para la red debía ser de como máximo aproximadamente 2.500 metros. Una alternativa más económica a éste era un cable coaxial más delgado con las características de un cable *RG58*. Este tipo de unión se denominaba también *Cheapernet* y permitía utilizar segmentos de cable de hasta 185 metros, mientras que la longitud total de la red no podía pasar de los 900 metros.

En teoría, con *Ethernet* era posible conectar hasta 100 estaciones de trabajo a un segmento y un total de 1.024 estaciones dentro de una misma red. Con el cableado *Cheapernet* se podían conectar un máximo de 30. Por lo tanto, la estructura de bus con cable coaxial era más apropiada para redes menores, extendidas por ejemplo en una

planta de oficina. Para redes y longitudes mayores ya se utilizan preferiblemente conexiones de tipo par trenzado o incluso líneas de fibra óptica (*LWL*, conductor de ondas luminosas). Para estos dos tipos se usa una topología de red de estrella, es decir, cada PC se comunica a través de una línea propia y de un *Hub* (distribuidor) o de un *Switch* propio. La formación de redes mayores pasa casi obligatoriamente por una segmentación de la red en redes parciales, más fáciles de administrar.

Para comunicar sólo dos PCs con cables de par trenzado, como hemos visto anteriormente, no es necesario utilizar un *Hub*, simplemente basta con "cruzar" las líneas de señal tal y como se practica también en las conexiones *RS232 NullModem*.

También resulta práctica una tarjeta *Hub*, como por ejemplo el modelo GH4050-PCI (10 Mbits/s) de la casa Genius, ya que con su ayuda es posible conectar hasta cinco ordenadores. Así pues cada PC no necesita un *Hub* particular sino que es la tarjeta, que se instala y configura como una tarjeta de red normal, la que lleva a cabo las funciones del distribuidor.



Una tarjeta Hub es ideal para redes pequeñas

Con un jumper de la tarjeta se puede configurar uno de los puertos como puerto *Uplink* para poder conectar allí otro *Hub* o *Switch*. Para unir directamente dos *Hubs* se utilizan conexiones cruzadas.

Para comunicar dos ordenadores con un cable de fibra óptica no se necesita nada más que unir la conexión *TX* (*Transmit*) de un PC, a través de una línea de fibra de vidrio, con la conexión *RX* del otro equipo y la operación inversa con la segunda línea, se trata por tanto de "cruzar" las señales.

15.3.2 Fast Ethernet y Gigabit Ethernet

Un desarrollo posterior de *Ethernet* (10 Mbits/s) se denomina *Fast Ethernet* y está diseñado para transmitir 100 Mbits/s. La última variante de *Ethernet* es *Gigabit Ethernet*, que puede proporcionar un máximo de 1.000 Mbits/s. Todas las variantes de *Ethernet* se basan en el procedimiento *CSMA/CD* por lo que pueden ser compatibles entre sí. Asimismo, la tasa de transferencia no se multiplica por diez o por cien automáticamente, algo que está fundamentado en el mismo protocolo (CSMA/CD). La tasa de transferencia depende además del número de PCs que están conectados a la red y qué tipo de datos (pocos archivos grandes o muchos archivos pequeños) se transmiten. Por consiguiente, la razón por la que se alcanzan tasas de transferencia muy por debajo de las posibles es la misma que en el estándar *Ethernet* original (10 Mbits/s).

Para detectar correctamente las colisiones es necesario observar determinados parámetros, entre los que se encuentra en primer lugar la longitud del tramo. La distancia entre las estaciones no debe ser tan grande como para impedir que una estación pueda detectar si la red está o no ocupada. Los telegramas *Ethernet* (paquetes de datos, frames) siempre tienen un tamaño de entre 64 y 1518 Bytes, independientemente de la estructura y tamaño de la red *Ethernet*. El tiempo para detectar una colisión se obtiene de la longitud mínima de un frame, 64 Bytes, denominado también *Slot-Time*, más un tiempo de bloqueo de 64 bits (en total 576 bits), los que significa que 576 tiempos de bit debe ser el tiempo de recorrido máximo de un extremo al otro de la red para que el mecanismo *CSMA/CD* pueda funcionar sin ningún contratiempo.

Parámetro	Ethernet	Fast Ethernet	Gigabit Ethernet
Tasa	10 Mbits/s	100 Mbits/s	1000 Mbits/s
Longitud típica	2000 m	200m	20 m
Duración por Bit	0,1 μs	0,01 μs	0,001 μs
Slot-Time	512 bits	512 bits	4096 bits

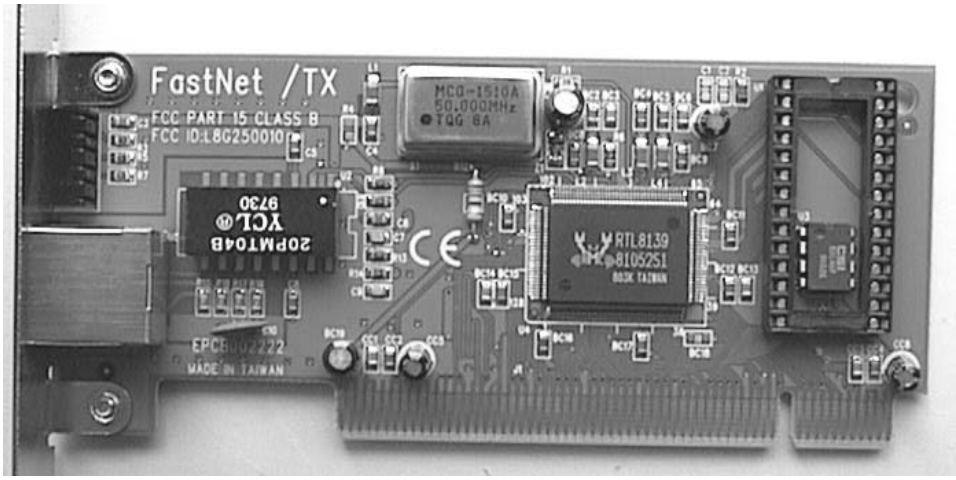
Por razones de compatibilidad (modo mixto) no debería ser utilizado otro formato de frame para conseguir que el estándar *Gigabit Ethernet* no se vea limitado a unos insuficientes 20 metros. Sin embargo, se alargó hasta 512 Bytes el frame de *Gigabit Ethernet* con frames adicionales (*Carrier Extensions*). Esto permitió detectar colisiones en una longitud máxima de 200 metros. Como consecuencia, cuando un ordenador emite datos a la red, el resto de equipos pueden detectar sin problemas la ocupación de la red.

El cable estándar de par trenzado pertenece a la categoría 5 y se aplica tanto al modelo *Ethernet* como al *Fast Ethernet*. En cambio, para *Gigabit Ethernet* se utilizan líneas de fibra óptica y un cable *Twinax*. Como vemos, la migración a *Gigabit Ethernet* conlleva el coste que supone cambiar todo el cableado. Por esta razón, está previsto para un futuro que el estándar *Gigabit Ethernet* también funcione con el cableado de cate-

goría 5, aunque todavía no hay una normativa clara sobre la longitud máxima del cable. Así pues, por razones económicas todavía es algo pronto para pasar de *Fast Ethernet* a *Gigabit Ethernet*.

Configuración

Fast Ethernet requiere el uso de conexiones de par trenzado o bien líneas de fibra óptica, y no funciona con cable coaxial. Desde un punto de vista tecnológico, si montamos una red con cableado de tipo coaxial invertiremos el dinero en un saco roto ya que el propio estándar hace imposible aumentar la tasa de transferencia de datos. Todas las tarjetas de red disponibles actualmente en el mercado admiten el estándar *Fast Ethernet*.

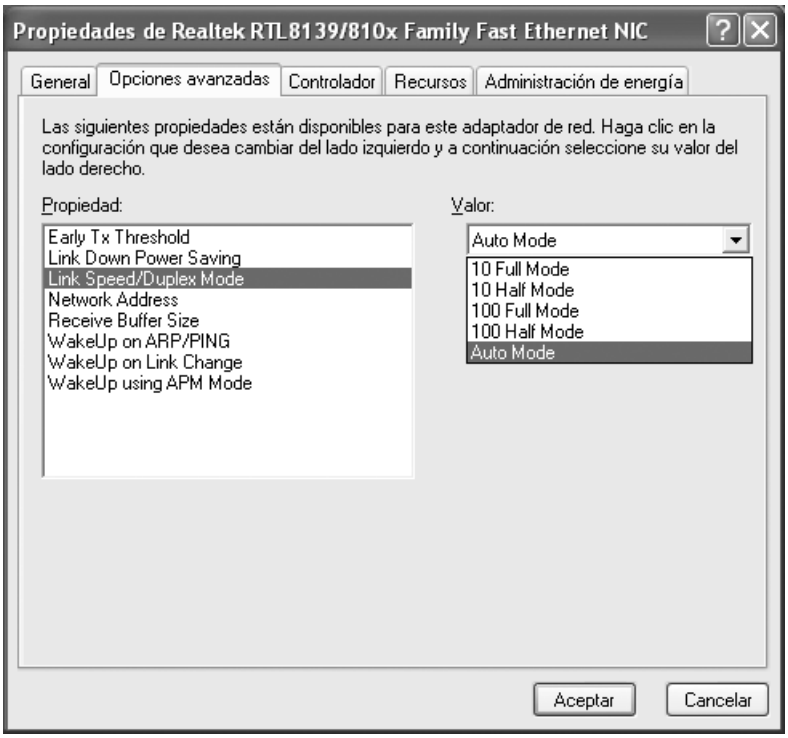


Tarjeta de red PCI para Ethernet estándar y Fast Ethernet

Normalmente las tarjetas de red, así como los *Hubs* y *Switches*, se ajustan de forma automática a la mayor velocidad posible, algo de lo que se encarga habitualmente el programa controlador de la tarjeta. Pero en algunos casos es posible ajustar manualmente este parámetro a través de un programa de instalación, un registro en el *Panel de control* o bien desde la ventana de propiedades de la tarjeta de red. Tampoco hay que dar por hecho que este ajuste automático (*Auto Negotiation*) funcione de la forma correcta con cualquier combinación de dispositivos, por consiguiente deberíamos predeterminar el *Ethernet* estándar o *Fast Ethernet* de forma manual.

Esta circunstancia es común a los *Hubs* y *Switches* compatibles con *Fast Ethernet*. Para su configuración estas unidades disponen de una conexión *RS232* para una terminal o un PC (con un programa de emulación de terminal), o bien la unidad se puede "gestionar" convenientemente en la red y ajustarla explícitamente para 10 o 100 Mbits/s. Sin embargo, en los *Hubs* más sencillos no existen estas posibilidades y lo más probable es que tengamos que cambiar la vieja tarjeta de red por un modelo de tarjeta *Fast Ethernet*, debido a que no funciona el ajuste automático en el *Hub*.

La configuración del modo idóneo debería ser también automática pero, en la práctica, funciona peor que el ajuste automático del modo *Ethernet* "correcto". Por estas razones es mejor no confiar en exceso en una mejora sustancial del rendimiento con *Fast Ethernet* si no está garantizado que todas las unidades se pueden comunicar de forma óptima.



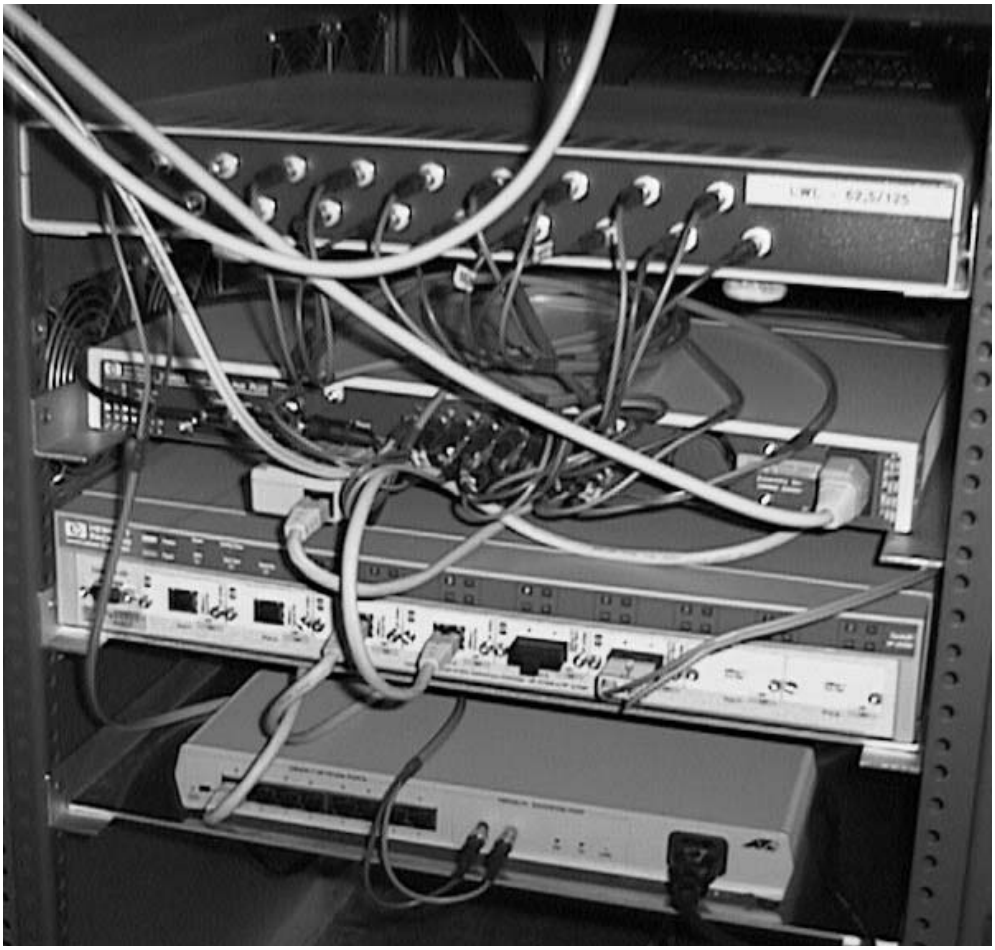
Configuración manual de los parámetros relevantes en una tarjeta de red

Las tarjetas LWL para fibra óptica no suelen disponer de un ajuste automático para *Ethernet* estándar o *Fast Ethernet*. Como existen tarjetas específicas para cada especificación deberemos elegir entre una tarjeta Ethernet estándar o una tarjeta F*ast Ethernet*. Las tarjetas *LWL* estándar también ofrecen un puerto para par trenzado de 10 Mbits/s. En 1999 la empresa SMC lanzó al mercado una tarjeta de red con soporte para transferencias en fibra óptica de 100 Mbits/s y, además, un puerto para par trenzado con ajuste automático (10 o 100 Mbits/s).

15.4 Hubs y Switches

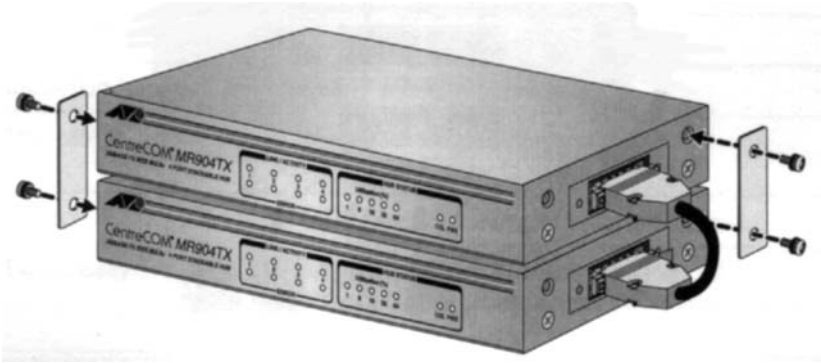
En las redes, los *Hubs* y *Switches* actúan a modo de distribuidores, y constituyen junto con los *Repeaters*, *Bridges*, *Routers* y *Gateways* los elementos del trabajo en *Internet* (aunque aquí no aludimos explícitamente a *Internet*).

Un *Hub* es el elemento de acoplamiento entre cada una de las estaciones de una red con tipología de estrella, representando el elemento central del sistema. Existen *Hubs* pasivos, activos e inteligentes. Los *Hubs* pasivos son simples puntos de unión entre dispositivos en los que no se realiza ninguna función eléctrica, y por lo tanto este tipo de *Hub* también se puede ver como una caja de conexión, conocida como *Patch-Box*.



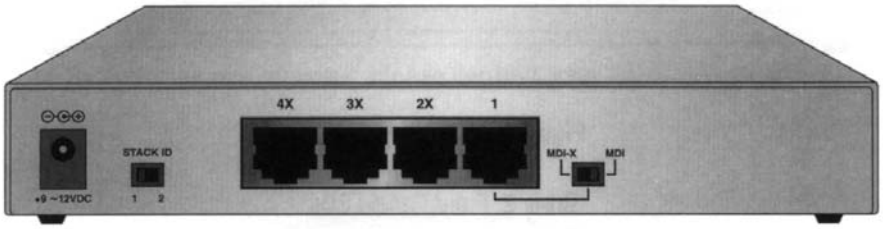
Todo de un vistazo: arriba un Patch-box para fibra óptica cuyas líneas conducen a un Hub de fibra óptica (debajo), que a su vez está conectado a un Switch. Abajo del todo podemos ver un Hub de 10 Mbits/s para fibra óptica con 8 puertos de par trenzado

Los *Hubs* activos regeneran, amplifican, las señales de datos y las distribuyen convenientemente entre las distintas estaciones. Normalmente poseen 8, 12 o incluso 24 conexiones (puertos). Si la cantidad de puertos disponibles no es suficiente se puede aumentar su número fácilmente con otros *Hubs* adicionales, apilándose uno encima de otro y se acoplan entre sí con una conexión *Backbone*. De esta forma, cada *Hub* se integra en el conjunto formando prácticamente una única unidad.



Agrupación de varios Hubs apilables por medio de una conexión Backbone

Pero también es posible conectar ente sí dos *Hubs* sin usar esta conexión especial, para ello entran en acción los puertos y la conexión se realiza con un cable con las líneas de señal cruzadas. En muchos casos hay un determinado puerto, generalmente el primero, marcado como puerto *Uplink* y junto a él, o en la cara posterior del dispositivo, se encuentra un interruptor que establece el cruce de líneas en caso de ser necesario.



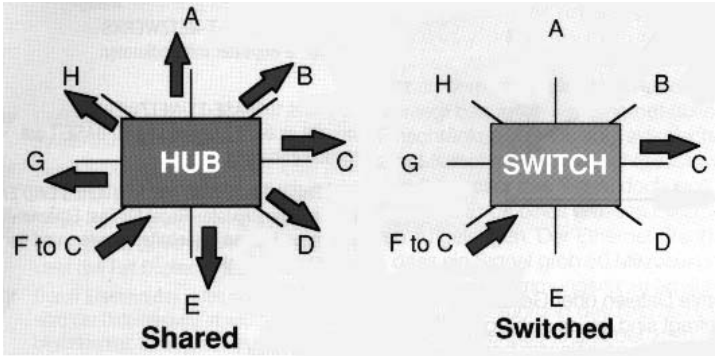
Un Hub con cuatro conexiones de par trenzado (10/100 Mbits) donde la primera se puede utilizar como puerto Uplink. Con el interruptor de la derecha se establece el modo MDI-X para cruzar internamente las líneas de señal

Los *Hubs* inteligentes*f* también se denominan *Managed Hubs* porque desempeñan funciones de un *Hub* activo y además disponen de otros mecanismos de diagnóstico, configuración y administración, funciones que no se dan en los demás *Hubs*.

15.4.1 ¿Hub o Switch?

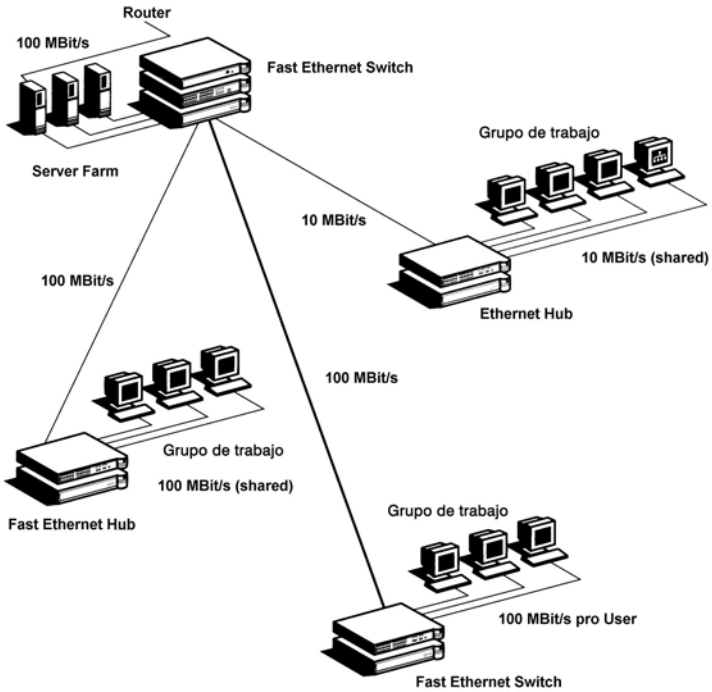
Un *Hub*, también llamado concertador o distribuidor de estrella, se puede considerar como un amplificador de señales (*Repeater*) que distribuye entre varios puertos la señal de entrada. A cada uno de esos puertos se conecta un ordenador cpara crear una estructura de estrella.

El ancho de banda total disponible en un *Hub* se reparte en función de las estaciones conectadas. Por esa razón, se utilizan *Switches* en lugar de *Hubs*, ya que los primeros facilitan en cada puerto la velocidad máxima posible.



Mientras que en un Hub se reparte el ancho de banda en función del número de estaciones conectadas, un Switch establece una conexión "directa" entre dos interlocutores, disponiendo así de todo el ancho de banda para transmitir los datos

Los *Switches* son idóneos para dividir una red *LAN* en varios segmentos o subredes, donde habitualmente los grupos de trabajo se establecen según aspectos lógicos y organizativos. A diferencia de un *Hub* o de un *Bridge*, el *Switch* hace posible un intercambio de datos paralelo entre subredes o eqeuipos. Un *Switch*, también conocido por *Multiport Bridge*, casi crea una conexión punto a punto entre dos dispositivos y donde se dispone de todo el ancho de banda del canal de transferencia.



Distribución de una red con Hubs y Switches para varios grupos de trabajo, cada uno con diferentes características y necesidades

La velocidad del *Switch* corresponde al ancho de banda total de la red multiplicado por el número de puertos. Por consiguiente, un *Switch Fast Ethernet* (100 Mbits/s) con 10 puertos podrá proporcionar una capacidad de 1 Gbits/s. Al igual que un *Bridge*, el *Switch* que trabaja en el nivel 2 del modelo *OSI*, es capaz de seleccionar las direcciones de hardware de los frames de datos para enviar la información únicamente al segmento de red para el que está predeterminada. Las direcciones de hardware se llaman también direcciones *MAC* (*Media Access Control*) y cada tarjeta tiene asignada una dirección particular. Las direcciones de red, por ejemplo las direcciones *IP* del protocolo *TCP/IP*, se reproducen sobre direcciones *MAC*.

Capa	Descripción	Nombre en inglés	Función
7	Capa de aplicación	Application Layer	Aplicación
6	Capa de representación	Presentation Layer	Representación
5	Capa de sesión	Session Layer	Control
4	Capa de transporte	Transport Layer	Transporte
3	Capa de mediación	Network Layer	Mediación
2	Capa de protección	Link Layer	Protección
1	Capa física	Physical Layer	Transferencia de bits

El modelo *OSI* consta de siete capas o niveles (*Layers*) donde cada uno tiene asignado un cometido específico. Durante el proceso de comunicación se recorren de forma consecutiva todas las capas de abajo hacia arriba (en el emisor) y a la inversa, de arriba hacia abajo (en el receptor).

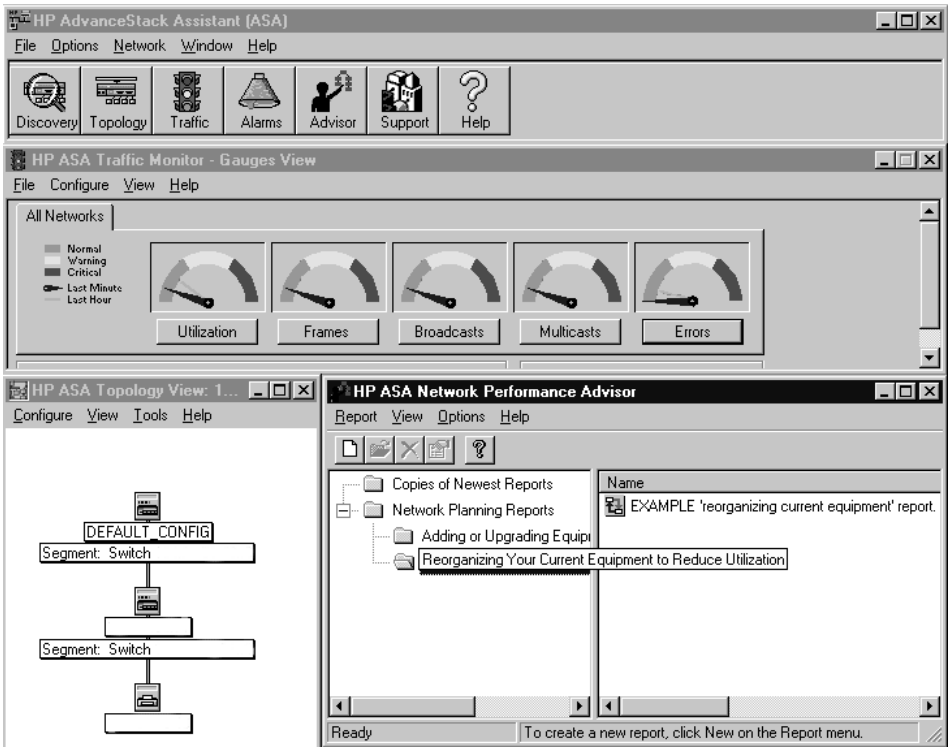
Básicamente, la información sólo se intercambia entre dos capas vecinas. Una capa sirve como medio de transporte a la que está situada encima, pone sus servicios a la disposición de la capa superior y a su vez reclama los servicios de la capa inferior.

15.4.2 Administración de red

Los *Managed Hubs* y *Switches* no sólo están diseñados para aplicar los parámetros de los dispositivos vía software sino también para poder controlar redes más complejas y configurarlas remotamente con la ayuda del software de gestión de redes correspondiente, como por ejemplo *LAN Manager* de Intel u *Open View* de HP (desde uno o varios de los equipos de administración).

Según el tipo de software existen numerosas funciones, por ejemplo la detección y análisis automático de los componentes, configuración de los parámetros de los dispositivos, representación gráfica de la carga de la red, recepción de mensajes de alarma ante determinadas situaciones problemáticas, creación de informes de alarma, definición de los valores límites del servidor, vistas de los contenidos de los paquetes, fijación del ancho de banda máximo y también la estructuración de redes *LAN* o *VPN*.

La funcionalidad de redes *LAN* o *Virtual Private Networks* (*VPN*) permite reunir las estaciones de trabajo en diferentes grupos de trabajo, en función de criterios organizativos o funcionales, y asignar a cada grupo determinadas características, que no tienen porqué ser iguales para cada uno de ellos.



Representación de la carga de trabajo y la topología (abajo a la izquierda) de una red con dos Switches y una impresora de red. El software del fabricante ofrece toda una serie de funciones (como por ejemplo Performance Advisor: Reorganizing) pero no reconoce ningún dispositivo de otro fabricante, por lo tanto deberemos utilizar el protocolo SNMP común a todos los fabricantes, y que constituye la base de casi todos los programas de gestión de redes

Prácticamente, todas las aplicaciones de gestión de red admiten el protocolo *Simple Network Management Protocol* (*SNMP*), que ya contiene una serie de funciones básicas de administración comunes. *SNMP* utiliza *TCP/IP* y se comunica con la ayuda del protocolo *UDP* (*User Datagramm Protocol*) a través de la capa de transporte (4) del modelo *OSI*. Algunas de las propiedades del protocolo *SNMP* son el *Configuration Management*, que capacita al equipo administrador para seleccionar y supervisar todos los datos de configuración de los dispositivos conectados a la red, que naturalmente también deberán ser compatibles con el protocolo *SNMP*.

La función *Performance Management* permite supervisar todo el tráfico de datos que se realiza a través de la red y al sobrepasar determinados valores umbral establecidos con anterioridad, por ejemplo el número máximo de colisiones que pueden surgir,

facilita la detección inmediata de problemas de rendimiento. La función *Fault Management* está diseñada para analizar automáticamente los errores. El protocolo *SNMP* cuenta además con otras opciones de gestión de usuarios (*Accounting Management*) y de seguridad (*Security Management*), como las que supervisan los permisos de acceso y contraseñas.

15.4.3 Otros dispositivos de interconexión

Las redes o los segmentos de red del mismo tipo, o incluso diferente, se pueden conectar entre sí con la ayuda de diferentes elementos. Según el ámbito de aplicación, estos elementos utilizan distintas capas del modelo *OSI*, y cuantas más diferencias haya en la forma de trabajar de las redes y aplicaciones que se acoplan más complejos serán los elementos para poder poner en práctica la transferencia y conversión de datos.

Repeater

Un *Repeater* puede presentar formas y estructuras muy diferentes. En el caso más sencillo no es más que un amplificador de las señales que intercomunica entre sí redes o segmentos de red del mismo tipo.

También es posible realizar una conversión a otro medio de transferencia, por ejemplo de cobre a fibra óptica, por lo que también se denomina conversor de medios. Con el uso del *Repeater* el sistema operativo y los programas no obtienen ningún tipo de información ya que el acoplamiento tiene lugar en la capa 1 del modelo *OSI*. Con un *Repeater* tampoco se distribuye una red en segmentos, únicamente sirve para aumentar la extensión de una red.

Bridges

Al igual que los *Repeaters*, los *Bridges* "realizan un puente" de señales pero de una forma más selectiva, es decir, poseen una función de filtro. ‡nicamente se transfieren aquellas señales que están dirigidas a una estación en particular. Como los *Bridges* intercambian los datos a través de la capa 2 (nivel MAC) del modelo *OSI*, puede diferir la capa física de las redes que se interconectan.

La selección de datos se realiza analizando las direcciones enviadas en los paquetes. Cuando se selecciona una determinada estación de la red, el *Bridge* "anota" en qué segmento de red se encuentra esa estación y registra su dirección en una tabla *Routing*. Si la tabla no posee todavía el registro correspondiente, el *Bridge* conduce el paquete de datos a todos los segmentos excepto al suyo propio y a partir de las reacciones podrá detectar qué segmento del sistema y qué estación ha aceptado los datos y en consecuencia incluye ese registro en la tabla *Routing*.

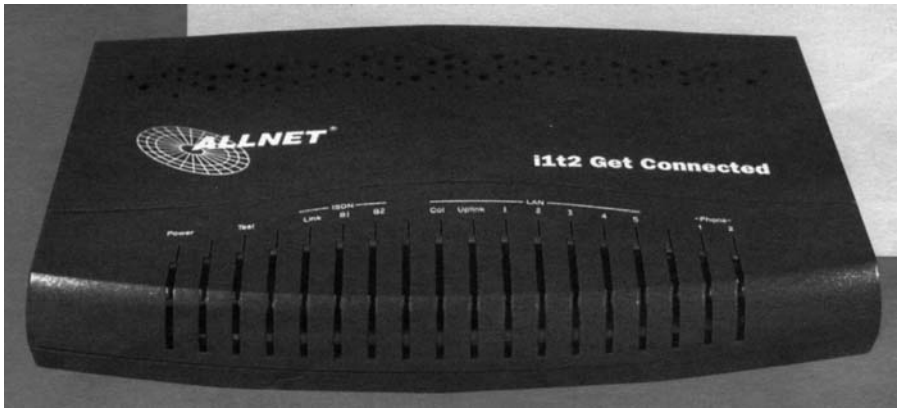
Router

Los *Routers* trabajan sobre la capa 3 del modelo *OSI* y, por consiguiente, las capas 1 y 2 de las redes que se acoplan pueden diferir entre sí. Tienen la facultad de poder pro-

cesar tanto diferentes tipos de red como protocolos distintos. Existen *Routers* como elementos de hardware y también como soluciones de software implementadas, como en *Windows NT/XP*, para lo que generalmente se pueden instalar dos tarjetas de red.

En muchas ocasiones, un *Router* está diseñado para un protocolo de red concreto (por ejemplo *TCP/IP*, *IPX/SPX*), aunque también existen *Routers* multiprotocolo, es decir, que entienden diversos protocolos. Windows NT/2000/XP se puede desenvolver con *TCP/IP* y con el protocolo *IPX/SPX*, lo que en Microsoft ya se designa como *Router* multiprotocolo. Con el protocolo de red *NetBEUI* de Microsoft no es posible acceder directamente a *Internet*, para ello se utiliza en su lugar el protocolo *TCP/IP*.

Para otras conversiones de protocolos, por ejemplo de *Ethernet* a *Token Ring*, se suele interponer un ordenador dedicado a esta tarea en exclusividad, que deberá disponer de un procesador potente y bastante capacidad de memoria para modificar y adaptar convenientemente la longitud y velocidad de los paquetes de datos.



Un Router RDSI con seis puertos de 10 Mbits (RJ45, par trenzado) es una solución interesante para pequeñas oficinas que desean utilizar en común un acceso a Internet vía TCP/IP. Incorporando un adaptador S/T podremos conectar además dispositivos analógicos como un fax o un teléfono

Los *Routers* pueden analizar las direcciones lógicas de una red y mediante la tabla *Routing* averiguar los caminos óptimos del emisor al receptor. En este aspecto diferenciamos entre *Routers* dinámicos y estáticos. En los últimos se establecen de forma manual los distintos caminos (rutas), para ello existe por ejemplo la instrucción "*route*" en *Windows NT Server* (4.0). La configuración de la tabla puede ser laboriosa pero tiene la ventaja de que es bastante segura. En un *Router* de hardware, la configuración de la tabla se suele realizar con un programa *Terminal*.

En los *Routers* dinámicos, las rutas se detectan automáticamente y por lo tanto se suprime el proceso de configuración manual. Para crear la tabla, los *Routers* se comunican entre sí e intentan definir los caminos más cortos posibles. Para ello se suma a la red el protocolo *Routing Information Protokoll* (*RIP*), que aproximadamente cada 30 segundos establece conexión con los otros *Routers*. Esto puede originar problemas de seguridad ya que los paquetes de datos se transportan prácticamente por todas las

redes participantes. Además del *RIP* existen otros protocolos de routing como por ejemplo *IGRP* (*Interior Gateway Routing Protocol* de la casa Cisco) o *NLSP* (*Netware Link State Protco*l), donde algunos *Routers* admiten varios de estos protocolos al mismo tiempo.

Brouter

Un Brouter es una combinación de un *Bridge* y un *Router*. Básicamente se trata de un *Bridge* al que se le ha añadido un protocolo propio de *routing*. El *Brouter* se hace cargo del filtrado y de la transferencia de datos y utiliza un direccionamiento directo. Por lo tanto, las redes *LAN* conectadas pueden usar protocolos diferentes. Al *Brouter* le acompañan tablas de red y *routing* que le permiten seleccionar caminos de red alternativos y redundantes.

Gateways

Con un *Gateway* es posible comunicar redes totalmente diferentes entre sí, redes que en principio no tienen nada en común, de tal forma que todas las capas *OSI* se tapan.

Un *Gateway* cumple las funciones de un *Router* y además dirige la conversión de los protocolos y códigos. Un *Gateway*, para el que generalmente se utiliza un ordenador propio, viene a ser un intérprete entre las redes que utilizan distintos protocolos. Por ejemplo, una red de ordenadores IBM trabaja con el código de caracteres *EBCDIC* y los PC de otra red con el código *ASCII*, por consiguiente las tareas de un *Gateway* deben extenderse por las siete capas del modelo *OSI*, algo que no es fácil de realizar.

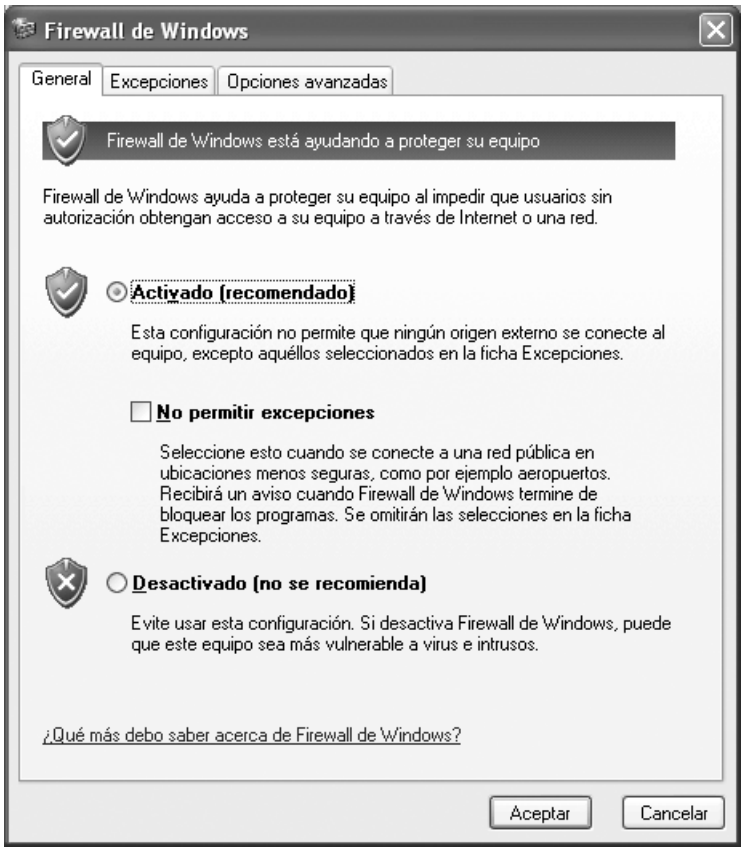
En primer lugar, un *Gateway* se hace cargo de los datos de la otra red, elimina la pila de protocolo, después de lo cual "implanta" los datos en el protocolo de la red de destino a través de una nueva pila de protocolo. Este proceso requiere de su tiempo por lo que la mayoría de las aplicaciones de un *Gateway* se limitan a determinadas funciones como transferencia de archivos y emulaciones de terminal.

Firewall

El *Firewall* entra en acción para proteger una red de las posibles incursiones externas no deseadas (por ejemplo de *Internet*). Su tarea consiste en supervisar todos los datos que se transfieren en función de diversos criterios configurables, y en caso de que esos datos no sean permitidos impide su acceso a la red local situada detrás del *Firewall*. Esto convierte al *Firewall* en una esclusa de *Internet* idónea, ya que nunca se puede estar seguro de qué viene de esa enorme red para escudriñar los datos de una red local o para ocasionar algún daño.

Por lo tanto, el *Firewall* debería constituir la única vía de comunicación hacia *Internet*, o también hacia otras redes internas. En primer lugar se encarga de que sólo exista una sola vía de comunicación a través de la cual fluyan todos los datos, y gracias a ello es capaz de documentar cuándo se ha transferido un determinado tipo de datos.

Las medidas de seguridad que se le otorgan a un *Firewall* pueden verse reflejadas de formas distintas, depende del nivel de seguridad exigido y de qué se puede considerar permitido. Algunas de estas posibilidades consisten en limitarse a determinadas direcciones *IP*, o impedir el acceso a logins y remitentes desconocidos. Tanto *Linux* como *Windows XP* ofrecen la posibilidad de usar sus propios *Firewalls* seguros e individuales.



El cuadro de diálogo de configuración del Firewall de Windows XP

16. Ampliar y modificar

Para los usuarios que no deseen cambiar de ordenador cuando las prestaciones del actual se quedan cortas, la solución más económica es ampliar o cambiar algunos componentes del PC. Una nueva tarjeta gráfica o quizás un poco más de memoria *RAM* y un disco duro adicional, si es posible un procesador más rápido o una nueva placa base son las típicas medidas de reconversión que veremos a continuación.

16.1 Medidas de seguridad

Antes de abrir la caja del ordenador siempre debemos que desconectar el PC de la toma de alimentación; para más seguridad deberíamos desconectarlo de la toma general y no de la base de enchufes, ya que podríamos equivocarnos de enchufe y correr un serio peligro.

Hay que mencionar que una fuente de alimentación *ATX* no se queda sin corriente si únicamente accionamos el botón de la parte frontal del equipo. En una fuente de alimentación *AT* la situación es distinta pero, en cualquier caso, es obligado desconectar el enchufe general antes de comenzar a quitar los tornillos.

Por razones de seguridad, antes de tocar con las manos una tarjeta, un módulo de memoria o sobre todo un procesador es conveniente descargarse de la posible carga electrostática a través de un metal con toma de tierra. Esta carga electrostática puede significar un peligro ya que una descarga de este tipo puede dañar los distintos componentes electrónicos del equipo. Pero tampoco es imprescindible utilizar una pulsera especial que se conecta a una toma de tierra para descargarnos de la electricidad estática, a pesar de que es una recomendación habitual en la mayoría de manuales.

16.2 El montaje de las tarjetas de expansión

Si hemos decidido adquirir una nueva tarjeta de expansión e instalarla en el ordenador es conveniente aclarar algunas cuestiones antes de realizar la compra: de qué tipo de tarjeta se trata (sonido, módem, gráfica o una controladora *SCSI*) es al principio una cuestión secundaria ya que la forma de proceder es casi siempre la misma.

- Para qué sistema de bus es la tarjeta?
- Disponemos de *slots* libres para esa tarjeta?
- Las dimensiones de la tarjeta se ajustan a las del PC?
- Impide el procesador u otro elemento de la placa base la instalación de la tarjeta?
- Cabe la posibilidad de que la tarjeta tape o bloquee alguna de las ranuras de expansión cercanas?
- Es preferible instalar una tarjeta *PCI*, normalmente más potente que una *ISA*, y además se configura automáticamente a través del *BIOS*.
- Si se trata de una tarjeta *ISA* es necesario que, por lo menos, se corresponda al estándar *Plug & Play*.

16.2.1 El sistema de bus

Podemos imaginar un sistema de bus como una concentración de señales diseñada en forma de conector especial, llamado ranura de expansión o slot en los PC, para albergar las distintas tarjetas de expansión. En principio, todas las ranuras de un sistema de bus tienen los mismos derechos, es decir, no es relevante en qué ranura insertamos la tarjeta ya que todas las líneas de señal van de una ranura a otra. Pero en el bus *PCI* y también en algunos sistemas más anticuados, como *EISA*, existen algunas excepciones, y también existen las denominadas señales específicas de ranura, que la placa base envía de forma individual a un único *slot*.

16.2.2 Montaje de la tarjeta

Después de constatar que el PC dispone de las ranuras libres y adecuadas es imprescindible comprobar si la fuente de alimentación del ordenador, una tarjeta adyacente o una unidad podrían impedir la instalación de otra tarjeta, bien sea porque ésta es demasiado larga o porque no se adapta a la ranura correspondiente. Si en el PC no es posible instalar ni tarjetas *ISA* ni tampoco *PCI* hemos llegado al final de las posibles ampliaciones.

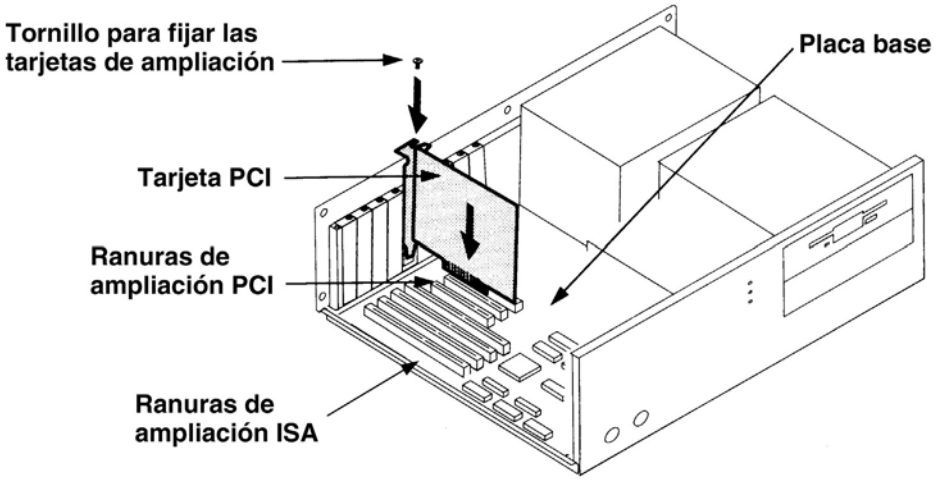
La forma de abrir la caja del ordenador (°desenchufar siempre antes la toma de corriente!) depende de cada modelo de caja (sobremesa, torre, etc.). No obstante, este paso no suele plantear muchas dificultades. A continuación se quita una de las pequeñas cubiertas de la cara posterior del ordenador (una pletina metálica) quitando su tornillo de sujeción. Más tarde se necesitará este tornillo para sujetar la tarjeta de expansión.

Antes de insertar la tarjeta en su ranura es preciso asegurarse si es necesario configurar algún parámetro con *jumpers* o interruptores *DIP* y, si es así, cuáles en concreto. Incluso en las nuevas tarjetas adaptadoras *SCSI* hay que colocar a veces un *jumper* para cambiar entre terminación automática (vía *BIOS*) y fija. Para estar seguros es indispensable consultar el manual de la tarjeta.

Generalmente, la tarjeta gráfica se coloca en un lugar lo más alejado posible de la fuente de alimentación. Con una tarjeta *AGP* no se tiene otra opción que insertarla en el único *slot* disponible. También podemos decir lo mismo con respecto a la tarjeta de sonido, uno de los componentes más sensibles del PC. Además, tampoco es conveniente colocar la tarjeta de sonido justo al lado de la tarjeta gráfica, ya que ésta unidad y también la fuente de alimentación generan señales de alta frecuencia que pueden notarse al reproducir y sobre todo al grabar con la ayuda de una tarjeta de sonido.

Una fuente de alimentación con un aislamiento incorrecto o defectuoso también puede influir en la tarjeta gráfica y perturbar la reproducción de las imágenes. Asimismo, la controladora del disco duro puede causar anomalías en la tarjeta de sonido, pero en cualquier caso hay que encontrar una solución para colocar la tarjeta, y las posibilidades son limitadas. Ante estas circunstancias sería necesario averiguar, por medio de pruebas, el lugar idóneo para cada tarjeta de expansión.

Si la tarjeta no se ajustase en la ranura de forma uniforme es adecuado mantenerla en una posición oblicua, introducir primero el extremo posterior y luego insertar el otro extremo (el lado de la chapa) ejerciendo algo de presión.



Montar una tarjeta en el PC es una operación realmente sencilla

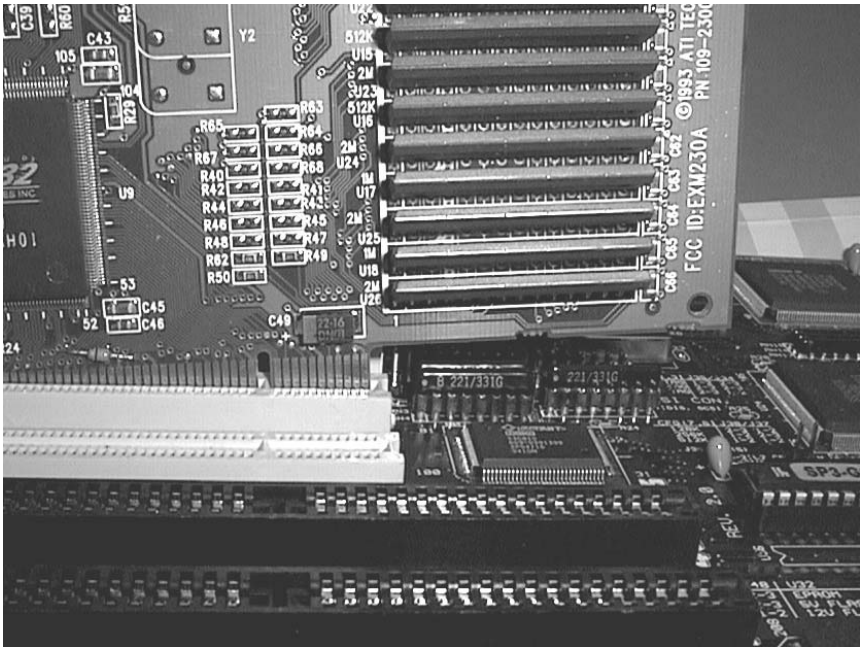
Ante todo, en esta operación es importante comprobar que los contactos se establecen correctamente. Después de insertar la tarjeta es conveniente encender el ordenador en modo de prueba. Si arranca con normalidad podemos empezar a estar satisfechos.

En caso contrario apagaremos inmediatamente el PC y repasaremos, de nuevo, tanto la colocación de la tarjeta como los parámetros configurados manualmente. Sólo hay que cerrar la caja del ordenador cuando se ha completado la instalación y se ha comprobado que la tarjeta funciona correctamente.

16.2.3 Problemas mecánicos al montar una tarjeta

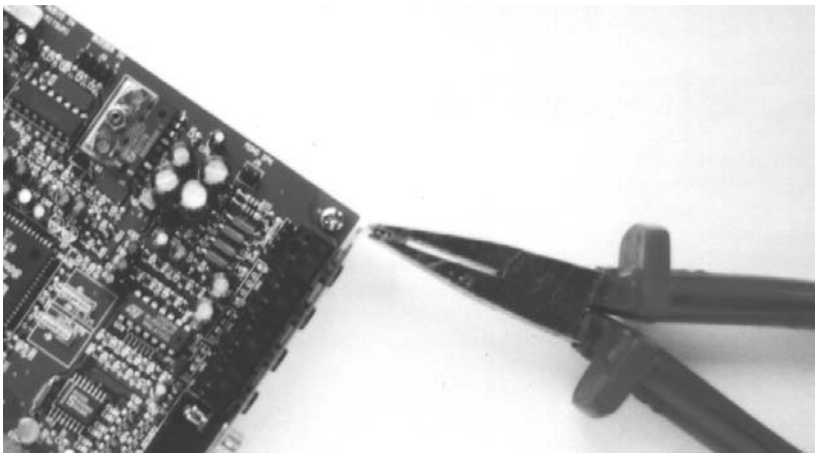
Todas las tarjetas tienen en común que los contactos de la conexión con la ranura están dispuestos a lo largo de una placa. Al igual que los circuitos impresos, cada contacto está sobre la pletina, es decir, a ambos lados de la tarjeta. Dichos contactos no están protegidos de la suciedad y la extracción e inserción continua de la placa en la ranura puede desgastar los contactos de la tarjeta.

Además nos podemos encontrar con tarjetas, sobre todo de los modelos antiguos *ISA*, que en el ámbito mecánico no estén bien acabadas y se muevan un poco dentro de la ranura de la placa base. Esta pequeña holgura es suficiente para que los contactos de la tarjeta no se unan con los contactos que les corresponden en la ranura y se produzca un cortocircuito que impediría arrancar el ordenador. Esto puede ocasionar serios daños tanto en la placa base como en la(s) tarjeta(s). Por esta razón, al insertar una tarjeta es conveniente comprobar que está bien colocada en la ranura y los contactos de la tarjeta coinciden exactamente con sus homólogos de la ranura. Al anclar la tarjeta con el tornillo (la pletina y la caja) puede suceder que, bien por razones mecánicas o debido a que la placa base no está bien montada, la tarjeta se desplace ligeramente hacia arriba y la parte más alejada de la pletina se salga de la ranura. En ese caso ya no se establece la conexión correcta y también puede causarse algún daño en los diversos componentes electrónicos. Algunas cajas de ordenador cuentan con agujeros alargados de ajuste que permiten mover un poco todo el marco en el que se sujetan las tarjetas y de esta forma es posible solucionar estos desajustes mecánicos entre la placa base y el asentamiento de las tarjetas.



Un extremo de la tarjeta se sale de la ranura debido a un desajuste mecánico

En ocasiones, la tarjeta no se puede sujetar bien con los tornillos sin que con ello se salga de su ranura. No es aconsejable mantener esta situación durante mucho tiempo ya que sólo con insertar un conector en la tarjeta o con mover el ordenador, la tarjeta se puede deslizar de su posición y provocar un cortocircuito. Por tanto, es necesario encontrar la manera de sujetar bien la tarjeta. Si la pletina de la tarjeta se eleva algo sobre el punto de anclaje de la caja podemos colocar una arandela. También se puede modificar algo la pletina de la tarjeta con unos alicates para que la línea de contactos se asiente correctamente en la ranura.

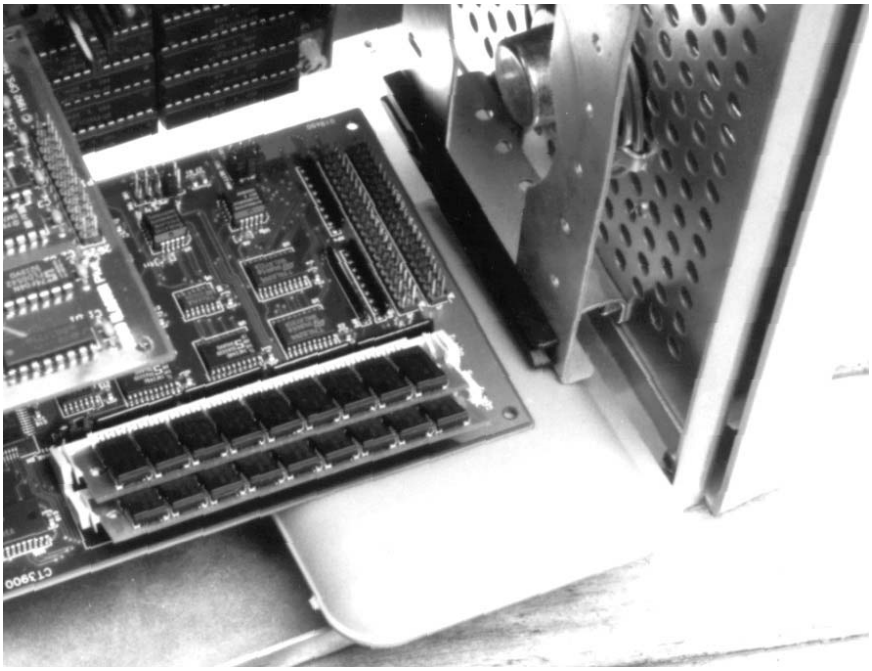


En ocasiones es necesario doblar la pletina de la tarjeta con unos alicates para poder introducir correctamente la tarjeta en la ranura y luego sujetarla a la caja

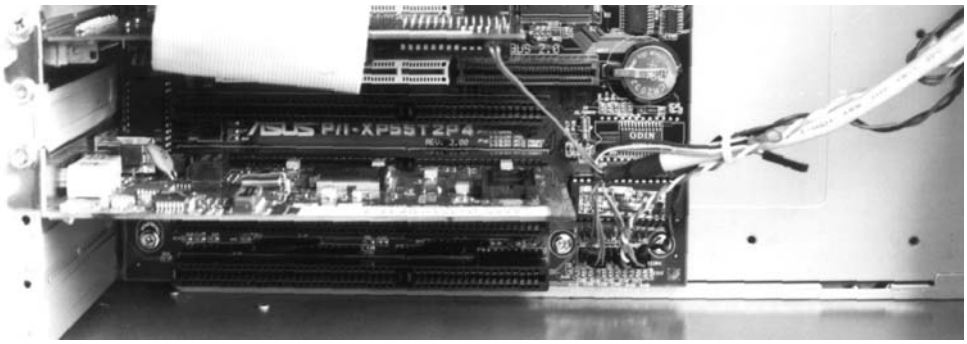
El montaje de una tarjeta gráfica *AGP* requiere una especial atención. Es necesario presionar con fuerza para insertarla en la ranura y, sin embargo, se sale con relativa facilidad. Por este motivo y para fijar de forma duradera la tarjeta AGP es recomendable utilizar una abrazadera adicional (de aspecto similar a una brida para cables), pero lamentablemente los fabricantes suelen olvidarse de ella. Con relativa frecuencia sucede que el monitor permanece inactivo únicamente porque la tarjeta *AGP* se ha salido de la ranura.

Hemos mencionado que la tarjeta no debe ser demasiado larga para que quepa en el interior del ordenador y el ventilador del procesador, o cualquier otro componente, no impida su correcta instalación y funcionamiento. Pero también nos podemos encontrar con el caso contrario y que la tarjeta sea demasiado corta. Con el estándar *ATX* este tipo de problemas mecánicos pudieron dejarse a un lado, aunque algunos fabricantes toman medidas realmente sorprendentes al montar sus equipos.

Las líneas que comunican la placa base con los indicadores y controles de la cara frontal del ordenador se agrupan en un único cable, todo muy ordenado. Pero a veces, este cable resulta demasiado corto y al instalar una tarjeta más larga no se puede apartar, como pasa en el ejemplo de la figura siguiente. La solución pasa por cambiar el cable por uno más largo o bien sustituirlo por varias líneas separadas, lo que conlleva un trabajo bastante delicado y que requiere de cierta experiencia.



Una mala solución mecánica: la tarjeta cuelga sin "relleno" en una caja torre porque no es lo suficientemente larga para alcanzar los soportes de plástico de la caja.



El cable que comunica la placa base con los indicadores y controles del panel frontal tiene una longitud corta, lo que impide instalar tarjetas más largas

16.3 La ampliación de la memoria

Los ordenadores actuales suelen disponer de 64, 128 o incluso 256 MB de memoria *RAM*. Ampliar la memoria de 128 MB, o menos, hasta los 256 MB es una medida de optimización recomendable y desde el punto de vista del rendimiento puede ser más productivo que sustituir un procesador *Pentium III* con 650 MHz por otro procesador (*Pentium 4)* de mayor frecuencia. Si nuestro ordenador actual posee menos de 128 MB de *RAM*, ampliarla sería la primera medida de optimización que deberíamos realizar antes de acometer otras mejoras.

Deberíamos ver cada componente del PC, como la memoria *RAM*, el procesador, la memoria caché, el disco duro o la tarjeta gráfica como elementos interrelacionados. Disponer de más memoria *RAM* no aporta muchas ventajas si el procesador o el disco duro son muy lentos. Encontrar y combinar las unidades idóneas es, en cierta manera, el método seguido por un buen "manitas" del PC, y también la razón principal por la que merece la pena montar uno mismo el ordenador a partir de componentes aislados.

16.3.1 La dotación correcta de memoria

Los viejos módulos *SIMM* de 30 contactos estaban formados por *chips DRAM* estándar. Posteriormente se utilizaron para dichos módulos tipos de memoria *Page Mode* y luego *Fast Page Mode* (*FPM*) que consiguieron disminuir los tiempos de acceso.

En un principio, los módulos *PS/2* se componían de chips FPM y más tarde se sustituyeron por módulos basados en *chips EDO-RAM* (*Extended Data Out*) o también *Burst EDO DRAM* (*BEDO*). Además del tipo, otro criterio de selección de la memoria es su tiempo medio de acceso, que normalmente oscila entre los 150 y 60 nanosegundos. El tipo de *chip* que se utiliza depende de la frecuencia (externa) del procesador, la frecuencia del sistema.

Al diseñar la ampliación de memoria es muy importante comprobar que todos los *chips* de memoria posean los mismos tiempos de acceso, ya que en caso contrario pueden surgir errores de memoria de tipo *Memory Error* o *Parity Error*. En contra de la teoría generalmente aceptada esto también puede suceder cuando los nuevos *chips* de memoria son más rápidos que los ya presentes y depende fundamentalmente del juego de *chips* de la placa base. Si los programas no arrancan o se bloquean es posible activar ciclos de espera (*Wait States*) para la memoria a través del *Setup* del *BIOS*.

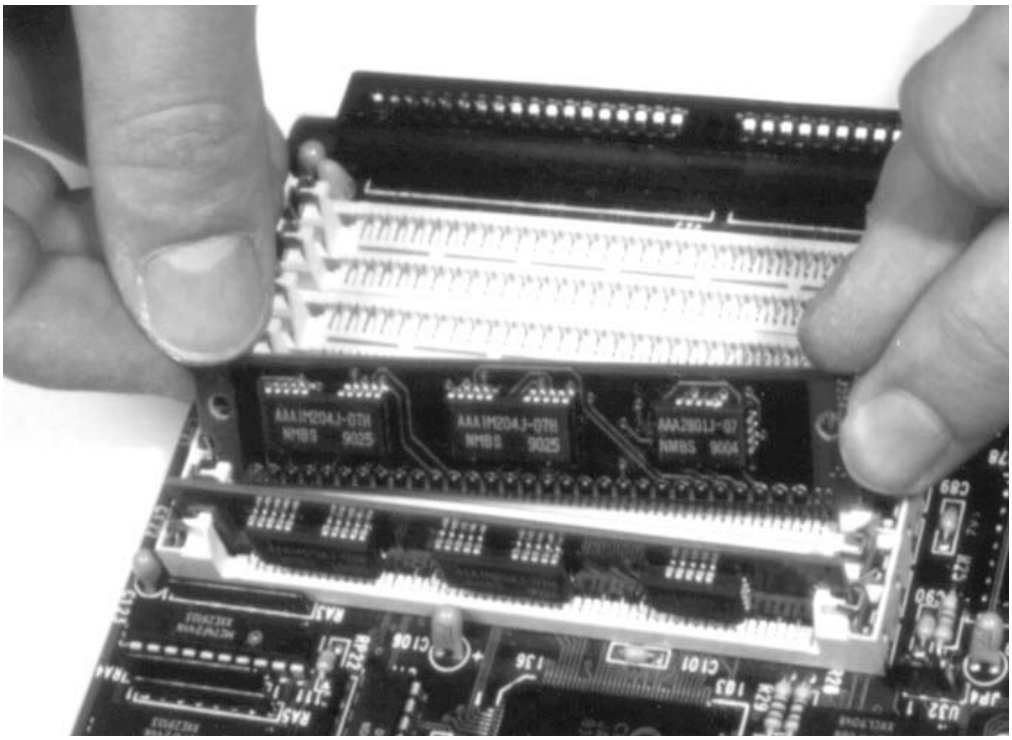
Los módulos con función *ECC* (*Error Correction Code*) ofrecen un sistema de detección y corrección de errores de memoria más eficaz. Las placas base a partir del *Pentium Pro* soportan dicha función. Generalmente, las placas base pueden manejar ambos tipos de módulos y, a veces, esta función se puede activar o desactivar a través del *Setup* del *BIOS*.

16.3.2 El montaje de los módulos

El montaje o más exactamente la inserción de los módulos es una operación sencilla. No obstante, en algunos modelos de ordenador (cajas pequeñas) es necesario quitar antes la fuente de alimentación o una unidad para poder acceder a las ranuras correspondientes. En cualquier caso, es imprescindible observar las medidas preventivas antes mencionadas.

Para montar un módulo SIMM, éste se mantiene en una posición inclinada (aprox. 45), se introduce en la ranura y luego se pone derecho, algo que sólo es posible si los contactos se acoplan correctamente y además la muesca lateral del módulo coincide

con la forma de la palanca de plástico de la ranura. El contacto 1, señalado en algunas placas base, se encuentra en el borde del módulo junto a la muesca de la placa.

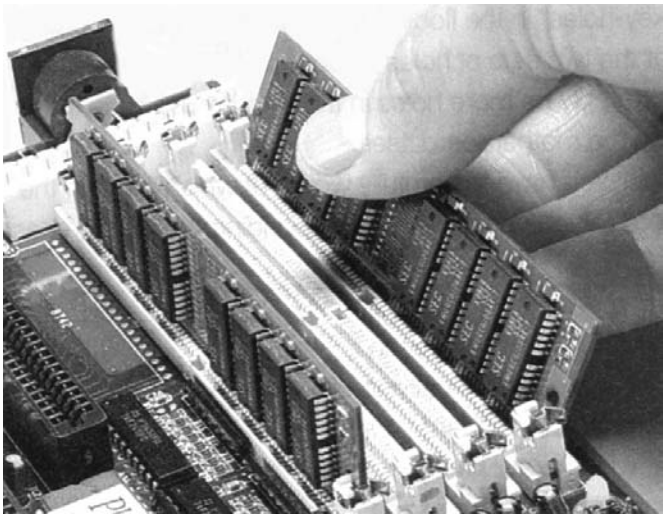


Para el montaje de un módulo SIMM es necesario tener en cuenta la muesca de la placa que identifica al contacto 1 (a la izquierda)

Es conveniente colocar siempre primero el módulo en la ranura situada más al fondo ya que luego, al disponer de menos espacio de maniobra, resultará mucho más difícil colocar el resto si no se tiene mucha experiencia. Debemos tener presente que como los módulos se presentan frente a la ranura en posición oblicua, los componentes ya colocados pueden impedir insertar el último de ellos.

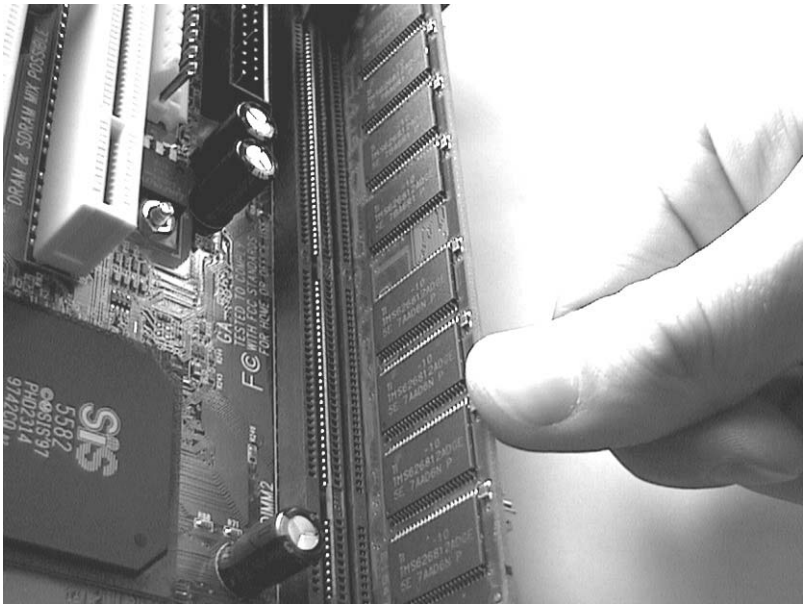
Como ya hemos comentado es necesario ocupar por completo un banco, y en la mayoría de los casos las ranuras que lo componen son adyacentes. Sin embargo, hay excepciones en algunas placas base con ranuras *PS/2 SIMM*. En esos modelos el banco 0 se compone de las ranuras 1 y 3, y el banco 1 de las ranuras 2 y 4. Primero hay que ocupar por completo el banco 0, luego el banco 1 y así sucesivamente.

Este orden es obligatorio en las placas base antiguas mientras que en las modernas suele ser indiferente si se insertan los módulos en el banco 0 o en el banco 1. En todo caso, ocupar los bancos respetando su orden es la mejor forma de atajar desde el principio cualquier problema posible.



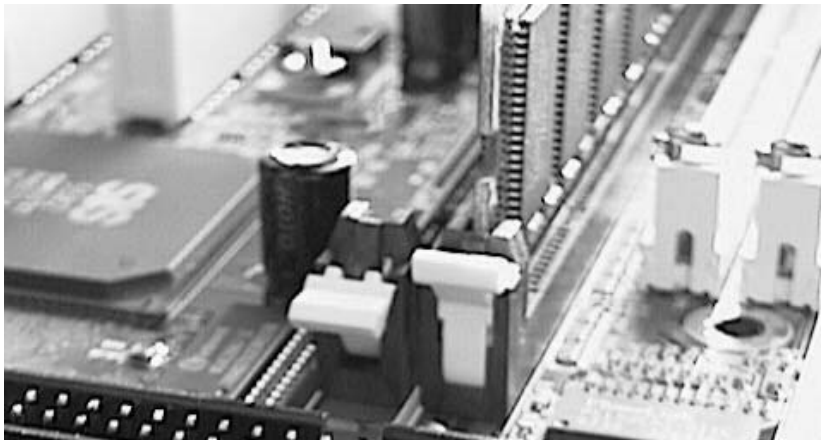
En el montaje de un módulo PS/2 SIMM, éste se sujeta en la ranura con dos abrazaderas de metal

Al colocar derecho el módulo sobre su ranura puede haber algún problema ya que es preciso presionar hacia los lados los brazos metálicos de sujeción. Generalmente, extraer los módulos es algo más difícil aunque a veces es necesario utilizar un destornillador a modo de palanca para hacer a un lado la abrazadera. Si no funciona así intentaremos apartar las dos abrazaderas hacia fuera con ambos pulgares para liberar el módulo de sus sujeciones.



Un módulo DIMM se inserta en la ranura presionando desde arriba

Los módulos *DIMM* que se utilizan en los ordenadores más modernos poseen otro mecanismo de sujeción. Estos módulos se insertan y se extraen de sus ranuras sin necesidad de colocarlos en posición oblicua. Una vez que el módulo se ha introducido correctamente, algo que a veces requiere emplear un poco de fuerza, el brazo de plástico se cierra automáticamente hacia arriba, tal y como se puede comprobar en la figura siguiente.



El módulo DIMM está correctamente asentado cuando el brazo de plástico se cierra hacia arriba

16.3.3 Solucionar problemas de memoria

Una vez se han insertado todos los módulos no suele ser necesario definir ninguna configuración con un *jumper* o a través del *Setup*, ya que el *BIOS* averigua automáticamente la cantidad de memoria instalada. Después de arrancar el ordenador, el BIOS muestra en pantalla la memoria disponible.

Después de modificar la memoria de trabajo en algunas versiones *BIOS*, por ejemplo de la casa AMI, aparece al principio un mensaje de error de tipo *Memory Size Error Run Setup*. Esto significa que el *BIOS* ha detectado un cambio en la memoria con respecto a la última vez que se inició. A continuación se activa el programa *Setup* del *BIOS* y si lo abandonamos sin hacer ningún tipo de modificación, con la opción *WRITE TO CMOS AND EXIT*, se acepta la nueva cantidad instalada y ya no aparece más el mensaje de error.

Naturalmente, también puede ocurrir que no se detecte de forma correcta el nuevo volumen de memoria o incluso que el monitor permanezca inactivo. Antes de que el *BIOS* muestre los valores de memoria correctos no tiene sentido utilizar un programa de testeo, en su lugar pueden servirnos de ayuda algunas de las pruebas de comprobación de la siguiente lista:

- Si el ordenador no funciona en absoluto, podemos comprobar en primer lugar la toma de alimentación, las conexiones y el estado de las tarjetas de expansión.

- Comprobar que los módulos se asientan correctamente en sus ranuras correspondientes.
- Repasar la disposición de los bancos de memoria con ayuda del manual de la placa base.
- Supervisar los tiempos de acceso.
- Si podemos acceder al *Setup* del *BIOS* es conveniente configurar a modo de prueba los valores más bajos para la memoria.
- Comprobar la necesidad de módulos con función de paridad o *ECC*.
- Si sólo funciona la memoria de un banco, intercambiaremos los módulos de las ranura para así averiguar, paso a paso, si existe algún módulo dañado.
- Si se utilizan módulos *PS/2 SIMM* y *DIMM*, comprobar la posición de los *jumpers* y la tensión de alimentación.

16.4 Instalación de unidades

A nivel general, y con excepción de las unidades de disquete, las unidades internas del PC utilizan una interfaz *EIDE*, a la que se puede conectar un máximo de cuatro unidades, o la interfaz *SCSI* (opcional), disponible en forma de adaptador *host*. El tipo de dicho adaptador depende de cuántas unidades se desean conectar. El adaptador *SCSI* de 8 bits permite conectar ocho dispositivos y el de 16 bits (*Wide*) dieciséis.

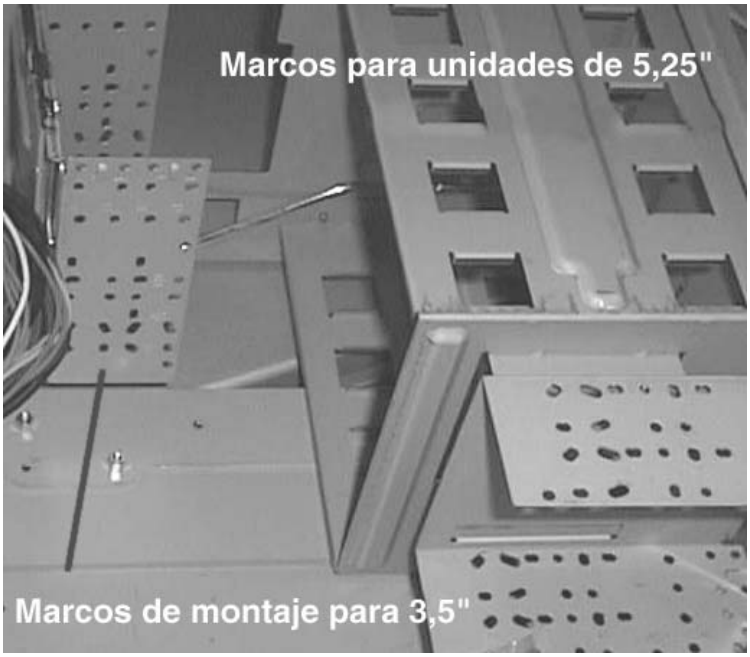
Las unidades que se suelen incorporar a los PCs son unidades *ZIP* y *LS-120*, grabadoras de *CD/DVD*, unidades de *DVD-ROM* y dispositivos para las copias de seguridad (*Streamer*). En principio, la forma de proceder con todas ellas es siempre la misma y, naturalmente, la caja del ordenador deberá disponer del área libre correspondiente para poder montar estos dispositivos.

Si no es éste el caso, los dispositivos externos representan una alternativa muy interesante hoy en día. Entonces para conectar los aparatos entran en escena los puertos *SCSI*, paralelo o también el *Universal Serial Bus* (*USB*).

Según el tipo de dispositivo, el área libre dentro de la caja deberá tener una anchura de 5.25 o 3.5 pulgadas, dimensiones tradicionales introducidas ya con las unidades de disquete de 5.25 y 3.5 pulgadas.

Estas bahías de la caja cuentan con una lámina de plástico y en las cajas más nuevas también se debe extraer una chapa de metal situada detrás, que constituye una pantalla de protección frente a las radiaciones del PC. Lo ideal es que la caja posea en su interior, justo detrás de la lámina de plástico, bahías, marcos de montaje o una especie de jaula metálica donde poder atornillar los dispositivos.

Los discos duros se suelen instalar dentro de estas estructuras y, especialmente, en las cajas pequeñas los discos duros se montan allí donde hay un hueco libre, que generalmente cuenta con los elementos de sujeción pertinentes.



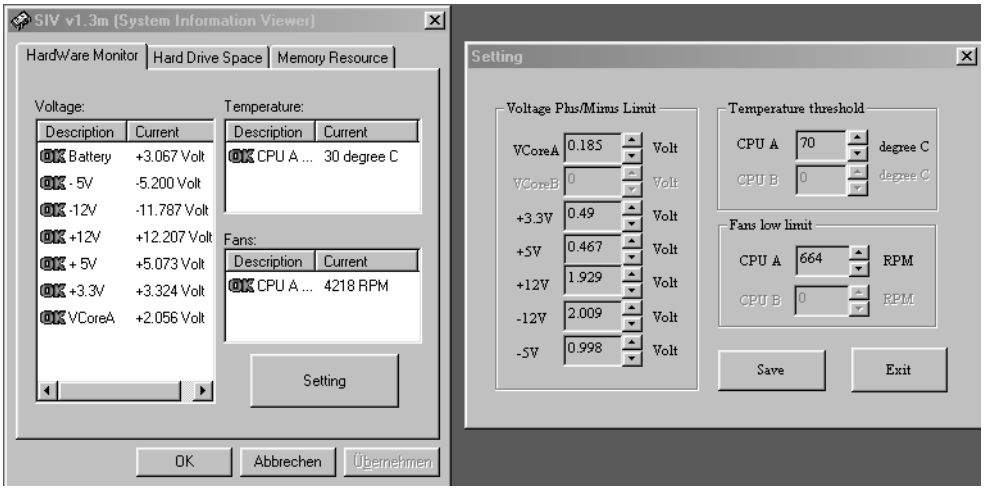
En estas cajas se pueden instalar un máximo de cuatro unidades de formato 5,25", debajo otra de 3,5", mientras que el disco duro se suele instalar en el marco interno de 3,5 pulgadas (izquierda)

Las cajas grandes tipo torre son especialmente idóneas para incorporar nuevas unidades. Cuanto mayor sea la caja más unidades se podrán instalar, y también podríamos dar por hecho que la fuente de alimentación es capaz de asimilar las nuevas unidades además de las ya instaladas de serie. Sin embargo, en la práctica no siempre es así, y no es tan raro encontrarnos con que el ordenador no trabaja de forma estable después de incorporar dispositivos adicionales. Entonces, la fuente de alimentación debe proporcionar más energía y, para ello, la tensión disminuye por debajo de los valores especificados, lo que puede provocar la inestabilidad del sistema.

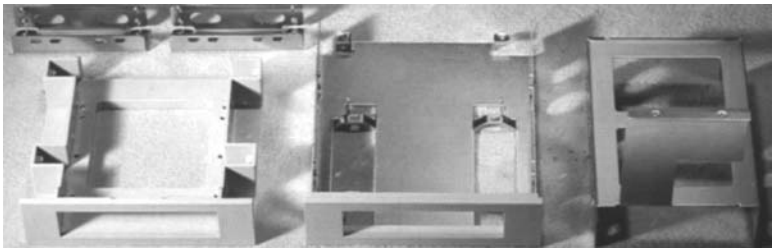
Si la placa base dispone de un *chip Supervisory* o de otro *System-Health-Chip* como puede ser el *Winbond* W83781D, con su programa complementario podremos comprobar en pantalla datos como la tensión de trabajo o la temperatura del procesador. En ese caso, como también a través del *Setup* del *BIOS*, es posible fijar los valores críticos correspondientes y si se sobrepasan se activa una señal de alarma para poder detectar el origen del problema.

Por tanto, es conveniente controlar con el programa esos valores, sobre todo al ampliar el sistema. Si la placa base no posee esta función podremos medir la tensión de alimentación con ayuda de un *tester*.

Puede darse la circunstancia de que haya que montar una unidad *ZIP* u otra de 3.5 pulgadas y sólo queden libres marcos de montaje grandes (5.25 pulgadas). Para estos casos existen marcos adaptadores o carriles donde colocaremos una chapa adicional.



Supervisión de las tensiones y de otros valores importantes en el funcionamiento del PC con el System Information Viewer



Marcos adaptadores y carriles de sujeción para instalar unidades de 3.5 pulgadas adicionales

16.4.1 Correctamente montado y conectado

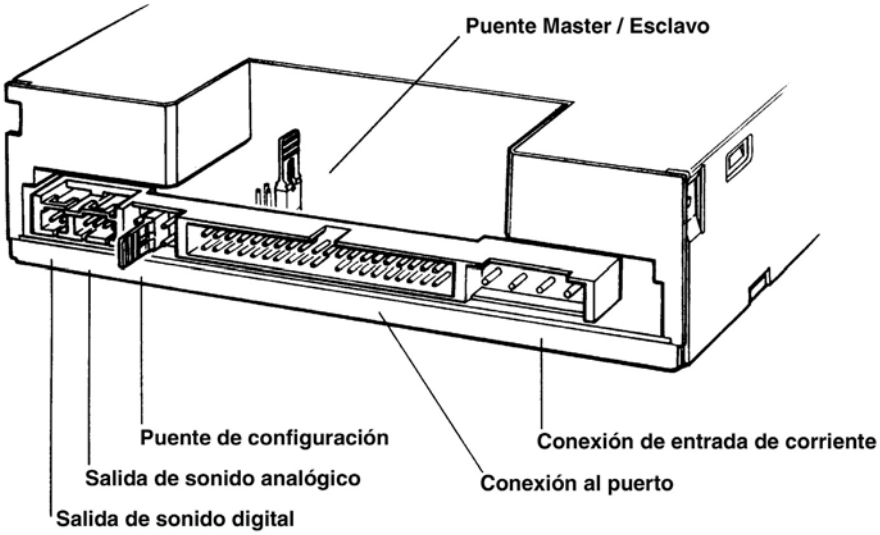
La conexión de dispositivos (*E)IDE* (*AT*, *ATA*, *ATAPI*) se realiza con un cable de 40 contactos, uno de los conectores se une a la conexión del controlador que se encuentra en una tarjeta específica o en la misma placa base. Una excepción a esta regla la constituyen los cables especiales de 80 líneas, para discos duros compatibles con el estándar *Ultra 66/100*, aunque los conectores continúan siendo de 40 contactos.

Es importante prestar atención a la marca correspondiente al *pin* 1, ya que en muchas ocasiones el conector también se puede acoplar en la posición inversa. Si los cables se conectan al revés no hay porqué temer daños de importancia pero seguramente se ahorrará gran cantidad de trabajo si observamos desde un principio la posición correcta de los conectores.

En cada puerto *EIDE* se puede conectar un máximo de dos dispositivos, uno de los cuales se configura como maestro con un *jumper* y otro como esclavo. El cable de 40 contactos, provisto de tres conectores, va de la placa base a un dispositivo y desde allí al siguiente. En esta disposición es indiferente qué posición en la cadena ocupa el dispositivo maestro y cuál el esclavo.

La muesca en el marco de plástico de estas conexiones IDE en una placa base evita cualquier equivocación al conectar los cables

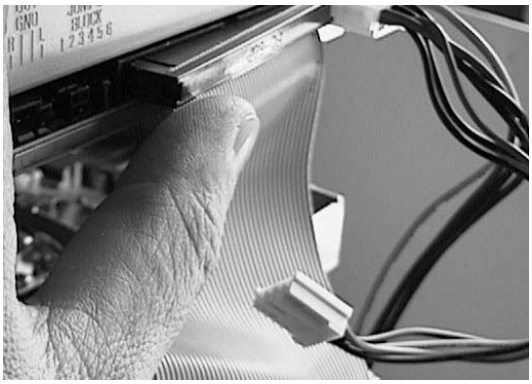
Lamentablemente no es posible definir a nivel general la configuración como maestro o esclavo para todas las unidades, ya que los fabricantes optan por diferentes soluciones. No obstante, el *jumper* correspondiente suele estar indicado en la cara posterior de la unidad, en caso contrario habrá que echar mano del manual del dispositivo. Por regla general, es aconsejable definir la configuración maestro/esclavo pertinente antes de montar la unidad en su marco ya que más adelante será difícil acceder a la posición de estos elementos. Para optimizar el rendimiento es conveniente configurar las unidades rápidas (discos duros) como dispositivos maestro.



Conexiones y el jumper maestro/esclavo en una unidad de CD-ROM/DVD-ROM

A veces representa un problema la longitud del cable, en especial si se desea montar un dispositivo en la parte superior de una caja tipo torre y resulta que el cable es demasiado corto. Por tanto, antes de sujetar con tornillos la unidad es conveniente comprobar con el cable del controlador si, realmente, es una opción factible. En esos casos podemos necesitar un cable nuevo más largo. Por regla general, el cable debe ser lo más corto posible para evitar de raíz problemas posteriores.

Para la toma de alimentación se utiliza, en función del tipo de unidad, el conector pequeño o el grande, que comunican los cables amarillo, negro, negro y rojo procedentes directamente de la fuente de alimentación. La asignación de líneas es idéntica en ambos conectores. Si ya no hay ningún conector disponible, en tiendas especializadas se puede adquirir un cable en Y, que a partir de una conexión proporciona dos. Este conector, y sus homólogos en las unidades correspondientes, está diseñado de tal forma que es imposible acoplarlo de forma incorrecta, a no ser que para ello se emplee una fuerza exagerada.



Al acoplar el cable del controlador de 40 contactos hay que observar siempre la posición correcta (pin 1) del conector. En primer plano se puede ver el conector de alimentación pequeño, la unidad de CD-ROM/DVD-ROM utiliza el conector grande

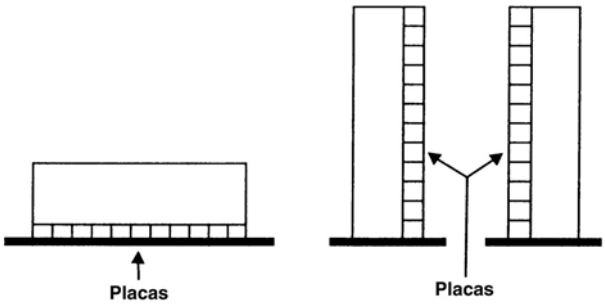
Algo que muchas veces se olvida es la conexión de audio de la unidad de *CD-ROM/DVD-ROM* con la tarjeta de sonido. Es cierto que se pueden reproducir perfectamente archivos de sonido *MIDI* o *WAVE*, que prácticamente están integrados como datos de sonido en la corriente de datos de la unidad *CD-ROM*, pero sería imposible escuchar un tono de un CD de audio normal. Para establecer esta conexión de audio existen multitud de variantes diferentes en la unidad y en la tarjeta de sonido. Con un poco de suerte, el paquete de la tarjeta o de la unidad incluye el cable apropiado. En caso contrario, o si el cable proporcionado no se adapta a la conexión de uno de los dos dispositivos, deberemos conseguir el cable adecuado por nuestra cuenta. Existen cables para este cometido que se adaptan a diferentes tipos de conexión de unidades de *CD-ROM/DVD-ROM* y tarjetas de sonido. Si se utiliza una unidad *DVD-ROM* y una tarjeta decodificadora *MPEG* el cable de audio no se suele conectar a la tarjeta de sonido sino a la tarjeta decodificadora.

16.4.2 Instalar discos duros y el sistema de ventilación

Para los discos duros de 2.5 pulgadas como los usados en portátiles existe un conector de 50 contactos en rejilla de 2 milímetros que también se fabrica como conector de 44 contactos. La asignación de los 40 contactos del conector *IDE* estándar se mantiene invariable y a ellos se suman unas líneas de señal específicas del fabricante (que son generalmente 4).

En las tiendas de informática hay adaptadores disponibles para hacer funcionar por ejemplo un disco duro (pequeño) de 2.5 pulgadas y conexión de 50 o 44 contactos unido a una conexión de controlador de 40 contactos.

Los cables adaptadores habituales no tienen en cuenta las posibles señales adicionales específicas del fabricante, utilizadas por ejemplo para cambiar a un modo de ahorro energético, ya que se trata generalmente de señales especiales para portátiles que cada fabricante maneja de forma particular.



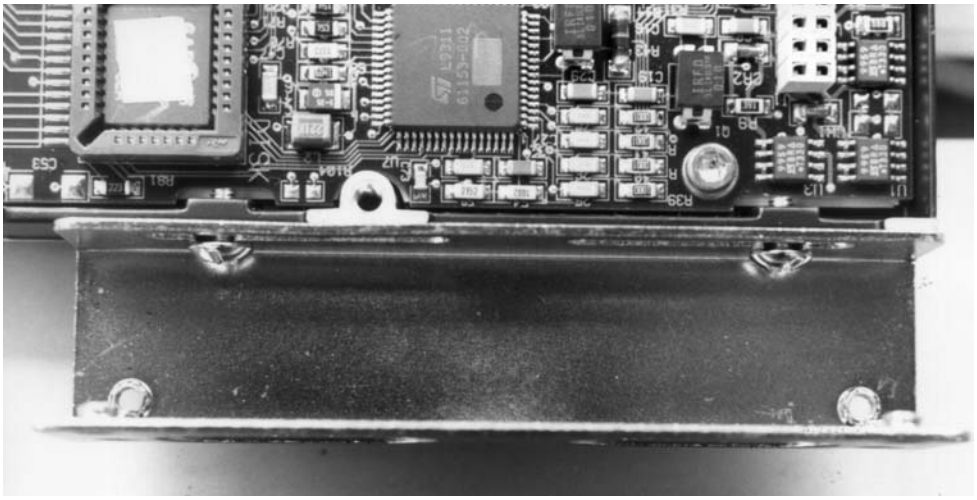
Lugares apropiados para las unidades de disco duro

Desde el punto de vista mecánico, los discos duros actuales son tan pequeños que para montarlos se necesitan ángulos y soportes, lo que en principio no es nada extraño. Sin embargo, a veces se requieren ángulos y tornillos específicos, ya que la cara inferior del dispositivo, donde se encuentran los componentes electrónicos, ofrece tan poco espacio de maniobra, que sujetar la unidad puede representar un serio problema si el paquete de entrega del disco no incluye los elementos de sujeción pertinentes.

Dependiendo de la forma de fabricación de la unidad, al apretar los tornillos entre el ángulo y la caja del dispositivo puede verse dañado algún componente electrónico si no se utilizan los tornillos apropiados.

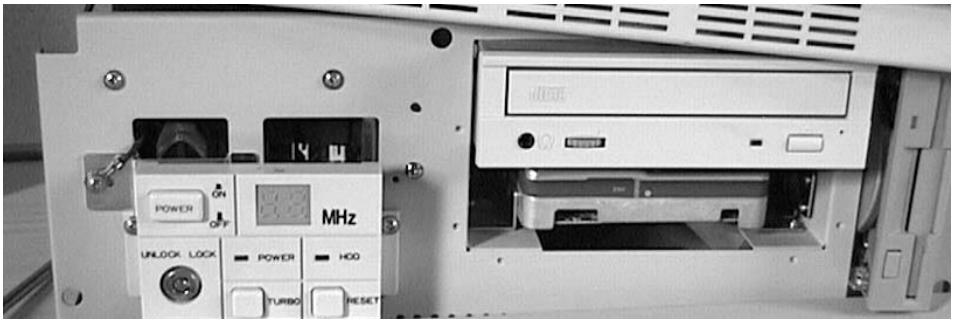
Si, por ejemplo, son muy largos podemos presionar en exceso alguna pieza de la cara inferior, que al estar en posición horizontal y mirar hacia abajo resulta difícil de ver.

La siguiente figura muestra un caso típico en el que los tornillos se detienen justo delante de los componentes electrónicos. Es fácil imaginar lo que sucedería si se utilizaran tornillos más largos.



Los tornillos que sujetan el disco duro no deben ser muy largos, de lo contrario podríamos dañar algún componente electrónico

Si después de largo tiempo de funcionamiento el disco duro registra pérdidas de datos puede obedecer a que la unidad se calienta demasiado. Este problema suele aparecer en los discos duros que trabajan a 7200 o incluso 10000 revoluciones por minuto.



El nuevo disco duro está montado con unos ángulos debajo de la unidad de CD-ROM

Sólo en los últimos tiempos han reaccionado algunos fabricantes de discos duros y han pegado en sus unidades una etiqueta en la que se advierte de la necesidad de una buena refrigeración. Aunque no siempre revelan cómo colocar el ventilador. También se pueden encontrar avisos de este tipo en los manuales de los discos duros, documentos que no siempre proporcionan los distribuidores.

A nivel general se constata que cuanto más rápido gira un disco duro más se calienta. En este problema también influye el tamaño de la unidad. Gracias a su mayor superficie los modelos mayores, con un formato de 5.25 pulgadas, pueden disipar el calor sobrante en su entorno con mayor facilidad que los modelos de 3.5 o 2.5 pulgadas.

Si un disco duro en marcha se calienta tanto que quema los dedos es aconsejable instalar un sistema de refrigeración. Para ello existen unos marcos con un ventilador para las bahías de 5.25 pulgadas, que se colocan debajo del disco duro en cuestión y refrigeran la unidad con la corriente de aire. También existen tarjetas de ventilación que se insertan en una ranura del PC, aunque no usan ningún contacto eléctrico sino que únicamente la utilizan como elemento de sujeción. Estos ventiladores proporcionan una ventilación general a todo el interior del PC. Algunos modelos más avanzados no sólo crean corrientes de aire en el interior de la torre sino que también lo expulsan hacia fuera a través de un hueco de la cara posterior. Una solución más económica consiste simplemente en colocar un ventilador normal debajo del disco duro. Para fijar el ventilador bastará con usar unas bridas (rígidas) sujetas al marco.

Al montar un disco duro hay que procurar que no quede directamente debajo o encima de otro dispositivo. Es conveniente dejar siempre un espacio intermedio para que el aire pueda circular con libertad.

Especialmente para discos duros rápidos y grabadoras de CD/DVD es recomendable montar un ventilador adicional

Montar una grabadora de *CD/DVD*, dispositivo que se calienta bastante, directamente debajo de un disco duro que trabaja a 10000 revoluciones por minuto no sólo puede provocar pérdidas de datos en el disco sino también serios problemas a la hora de grabar los correspondientes CDs o DVDs.

16.5 Instalación de la placa base

Instalar una placa base nueva ya es una operación importante que en cualquier caso requiere hacer un repaso previo de todos los condicionantes mecánicos y eléctricos antes de comenzar el auténtico montaje. El primer aspecto a tener en cuenta es si se va a montar una placa base *AT* o ATX. En sistemas *NLX* y *LPX* no suelen darse posibilidades razonables de ampliación ya que las placas base correspondientes suelen ser

modelos exclusivos difíciles de encontrar y su precio supera el de los dos modelos estándar. El tipo de placa base que se va a montar no es relevante respecto a la forma de proceder, en principio es siempre la misma, si se trata como en este caso de desmontar y montar componentes en un ordenador de estructura convencional (*AT*, *ATX*).

16.5.1 Desmontar la placa base

Según el modelo de caja será necesario desmontar varios componentes para poder acceder a la placa base. Para asegurarse más tarde y poder colocar de nuevo todas las conexiones correctamente y sin complicaciones, es aconsejable marcar todos los puntos de relevancia, por ejemplo los cables correspondientes a las unidades e interfaces.

En este PC hay que realizar algunas operaciones antes de poder cambiar la vieja placa base por una nueva placa base

Otra medida de seguridad antes de desinstalar la placa base será anotar, o mejor imprimir, los datos importantes del *Setup* del *BIOS*. Qué tipo de unidades de disquetes se utilizan se sabrá también luego, pero los parámetros de los discos duros *IDE* no siempre estarán presentes, y son imprescindibles para el *Setup* del *BIOS* de la nueva placa base ya que no está garantizado que el *BIOS* de la nueva placa active automáticamente los datos utilizados anteriormente.

Además, si todavía se utilizan viejas tarjetas *ISA*, llevaría su tiempo redefinir las configuraciones en *PNP/PCI Configuration Setup* y para no tener que empezar todo desde el principio es conveniente tener preparados estos viejos datos de instalación.

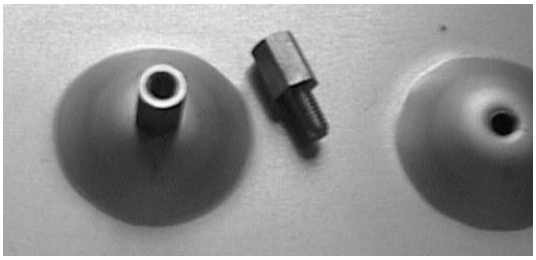
A un lado de la placa base suelen estar las conexiones del botón *RESET*, el interruptor principal y de los diversos diodos luminosos. Lamentablemente no siempre están claramente rotulados sobre la placa, o sólo con conceptos algo confusos. Por esta razón también es conveniente anotar antes a qué corresponde cada conexión.

Las conexiones de los indicadores y controles de la placa base no siempre están indicados con tanta claridad como en este caso

Una placa base sólo está sujeta a la caja del ordenador con uno, dos o tres tornillos, dependiendo de su tamaño. En otro caso serán las tarjetas de expansión las que se encarguen en parte de mantener sujeta la placa base, algo totalmente desaconsejable. Después de sacar los tornillos, naturalmente quitadas también las tarjetas, la placa base se suele extraer de los agujeros longitudinales por la izquierda (caja tipo sobremesa) o hacia arriba (caja tipo torre) y luego de la caja del ordenador. Los separadores de plástico, insertados a presión en los agujeros de la placa y encargados de mantener la distancia de separación con la chapa de la caja, se deben extraer con cuidado de la placa vieja con la ayuda de unos alicates. Estas piezas se suelen entregar junto con la caja del ordenador y es difícil encontrarlas por separado, por tanto se necesitarán de nuevo para colocarlos en la nueva placa base.

16.5.2 Sujeción de la placa base

Los agujeros de montaje previstos en la nueva placa base deben coincidir con los agujeros correspondientes de la caja. Es útil orientarse por la conexión del teclado ya que siempre corresponde con el agujero de la caja correspondiente. Si parece necesario hacer algún agujero en la caja algo falla, ya que tanto la caja como la placa disponen de agujeros de montaje que están adaptados, o deberían estarlo, para admitir diversas combinaciones.

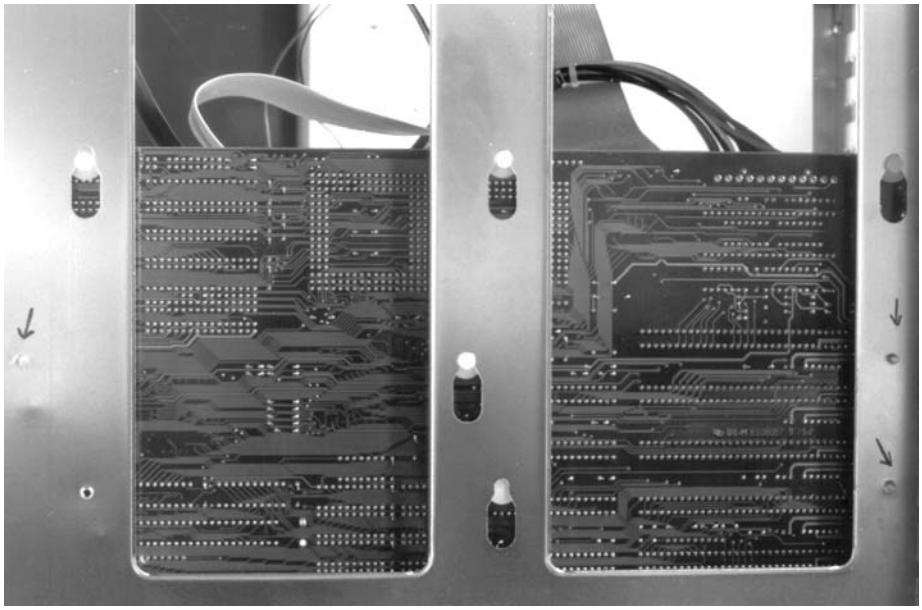


Pequeños pero importantes, los bornes de sujeción para montar la placa base

Para sujetar la placa base a la caja se utilizan en determinados lugares (como mucho 3, allí donde las placas viejas tenían tornillos) unos bornes de separación metálicos que se sujetan luego con rosca a la caja. Luego la placa se sujeta con tornillos a estas piezas. Allí donde sea posible es conveniente colocar separadores de plástico ya que la placa base se puede combar de forma ostensible cuando luego se insertan las tarjetas de expansión. Si la placa base se curva demasiado se puede romper alguno de sus circuitos.

La cara de la placa donde están los circuitos no debe tocar nunca el marco de metal de la caja ya que esto podría provocar un cortocircuito que acabaría con la placa base. En algunas placas también nos podemos encontrar con que los hilos de los componentes por el lado de la soldadura no estén cortados lo suficientemente cortos, y por tanto existe el peligro de que se produzca un contacto fatal entre la placa y la caja.

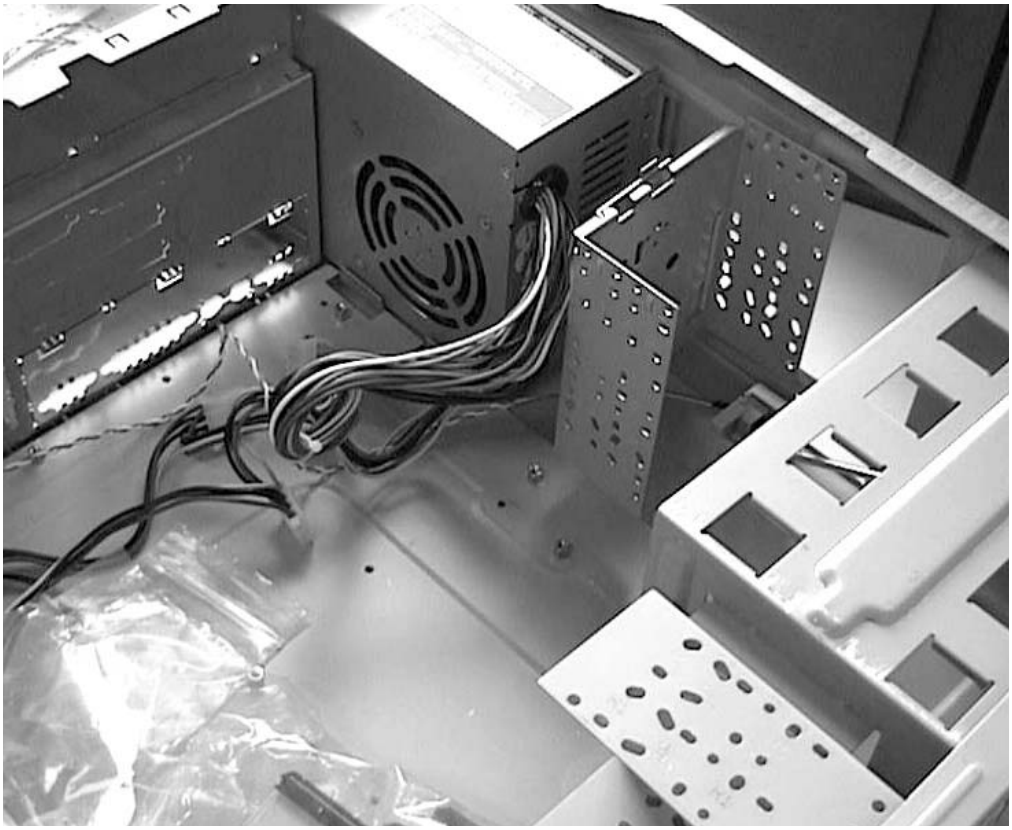
Naturalmente también hay que impedir que un posible curvamiento de la placa la acerque peligrosamente a la caja de metal. En algunos lugares, donde se colocan los tornillos y bornes de metal existe un contacto real entre la placa y la caja del ordenador, pero esta circunstancia ya se ha tenido en cuenta; la masa de la placa está unida a la masa de la caja, que a su vez está conectada al conductor de tierra.



Parte posterior de una placa base montada en una caja tipo torre.
Los tres bornes de separación están atornillados donde indican las flechas.
En los agujeros longitudinales de la caja se ajustan los separadores de plástico

Si la placa base no presenta problemas de índole mecánico, se adapta a la estructura de la caja sin necesidad de hacer agujeros y también se pueden montar las tarjetas, todo indica que el resto de la instalación se desarrollará sin complicaciones. A nivel

mecánico cada componente está (teóricamente) diseñado para que armonice con el resto del equipo, de tal forma que nada puede fallar si se utilizan los materiales y cables originales.



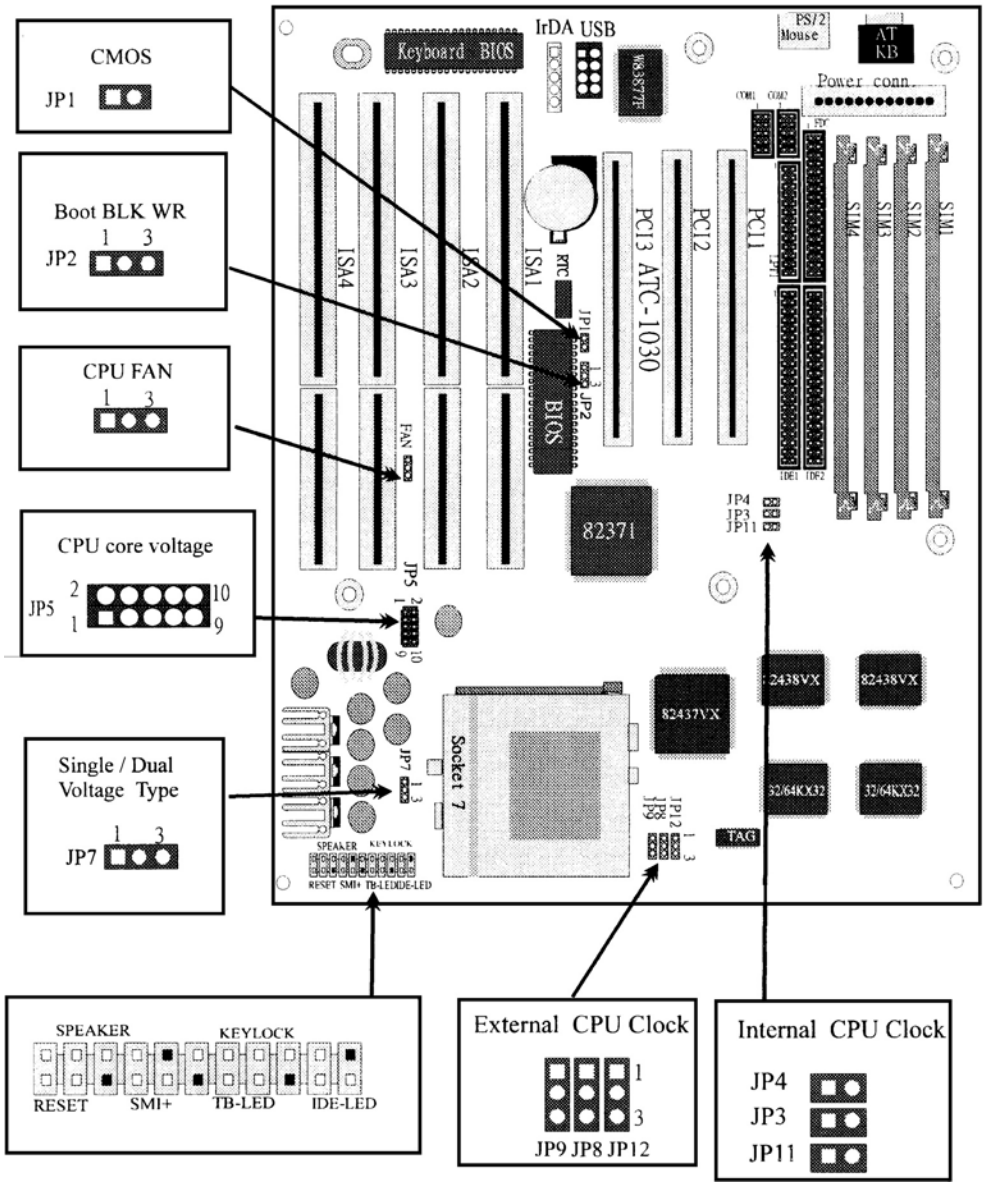
Caja ATX tipo sobremesa vista desde dentro. A diferencia de las placas y cajas ATX, en este caso la placas con sus separadores de plástico no se introduce con ayuda de las guías longitudinales sino desde arriba directamente en los agujeros previstos y luego se sujeta con uno o dos tornillos (quedando debajo los bornes de separación)

16.5.3 Montaje de la placa base

Antes de montar una placa base nueva hay que consultar siempre el manual adjunto y comprobar si todos los *jumpers* están en la posición correcta, prestando especial atención a los que definen las configuraciones del procesador. A continuación hay que establecer la conexión eléctrica con las conexiones *P8* y *P9* o el conector específico *ATX* a la fuente de alimentación.

Por razones de seguridad es aconsejable no montar todavía el procesador, los módulos de memoria ni las tarjetas de expansión, encienda antes el ordenador a modo de prueba y se deberá oír el sonido típico del ventilador de la fuente de alimentación.

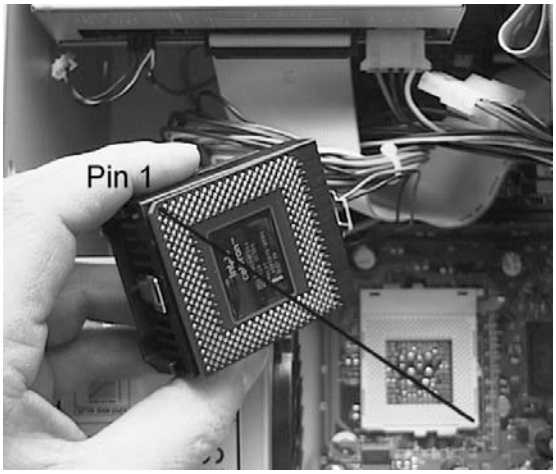
No obstante este método sólo se aplica en las placas *BAT* ya que para la prueba no es necesario conectar ningún indicador o control a la placa. Al contrario, en las placas base *ATX* es preciso conectar por lo menos la tecla de la conexión *PRW* o como mínimo puentear con un destornillador los dos *pins* adyacentes para generar el impulso que activa la fuente de alimentación *ATX*.



Jumpers y conexiones en una placa base. Antes de instalar la placa hay que comprobar la posición de los jumpers. Sin embargo, para la primera prueba no es necesario conectar ningún elemento de control a la placa base

16.6 El montaje del procesador

En el paso siguiente se monta el procesador. Por ejemplo, para procesadores para el *socket 7* y *Celeron* en cápsula *PPGA* existe un zócalo con una pequeña palanca a un lado que se levanta para liberar los contactos del zócalo. A continuación se inserta el procesador y se cierra la palanca hacia abajo. En esta operación es vital observar la posición del *pin* 1, en esa posición el procesador tiene una esquina limada. Sólo empleando una fuerza excesiva se podría colocar el procesador del modo equivocado, aun cuando la forma física del zócalo esté diseñada para evitar cualquier error en este sentido.



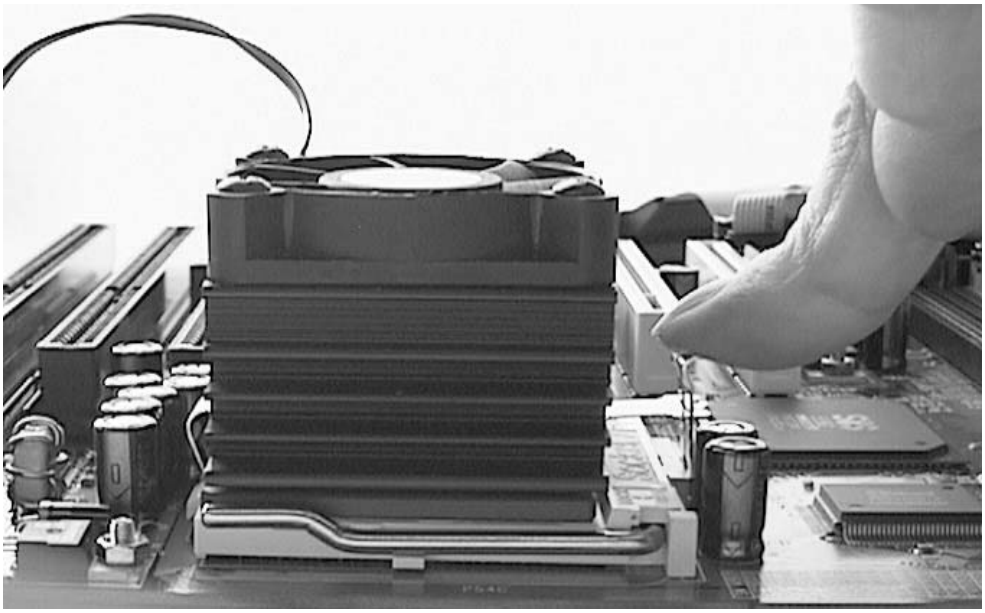
El procesador Celeron de 370 contactos, en el que no suele estar marcada la posición del pin 1, se podría insertar en el zócalo de forma equivocada si se coloca a la fuerza

Al colocar el ventilador sobre el procesador hay que actuar con sumo cuidado,ya que existen varios tipos que se montan de forma diferente. En algunos casos, el cuerpo del ventilador se acopla entre el zócalo y la cápsula del procesador con ayuda de un marco de plástico o de unos pequeños soportes. Si ese marco no está bien asentado debajo del procesador, es posible que no se cree el contacto preciso debido a que hay mucha separación entre el procesador y el zócalo.

Con el tiempo, el calor generado por el procesador hace más frágil el marco de plástico que hasta se puede desprender del cuerpo del ventilador. Esta circunstancia es especialmente crítica en las cajas tipo torre ya que esta pieza tan vital se balancearía en algún lugar dentro de la caja sin tocar, ni ventilar, en absoluto el procesador. Para optimizar el contacto térmico entre el procesador y el cuerpo del ventilador se puede utilizar una pasta conductora de calor, que muchas veces se distribuye junto con el procesador y ventilador adecuado cuando estos componentes se adquieren en un kit específico.

Por ejemplo, un procesador *Pentium* normal sólo puede sobrevivir unos minutos sin refrigeración. No obstante se puede llevar a cabo una prueba sin ventilador durante

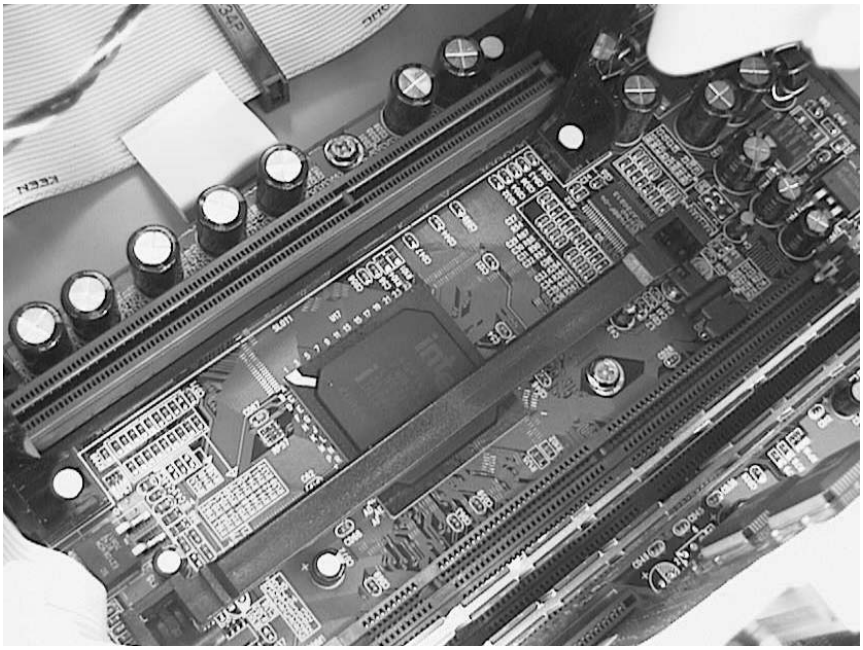
unos momentos para comprobar si la causa de que el procesador no funciona se debe a una mala conexión entre el zócalo y el procesador, provocada por el cuerpo del ventilador. Algunos ventiladores se sujetan con una grapa a un saliente del zócalo, a veces para encajar esta especie de grapa hay que presionar con bastante fuerza. Sin embargo, nunca se debe utilizar un destornillador para encajar el ventilador, porque si al presionar la grapa resbala el destornillador y golpea sobre la placa podríamos dañar algún circuito impreso y la placa base ya no se podría arreglar.



Un buen ventilador para procesadores sockel 7 se caracteriza por sujetarse al zócalo con grapas de metal flexibles y el ventilador se asienta sobre rodamientos

En un procesador *Pentium II/III* o *Celeron* de diseño *slot 1* el montaje es algo más laborioso ya que primero hay que montar sobre la placa los elementos de sujeción y el ventilador se ensambla junto con el procesador. No obstante estos procesadores tampoco se pueden insertar mal en el zócalo.

El *Pentium II* en cápsula *SECC2* utiliza un bloque de refrigeración bastante grande con un ventilador. El montaje de un procesador *Pentium II* se parece a un "pequeño puzzle de piezas". Con ayuda de un marco de plástico y una grapa se sujeta el bloque de refrigeración al módulo del procesador. La placa base del *Pentium II* dispone de un elemento de sujeción especial que se encarga de asentar fijamente el módulo del procesador junto con el bloque de refrigeración. Dadas las dimensiones y el peso de este *Pentium II*, el dispositivo de sujeción de la placa es imprescindible y debe ser fijo y seguro. Sin un elemento de sujeción óptimo, el bloque del *Pentium II* puede salirse del zócalo durante un transporte del ordenador, y en determinadas circunstancias destruir el procesador al encender el aparato.



Slot 1 y elementos de sujeción para un Pentium II

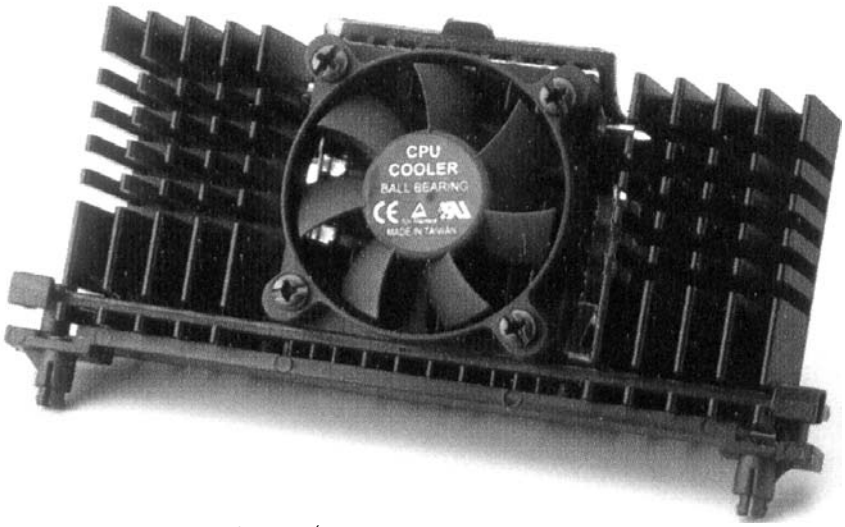
El procesador *Celeron* con diseño *slot 1* no se distribuye en forma de módulo cerrado sino más bien como una placa sin tapa. El bloque de refrigeración se sujeta con unas grapas justo encima del procesador y no utiliza elementos de sujeción tan complejos como el *Pentium II*. Por esta razón los modelos *Pentium II* y *Celeron* necesitan unos bloques de refrigeración y sujeciones diferentes, lo que significa que el fabricante de la placa base deberá proporcionar accesorios diferentes para cada procesador

A esto hay que añadir que para los *Pentium II* y Pentium III se simplificó el diseño del cartucho (*SECC2* en lugar de *SECC1*), pero también nos encontramos de nuevo con otros tipos de bloques de refrigeración y sujeciones.

Dado el panorama es aconsejable informarse antes de la compra en qué medida son compatibles a nivel mecánico el procesador, el bloque de refrigeración y la placa base. Incluso en productos de fabricantes de prestigio no existe a veces una armonía óptima entre los componentes, lo que puede llevar a que el PC no funcione la primera vez que se encienda.

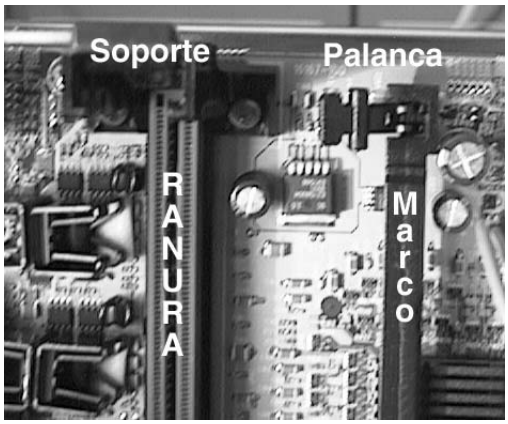
Según datos de Intel para sujetar los diferentes zócalos de procesador se debería usar un *Universal Retention Modul* (*URM*), compatible con todos los procesadores aparecidos hasta la fecha del *Pentium 4*.

Los procesadores para el *slot 1* se insertan en el zócalo presionando con más fuerza, debiendo comprobar siempre si los contactos se han introducido lo suficiente en el zócalo y que el procesador y el ventilador se asientan firmemente en sus sujeciones.



Ejemplo de bloque de refrigeración utilizado para procesadores Pentium II y Pentium III

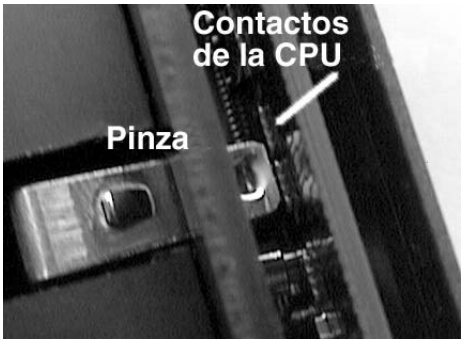
En principio el montaje de un procesador *Athlon* en un zócalo *slot A* es igual al de un procesador *Pentium* en un zócalo *slot 1*, es decir, primero se monta sobre la placa el elemento de sujeción, luego se ensambla el cuerpo del ventilador al procesador y finalmente se inserta con cuidado todo ese "bloque" en el zócalo, comprobando que se adapta correctamente a las sujeciones de plástico.



Es imprescindible montar el marco derecho ya que después de colocar el procesador Athlon la palanca se debe acoplar correctamente sobre las "patillas" del bloque, de otro modo el bloque no quedará bien asentado y fácilmente podrá deslizarse fuera del zócalo

Este dispositivo de sujeción consta de tres partes: concretamente, dos pequeñas, que se colocan en la placa base a ambos lados del zócalo, y un marco que se halla frente al zócalo y que sostiene el módulo. Muchas veces este marco no se coloca correctamente o incluso se deja a un lado, ya que sólo se ajusta en una posición a la placa base, que para ello posee dos agujeros de distinto tamaño, para que luego las dos pequeñas

palancas del marco puedan anclar perfectamente el módulo del procesador. Al montar el cuerpo de refrigeración acecha otro peligro; la grapa de sujeción puede quedar demasiado cerca de las conexiones del procesador y si llegaran a tocarse probablemente se dañaría el *Athlon*. Por consiguiente, al montar el cuerpo de refrigeración hay que extremar las precauciones.



La grapa de sujeción de un cuerpo de refrigeración puede quedar muy cerca de los contactos del Athlon lo que podría significar el fin prematuro del procesador

Antes de montar en la caja la placa base y el procesador es recomendable realizar una pequeña prueba para supervisar la capacidad de funcionamiento de los componentes importantes del ordenador. Para la prueba de la placa base:

- Comprobar los *jumpers* de la placa base relativos al procesador y si es necesario cambiar su posición.
- Montar el procesador.
- Montar el ventilador del procesador (atención, ya que es muy posible que la grapa toque un circuito impreso; si ejerce excesiva presión mecánica deberíamos dar la vuelta al cuerpo de refrigeración; existen varios tipos de bloques de refrigeración).
- Conectar las toma de alimentación del procesador y del ventilador.
- Conectar la toma de alimentación *(ATX* o *P8* y *P9*).
- Conectar y activar la fuente de alimentación. En una placa *ATX* puentear brevemente con un destornillador los contactos *Power On*; la placa base debe arrancar sin problemas.

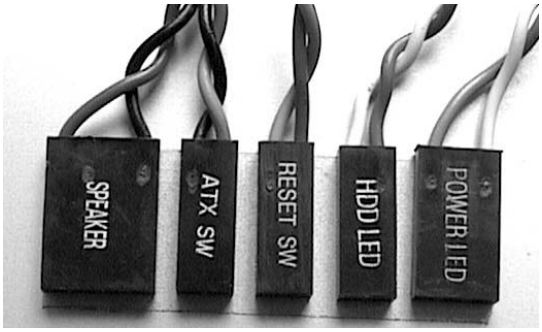
Si ya está conectado el altavoz se deberían oír tonos (generalmente tres) ya que falta montar la memoria. Si la placa base posee ranuras *DIMM* o *PS/2* será necesario consultar de nuevo el manual de la placa base ya que, posiblemente, sólo son posibles determinadas combinaciones de módulos y, además, es posible que debamos establecer mediante *jumpers* la tensión de trabajo apropiada para los módulos. A continuación le toca el turno a la tarjeta gráfica, que se debe montar y conectar al monitor. Después de iniciar el ordenador deberá aparecer una imagen en pantalla y se deben detectar los módulos de memoria.

16.7 Los indicadores y elementos de control

En un paso siguiente se acoplan las conexiones de los indicadores y elementos de control que aparecen en la cara frontal del PC. En esta operación no hay mucho margen de error ya que para los botones *RESET* y *Power On* (placas base *ATX*) es indiferente la polaridad.

Naturalmente es preciso unir los contactos específicos de la placa base que tienen asignada esta función, conexiones que no siempre se identifican fácilmente porque a veces todas las conexiones de los indicadores y controles están muy juntas sobre la placa.

Si nos equivocamos al conectar las líneas, por ejemplo si conectamos el interruptor *Turbo* a la conexión *RESET* de la placa base, el PC pasará a un estado *RESET* continuo y no aparecerá ninguna imagen en pantalla.



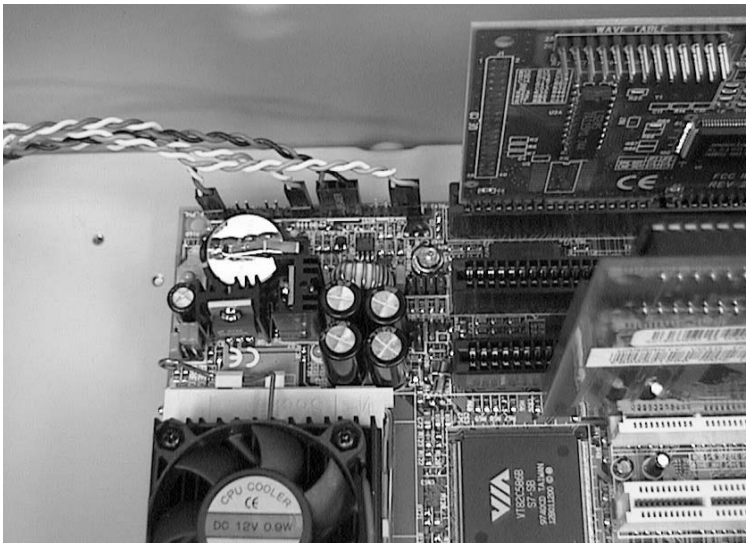
Conexiones habituales (mínimas) de una caja ATX para una placa base compatible, que puede poseer muchas otras posibilidades de conexión adicionales que sin embargo no tienen porqué estar presentes en el panel frontal de la caja.

No es del todo extraño que una placa base disponga de algunas conexiones específicas (por ejemplo de *Sleep Mode*) para las que no existe el indicador correspondiente en la caja del ordenador (una tecla o un interruptor), ni tampoco el caso inverso, es decir, que el panel de la caja posea ciertos indicadores o controles cuya réplica no se encuentre disponible en la placa base.

Sólo las placas base y cajas antiguas poseen todavía una conexión *TURBO* y el correspondiente *LED*. En este aspecto lo más importante es que el procesador pase de hecho al modo rápido cuando el modo *TURBO* (*LED* encendido) esté activado.

La forma más sencilla de comprobar esta función es con un programa de test especial como *SYSINFO* (de la casa Norton), al accionar el interruptor el procesador deberá cambiar su frecuencia de trabajo de la forma correspondiente.

Si la caja del equipo no incluye el botón *TURBO* deberemos colocar el *jumper* de la placa base responsable de este cambio en la posición *TURBO* para que este modo de funcionamiento esté activado de forma permanente y no cause ningún tipo de problema en un futuro.

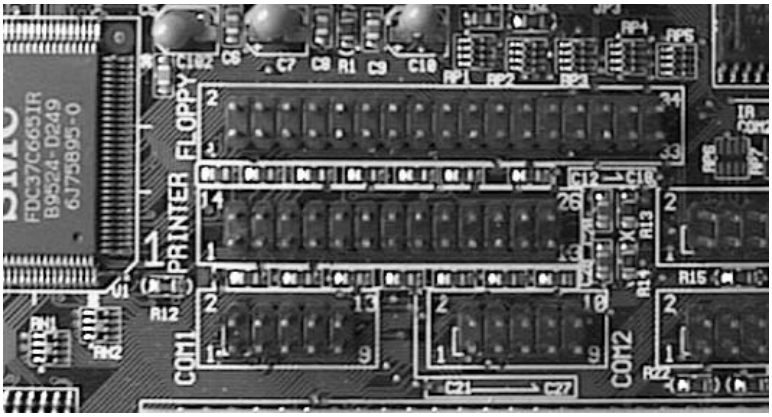


Indicadores y elementos de control conectados en la placa base

16.8 La conexiones de los distintos puertos

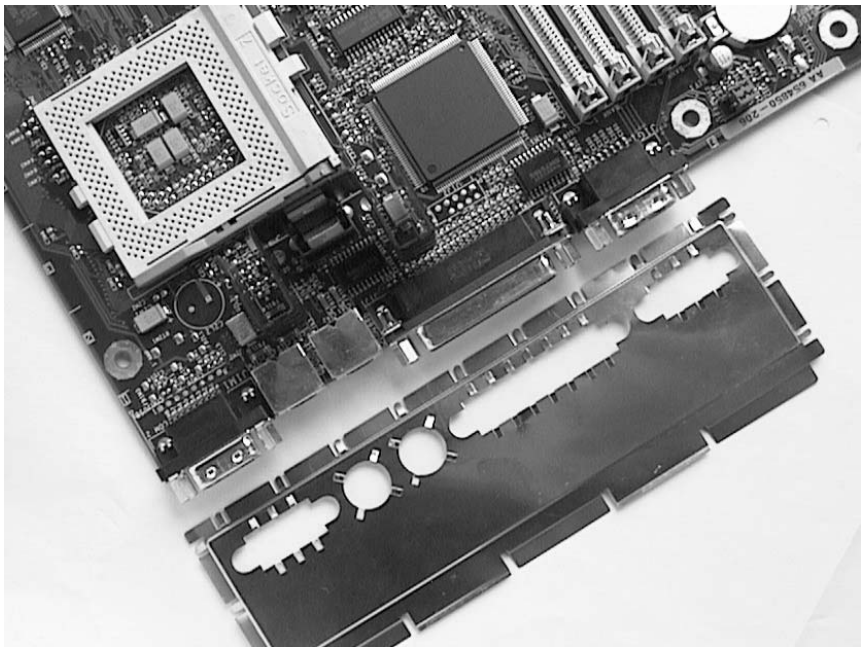
Si resulta difícil acceder a algunas conexiones de la placa base, por ejemplo al puerto del ratón *PS/2*, al puerto de la impresora o a los puertos *RS232*, hay un cierto riesgo de colocar mal los contactos, por lo tanto es preciso proceder con mucha atención.

Para las interfaces es aconsejable utilizar siempre los cables entregados junto con la placa base y no otros aparentemente idénticos ya que las conexiones están diseñadas de forma específica para esos cables y no se debe dar por hecho que la asignación de líneas de los cables alternativos, aunque similares, sea realmente la misma.



Al conectar el cable de cinta en este tipo de conexión, con las patillas muy juntas entre sí y sin una posición evidente, es bastante fácil equivocarse

En una placa base *ATX* las conexiones de interfaz se encuentran directamente sobre las interfaces y, por tanto, no será necesaria la operación de cableado. Sin embargo, aquí puede aparecer otro problema. El área donde se encuentran las conexiones de las interfaces integradas en la placa está bien definida pero no (a su debido tiempo) los huecos pertinentes de la caja. Como consecuencia, los fabricantes de cajas deben proporcionar los huecos y cubiertas de chapa necesarios para los diferentes modelos de placas. Por eso, al comprar una caja y/o una placa base *ATX* hay que comprobar si el hueco previsto para el área coincide con las conexiones de la placa. Sólo a partir de la especificación *ATX 2.1* se unificaron estos criterios.



La cubierta del área en ATX debe coincidir en la placa base y la caja ATX

16.9 Resumen de los distintos pasos a seguir

A continuación enunciamos los pasos a seguir al ampliar o modificar la configuración del ordenador:

- Anotar o imprimir los valores importantes del *Setup* del *BIOS*.
- Desconectar la toma de corriente.
- Desconectar todas las conexiones externas (teclado, tarjeta gráfica, impresora, monitor).
- Abrir la caja del ordenador (guardar bien los tornillos).
- Extraer la tapa de la caja.
- Anotar la posición de las conexiones de la placa base y desconectarlas.

- Desconectar los cables de alimentación *P8, P9*, marcando antes las posiciones de ambos conectores (las líneas negras juntas en el interior), o bien desenchufar la conexión *ATX*.
- Retirar el cable del ventilador del procesador.
- Quitar los tornillos de sujeción y extraer las tarjetas de expansión.
- Desconectar el resto de conexiones marcando la posición del *pin* 1.
- Aflojar y quitar los tornillos que sujetan la placa base.
- Tirar de la placa base (observando las guías longitudinales de la cara posterior) y extraerla de la caja.
- Colocar a modo de prueba la nueva placa base dentro de la caja, controlar la posición de las sujeciones, orientarse por la conexión del teclado.
- Extraer los separadores de plástico de la placa base vieja y colocarlos en los lugares correspondientes de la nueva.
- Introducir la placa base en la caja, observar las muescas (chapa de la caja) y desplazar la placa base hacia el interior.
- Es posible que la placa base quede suspendida por un lado debido a que los agujeros no coinciden plenamente. Si no queda fija y estable, utilizaremos los separadores de plástico y adaptaremos la estructura metálica de la caja.
- Sujetar la placa base con los tornillos; los separadores metálicos están previstos para albergar los tornillos.
- Montar el procesador y el cuerpo de refrigeración.
- Conectar la toma de alimentación (*ATX* o *P8, P9*) y el ventilador del procesador a la placa.
- Unir a la placas las conexiones del panel de control (*Reset, Keyboard lock, Speaker, Power*). Es posible que algunas conexiones de la placa queden sin utilizar debido a que el panel de la caja no disponga de los elementos correspondientes.
- Realizar el primer test.
- Montar los módulos de memoria RAM.
- Establecer las conexiones de las unidades de disquetes, *CD-ROM/DVD-ROM* y disco duro. El disco duro principal: unidad primaria *IDE*. El *CD/-DVD*: unidad secundaria *IDE*. La unidad de disquetes: *FDD* (*Floppy Disk Drive*).
- Instalar la tarjeta gráfica y el resto de tarjetas de expansión.
- Realizar el segundo test.
- En el *Setup* del *BIOS* realizar una búsqueda automática del disco duro. Si el sistema no lo detecta deberemos comprobar la conexión de la toma de corriente y los puertos (*Primary*, *Secondary*), y registrar los (viejos) parámetros de forma manual.

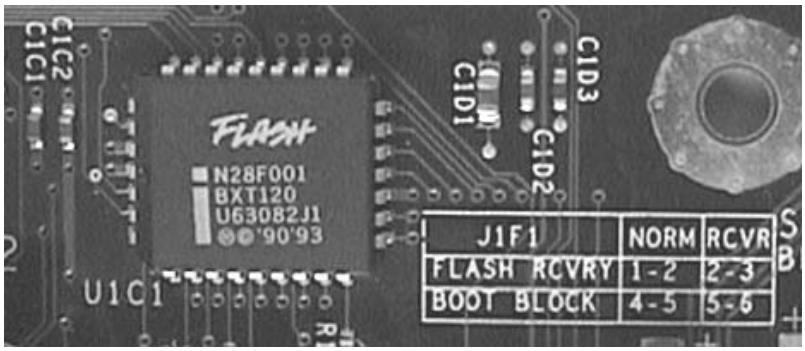
- Conectar las interfaces serie y demás, montar si es preciso las cubiertas de metal (los cables viejos pueden tener una asignación de líneas diferente).
- Comprobar y si es preciso adaptar los indicadores (*Power, HDD*) e interruptores (*Reset*).
- Conectar de nuevo los dispositivos externos.
- Arrancar el sistema operativo e instalar los nuevos controladores.
- Optimizar las configuraciones del *Setup* del *BIOS*.
- Instalar el software adicional que haga falta y llevar a cabo otras pruebas adicionales.
- Por último, cerrar la caja del ordenador.

17. El BIOS y el Setup

BIOS significa *Basic Input Output System*, o sistema básico de entrada y salida. Como ya sabemos, un ordenador está compuesto de hardware y software. El hardware es toda la parte física del equipo: placa base, procesador, memorias, disco duro, etc. Mientras que el software es la parte lógica que coordina todo su funcionamiento.

El *BIOS* es, justamente, la primera capa de software del equipo, que está almacenado en un pequeño *chip* de la placa base, y tiene la función de dar la señal de inicio del sistema, reconociendo los dispositivos instalados en el equipo y realizando el *BOOT* (arranque). Incluso después de tener el sistema operativo cargado, el *BIOS* continúa aportando mucha información y ejecutando tareas indispensables para el funcionamiento general del sistema.

Muchas de las funciones ejecutadas por el *BIOS* pueden personalizarse. El Setup es el programa que nos permite configurar estas opciones del *BIOS*. La velocidad de trabajo de las memorias, el modo de funcionamiento de los discos duros y, en muchos casos, incluso la velocidad del procesador se puede configurar a través del *Setup*. Una mala configuración del *Setup* del *BIOS* puede hacer que el sistema sea hasta un 90 o 95% más lento de lo que sería con una configuración optimizada.



El chip de la placa base que contiene el BIOS

En casi todos los modelos de *BIOS* encontramos una opción de configuración del *Setup* usando los valores sugeridos por defecto por el fabricante de la placa base.

Estos valores aseguran que el sistema funcione con el máximo de estabilidad posible aunque, sin embargo, a costa de sacrificar un poco de rendimiento.

Generalmente, con unas configuraciones optimizadas de los valores del *Setup* es posible obtener una ganancia de rendimiento de un 10 a un 20% sobre los valores por defecto. Muchas veces, es necesario cambiar algunos valores del *Setup* para resolver algún conflicto entre distintos dispositivos, o para poder instalar algún periférico. Hasta hace poco, un ejemplo clásico era la opción *Assign IRQ to VGA* que debía estar habilitada para que la mayoría de las tarjetas gráficas 3D funcionasen correctamente, pero que en muchas placas base aparecía deshabilitada al usar valores por defecto del fabricante.

17.1 CMOS

CMOS equivale a *Complementary Metal Oxide Semiconductor*. En los primeros PCs, como los antiguos *XT* y algunos *286*, todos los datos referentes a la configuración de las direcciones de *IRQ*, cantidad y velocidad de las memorias, discos duros instalados, etc. estaban configurados a través de los *jumpers* de la placa base. No es necesario decir que la configuración de estos *jumpers* era un trabajo extremadamente complicado. Para facilitar la vida de los usuarios se creó el *Setup*, que permite configurar de una forma más fácil el sistema.

La función del *CMOS*, una pequeña cantidad de memoria RAM de generalmente 128 o 256 Bytes, es almacenar los datos del *Setup* para que éstos no se pierdan al desconectar el equipo. Cuando se inicializa el equipo, el *BIOS* lee estos valores y actúa de acuerdo con ellos. Sin embargo y por ser un tipo de memoria RAM, el *CMOS* es volátil, o sea que pierde sus valores cuando deja de recibir corriente eléctrica. Justamente por esto, se usa una pequeña batería en la placa base destinada a alimentar el *CMOS*. Claro que esta batería no dura siempre, de modo que cada 2 o 3 años es preciso cambiarla.

17.2 POST

Durante el arranque, el *BIOS* realiza una serie de pruebas intentando detectar con exactitud los componentes de hardware instalados en el equipo. Esta prueba se conoce por *POST*, acrónimo de *Power-On Self Test*. Los datos del *POST* se muestran durante la inicialización en forma de tabla y que aparece antes de cargar el sistema operativo, indicando la cantidad de memoria instalada, así como los discos duros, unidades de disquetes, puertos serie y paralelo y unidades de *CD-ROM/DVD-ROM* instalados en el equipo. Además de detectar el hardware instalado, otra función del *POST* es verificar si todo está funcionando correctamente. Si se detecta algún problema en un componente necesario para el funcionamiento del sistema, como las memorias, el procesador o la tarjeta gráfica, el *BIOS* emite una cierta secuencia de pitidos sonoros, alertando del problema.

17.3 Actualización del BIOS

A pesar de su importancia, el *BIOS* no deja de ser un programa como otro cualquiera, pero que, en vez de guardarse en el disco duro, queda almacenado en un *chip* de memoria de la placa base. Antiguamente, se usaban *chips* de memoria *ROM* para almacenar el *BIOS*. La memoria *ROM* era solamente de lectura, con el contenido inalterable.

En las placas base más actuales, sin embargo, el *BIOS* es grabado en memorias *Flash*. El uso de este tipo de memoria permite que el *BIOS* sea modificado, para poder corregir eventuales *bugs* o aumentar el grado de compatibilidad de las distintas placas base, añadiéndoles nuevos recursos.

Esta actualización se conoce por *upgrade* del *BIOS*. Los fabricantes ponen tales *upgrades* en sus páginas de *Internet* para su descarga gratuita, donde la actualización viene en forma de un archivo binario (.bin) y una utilidad que realiza la grabación de los datos en el *chip* que almacena el *BIOS*. Si ya estamos en posesión de los archivos necesarios, basta con iniciar el equipo a través de un disquete de arranque y ejecutar la aplicación encargada de realizar la grabación.

Durante la actualización, los datos del *BIOS* son completamente reescritos en un proceso que suele durar unos pocos minutos. El problema es que se interrumpa esta actualización por un pico de tensión que reinicie el equipo, ya que entonces el *BIOS* quedará dañado, y muy probablemente imposibilite el inicio del equipo para reiniciar la actualización, inutilizando la placa base. Justamente por esto, los fabricantes no suelen dar ningún tipo de garantía sobre los daños causados por una actualización del *BIOS*, alegando que es el usuario el que acepta correr sus propios riesgos.

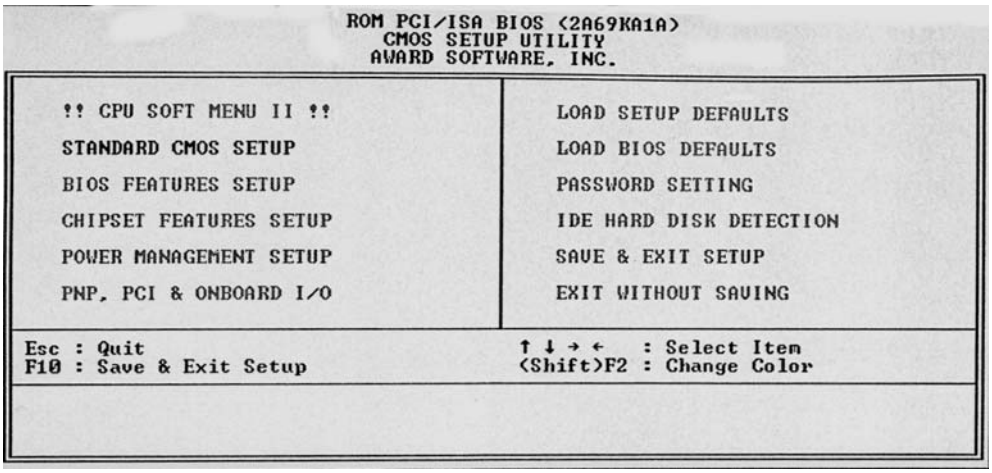
17.4 Configurar el CMOS Setup

Decimos que el BIOS es la primera capa de software del sistema ya que cuando se conecta el equipo, el *BIOS* es la primera cosa que se carga. La función del *BIOS* es configurar varios recursos de la placa base, principalmente, las direcciones *IRQ* y *DMA* usadas por los periféricos instalados y, enseguida, dar la señal de inicio del equipo, cargando el sistema operativo y pasándole el control del equipo a él.

Por su parte, el *Setup* es un pequeño programa que permite configurar el *BIOS*. La función de la batería de la placa base es, justamente, mantener las diferentes configuraciones establecidas en el *Setup* cuando se desconecta el equipo. Para acceder al *Setup* es suficiente con pulsar la tecla <Supr> durante el recuento inicial de la memoria. En algunas placas base es necesario pulsar la tecla <F1> o alguna otra combinación de teclas. En caso de duda, deberíamos consultar el manual de la placa base.

Aquí trataremos sobre las opciones con las que sólo deberíamos probar si sabemos exactamente qué podemos provocar con cada una de las configuraciones, ya que resulta muy fácil desconfigurar el PC de manera que deje de funcionar.

En un caso así, normalmente no queda otro remedio que eliminar la *RAM CMOS* que guarda las configuraciones del *Setup* del *BIOS*.



Una pantalla inicial de ejemplo del Setup del BIOS

Una correcta configuración del *Setup* del *BIOS* es un requisito fundamental para que funcione bien un PC, antes de tomar ninguna medida de configuración del hardware en el sistema operativo o en el software instalado en el equipo.

Las configuraciones del *BIOS* que establecen los fabricantes de PCs son aquellas con las que el PC funciona de forma normal, pero no aprovechan al máximo su funcionalidad o rendimiento. Sólo mediante a una configuración del hardware personalizada a través del *Setup* del *BIOS*, que puede ir perfectamente desde una modificación relativamente fácil como la configuración del modo *PIO* para un disco duro *EIDE* hasta la actualización del *BIOS*, se puede configurar un PC de forma óptima.

17.4.1 BIOS Features Setup

Las opciones avanzadas del *Setup*, es decir, aquellas funciones que van más allá de las posibilidades de configuración del *Setup* estándar, se encuentran en la opción *Advanced Setup*, *BIOS Features Setup* o *Chipset Features Setup*.

Para un rendimiento óptimo del PC es inevitable enfrentarse con este *Setup* de forma más detallada. Las funciones que pueden encontrarse en el *Setup* avanzado del *BIOS* dependen de varias circunstancias, como la versión del *BIOS*, el *chipset*, la placa base y el tipo de PC. Por esta razón, a continuación se han reunido las funciones que son más habituales en los distintos modelos de PC. Algunas de las diferentes opciones también pueden aparecer bajo otra denominación ligeramente distinta, y las pantallas del *Setup* pueden diferir de las indicadas en cuanto al aspecto, aunque eso no varía para nada la mayoría de funciones que ofrece. Tampoco podemos excluir nunca que uno se encuentre con un error del *BIOS* que no pueda eliminarse si no es con una actualización del *BIOS*.

Virus Warning o BootSector Virus Check o Virus Protection (detección de virus)

El sector de arranque del disco duro o de un disquete, particularmente, están en peligro de ser atacados por los virus, ya que los virus colocados ahí pueden destruir completamente el soporte de datos, de modo que a menudo no queda otro remedio que volver a darle formato. Cierta protección ante este tipos de virus es la que se obtiene al activar la opción *Virus Warning* (en otras versiones de *BIOS* también se denomina *BootSector Virus Protection*), por lo que debería utilizarse. No obstante, esto debería ocurrir antes de que se instale el primer software en el equipo, ya que si se activa posteriormente podría ser demasiado tarde, porque posiblemente ya se haya colado un virus que ya no podrá detectarse. El programa de detección de virus del *BIOS* puede hacerse notar, por ejemplo, cuando se instalan o ejecutan programas administradores de memoria o software de red.

Configuración de la memoria caché

Las dos configuraciones para la memoria caché *CPU Internal Cache* y *External Cache* se limitan a activar la caché interna (caché L1) y la caché externa (caché *L2*). Por razones de rendimiento deberíamos tener activados necesariamente estos dos puntos con *Enabled*, puesto que la memoria caché tiene una influencia decisiva en la velocidad de ejecución del PC.

No obstante, en este punto no tiene ninguna importancia en qué forma estén realizadas las dos cachés y qué capacidad nos ofrezca cada una de ellas, ya que por regla general eso se detecta automáticamente, y sólo en las placas base más antiguas puedan encontrarse todavía *jumpers* para la caché externa.

Algunas versiones del *BIOS* también permiten configuraciones más especiales para la memoria caché, pudiendo tener las distintas opciones la denominación *SRAM* (*RAM* estática = *RAM* de caché) en lugar de caché. A continuación se explican las distintas posibilidades de configuración habituales para la caché en el *Setup* del *BIOS*.

Cache Update Policy

En este punto es posible cambiar entre el modo *Write Through* y el modo Write Back, más eficiente. En algunas versiones de *BIOS* podemos encontrar una opción de menú equivalente en el *Setup* del *BIOS* que permita cambiar entre los dos modos de funcionamiento diferentes, pero si se intenta hacer el cambio, en algunos casos no ocurre nada, ya que esta opción aparece en forma de texto, pero su función simplemente está bloqueada, lo cual sucede en algunas placas *486* antiguas que no soportan la caché *Write Back*.

Cache Write Hit Wait State (acceso de escritura a la caché)

El principio de esta configuración se rige por la frecuencia de reloj del procesador y el tiempo de acceso de las *SRAMs* utilizadas para la caché externa, ya que puede ser

necesario utilizar esta opción para establecer ciclos de espera adicionales a fin que la memoria caché pueda utilizarse de una forma perfecta.

Cache Read Hit Burst o SRAM Read Burst Control (acceso de lectura a la caché)

Esta opción permite establecer el ciclo de acceso para el modo *Burst* de lectura, que en lo esencial depende del tiempo de acceso de las *SRAMs* utilizadas. A diferencia de un acceso normal a la memoria, en el modo *Burst* sólo se envía una dirección, la dirección inicial, tras la que siguen los datos como bloque y la electrónica de la caché aumenta la dirección correspondiente por sí misma.

CPU L2 Cache ECC Checking (detección de errores de caché)

Esta opción no se da habitualmente en las placas base *socket 7*, sino sólo en las *slot 1* o también en las del *Celeron* en carcasa *PPGA*. Si debe activarse o no la detección de errores y el mecanismo de corrección de memoria *Error Check Correction* (*ECC*) para la caché *L2* es algo que depende, en último término, de si los *chips* de la caché soportan esta opción, aunque eso queda oculto para el usuario. Por regla general, los procesadores *Pentium II* y *Pentium III* pueden arreglárselas con ambas configuraciones, y en el caso de un *Celeron* la configuración más habitual es *Disabled*.

En algunas versiones de *BIOS* (también en las placas base *socket 7*) en la opción *BIOS Features Setup* también puede encontrarse una configuración similar con el nombre *Memory Parity/ECC Check*, pero esta no se refiere a la caché sino a la memoria *RAM* y, por lo tanto, también depende del *chip* o de los módulos que se utilicen. Generalmente, en caso de duda simplemente desactivaremos el *ECC Check*, lo que no provoca problemas si la frecuencia de reloj es correcta y los *chips* funcionan bien.

Quick Power On Self Test (autotest rápido)

El autotest del PC se efectúa en poco tiempo si esta opción está activada (*Enabled*). Sin embargo, esta configuración sólo debería realizarse cuando el PC ya funciona de forma óptima, ya que en el *Quick POST* algunos pasos como el test de memoria sólo se resuelven en parte. Además, con el *Quick POST* también puede darse el caso de que no se detecten las tarjetas de expansión o incluso los dispositivos *SCSI* conectados en una controladora *host SCSI*.

Boot Sequence (secuencia de arranque)

A través de esta opción se establece en qué orden el *BIOS* debe buscar el sistema operativo en las distintas unidades. Habitualmente aquí se indica la secuencia *C, A*, con lo cual se arranca desde el disco duro y sólo si eso falla, por alguna que otra razón, se accede a la unidad de disquetes *A*, para cargar el sistema desde esta unidad. En versiones de *BIOS* más recientes también se establece a menudo una unidad de *CD-ROM* como fuente de arranque. Pero para ello el *CD-ROM* tiene que crearse de forma espe-

cial (tiene que ser iniciable), para lo cual el software de grabación de *CDs* tiene que ofrecer el soporte pertinente en caso de que deban crearse *CDs* de inicio..

Swap Floppy Drive (cambiar asignación de unidades de disquete)

Habitualmente, la unidad de disquetes *A* corresponde al tipo de 3.5 pulgadas y la unidad *B* al anterior tipo de 5.25 pulgadas, lo cual naturalmente, sólo es corriente en los PCs muy antiguos. Este orden de unidades viene establecido por la conexión en la controladora correspondiente, utilizando la unidad de disquetes *A* un cable con contactos trenzados. Si es preciso cambiar *(swap*) esta asignación, por ejemplo porque se espera un disquete de 5.25 para instalar un programa en la unidad *A*, deberá activarse esta opción de menú y no tendrán que cambiarse los cables de su lugar. En los PCs con una única unidad de disquetes, casi todos, lógicamente no se activa esta opción ya que raramente ocurre que para *A* no se utilice un cable de unidad trenzado.

Boot Up Floppy Seek (detectar unidad de disquete)

Si la opción *Boot Up Floppy Seek* está definida como *Disabled*, durante el proceso de arranque no se comprueba de qué tipo de unidad de disquete se trata. Dado que esta medida sólo es pertinente para la correcta detección de una unidad de 360 KB antigua, esta opción debería dejarse configurada siempre como *Disabled*. En caso de error, en caso de que la unidad de disquete (aparentemente) no funcione, la configuración *Enabled* se encargará de efectuar un breve acceso a la unidad y, por lo tanto, de que se ilumine brevemente el diodo luminoso, lo cual puede suministrar una valiosa información para la localización del error.

Boot Up Num Lock Status (configuración de teclado)

El bloque derecho del teclado puede utilizarse para introducir números o para el manejo del cursor con las pertinentes teclas de desplazamiento. El cambio de un modo de funcionamiento a otro se realiza a través de la tecla <Bloq Num> del teclado. Cuando está activado el bloque numérico está encendido el indicador correspondiente en el teclado, y el bloque de teclas sirve para introducir números. Si se quiere que este modo de funcionamiento se configure inmediatamente después de arrancar, lo cual, naturalmente, puede deshacerse de nuevo en cualquier momento con la ayuda de tecla <Bloq Num>, se debe definir *On* en la opción de menú *Boot Up Num Lock Status*.

IDE HDD Block Mode (modo de funcionamiento para IDE)

Los datos de un disco duro *IDE* pueden transferirse en bloques (de 512 Bytes) en lugar de hacerlo individualmente, lo cual debería comportar un incremento de velocidad en la transferencia de los datos. Muchas veces sólo hay que activar o desactivar (*Enable*, *Disable*) el modo *Block*, pero también existen versiones de *BIOS* en las que puede indicarse explícitamente un número de bloques (*Multi Sector Transfers*).

El número posible de bloques depende del disco duro utilizado y en el fondo sólo se obtiene de la documentación que acompaña al disco duro. En caso de que no se conozcan estos datos, lo cual ocurrirá en la mayoría de los casos, es preferible dejar esta configuración tal y como está, ya que de todos modos sólo surte efecto para el sistema *DOS* y además, en *Windows*, puede ser la causa de serios problemas (por ejemplo, al grabar *CDs* o *DVDs*).

Typematic Rate (opciones del teclado)

A menudo, en los *Setups* de los *BIOS* hay implementadas varias posibilidades de configuración para el comportamiento del teclado. Los parámetros que se establecen en ellas son válidos para todos los programas instalados y se indican a continuación con su opción predeterminada (*default*), tal y como aparecen normalmente:

- *Typematic Rate Programming o Typematic Rate Setting: Disabled*
- *Typematic Rate (Chars/Sec): 10*
- *Typematic Rate Delay (Msec): 250*

Si la opción de menú *Typematic Rate Programming* está definida como *Disabled* las otras dos opciones no tienen ningún sentido, ya que no están activadas. Sólo con la opción en *Enabled* se activan los parámetros de las opciones *Typematic Rate Delay* y *Typematic Rate*. La opción *Typematic Rate Delay* es el tiempo en milisegundos que debe transcurrir hasta que se puede representar repetido el carácter pulsado cuando se mantiene pulsada su tecla correspondiente. La opción *Typematic Rate*, en cambio, establece la velocidad de repetición del carácter en caracteres por segundo.

Los sistemas operativos de la familia *Windows* ofrecen sus propias posibilidades de ajuste para el teclado y las configuraciones allí realizadas sobrescriben las indicadas en el *Setup* del *BIOS*, con lo cual son válidas para todos los programas que se ejecutan bajo *Windows*.

Security Option (solicitud de contraseña)

Podemos proteger el PC con una contraseña, lo cual generalmente se realiza a través de una entrada adicional en el menú principal del *Setup* del *BIOS* (por ejemplo, *Password Setting*). En este punto (**Security Option**), en cambio, podemos establecer si se debe solicitar una contraseña con cada arranque (*System*) o sólo cuando se llama al *Setup* del *BIOS* (*Setup*). La opción *Security Option* es en general la vía más fácil y efectiva de proteger el PC ante los accesos de extraños.

Gate A20 Option o Port 92 o Fast A20 G (cambio de modo)

La función para cambiar de *Real Mode* (procesador *8086*, modo *DOS*) al *Protected Mode* ya tiene más de 13 años de antigüedad y de vez en cuando aparece con diferentes nombres, incluso en algunas placas nuevas en los *Setups* de los *BIOS*. El método de cambio *Gate A20* puede establecerse con esta opción. La opción predeterminada

Fast, a ser posible, no debería modificarse ya que la configuración *Normal* sólo comporta mermas en el rendimiento.

PCI/VGA Palette Snoop (opción para tarjetas de vídeo ISA)

Esta configuración en principio deberá definirse como *Enabled* sólo cuando se utilizan tarjetas de vídeo *ISA* (sintonizador de TV, capturadora de vídeo), pero no tarjetas gráficas. Con ella, los accesos de *E/S* al registro de paletas también se reproducen en el bus *PCI*.

Esto es necesario, por ejemplo, cuando se controla una tarjeta de vídeo con ayuda de los conectores *Feature* de la tarjeta gráfica *PCI/AGP*. De no ser así, la tarjeta de vídeo no recibiría ninguna información sobre el cambio de color y, posiblemente, aparecerían colores incorrectos o no aparecería ninguna imagen de vídeo en absoluto.

OS Select For DRAM > 64 MB u OS/2 Onboard Memory > 64 M (anular la limitación de memoria para OS/2)

Cuando se utilizan más de 64 MB de *DRAM* con el sistema operativo *OS/2* hasta la versión 3.0 (*Warp*, *Fix Level 3*) debe activarse esta opción, ya que de lo contrario este sistema operativo no puede utilizar la memoria instalada por encima de los primeros 64 MB. Para *Windows*, esta configuración no es relevante y debería definirse con las opciones *Non-OS2* o *Disabled*.

Report No FDD For Win 95 (opción para Windows 9x)

Una posibilidad de configuración, en realidad superflua, que sólo debe definirse con *Yes* en caso de que no exista una unidad de disquetes (*Floppy Disc Drive*) y se utilice *Windows 9x*.

Delay IDE Initial <sec> o Delay for HDD (retraso para dispositivos IDE)

La detección de discos duros *(E)IDE* puede prolongarse en un intervalo a definir (normalmente de 0 a 15 segundos), lo cual sólo es necesario para modelos de disco duro lentos y antiguos y, por lo tanto, debería configurarse a *0*. Sin embargo, puede ser perfectamente conveniente activar esta opción para otros dispositivos (*ATAPI*), que estén conectados en un puerto *(E)IDE* y no sean detectados por el *BIOS*, como una unidad *ZIP* o de *CD-ROM/DVD-ROM*.

Shadow RAM (copiar el contenido del ROM del BIOS)

Tanto para el *BIOS* de la tarjeta gráfica (*Video BIOS*) como para otros campos del segmento de los adaptadores puede establecerse la RAM Shadow. Si el área en la que se encuentra el *BIOS* de una tarjeta de expansión o del sistema (*System BIOS Shadow*) está definida como *Enabled*, se copian las rutinas del *BIOS* en el área *RAM* correspondiente únicamente con el *ROM* del *BIOS* de 8 bits de ancho.

Proporciona un acceso más rápido debido a su arquitectura de 32 bits (a partir del *386*) y a los breves tiempos de acceso de las DRAMs respecto a las *EEPROMs* o *PROMs Flash*. Lo importante es que el área establecida como *RAM Shadow* esté protegida contra escritura y no pueda utilizarse como un área RAM corriente. ‡nicamente en los PCs antiguos está prevista para ello una opción de menú, mientras que el resto lo hacen automáticamente al activar la *RAM Shadow*.

En general, la función *Shadow RAM* ha dejado de ser de importancia técnica para el rendimiento, ya que se utilizan controladores de dispositivos propios en lugar de las rutinas del *BIOS*. Además, las tarjetas *PCI* utilizan direcciones asignadas automáticamente por el *Plug & Play* del *BIOS*, por lo que éstas no son fijas ni tampoco pueden trasladarse manualmente a un área de *RAM Shadow*.

17.4.2 Chipset Features Setup

Lo que los distintos fabricantes de *BIOS* ofrecen en la opción *Chipset Features Setup*, también llamado *Advanced Setup*, en cuanto a posibilidades de configuración, lamentablemente es muy diferente en comparación con otras páginas del *Setup* del *BIOS*, ya que aquí en principio pueden configurarse las cosas específicas del juego de *chips*.

Por ejemplo, en caso de que la página principal del *Setup* del *BIOS* no contenga ninguna opción de menú especial para los *On Board Devices* (controlador *IDE*, puerto paralelo, etc.), las posibilidades de configuración necesarias para ello suelen encontrarse en la opción *Chipset Features Setup*. O si no existe una página específica para las configuraciones de microprocesadores slot 1 y *Celeron*, a menudo uno encontrará lo que busca en la opción *Chipset Features Setup*.

Configuraciones de memoria DRAM (EDO, SDRAM)

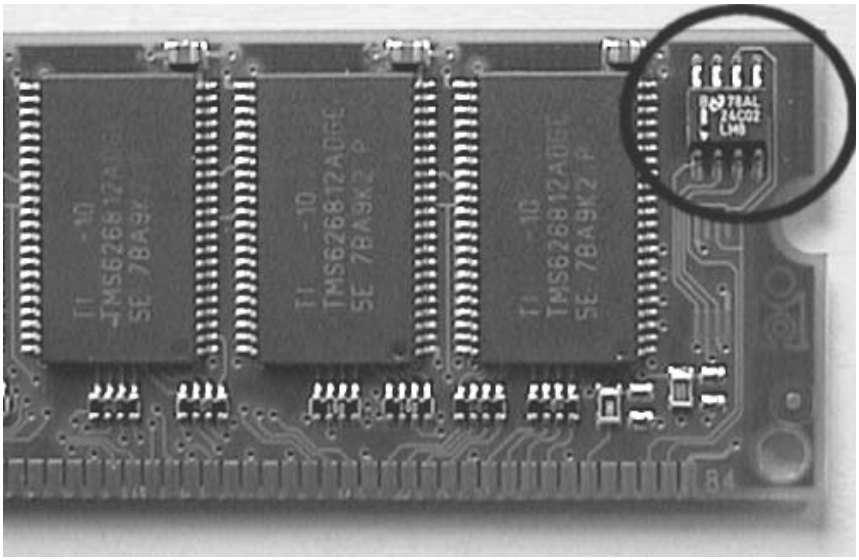
Para un "ajuste refinado" de la *DRAM* suelen encontrarse varias entradas en la opción *Chipset Features Setup*. En función de los módulos de memoria soportados e instalados, las opciones que podemos encontrar aquí son diferentes y también pueden llegar a ser bastante complejas.

En el más sencillo de los casos se define la opción de menú *Auto Configuration* con *Enabled*, con lo cual posiblemente en algunas versiones de *BIOS* todavía falte por indicar el tiempo de acceso de las *DRAMs* utilizadas. El resto de configuraciones para la memoria las establece luego el *BIOS* automáticamente y, en la mayoría de los casos, de este modo se obtiene un sistema estable.

Las modificaciones manuales de los parámetros de temporización sólo deberían realizarse en caso de que el PC no pueda trabajar con los módulos de memoria (bloqueos repentinos del sistema, errores de memoria o similares).

A partir de placas base *Pentium II* y *Super Socket 7*, junto a las configuraciones para la *RAM EDO* (*Extended Data Output*) podíamos encontrar las configuraciones para la *SD-*

RAM, en caso de que la placa base soportase ambos tipos de memoria, pero en principio obedecen a la misma regularidad (por lo menos en el Setup) que las *DRAMs EDO* o que las *DRAMs Fast Page Mode* (*FPM*), que actualmente son muy difíciles de conseguir.



Los DIMMs actuales contienen una EEPROM, que se encarga de la detección del tipo de memoria y activa los parámetros adecuados

A pesar de todo, los parámetros para *SDRAMs*, que representan el estándar en todas las placas base recientes, en realidad no deberían definirse manualmente sino automáticamente con la ayuda de la *EEPROM* integrada en el mismo módulo de memoria.

Sin embargo, en caso de que eso no sea posible, porque el módulo *SDRAM* no contiene la *EEPROM* o porque sencillamente esta función no funciona, primero deberíamos empezar con los parámetros menos peligrosos (por ejemplo, *SDRAM RAS Precharge Time: Slow*).

No obstante, lo que para *SDRAMs* suele funcionar sin problemas es el punto *Auto Detect DIMM/PCI Clock*, que al ser activado configura la frecuencia óptima para las memorias *SDRAMs*, que habitualmente se corresponde con la frecuencia del sistema.

Memory Hole At 15M-16M (establecer huecos de memoria)

En la mayoría de los casos este punto se deja desactivado con *Disabled*, ya que de lo contrario se establece un "agujero" en el área de memoria de 15-16 MB que suele utilizarse para las memorias *ROM* de las tarjetas *ISA*.

Existen muy pocas tarjetas *ISA* (controladoras especiales, tarjetas de vídeo, tarjetas gráficas ATI antiguas) que requieran el uso de esta función.

Configuraciones del bus ISA

Para la utilización de tarjetas *ISA* antiguas a menudo existen dos configuraciones a definir para la opción *I/O Recovery Time*, que se configura por separado para los accesos de 8 y de 16 bits. Sin embargo, este tipo de retrasos adicionales sólo son necesarios en caso de que las tarjetas no funcionen de forma estable, lo cual es necesario, por ejemplo, para algunas tarjetas de red antiguas compatibles *NE2000*.

Habitualmente existe una posibilidad de configuración para la frecuencia del bus *ISA* (*ISA Bus Clock*) sólo para placas base que no utilizan un *chipset* de la casa Intel.

La frecuencia del bus *ISA* normalmente es *PCICLKI/4*, lo cual equivale a 8.25 MHz. En función del juego de *chips* y del tipo de *BIOS* puede aumentarse la frecuencia del bus *ISA*, aunque, naturalmente, no puede llegar tan lejos como para que las tarjetas *ISA* dejen de funcionar. No obstante, la mayoría de tarjetas ISA resisten perfectamente una frecuencia de 10 MHz (*PCICLKI/3*).

Configuraciones del bus PCI

En la opción *Chipset Features Setup* también pueden aparcerer varias opciones para la configuración del bus *PCI*, aunque deberían encontrarse en otra página del *Setup* con el nombre *PCI Configuration* o *PNP/PCI Configuration*.

PCI-Slot IDE 2nd Channel

Este punto está desactivado en la mayoría de los casos y sólo es importante cuando en una ranura *PCI* se encuentra instalada una tarjeta controladora *(E)IDE* adicional, que se activa pertinentemente con *Enabled*. No obstante, entonces, la controladora *EIDE Onboard* tiene que estar desactivada.

PCI Bursting

Para la transferencia de datos entre componentes *PCI* y la memoria del PC puede activarse o desactivarse el rápido modo *Burst*. Por razones de rendimiento, es recomendable utilizar el modo *Burst*, pero ocurre que algunas tarjetas gráficas, especialmente de la primera generación *PCI*, no se las arreglan bien con ese modo y, por tanto, es preciso desactivar el modo *Burst Write*.

Algunos PCs permiten otras opciones de optimización *PCI*, que en otros sistemas *Pentium* ya vienen establecidas de fábrica y, por tanto, tampoco aparecen en el *Setup*. Entre las opciones más importantes están las configuraciones de *búfer*.

PCI Buffer y CPU Buffer (activar memoria intermedia)

Estas memorias intermedias sirven para la comunicación entre el procesador, la memoria *RAM* y los componentes *PCI* entre sí. A ser posible deberían estar activadas. Por ejemplo, si el procesador tiene que enviar datos al bus *PCI*, coloca los datos en el *Write*

Buffer correspondiente (*CPU to PCI write buffer*) y puede seguir trabajando inmediatamente sin tener que esperar a que el dispositivo *PCI* (por ejemplo, una tarjeta gráfica o una controladora *SCSI*) haya aceptado los datos. Eso lo hará luego el dispositivo *PCI* por su cuenta.

Peer Concurrency y PCI Streaming (opciones PCI)

El punto *Peer Concurrency*, por regla general, debe definirse con *Enabled* y hace que el procesador también pueda acceder a la memoria y a la caché L2 cuando un *Busmaster PCI* está transfiriendo datos (a un *Target*). En caso contrario (*Disabled*) los datos no se anidan, sino que se transfieren consecutivamente, lo cual tiene como consecuencia una pérdida de rendimiento.

No obstante, en algunas tarjetas *PCI* (normalmente antiguas) activar la opción *Peer Concurrency* también puede provocar que la transferencia de datos, por ejemplo de una controladora *SCSI* (por ejemplo, de NCR), de repente se quede bloqueada. En estos casos deberá desactivarse esta opción, lo cual también afecta de modo similar a la entrada *PCI* que pueda existir. Al desactivarla, los accesos a los datos no se efectúan directamente uno detrás de otro, sino que se intercala una pausa (ciclo de espera) entre ellos.

Passive Release (opción PCI/ISA)

Esta entrada con la opción de configuración predeterminada *Enabled* se encarga de que un *Busmaster ISA* pueda interrumpir y/o bloquear una transferencia *PCI*, cosa que en un principio parece conveniente. Si el PC está equipado exclusivamente con tarjetas *PCI*, en principio no hay nada en contra de activar esta opción, aunque eso puede provocar problemas en relación con las tarjetas *ISA*.

Las tarjetas de red *ISA* con capacidad *Busmaster* empiezan una transferencia, aunque la misma nunca llega a su fin, de modo que el PC se queda colgado de repente. Después de configurar *Disabled*, el bus *PCI* se queda bloqueado hasta que ha terminado la transferencia *ISA*, lo cual no es favorable para el rendimiento, pero de todos modos hace posible que funcionen las tarjetas *ISA*.

Configuraciones del bus AGP (Accelerated Graphics Port)

Si en la placa base existe una ranura para una tarjeta gráfica *AGP*, existe por lo menos una posibilidad de configuración para ella en el *BIOS*: *AGP Aperture Size* o *Graphics Aperture Size*, aunque puede aparecer en español en alguna *BIOS* de la casa Phoenix.

AGP Aperture Size (MB) o Graphics Aperture Size

Con ayuda de esta opción se establece un tamaño máximo de memoria en MB en la *RAM* del sistema que puede ser utilizado por la tarjeta gráfica *AGP* para las texturas. El tamaño puede definirse habitualmente en 4, 8, 16, 32, 64, 128 o 256 MB, dependien-

do el tamaño óptimo de si la tarjeta *AGP* ya contiene una memoria de texturas propia y si es así, de qué tamaño tiene.

El tamaño de memoria definido aquí se utiliza dinámicamente, lo cual significa que el área se adapta a la cantidad de datos, y en absoluto tiene 4 u 8 MB de forma constante, sino que sólo indica el máximo que se va a usar. En la práctica, el valor óptimo sólo puede averiguarse experimentalmente, en caso de que el manual de la tarjeta gráfica *AGP* no proporcione la información exacta sobre este apartado.

Cuanto más reciente es la placa base, más posibilidades de configuración *AGP* se encuentran en el *Setup* del *BIOS*, como las que podemos ver a continuación.

AGP-2x Mode: Disabled/Enabled

El modo de transferencia *AGP* estándar se denomina 1x, y todas las tarjetas gráficas *AGP* antiguas soportan únicamente este modo, mientras que las posteriores ofrecen el modo 2x y, por tanto, debe activarse esta opción en el *BIOS*. No fue hasta los modelos del año 1999 que se soportó también el modo 4x, como las tarjetas con el *chip* RIVA TNT2, como por ejemplo del tipo Maxi Gamer de la casa Guillemot o Viper V770 de la casa Diamond. Así pues, en el *Setup* del *BIOS* de las placas base actuales debería dejarse activado este modo, porque de no ser así no puede utilizarse el modo 4x, con el que tanta publicidad hacen los fabricantes de tarjetas gráficas. Uno de los primeros *chipsets* que pudo ofrecer el modo 4x no procedía de Intel, sino de la casa VIA (VT82694) y el primer *chipset* Intel que permitió éste modo fue el *Camio*.

AGP Master 1 WS Write: Disabled/Enabled

Con esta opción se establece para la tarjeta gráfica *AGP* (*master*) un ciclo de espera adicional (*WS = Waite State*) para las operaciones de escritura.

AGP Master 1 WS Read: Disabled/Enabled

Aquí, de forma análoga, se establece un ciclo de espera para las operaciones de lectura *AGP*. Ambas opciones deberían activarse únicamente en caso de que durante el funcionamiento de la tarjeta *AGP* se manifestasen problemas en forma de errores de píxeles o de imagen en general. Pero, habitualmente, no hay que definir ningún ciclo de espera. Cuando se dan este tipo de problemas la causa puede ser una frecuencia del bus *AGP* demasiado elevada (66 MHz es lo normal) o un modo del bus *AGP* incorrecto (1x, 2x, 4x).

17.4.3 Power Management Setup

Las posibilidades de configuración para las funciones de ahorro de energía de un PC, a veces también llamadas *Green PC Functions*, varían mucho según las distintas marcas. Además, no todos los componentes de un PC tienen incorporada esta capacidad de ahorro de energía.

Generalmente, las funciones de *Power Management* aspiran a que cuando el PC está conectado pero no está siendo utilizado en ese preciso momento, algunos componentes del mismo puedan desactivarse totalmente o bien pasar a un estado de ahorro de energía. Para poder ahorrar una gran cantidad de energía es preciso que numerosos componentes, como la fuente de alimentación del PC, el monitor, el procesador, el disco duro y la placa base estén equipados específicamente para ello con el *BIOS*.

Para los distintos componentes se configura un modo especial en el *Setup* del *BIOS* con un valor de tiempo predeterminado (normalmente de 1 minuto a 1 hora), que entra en acción tras ese tiempo. Después de pulsar una tecla, o cuando se produce un evento, el PC vuelve a pasar al modo de funcionamiento normal.

A veces, en el *Setup* es posible activar un gran número de funciones de ahorro de energía diferentes, aunque están asociadas a inconvenientes inesperados, de modo que a menudo es mejor dejar desactivada la opción *Power Management* en el *Setup* (*Disabled*). Uno de estos inconvenientes, por ejemplo, se da cuando deben realizarse operaciones matemáticas y el teclado o el ratón no se utilizan durante mucho tiempo. Después del tiempo establecido se reduce automáticamente la frecuencia del procesador y el cálculo tarda mucho más, lo cual provoca en último término un consumo de energía superior. Otra situación en la que uno debería prescindir necesariamente de las funciones de ahorro de energía, es en la grabación de *CDs* o *DVDs*. Aquí, si se presenta un evento de ahorro de energía puede comportar una interrupción del proceso de escritura, con lo cual normalmente el *CD* o *DVD* resultante queda inservible.

La forma más eficaz y menos peligrosa de ahorrar energía consiste, simplemente, en apagar manualmente el monitor cuando ya no se necesita. A pesar de todo, no puede faltar una breve explicación de las funciones de ahorro de energía habituales, aunque en este punto no tiene sentido entrar una a una en todas las funciones posibles, que pueden variar mucho según el fabricante y la versión de la placa base o del *BIOS*. En caso de duda es preferible hacer caso omiso de las funciones de ahorro de energía del *BIOS* y en lugar de eso configurar las cosas que a uno le parecen importantes a través del propio *Windows*.

En principio, para el procesador son concebibles dos mecanismos distintos. Todos los procesadores a partir del microprocesador *486* (con una *S* en su nombre) poseen un modo de ahorro de energía integrado. Sin embargo, este sólo puede activarse en el *BIOS* cuando realmente hay instalado un procesador de ese tipo. Para los otros procesadores más antiguos, el *BIOS* simplemente reduce la frecuencia (por ejemplo, a 8 MHz), ya que una frecuencia menor también significa un menor consumo de energía. Este método funciona con cualquier procesador, ya que éste no tiene que disponer de un modo de ahorro de energía propio.

La reducción de energía para el monitor sólo resultará eficaz en un tipo correspondientemente equipado para ello con función *Power Saving*. Entonces, después del tiempo establecido en el *Setup*, el *BIOS* desactiva brevemente las señales de sincronización (*V-Sync* y *H-Sync*) a través de la tarjeta gráfica, con lo cual el monitor se des-

conecta completamente o bien reduce su consumo de energía. No obstante, ese monitor debería ser necesariamente compatible con *DPMS* (*Device Power Management System*), ya que un monitor sencillo puede resultar dañado mediante los pulsos de las señales de sincronización. Si está activado el punto *HDD Power Down*, después del tiempo establecido el disco duro *(E)IDE* recibe del *BIOS* el comando *IDE Power Down* definido, con lo cual pasa automáticamente al modo *Standby*. En cambio, la alimentación del disco duro sólo se desactiva cuando en el PC se encuentra una fuente de alimentación especial.

```
                    ROM PCI/ISA BIOS
                POWER MANAGEMENT SETUP
                 AWARD SOFTWARE, INC.

Power Management : Enabled        Power Down Activities
Video Method     : DPMS           IRQ3 (COM2)            : On
                                  IRQ4 (COM1)            : On
Doze Mode        : 10 Min         IRQ5 (LPT2)            : Off
Standby Mode     : 20 Min         IRQ6 (Floppy Disk)     : Off
Suspend Mode     : 1  Hour        IRQ7 (LPT1)            : Off
HDD Power Down   : 15 Min         IRQ8 (RTC Alarm)       : Off
                                  IRQ9 (IRQ2 Redir)      : Off
                                  IRQ10(Reserved)        : Off
                                  IRQ11(Reserved)        : Off
                                  IRQ12(PS/2 Mouse)      : Off
                                  IRQ14(Hard Disk)       : Off
                                  IRQ15(Reserved)        : Off

                                  ESC: Quit      ↓→↑←:Select Item
                                  F1 : Help        PU/PD/+/-:Modify
                                  F5 : Old Values (Shift)F2 :Color
                                  F7 : Load Setup Defaults
```

En la opción Power Management Setup se encuentran las configuraciones para las funciones de ahorro de energía, que también pueden desactivarse completamente con Disabled

Todos los discos duros *IDE* (a partir de 1993) deberían ser capaces de ejecutar el comando *IDE Power Down*, pero en la práctica lamentablemente no siempre es así. Sin embargo, para los discos duros *SCSI* no puede utilizarse las opciones del punto *HDD Power Down*. Generalmente, en el Setup pueden encontrarse tres grados de *Power Managemen*t habituales:

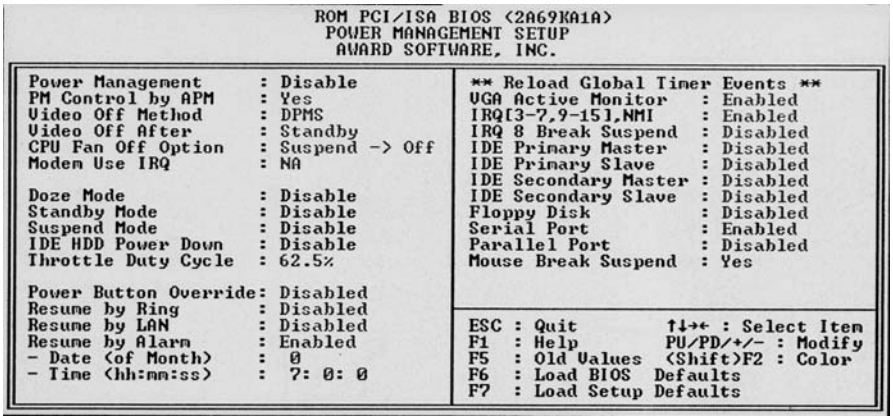
- *Doze Mode*: el consumo de energía del procesador se reduce aproximadamente el 80% una vez transcurrido el tiempo configurado.
- *Standby Mode*: el consumo de energía del procesador se reduce aproximadamente el 92% una vez transcurrido el tiempo configurado. Con la configuración *DPMS* para el monitor se reduce de forma análoga el consumo de energía del monitor.
- *Suspend Mode*: el consumo de energía del procesador se reduce aproximadamente el 99% una vez transcurrido el tiempo configurado, y con la configuración *DPMS* para el monitor se reduce el consumo del monitor en dependencia del tipo en cuestión. Por lo demás, el disco duro pasa al modo *Sleep* o, si la placa base y la fuente de alimentación están equipadas para ello, se desactivan completamente.

El PC no sólo puede ser devuelto a su modo de funcionamiento normal pulsando una tecla del teclado, sino también mediante eventos que se presentan externamente, desarrollados a través de las interrupciones habituales. Un ratón puede estar conectado en el puerto *COM1* o *COM2*, que generalmente tiene asignada la interrupción 4 o 3. En cambio, la responsable de un ratón *PS/2* es normalmente la *IRQ* 12. Después de activar (*On*) la opción de menú pertinente es posible "devolver la vida" al PC acto seguido, mediante la interrupción correspondiente que se desencadena con el movimiento del ratón.

Por lo demás, según el tipo de *BIOS* y siempre y cuando se haya activado (*On*) la opción de menú correspondiente, también otras unidades del PC, como el reloj en tiempo real (*RTC*) o una unidad de disquetes pueden volver a activar el sistema. Los eventos externos para el puerto *LPT* (*Line Printer*, puerto de impresora) sólo pueden procesarse si está configurado para el funcionamiento bidireccional (*EPP*, *EPC*) y no se ha definido la configuración *Normal*. También son eventos externos para los puertos *COM*, además del accionamiento del ratón, las llamadas entrantes de un módem. Si una tarjeta de red soporta la función *Wake On LAN*, también es posible "despertar" al PC a través de la red, para lo cual se necesita una conexión por cable adicional entre la tarjeta y la placa base.

APM y ACPI (funciones avanzadas de ahorro de energía)

Dado que para las funciones de ahorro de energía que pueden configurarse en el *Setup* del *BIOS* desde hace tiempo no se ha consolidado ningún estándar general, Intel y Microsoft definieron el *Advanced Power Management* (*APM*), por lo que esta opción de menú (*PM Control By APM*) no aparece hasta en las versiones más recientes de *BIOS*. Si se activa este punto, el *BIOS* trabaja según el estándar *APM*, de lo cual se aprovecha *Windows* y puede activar y desactivar las unidades de acuerdo con eso.



El gran número de funciones de Power Management y APM, así como también de ACPI, lamentablemente no cambian para nada el hecho de que en la mayoría de los casos apenas resulta conveniente utilizarlas en el equipo

Un perfeccionamiento de ellas es el *Advanced Configuration and Power Management* -abreviado *ACPI*-, y a partir de la versión *Windows 98*, con él deberían abrirse útiles propiedades para el PC, igual que ocurre desde hace tiempo con los ordenadores portátiles. Entre ellas están las funciones *Suspend to RAM* y *Suspend to Disk*, que deben encargarse de que el estado actual del sistema pueda retenerse y almacenarse de forma automática.

Cuando, posteriormente, se reinicia el PC se restablece el estado anterior con la ayuda del *BIOS* y se ejecuta una rutina de inicio especial de *Windows*, de modo que se evita el fastidioso arranque del sistema y, por así decirlo, en un tiempo relativamente breve uno puede continuar trabajando en el punto donde lo había dejado. Seguramente esta sería una función deseable, si también funcionara en la práctica, lo cual queda justificado en primer lugar por la falta de soporte *ACPI* del propio *Windows 98* de la época. En segundo lugar, para cada unidad de hardware que interviene en el *ACPI* tiene que existir un controlador apropiado, que habitualmente tendría que proporcionar el fabricante del hardware, aunque eso es algo con lo que actualmente uno no puede contar de forma segura.

Lamentablemente, problemas como por ejemplo que el PC simplemente no pase al modo *Sleep* o *Standby* o que no pueda volver a iniciarse correctamente, siguen siendo características típicas de *ACPI*, de modo que para mayor seguridad este modo no debería estar activado en *Windows 98*, ya que sólo puede anularse si se define manualmente como 0 la clave *ACPI* en el registro.

Además, el soporte *ACPI* en el *BIOS* debería desactivarse igualmente, aunque eso no es posible en muchas placas base. La función *ACPI* del *BIOS* sólo puede activarse a través de *Windows 98*, pero no es tan sencillo deshacer el proceso sin más. Fue necesario esperar a *Windows 2000* para que esto tomase un giro favorable. En el *Setup* del *BIOS*, la función *ACPI* no tiene que encontrarse obligatoriamente en la página *Power Management*. En un *BIOS* de la casa Award, por ejemplo, la entrada equivalente (*ACPI I/O Device Node*) se encuentra en la página *PNP/PCI-Configuration*.

17.4.4 Plug & Play y PCI Setup

Puede extrañar que para unidades *Plug & Play* existan posibilidades de configuración en el *Setup* del *BIOS*, ya que *Plug & Play* significa precisamente "conectar y empezar" y esta característica debería dispensar al usuario de los trabajos de configuración manuales. En realidad, el sistema *Plug & Play* sólo funciona sin grandes problemas cuando en el PC se encuentran exclusivamente tarjetas *PCI*, ya que el *BIOS Plug & Play* asigna automáticamente a estas tarjetas los recursos del PC necesarios (por ejemplo, las interrupciones y direcciones).

Los problemas empiezan a aparecer cuando en el PC también se encuentran tarjetas *ISA* que son *Plug & Play* y, posiblemente, también tarjetas *ISA* antiguas que tienen que configurarse mediante *jumpers*. Este también es el motivo por el que existen opciones de configuración *Plug & Play* en el *Setup* del *BIOS*.

El sitio en el que aparecen puede variar de nuevo: generalmente la página en cuestión, que puede seleccionarse desde el menú principal del *BIOS*, se llama *PNP/PCI Configuration*. A veces en ella también pueden encontrarse las configuraciones para los *Onboard Devices (Onboard I/O)*, que se tratarán más adelante.

Cualquier PC o cualquier *BIOS* corriente hoy en día puede arreglárselas tanto con tarjetas *Plug & Play PCI* como *ISA*, lo cual hace algún tiempo no era natural ni mucho menos, ya que el bus *PCI* se desarrolló antes que el bus *ISA Plug & Play* en el tiempo. Por lo tanto, existen PCs con lo que se denomina *BIOS Plug & Play* que no ofrece soporte *ISA Plug & Play* y, por consiguiente, en el *Setup* del *BIOS* no se encuentran ninguna posibilidad de configuración para ese sistema, sino solamente una página del *Setup* llamada *PCI Configuration*. No obstante, eso no es grave, por lo menos si todavía se utiliza *Windows 95*, ya que este sistema operativo incorpora una funcionalidad *Plug & Play* propia.

Lo que resulta grave en ese tipo de versiones del *BIOS* (muy antiguas), es el hecho de que aquí, a menudo, uno tiene que ocuparse de configuraciones especiales específicas de *PCI* y, posiblemente, también tendremos que definir *jumpers* en la placa base para la asignación de interrupciones *PCI* en las distintas ranuras, que tienen que corresponderse con las configuraciones del *Setup* del *BIOS*.

En una placa base *PCI* existen tres o cuatro ranuras *PCI* para tarjetas de expansión. En diseños *PCI* anteriores por lo menos una ranura tiene capacidad *master*, y el resto están previstas exclusivamente para tarjetas en modo esclavo. Un *master* puede encargarse del control del sistema en lugar del microprocesador de la placa base.

```
                    ROM PCI/ISA BIOS (P/I-SP3G)
                        CMOS SETUP UTILITY
                     PCI CONFIGURATION SETUP

Slot 1 (RIGHT)                      Note :
   Latency Timer : 80 PCI Clock     All of PCI adapters should use INTA.
   Using IRQ     : 10               BIOS will route each INTA# to
   Trigger Method: Level (Auto)     corresponding IRQ automaticalley.

Slot 2 (MIDDLE)
   Latency Timer : 80 PCI Clock
   Using IRQ     : 11
   Trigger Method: Level (Auto)

Slot 3 (LEFT)
   Latency Timer : 80 PCI Clock
   Using IRQ     : 15
   Trigger Method: Level (Auto)

NCR Latency Timer: 80 PCI Clock       ESC: Quit      ↓→↑←:Select Item
NCR Using IRQ    : 9                  F1 : Help        PU/PD/+/-:Modify
                                      F5 : Old Values (Shift)F2 :Color
                                      F7 : Load Setup Defaults
```

Un PCI Configuration Setup de una versión de BIOS sin soporte ISA Plug & Play.
Aquí deben definirse tanto los datos específicos de interrupciones como los de las ranuras PCI

En cambio, todos los diseños a partir del juego de *chips* Triton permitieron instalar una tarjeta *PCI* con capacidad *master* en las cuatro ranuras, lo cual facilita enormemente la configuración, especialmente teniendo en cuenta que ya no es preciso definir ningún *jumper* en la placa base.

En la figura anterior del *Setup* puede observarse una asignación de las interrupciones *PCI* a modo de ejemplo. Aquí se ha realizado una simplificación de la configuración con respecto a los ordenadores *PCI* de la primera generación, ya que para todas las ranuras *PCI* y, por tanto, para todos los dispositivos *PCI* se utiliza siempre la interrupción *PCI INTA*, que a su vez se sirve de diferentes interrupciones *ISA*. Esto es posible porque las interrupciones *PCI*, a diferencia de las interrupciones *ISA*, trabajan con un nivel (*level*) en lugar de hacerlo con un *Edge Triggering* y, por tanto, pueden reproducirse varias interrupciones *ISA* a través de una única interrupción *PCI*.

Como método de *Triggering* está configurado de forma predeterminada la opción *Level*, y las ranuras *PCI* tienen asignadas las *IRQs* 10, 11 y 15. En cambio, si debe llevarse a cabo un *Edge Triggering* para las interrupciones, deberá configurarse la opción de menú correspondiente con *Edge* y, en la placa base, deberá definirse un puente con la misma configuración *IR*Q. No obstante, el *Edge Triggering* no es nada habitual para las tarjetas *PCI*, de modo que en la mayoría de los casos se puede aceptar tranquilamente la opción predeterminada.

Para cada una de las ranuras *PCI*, así como para la controladora *SCSI* que pueda existir, puede definirse un valor para el *Latency Timer* (teóricamente, 0-256). El *Latency Timer* especifica el tiempo durante el que un dispositivo *PCI* puede acceder al bus, a pesar de que el bus ya esté solicitado por un dispositivo.

En una palabra: esta configuración establece un retraso adicional respecto a la frecuencia de reloj *PCI*. Generalmente, el fabricante hincha un poco los valores para que la configuración funcione sin problemas con distintas tarjetas *PCI*.

PNP/PCI Configuration

En los PCs que además de tarjetas *PCI* soportan también tarjetas *ISA Plug & Play*, la página de configuración *PCI* se amplió con las configuraciones *ISA*. Los recursos del PC (canales *IRQ* y *DMA*) tienen que asignarse a las tarjetas con capacidad *Plug & Play* (*PCI*, *ISA Plug & Play*) o a las unidades *ISA* a configurar con *jumpers*.

En el más sencillo de los casos es cuestión de definir una entrada como *PnP BIOS Auto-Config* o *Ressources Controlled By* con *Enabled* o *Auto*, y el mismo *BIOS* llevará a cabo de forma automática la asignación de interrupciones, sin que uno pueda reconocer en un principio, qué *IRQ* utiliza cada unidad.

Habitualmente, esto no se muestra en el monitor hasta que no se inicia el PC. En caso de que todas las tarjetas funcionen con esta configuración, uno debería darse por más que satisfecho con ella.

```
                    ROM PCI/ISA BIOS (2A59CG01)
                      PCI CONFIGURATION SETUP
                       AWARD SOFTWARE., INC.

PnP BIOS Auto-Config: Disabled
1st Available IRQ   : 10
2nd Available IRQ   : 11
3rd Available IRQ   : 9
4th Available IRQ   : 5
PCI IDE IRQ Map To  : ISA

                                ESC: Quit      ↓→↑←:Select Item
                                F1 : Help        PU/PD/+/-:Modify
                                F5 : Old Values (Shift)F2 :Color
                                F7 : Load Setup Defaults
```

Cuando está desactivado el PnP BIOS Auto-Config, puede realizarse manualmente la asignación de interrupciones

Sin embargo, eso sólo funciona sin conflictos cuando todas las tarjetas utilizadas en el PC tienen la capacidad *Plug & Play*. En la mayoría de los casos, las interrupciones para las unidades *Plug & Play* se asignan en el orden que pueden verse en la figura siguiente si la opción *PnP BIOS Auto-Config* se define con *Disabled* y, en el paso siguiente, puede modificarse manualmente esa asignación.

```
[snd16.drv]
IOAddress=530
Interrupt=11
DMADAC=0
DMAADC=0
....................
LeftSynthVol=12336
RightSynthVol=12336
....................
[mpu401.drv]
port=330
int=9
```

A pesar del Plug & Play no debería olvidarse el archivo SYSTEM.INI, en el que posiblemente puedan encontrarse entradas de hardware, como las que realizan por razones de compatibilidad las tarjetas de sonido PCI modernas (Sound Blaster) y que en último término son las responsables de los distintos problemas con Windows

Hasta que no se mezclan tarjetas de tres tipos diferentes no surgen problemas y, por eso, muchas veces deben realizarse configuraciones de *Plug & Play* manuales. En relación con esto también debería pensar que posiblemente en el archivo *SYSTEM.INI* de *Windows* se hayan realizado asignaciones de interrupciones porque una tarjeta *ISA* (antigua) con *jumpers* puede integrarse en el sistema a través de esta entrada.

Otro ejemplo de un *PNP/PCI Configuration Setup* se muestra en la figura siguiente. Aquí las *IRQs*, e incluso los canales *DMA*, pueden dejarse para el mecanismo de configuración automática (*PCI/ISA PnP*) o denominarse tarjetas *ISA Legacy*, como estas tarjetas que deben configurarse mediante *jumpers* o bien en el programa de configuración específico del fabricante.

```
                    ROM PCI/ISA BIOS (2A69KA1A)
                     PNP, PCI & ONBOARD I/O
                      AWARD SOFTWARE, INC.

Force Update ESCD          : Disabled     Onboard USB Assigned IRQ : Enabled
PNP OS Installed           : No           Onboard FDD Controller   : Enabled
Resources Controlled By    : Manual       Onboard Serial Port 1    : 3F8/IRQ4
** IRQ DMA Assigned to PCI/ISA PnP **      Onboard Serial Port 2    : 2F8/IRQ3
  IRQ 3:N  IRQ 4:N  IRQ 5:Y  IRQ 7:N       - Onboard IR Function    : IrDA
  IRQ 9:Y  IRQ10:Y  IRQ11:Y  IRQ12:Y       - RxD TxD Active         : Hi,Lo
  IRQ14:N  IRQ15:N  DMA 0:Y  DMA 1:Y       - IR Transmission Delay  : Enabled
  DMA 3:Y  DMA 5:Y  DMA 6:Y  DMA 7:Y       Onboard Parallel Port    : 378/IRQ7
Assign IRQ For PCI VGA     : Auto         - Parallel Port Mode     : ECP+EPP
PCI IDE Card 2nd Channel   : Enabled      - ECP Mode Use DMA       : 3
PCI IDE Card IRQ Map To    : PCI-AUTO     - EPP Mode Select        : EPP1.7
- Primary IDE INT#         : A
- Secondary IDE INT#       : B
USB Keyboard Support Via   : OS
Onboard IDE-1 Controller   : Enabled
- Master Drive Mode        : Auto         ESC : Quit          ↑↓→← : Select Item
- Slave Drive Mode         : Auto         F1  : Help          PU/PD/+/- : Modify
Onboard IDE-2 Controller   : Enabled      F5  : Old Values    (Shift)F2 : Color
- Master Drive Mode        : Auto         F6  : Load BIOS  Defaults
- Slave Drive Mode         : Auto         F7  : Load Setup Defaults
```

Esta página del Setup no es particularmente clara, ya que en este lugar también se configuran los Onboard Devices. Lo importante aquí es que los recursos del PC puedan asignarse a las unidades con capacidad Plug & Play o a las tarjetas ISA convencionales

Puede darse perfectamente el caso de que en un PC no tenga suficientes canales de interrupción disponibles para las distintas unidades. No obstante, en un caso así puede ayudarle un pequeño truco: asigne el máximo número posible de canales *Legacy ISA*, con lo cual el *BIOS* tendrá que asignar automáticamente la misma *IRQ* a varias unidades *PCI*. Esto funciona sin problemas, ya que el bus *PCI* utiliza el *Level Triggering* de *IRQ*, lo cual significa que varias unidades *PCI* pueden compartir un canal de interrupción (*Shared Interrupt*).

En algunas versiones de *BIOS*, por ejemplo de la casa AMI, en el *ISA Plug & Play Setup* existe otro punto llamado *Configuration Mode* que siempre provoca confusiones. Ofrece las tres posibilidades de configuración siguientes:

- *Use ICU*: si en el PC además del las tarjetas *ISA Plug & Play* también se utilizan tarjetas *ISA* convencionales, el *BIOS* no obtiene información sobre los recursos del PC que ocupan estas tarjetas. Es por ello que se realizan las definiciones correspondientes con una *ISA Configuration Utility* (*ICU*). Si en el *Setup* está activada laopción *Use ICU*, el *BIOS* busca en el disco duro

los archivos de configuración correspondientes, que se administran con un administrador de configuración.

- *Use Setup Utility*: esta es la opción predeterminada para la configuración automática de las unidades *Plug & Play*. No se utiliza ninguna *ICU* ni un administrador de configuración. Si en el PC también se encuentran tarjetas convencionales -no declaradas explícitamente como *Plug & Play*-, esta es igualmente la configuración correcta, y la asignación de las *IRQs* necesarias se lleva a cabo de forma manual.
- *Boot With P&P OS*: este punto puede definirse como *Enabled* o *Disabled*. Si está activado, sólo se activan aquellos componentes (tarjeta gráfica, disco duro) necesarios para arrancar el sistema operativo *Plug & Play* (*Windows 9x* en adelante). Del resto de configuraciones se encarga luego el sistema operativo. Esta configuración es posible tanto con *Use Setup Utility* como con *Use ICU*.

Extended System CMOS DataRAM (ESCD)

Otro punto importante para la configuración *Plug & Play* es el punto *Force Update ESCD*. El *Extended System CMOS DataRAM* (*ESCD*) contiene la información (sobre recursos) de los dispositivos *Plug & Play* utilizados y se encuentra con un área aparte en la *EPROM Flash*, que también contiene el *BIOS* del sistema.

Normalmente esta opción debe definirse con *Disabled*, porque de lo contrario los datos *Plug & Play* se volverían a escribir automáticamente justo después del reinicio y en función del dispositivo *Plug & Play* utilizado en cada caso.

Al actualizar el área *ESCD* se tienen en cuenta tanto los parámetros definidos automáticamente de las diferentes tarjetas como los posibles parámetros definidos de frorma manual en el *Setup* del *BIOS*, así como los definidos en *Windows* para los distintos componentes de hardware en cuestión. El proceso de actualización normalmente puede observarse en el monitor a través del mensaje *Updating ESCD*.

A pesar de todo, este tipo de configuración *Plug & Play* sólo debería realizarse en casos de emergencia, cuando no es posible utilizar determinados componentes de hardware *Plug & Play* en el PC con ningún otro método, puesto que al hacerlo es posible escribir datos incorrectos en las unidades que ya funcionan (porque otra tarjeta es defectuosa), y que entonces dejarían de funcionar. Este punto, algo peligroso, existió sólo a partir de las nuevas versiones de *BIOS* y antes únicamente estaba implementado en los programas *Flash Writer* de actualización del *BIOS*.

No obstante, también pueden existir razones de peso para la ejecución de esta función. Si en un PC se constata realmente un problema con un componente *Plug & Play*, es conveniente realizar un aislamiento para poder determinar el causante del problema y no empeorar más la situación. En primer lugar se montan en el PC sólo aquellas tarjetas *Plug & Play* que son absolutamente necesarias para el arranque, y si no funcionan correctamente, se inicia una actualización del *ESCD*. A continuación se añade

la siguiente tarjeta, y se comprueba su correcto funcionamiento, se realiza una nueva actualización, y así sucesivamente, hasta que vuelve a aparecer el problema. De este modo se averigua la tarjeta incorrecta, con la que, naturalmente, ya no se puede efectuar una actualización del *ESCD*, porque de lo contrario tendríamos que volver a empezar desde el principio.

17.4.5 On Board Devices, Integrated Peripherals

Con las placas base *PCI* se convirtió en costumbre que numerosos componentes (que de lo contrario se integrarían en el sistema mediante tarjetas de expansión) ya se encuentren en la placa base. Algunos de estos *On Board Devices* y sus opciones, que se encuentran en otras páginas del *Setup* del *BIOS*, ya se han descrito en otros puntos, y dado que son más bien poco relevantes en comparación con otras configuraciones (por ejemplo, las del *Chipset Setup*), a continuación sólo hablaremos de ellas de forma breve y concisa.

Para las entradas del *Setup* relativas a los componentes *onboard* se recomienda en general activar sólo las que se utilicen realmente, ya que de lo contrario se ocuparían innecesariamente recursos (direcciones *E/S*, canales de interrupción y *DMA*), que posiblemente sean necesarios en otra parte. La posibilidad de desactivar distintas unidades también es importante porque en caso de error no se necesita una nueva placa base de inmediato, sino que es posible desactivar en el *Setup* del *BIOS* el componente defectuoso y reemplazarlo a través de una tarjeta de expansión adicional. Aunque en algunos casos, para ello es necesario colocar en la placa base el *jumper* correspondiente para su configuración.

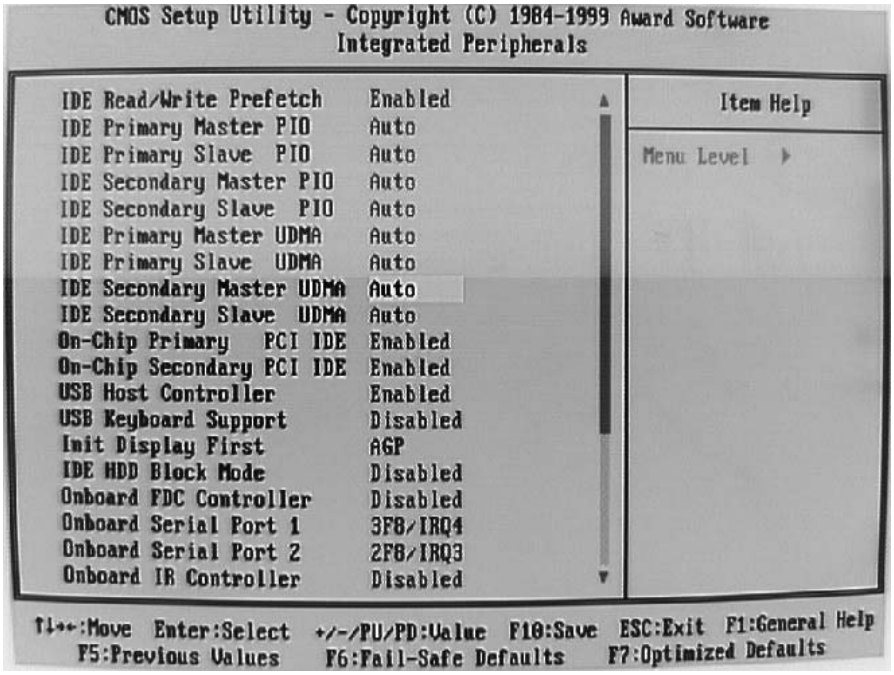
ROM PCI/ISA BIOS (2A59GA29)
INTEGRATED PERIPHERALS
AWARD SOFTWARE, INC.

IDE HDD Block Mode	: Enabled	ECP Mode Use DMA	3
IDE 32-bit Transfer Mode	: Enabled		
IDE Primary Master PIO	: Auto		
IDE Primary Slave PIO	: Auto		
IDE Secondary Master PIO	: Auto		
IDE Secondary Slave PIO	: Auto		
On-Chip Primary PCI IDE	: Enabled		
On-Chip Secondary PCI IDE	: Enabled		
PCI Slot IDE 2nd Channel	: Enabled		
USB Controller	: Disabled		
Onboard FDD Controller	: Enabled		
Onboard Serial Port 1	: 3F8/IRQ4		
Onboard Serial Port 2	: 2F8/IRQ3	Esc: Quit	↑↓→← :Select Item
UART 2 Mode	: Standard	F1 : Help	PU/PD/+/- : Modify
		F5 : Old Values (Shift) F2: Color	
Onboard Parallel Port	: 378H/IRQ7	F6 : Load BIOS Defaults	
Onboard Parallel Mode	: ECP	F7 : Load Setup Defaults	

Las configuraciones para las unidades On Board se encuentran en una página específica -como aquí- o en el Chipset Feature Setup

Configuraciones IDE

El punto *IDE Primary Master PIO* y los tres siguientes establecen el modo de funcionamiento para los, como máximo, cuatro discos duros conectados. Para este paso de la configuración tenemos los 5 modos *PIO* (*Parallel Input Output*) y el modo *UDMA*. El modo *PIO* 0, el más lento, debe utilizarse para discos duros *IDE* antiguos y el modo 4 para todos los discos *EIDE* habituales. Los más recientes también soportan *UDMA*.



En versiones del BIOS puede activarse opcionalmente para cada puerto el modo UDMA (DMA 33/66/100) para discos duros o unidades de DVD-ROM equipados convenientemente para ello. Generalmente, aquí sólo existen dos posibilidades: Auto o Disabled

En la opción *Auto* el controlador *EIDE* averigua por su cuenta el modo de funcionamiento óptimo, si más no lo intenta en cualquier caso. En caso de que conozcamos los datos de nuestro(s) disco(s) duro(s), deberíamos configurar manualmente el modo soportado para una mayor seguridad. Especialmente los discos duros *IDE* antiguos pueden señalizar al controlador modos *PIO* "imposibles", de modo que no pueda utilizarse el disco duro.

Las interfaces *EIDE* sólo están activadas cuando lo están las correspondientes entradas *Onboard Primary PCI IDE* para el primer puerto *EIDE* y *Onboard Secondary PCI IDE* para el segundo.

Habitualmente, el disco duro se encontrará como maestro en el puerto primario y una unidad de *CD-ROM/DVD-ROM* como maestro en el segundo puerto, de modo que ambas opciones de menú deberán definirse con *Enabled*.

Controladoras y puertos

Otros componentes *Onboard* habituales, y para los que pueden encontrarse entradas en el *Setup*, son los siguientes:

- *Onboard IDE Controller* (controladora de discos duros *EIDE*).
- *Onboard FDD Controller* (*Floppy Disk Drive*, controladora de la unidad de disquetes).
- *Onboard Serial Port A* (primer puerto serie): opción predeterminada COM1/3F8h.
- *Onboard Serial Port B* (segundo puerto serie): opción predeterminada COM2/2F8h.
- *Onboard Parallel Port* (puerto paralelo de impresora): opción predeterminada 378/IRQ7.

Las opciones predeterminadas indicadas representan los valores habituales en general para estos puertos y raramente se modifican. Sin embargo, para el puerto paralelo pueden observarse diferentes modos.

Onboard Parallel Mode

De forma predeterminada, el puerto paralelo o *LPT*, sirve para el control de una impresora. Las funciones de este puerto se ampliaron con la norma *IEEE1284* y en caso de que el *BIOS* del PC soporte estas ampliaciones, por regla general, en el *Setup* se encontrarán las siguientes posibilidades de configuración:

- *Normal* o *SPP* (*Centronics Mode*).
- EPP (*Extended Parallel Port*).
- *ECP* (*Enhanced Capability Mode*).
- *EPP* + *ECP* (*Extended Parallel Port* y *Enhanced Capability Mode*).

Las distintas funciones del puerto paralelo no son fáciles de descubrir para el usuario, ya que aquí han desembocado los conceptos más diversos de los fabricantes, lo cual se refleja en varias especificaciones. Es por eso que el usuario debería saber con exactitud qué modo de funcionamiento soporta el dispositivo que quiere conectar, dato que debería figurar en el manual correspondiente, para poder configurar el modo de funcionamiento apropiado en el *Setup*.

En el más fácil de los casos, este punto se define con *Normal* (*SPP*, *Standard Parallel Port*), con lo cual todas las impresoras tendrán que responder unidireccionalmente y, con lo cual, debería realizarse una primera prueba en caso de que tenga la sospecha de que el puerto pueda ser defectuoso. Esto no ocurre tan raramente como se cree, ya que enchufando o desenchufando el cable con el PC conectado pueden producirse daños eléctricos en el puerto.

17.4.6 Configuraciones del procesador o CPU

En las placas base *socket 7*, habitualmente no existen en el *BIOS* datos de configuración (frecuencia de reloj, tensión) para la *CPU*, sino solamente en las *CPUs* de Intel para el *slot 1* y para las *CPUs Celeron* con zócalo *PPGA* (370 polos). Estas *CPUs* pueden señalizar al juego de *chips* la tensión de alimentación que necesitan a través de *pins* especiales y el juego de *chips* las conecta automáticamente en consecuencia.

Dado que en las *CPUs socket 7* esta funcionalidad no suele existir, esta configuración se realiza mediante *jumpers* en la placa base. Allí debe definirse igualmente mediante puentes la frecuencia de la placa base (frecuencia del sistema, típicamente 66 o 100 MHz) y el multiplicador por el que la *CPU* debe multiplicar esa frecuencia, que se denomina *Core/Bus Ratio*.

A partir de las placas base *Super Socket 7* a veces también puede seleccionarse un modo *AUTO* para la *CPU* mediante *jumpers* en la placa base. Además, aquí posiblemente exista igualmente una función *Auto Detect* para la configuración de la tensión de la *CPU* (procesador). Sin embargo, esa función no siempre funciona correctamente, lo cual puede ser debido a que este zócalo puede incorporar un gran número de *CPUs Pentium* (compatibles). Por lo tanto, la detección automática de *CPU*, aquí mediante la lectura de la identificación de la *CPU* por parte del *BIOS*, y los parámetros a activar en cada caso dependen de la revisión del *BIOS* y lo mejor suele ser definir los datos manualmente (justamente mediante *jumpers*).

La definición de la frecuencia en las *CPUs Pentium II/III/4* y *Celeron* no es necesaria en principio, ya que algunas de estas *CPUs* internamente están conexionadas de tal forma que siempre multiplican internamente la frecuencia de la placa base, por ejemplo, por cinco, lo cual significa para una *CPU Celeron*, especificada como tipo de 333 MHz, que tiene que trabajar con la frecuencia de sistema de 66 MHz que ha predefinido el fabricante. Por consiguiente, no tiene importancia el factor de multiplicación (*Core/Bus Ratio*) predefinido aquí, puesto que la *CPU* trabaja siempre con 333 MHz, a no ser que se cambie la frecuencia del sistema a 100 MHz, con lo cual la *CPU Celeron* funciona a 500 MHz. Además, esto funciona (por lo menos con los tipos de formato *slot 1* y placas base con un juego de *chips* BX) excelentemente si los módulos de memoria no "se rompen" debido al incremento de la frecuencia del sistema o la frecuencia *PCI* y *AGP* se van fuera de sus especificaciones.

En caso de que la *CPU* esté provista internamente de una protección contra sobrefrecuencia y con un multiplicador fijo entre la frecuencia externa y la interna, la única magnitud variable que queda es la frecuencia del sistema. Sin embargo, no puede determinarse en general en qué *CPUs* y variantes de fabricación el multiplicador es fijo o no lo es.

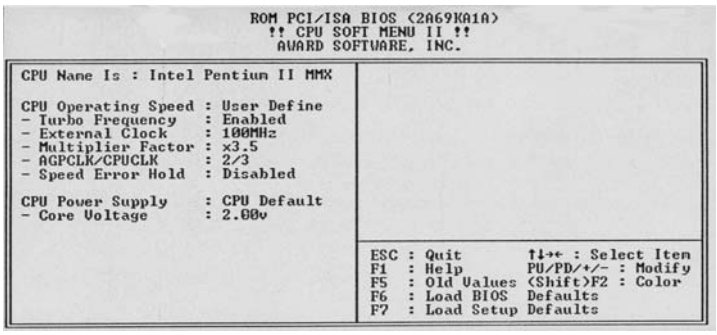
Si los parámetros de la *CPU* pueden "optimizarse" cómodamente a través del *Setup* del *BIOS*, por lo común uno más bien tiende a probar otra configuración antes que a destornillar primero el PC y buscar los *jumpers* apropiados en la placa base.

No obstante, las entradas incorrectas pueden perfectamente destruir (!) la *CPU*, de modo que estos parámetros sólo deberían modificarse si uno tiene totalmente claros los datos de la misma. Por esta razón, en algunas placas base, esos parámetros o una página específica del *Setup* del *BIOS* sólo pueden manipularse si, previamente, se ha cambiado un *jumper* (por ejemplo, *Configure Mode*) de la placa base, como ocurre en las placas con el *chipset* 440LX. Una vez finalizadas las configuraciones, el *jumper* deberá volverse a poner en la "posición normal".

En una placa base en que pueden determinarse los parámetros de la *CPU* a través del *BIOS*, existe otro peligro, ya que en caso de que se haya realizado una configuración no válida de la frecuencia, posiblemente el PC no podrá efectuar un reinicio y, entonces, ya no será posible configurar de nuevo el *Setup* del *BIOS* para poder realizar una corrección. En cambio, en una placa base *Pentium/Celeron* configurable a través de *jumpers* eso no representa ningún problema, puesto que aquí sencillamente pueden definirse los *jumpers* de nuevo. Así pues, lo importante es que después de una modificación de los datos de la *CPU* a través del *Setup* del *BIOS* no se pulse el botón *Reset* o simplemente se desconecte el PC. En lugar de eso, es preciso finalizar el *Setup* del *BIOS* correctamente y esperar un reinicio del PC. En caso de que el PC entonces no se inicie de forma correcta, normalmente puede volverse al *Setup* del *BIOS* y efectuar una nueva configuración. Pero en el (temido) caso de que ocurra que el PC no diga ni pío*f* debido a una configuración de frecuencia demasiado "optimista", sólo se puede apagar y encender el PC repetidas veces (lo normal son 3 ó 4 veces) y esperar a que el *BIOS* active automáticamente las configuraciones predeterminadas y a continuación puedan configurarse de nuevo los parámetros de *CPU* apropiados.

CPU Soft Menu (parámetros de la CPU)

A partir de las placas base *Pentium II* en la opción *Chipset Features Setup* son posibles configuraciones para la *CPU*, o bien existe una página específica para ellas: *CPU Soft Menu*. Los fabricantes recomiendan confiar en la configuración automática para los parámetros de la *CPU*, aunque eso, como hemos explicado, no siempre funciona correctamente, de modo que para la definición manual debería configurarse el punto *CPU Operating Speed* con la opción *User Define*.



Para la definición manual de los datos de funcionamiento de la CPU debemos obrar con especial precaución, ya que unas configuraciones incorrectas pueden destruir la CPU por completo

Turbo Frequency (aumento de la frecuencia)

Generalmente, la opción *Turbo Frequency* puede activarse, con lo cual el procesador funcionará con una frecuencia de aproximadamente el 2.5% más. Eso queda fuera de su especificación, pero habitualmente funciona debido al mínimo aumento de la frecuencia de operación.

External Clock o CPU Clock Frequency (frecuencia de reloj de la CPU)

Aquí se configura la frecuencia de sistema (externa) de habitualmente 100, 200, 400 MHz. Valores como 122 MHz, 133 MHz o incluso 166 MHz no corresponden al estándar, y quien piense que un mínimo aumento de 100 MHz a 122 MHz funcionará, a menudo está equivocado, ya que el aumento de frecuencia puede surtir efecto en la frecuencia *PCI*, *AGP* e *ISA*.

Multiplier Factor o CPU Clock Ratio (multiplicador de frecuencia)

Es el factor por el que se multiplica la frecuencia de reloj del sistema para obtener la frecuencia especificada para la *CPU* utilizada. Por ejemplo, para una *CPU Pentium II* de 300 MHz con una frecuencia del sistema de 66 MHz debe definirse un multiplicador de 4.5 o preferiblemente de 3 cuando la frecuencia de sistema configurada es de 100 MHz si los módulos de memoria corresponden a *DIMMs PC-100*.

En principio, para la *CPU* en sí es indiferente con qué combinación de frecuencia de sistema y factor de multiplicación se obtenga la frecuencia necesaria. Eso sólo afecta a la memoria y posiblemente también a las frecuencias de los diferentes sistemas de bus. Según la fecha de fabricación de la placa base o de la versión de *BIOS* pueden encontrarse diferentes factores en el *Setup* del *BIOS*, y aunque el manual de la placa base no indique explícitamente la *CPU* que debe utilizarse, puede emplearse mediante la correspondiente combinación de frecuencia de sistema y factor de multiplicación. No obstante, lo óptimo es cuando puede elegirse una frecuencia de sistema lo más alta posible. Mediante una actualización del *BIOS*, a veces, es posible incorporar posteriormente un multiplicador que no existe en el *Setup* del *BIOS* pero que es necesario.

AGPCLK/CPUCLK (relación frecuencia AGP con frecuencia CPU)

Esta opción de menú no existe en todas las placas base, sobretodo en las más antiguas. Permite definir la frecuencia *AGP* en dependencia de la frecuencia externa de la *CPU*, es decir, la frecuencia de sistema o frecuencia *FSB* (*Front Size Bus*), como también se denomina en ocasiones. Con una frecuencia de sistema de 100 MHz y la configuración *AGPCLK/CPUCLK: 2/3* que se muestra en la figura anterior, la frecuencia *AGP* corresponde a los 66 MHz especificados.

Speed Error Halt (paro en caso de configuración errónea)

Esta configuración con la opción *Enabled* se encarga de interrumpir el proceso de inicio cuando el BIOS constata que la configuración de frecuencia realizada para la *CPU*

no coincide con la que tiene especificada. Como ya hemos visto, uno no puede confiar necesariamente en que el *BIOS* detecte de forma automática el tipo utilizado correctamente. Con la opción *Disabled* se aceptan las configuraciones definidas sin ninguna verificación, lo cual también puede provocar que el PC ya no pueda iniciarse porque el procesador no se las arregla con la definición de la frecuencia.

CPU Power Supply o CPU-Voltage (tensión CPU)

La detección automática de la tensión de alimentación que necesita la *CPU* funciona de forma mucho más fiable que la detección de la frecuencia, porque las *CPUs* poseen para ello los *Voltage Identification Pins* (*VID*), de modo que, por regla general, uno puede confiar en la configuración predeterminada para la *CPU* y conservar esa configuración. Quien sepa más que el mecanismo automático y eleve la tensión, aquí tiene más probabilidades de arriesgarse a destruir la *CPU* que si ha elegido una frecuencia demasiado alta. Las tensiones que están disponibles dependen de nuevo del tipo de placa base y de *BIOS*.

17.5 El proceso de actualización del BIOS

En todos los PCs de hoy en día es habitual que el *BIOS* se encuentre en un componente de memoria *Flash*, que a diferencia de una *ROM* normal puede volverse a escribir mediante un software especial, que habitualmente se suministra en un disquete con la placa base. La ventaja de esto es que puede llevarse a cabo una (necesaria) actualización del contenido del *BIOS* de modo relativamente fácil, lo cual era muy molesto anteriormente, puesto que las *ROMs* grabadas (si existían) generalmente sólo podían obtenerse después de grandes esfuerzos del fabricante del *BIOS*.

Sin embargo, desde que se utiliza memoria *Flash* para el *BIOS* se puede constatar que desde ese preciso momento en adelante existe una oleada de actualizaciones del *BIOS*, lo cual también es debido a que las versiones del *BIOS* tienen que desarrollarse bajo una enorme presión de tiempo, con lo que se deslizan errores que tienen que solventarse (rápidamente) mediante las versiones siguientes. Esta situación también es el motivo más frecuente por el que se recomienda una actualización. En las páginas de *Internet* de los fabricantes de placas base, generalmente puede encontrarse para cada versión de *BIOS* un archivo que describe qué es lo que se ha cambiado en concreto.

Uno debería guardarse de actualizar el software del *BIOS* sólo por diversión, ya que puede pasar perfectamente que, después de hacerlo, nada funcione porque el PC sin el *BIOS* correcto está muerto. Así pues, en caso de estar satisfecho con el funcionamiento de la versión de *BIOS* existente, no hay motivo para preocuparse más por una actualización del *BIOS*. A fin de cuentas, nadie puede garantizarle de que una nueva versión del *BIOS* funcionará mejor que la anterior.

La regrabación del *BIOS* actualmente parece ser ya tan natural como una actualización de controladores para *Windows*, y con demasiada frecuencia los errores del PC se atri-

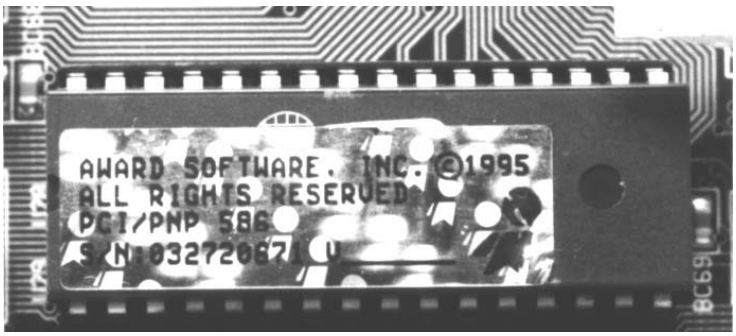
buyen al *BIOS* de forma errónea. Algunos fabricantes ya tienen en cuenta la circunstancia de que debido a numerosos *chips* de *BIOS* "destruidos" les solicitarán su ayuda, y equipan las placas base con dos *chips* de *BIOS* para que los técnicos en ajustar el PC puedan realizar otro intento. Por lo tanto, sólo se aconseja una actualización del *BIOS* cuando existe una versión más reciente del *BIOS* del fabricante, en la que respecto a la versión incorporada se han ampliado funciones o se han arreglado algunos errores de la versión anterior. La forma más fácil de enterarse de si existe una nueva versión es consultar la página de *Internet* del fabricante de la placa base (no del fabricante del *BIOS*), donde generalmente podremos bajar la versión que necesitemos.

Además de la mera corrección de errores existen otros motivos a favor de una actualización del *BIOS*. Por ejemplo, si con la nueva versión uno dispone de funciones que el *BIOS* aún no soporta, como el modo *Ultra-DMA* para discos duros *EIDE*. Pero la versión del *BIOS* tiene que estar elaborada explícitamente para la placa base existente *(chipset*), y bajo ningún concepto puede utilizarse una versión presuntamente similar sin ser la adecuada en cada caso.

17.5.1 Chips de BIOS

En todo PC se encuentran una o varias *ROMs* (hasta cuatro en algunas placas base antiguas), que contienen el software necesario para la comunicación del sistema operativo con el hardware: el *BIOS*. El contenido de las memorias *ROM* no se puede modificar. Si alguna vez se requiere una nueva versión, que por ejemplo dé soporte para nuevas unidades, es preciso cambiar los componentes. Si se tienen dos *ROMs*, generalmente una está marcada con *EVEN* o *LOW* y la otra con *ODD* o *HIGH*.

La *ROM EVEN* se controla con las direcciones pares (las inferiores) y la *ROM ODD* con las direcciones impares del procesador. El software de la *ROM*, que también se denomina *Firmware*, en las placas base modernas puede bajarse a la *ROM* con ayuda de la unidad de disquetes si fuera necesaria la instalación de una nueva versión. Las placas base modernas utilizan habitualmente un único componente de memoria para almacenar el contenido del *BIOS*.



Los PCs actuales poseen un único componente de memoria Flash en el que encontramos el software adecuado del BIOS

En este caso, en la placa base se encuentra una *EEPROM* (*Electrically Erasable Programmable Read Only Memory*) o una memoria *Flash*; ambos tipos pueden borrarse eléctricamente. En contrapartida, las *ROMs* normales no pueden borrarse, ya que el programa prácticamente se ha marcado en el *chip*. En cambio, las *EEPROMs* sólo pueden borrarse mediante la luz ultravioleta de un aparato especial para borrar.

Estos componentes tienen una ventana que suele estar tapada por la etiqueta del *BIOS*, e igualmente se utilizan como componentes de memoria para el *BIOS*. No obstante, eso sólo es así en placas base antiguas, ya que las memorias *Flash* contienen además información *Plug & Play* que el sistema actualiza automáticamente, lo cual no es posible cuando se utilizan *PROMs* o *EEPROMs*.

Por este motivo, en principio el contenido de una memoria *Flash* también puede programarse en un *EEPROM* a través de un aparato de programación especial, pero a continuación no funcionará correctamente el mecanismo *Plug & Play*.

Si un *BIOS* se descubre como versión *Plug & Play* (durante la inicialización del PC), con toda seguridad puede darse por sentado que en la placa base se utiliza una memoria *Flash* para el *BIOS*.

Las memorias *Flash* son un perfeccionamiento de las *EEPROMs* y, por tanto, en principio también utilizan la misma tecnología. La lógica de decodificación adicional, con la que puede modificarse el contenido por bloques y no siempre por completo (como por ejemplo, en las *EEPROMs*), un automatismo de estado (*State Machine*) para la programación y bombas de carga para la creación de tensión de programación son elementos ya integrados en el propio componente de en una *EEPROM* y totalmente en una memoria *Flash*.

Es por ello que las *EEPROMs* necesitan (según el tipo) un poco de lógica adicional, situada en la placa base, mientras que las memorias *Flash* se las arreglan sin ella. Por esta razón, el hecho de que en la placa base se encuentre un componente de memoria *EEPROM* o *Flash*, en principio no tiene una mayor importancia para realizar la actualización del *BIOS*.

Sin embargo, sí que es importante (además del soporte por parte de un programa de escritura apropiado) la tensión de programación, que puede ser de 12 o 5 voltios, dependiendo del tipo de componente en cuestión.

17.5.2 La programación

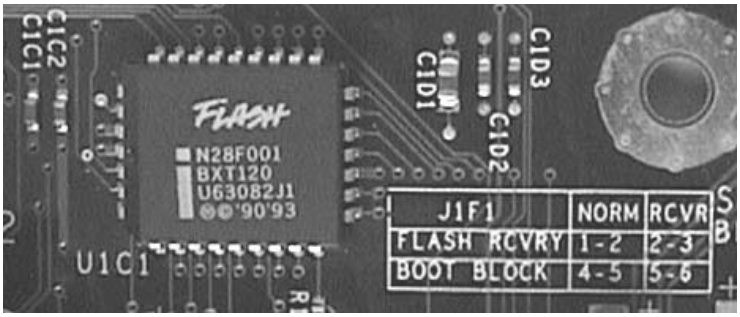
Un programa que efectúa la programación de la memoria *Flash* es, por ejemplo, el programa *AWDFlash.exe* de la casa Award o bien un *Flash Memory Writer* de la casa ASUS, si se trata de una placa base de esta marca.

Para placas base que trabajan con un *BIOS* AMI, el programa de escritura correspondiente se llama *AMIFlash.exe*, y según el fabricante del *BIOS* o de la placa base estos programas pueden tener distintos nombres.

Normalmente, en el disquete de utilidades de la placa base se encuentran, por ejemplo, los siguientes archivos:

- *PFLASH.EXE*: se trata del programa para escribir la memoria *Flash*.
- *README.TXT* o *FILELIST.TXT*: se trata de un archivo de texto con instrucciones e información adicional sobre el software.
- *TX5Ixxxx*: el archivo BIOS de la versión del BIOS existente. También puede darse el caso de que este archivo todavía esté por crear, como veremos más adelante.

En la mayoría de los casos, el proceso de programación se realiza sin problemas, cosa que no siempre puede decirse de algunas versiones antiguas (generalmente de los primeros tiempos de la programación de las memorias *Flash*, aproximadamente a partir de 1993). No obstante, como hemos mencionado, siempre tienen que existir buenas razones para una nueva programación, ya que también es posible fracasar totalmente y, normalmente, no se da una nueva posibilidad de programar. Pocas placas base poseen la función *BIOS-Recovery*, que permite recuperar un *BIOS* que funciona después de fracasar en un intento de programación.



La PROM Flash contiene el BIOS del sistema y también la inscripción para el puente (muy útil) que desencadena la función BIOS-Recovery (Flash RCVRY) en caso de que no haya funcionado la actualización del BIOS realizada con anterioridad

Antes de la programación deberíamos buscar un puente en la placa base que (en caso de existir) tiene dos posiciones, que se llaman *Enable Programming* y *Disable Programming/Normal Read* o algo similar. Esta última posición está prevista para el funcionamiento normal de la placa base y, por eso, tiene que definirse el puente en la posición *Enable Programming* para poder efectuar la programación. Una vez resuelto este proceso no hay que olvidar volver a colocar el puente en la posición normal después de la programación. El manual de la placa base o un archivo *README.TXT* deberían contener información sobre la correspondiente posición del puente.

En muchas placas base no existe este puente, lo cual facilita la cuestión (no es preciso desmontar el PC para buscar ningún puente), pero lamentablemente eso también significa que el *BIOS Flash* puede escribirse en cualquier momento, y en la actualidad existen virus que se dedican precisamente a eso, con lo cual el PC queda inservible.

Se inicia el PC en *Real Mode* (*DOS*), es decir, sin cargar ningún administrador de memoria (*HIMEM.SYS*, *EMM386*) y en ningún caso en un modo *Windows*. Es preferible iniciar el PC desde un disquete que no contenga ningún archivo *CONFIG.SYS* y en cuyo *AUTOEXEC.BAT* únicamente se cargue el controlador de teclado. Por lo demás, el disquete debería contener el programa de escritura *Flash* y el nuevo archivo *BIOS* que, si se ha obtenido en *Internet*, posiblemente tenga que descomprimirse. Un archivo de *BIOS* completo suele tener un tamaño de 128 KBytes (131.072 Bytes), por lo que debe caber sin problemas en un disquete. Sin embargo, en caso de que el archivo de *BIOS* sea más pequeño, algo le faltará, por lo que debería comprobarse otra vez si realmente se ha obtenido el archivo (binario) correcto y si éste ya se encuentra descomprimido.

Los programas actuales de escritura *Flash* se niegan automáticamente a trabajar y emiten el correspondiente mensaje de error en caso de que durante el inicio se cargue algún controlador que pudiera imposibilitar la correcta programación. Esto es algo que los programas de escritura *Flash* más antiguos no siempre hacen, por lo que entonces la memoria *Flash* no puede programarse correctamente y a continuación el PC sólo presenta una imagen negra en el monitor.

Para poder iniciar desde el disquete posiblemente deba modificarse previamente en el *Setup* del *BIOS* el arranque desde la unidad *A:* o el orden de arranque (*A:, C:*) establecido. A continuación se inicia el programa de escritura (*PFLASH.EXE* o similar) desde el disquete, con lo cual se presenta, por ejemplo, la siguiente imagen.

```
                    ASUSTek PNP BIOS
                FLASH MEMORY WRITER V1.5
        Copyright (c) 1995, ASUSTek Computer Inc.

Flash Type - SST 29EE010
Current BIOS Revision: #401A0-0104
Choose one of the following:

1. Save Current BIOS to File
2. Update BIOS Main Block From File
3. Advanced Features

Enter Choice: [1]

Press ESC To Exit
```

La pantalla de un programa de escritura Flash para la actualización del contenido del BIOS

En caso de que en la entrada *Flash Type* aparezca la palabra *unknown,* significa que en la placa base no hay instalada ninguna memoria *Flash* o que la de tipo similar que hay instalada no puede cooperar con el programa de escritura *Flash* existente. En *Current BIOS Revision* debería encontrarse igualmente una palabra significativa en lugar de *#FFFF* o algo similar. Esto también indica que uno debería salir rápidamente del programa pulsando la tecla <Esc>, ya que no puede realizase ninguna programación. Por lo tanto, una pantalla como la de la figura anterior es el primer paso para realizar la nueva programación del *BIOS* con éxito. Pero antes de llevarla a cabo deberíamos escribir en un archivo la versión existente en el componente de memoria (opción de menú 1), para lo cual se utiliza automáticamente el directorio actual del disquete. En caso de que la nueva versión no trabajara del modo deseado, entonces podría recuperarse la anterior. Como nombre para el archivo lo más razonable es indicar el nombre de la versión del *BIOS* en cuestión (por ejemplo, *TX5I0104.AWD*).

En principio, también podrían copiarse en el disco duro todos los archivos necesarios, pero puede ocurrir que si fracasa la programación ya no sea posible acceder a esos datos. Naturalmente, eso también puede suceder en la unidad de disquete, aunque es menos probable. Además, en el peor de los casos uno siempre sigue teniendo los datos a punto en el disquete. Con el dispositivo de programación correspondiente podría programarse entonces la *PROM Flash* con los datos del disquete.

La programación del componente *Flash* se desencadena a través de la opción *Update BIOS Main Block From File*, y el archivo a programar tiene que encontrarse en el directorio actual del disquete. Después de elegir esta función se presenta una pregunta acerca del nombre del archivo de *BIOS*, que deberá indicarse a continuación (por ejemplo, *TX5I0105.AWD*). Tras pulsar la tecla <Entrar> se actualiza el *BIOS*, y en caso de que no se presente ningún mensaje de error, ya habrá pasado lo peor.

Pero si se presenta, el peligro está a la vuelta de la esquina, ahora el PC no puede reiniciarse ni desconectarse de ningún modo. En lugar de eso, debe realizar otro intento de programación. Si tampoco tiene éxito, debería realizar otro intento con la versión de *BIOS* (antigua) guardada en el disquete. Si eso también falla, sólo queda una posibilidad: que el puente (*Enable Programming*) no se encuentra en la posición correcta. Sin embargo, como el PC no puede desconectarse, es preciso colocarlo en la posición correcta con el PC en marcha, para lo cual, naturalmente, se necesitan las máximas precauciones. Si todo eso no da ningún fruto, ahora es preciso desconectar el PC, y posiblemente se habrá creado un problema (que, por cierto, también puede darse cuando la actualización parece haber funcionado bien): el PC se niega a trabajar. Aún tendrá suerte dentro de la desgracia, si la placa base dispone del modo *Recovery*.

No obstante, también puede darse el siguiente caso, no tan trágico, en el que la programación se interrumpe (al principio) con el siguiente mensaje:

```
Boot Block of New BIOS is different from old one !!!
Please Use ´Advanced Features´ to Flash whole BIOS
```

Lo que hay que hacer ahora se lo explicamos en el texto siguiente, pero primero realizaremos unas observaciones para intentar comprenderlo mejor: una *PROM Flash* está dividida internamente en bloques, que según el fabricante pueden presentar un tamaño diferente. Para un *BIOS* uno se aprovecha de esta circunstancia y también divide el software del *BIOS* en bloques.

En la tabla siguiente mostramos el aspecto que puede tener esto, por ejemplo, con un tipo estándar como el 28F001BX-T de la casa Intel.

Direcciones/Hex	Tamaño	Aplicación
FE000-FFFFF	8 KBytes	Bloque de arranque
FD000-FDFFF	4 KBytes	Área de memoria Plug & Play, ESCD
EC000-FCFFF	4 KBytes	Logotipo OEM para pantalla
E0000-FBFFF	112 KBytes	BIOS del sistema

La asignación típica de una PROM Flash (128 KBytes x 8 bits)

El bloque de arranque contiene (como su nombre indica) información para el proceso de inicio del PC, que se coloca por separado del software de *BIOS* propiamente dicho en la *PROM Flash*.

Tanto el bloque de arranque como el resto de rutinas del *BIOS* tienen que concordar entre sí (número de versión), y dado el gran número de versiones posibles puede darse el caso de que también tenga que actualizarse el bloque de arranque, como puede deducirse del mensaje de error anterior (que también puede presentarse con el mensaje *Boot Block Error* o algo parecido).

Para este caso, un programa de escritura *Flash* moderno contiene una opción de menú especial que, generalmente, se encuentra en *Advanced Features*, y que después de ser activada ofrece las opciones que se muestran en la figura siguiente. No obstante, en caso de que el programa de escritura *Flash* utilizado no ofrezca esta opción, en principio habrá llegado al final de la supuesta actualización del *BIOS*, y deberá salir del programa con ayuda de la tecla <Esc>.

Sólo si el fabricante (de la placa base o del *BIOS*) puede ofrecer en su servidor de *Internet* un programa de escritura más reciente y apropiado para la placa base, es aconsejable un nuevo intento de programación posterior. De lo contrario, es preferible abstenerse de ello.

Eligiendo el punto 2 se programa de nuevo el *BIOS* junto con el bloque de arranque completo. Para ello deben tenerse en cuenta las mismas cosas que hemos descrito arriba para la actualización del *BIOS* sin el bloque de arranque. Esta actualización también hace que se borren las *Extended System CMOS DataRAMs* (*ESCD*), que contienen la información (sobre los recursos) de los dispositivos *Plug & Play* utilizados en el equipo.

```
                    Advanced Features


Flash Type - SST 29EE010
Current BIOS Revision: #401A0-0104
Choose one of the following:


1. Clear PNP ESCD Parameter Block
2. Update BIOS Including Boot Block and ESCD
Enter Choice: [2]


Press ESC To Exit
```

Las Advanced Features de un programa de escritura Flash permiten borrar los parámetros Plug & Play actuales y realizar una completa actualización del BIOS

Este bloque de parámetros también puede eliminarse por separado con la ayuda de la primera opción de menú. La aplicación de esta función, que en el fondo no tiene nada que ver con una actualización del *BIOS* o una nueva programación, ha demostrado ser sumamente útil cuando el PC por alguna que otra razón no se las arregla con determinados dispositivos *Plug & Play* y, por lo tanto, no puede iniciarse correctamente. Esta área *CMOS-RAM* extendida no se encuentra en la *CMOS-RAM* habitual (combinada con el *chip* de fecha y hora), sino en la memoria *Flash* del *BIOS*, y esta área se borra al seleccionar la opción mencionada.

Los parámetros *ESCD*, que se encuentran en esta área extendida, se vuelven a escribir automáticamente a continuación con el reinicio (en función de los distintos dispositivos *Plug & Play* utilizados en cada caso). Generalmente, el proceso de la actualización puede observarse en el monitor con el mensaje *Updating ESCD*.

La nueva programación del componente *BIOS* se representa de forma diferente, en función del fabricante del programa de escritura *Flash*. A menudo se indica el proceso en forma de una barra de progreso que finaliza con el mensaje *Programming Flash Memory - OK*. Pulsando la tecla <Esc> se termina el programa de escritura, y tras reiniciar el PC debería estar activado el nuevo *BIOS*.

El reinicio que debe realizarse a continuación debe efectuarse como arranque en frío, es decir, desconectando y volviendo a conectar el PC. A diferencia del arranque en caliente (combinación de teclas <Ctrl> + <Alt> + <Supr>), un *Reset* (arranque en frío) completo del PC siempre provoca una reinicialización completa del sistema.

Tras aparecer el mensaje del *BIOS* (nuevo) primero se ejecutará el *Setup* del *BIOS* y, para ello, se activarán los valores predeterminados con *Load Setup Defaults* (por ejemplo, con la tecla <Retroceso> o <Supr>). Este proceso evita que se interpreten mal

las entradas de la antigua versión del *BIOS* y también que se activen primero todas las configuraciones básicas. Si hasta aquí todo es correcto se desconectará el PC y se volverá a colocar el puente (*Disable Programming/Normal Read*) en su posición normal. Acto seguido se realizará un arranque completo del PC, y entonces sí que se podrá pasar a reemplazar las distintas entradas del *Setup* del *BIOS*, es decir, los *Setup Defaults*.

Puede ocurrir perfectamente que para una única placa base existan versiones de *BIOS* de diferentes fabricantes, por ejemplo de las casas Award y AMI. En ocasiones es conveniente realizar un cambio, para lo cual deberá efectuarse previamente un control de las características correspondientes, de forma habitual mediante un archivo *README* o como texto en el servidor de *Internet* de la marca en cuestión.

Resumen del procedimiento para la actualización del BIOS

- Pensar si es realmente conveniente una actualización.
- Comprobar el tipo de *Flash* o de *EEPROM* en la placa base.
- Consultar en el manual qué tipos pueden utilizarse.
- Conseguir una nueva versión de *BIOS* y, eventualmente, un programa de escritura *Flash*.
- Definir el posible puente para la tensión de programación correcta.
- Comprobar si la placa base o el *BIOS* soportan el *Recovery Mode*, lo cual ofrece cierta seguridad en posteriores problemas durante la programación.
- Crear un disquete de inicio con todos los archivos necesarios.
- Modificar el orden de arranque en el *Setup* del *BIOS* para iniciar el equipo desde la unidad de disquetes.
- Si existe, colocar el puente de la placa base en la posición de *Enable Programming*.
- Iniciar el PC desde el disquete de inicio y activar el programa de escritura *Flash*.
- Proseguir sólo si se detectan correctamente tanto el componente de *BIOS* que se encuentra en el PC como la versión de *BIOS*.
- Copiar el contenido actual del *BIOS* en un archivo.
- Iniciar la nueva programación, para lo cual posiblemente tengan que activarse las *Advanced Features* (*Update BIOS Including Boot Block*).
- Finalizar el programa de escritura *Flash* y realizar un arranque en frío del PC con una pulsación del botón *Reset*.
- Activar el *Setup* del *BIOS* y elegir la opción *Load Setup Defaults*.
- Si existe, volver a colocar el puente en la posición normal.
- Reiniciar el PC y realizar las configuraciones optimizadas para el equipo en el *Setup* del *BIOS*.

18. Sistemas operativos

Antes de entrar en materia con el estudio de los sistemas operativos, vamos a realizar un pequeño repaso sobre algunas de las nociones básicas de hardware que debemos conocer para entender mejor la funcionalidad de los sistemas operativos.

18.1 Nociones básicas de hardware

De una manera bastante general podemos ver que todo PC está compuesto por 3 componentes principales:

- **El procesador**: también se le puede conocer por el nombre de *Unidad Central de Procesamiento* o como *CPU* del inglés *Central Process Unit*. Es el encargado de la coordinación de todas las acciones del equipo y del procesamiento de los datos.
- **Memoria principal**: es el lugar donde se almacenan todos los datos con los que trabaja el procesador, además de los programas que se encuentran en ejecución.
- **Unidades de Entrada/Salida (E/S)**: son los encargados de suministrar y recibir los datos de la memoria o del procesador, desde o hacia el entorno exterior. Estos dispositivos se pueden comunicar entre ellos gracias a la ayuda de los sistemas de interconexión existentes en la placa base.

18.1.1 Los registros del procesador

Aunque casi siempre se habla del procesador como una estructura cerrada, en realidad está compuesto de diferentes módulos. Uno de los módulos principales son los registros. Éstos cumplen la función de almacenamiento de los datos aunque de una manera mucho más rápida que si se usase la memoria *RAM*. Los registros del procesador se dividen en dos grandes bloques según su función:

- **Registros de control y estado**: son de uso casi exclusivo del procesador para su funcionamiento interno (control de las operaciones) y por algunas rutinas especiales de los sistemas operativos (para el control de la ejecución de los distintos programas).

- **Registros de usuario**: estos registros pueden ser utilizados por los programas de los usuarios que necesiten la utilización de pequeños espacios de almacenamiento de gran velocidad. Sin embargo, es necesario notar que no todos los lenguajes de programación permiten la creación de programas que puedan interactuar con estos registros, tan sólo algunos como el famoso lenguaje *C*.

18.1.2 La ejecución de las instrucciones

Todo programa es una sucesión de instrucciones que le indican al procesador la acción que debe realizar, pero éstas no pasan directamente del programa al procesador, antes de esto se llevan a cabo unos pasos intermedios.

Al ejecutar un programa éste es cargado en la memoria principal, la cual le pasa una solicitud al procesador por cada instrucción almacenada y éste se encargará de su posterior ejecución y de una nueva petición a la memoria para la instrucción siguiente. Aunque pueda parecer un proceso lento, en realidad, es ejecutado en fracciones de segundo y por ello nos parece que se lleva a cabo de forma instantánea. Esta es una visión simplificada del proceso, ya que conlleva una serie de pasos más en los cuales juegan un papel trascendental algunos registros del procesador.

18.1.3 Las interrupciones

La comunicación entre el procesador y la memoria principal se realiza a gran velocidad, sin embargo, la mayoría de programas también interactúan con alguno de los tipos de dispositivo de *E/S*. Durante este proceso, la ejecución de los programas se ve interrumpida ya que la comunicación con dichos dispositivos es significativamente más lenta. Incluso en algunas ocasiones, esta operación puede ser todavía más lenta ya que puede necesitar de la intervención humana.

Algunos sistemas operativos ejecutan todos los trabajos en forma de lotes (*batch*), es decir que no pueden ejecutar otro trabajo hasta que el programa en ejecución no termine su trabajo Para solucionar este problema la mayoría de sistemas operativos modernos utilizan el concepto de interrupción.

Este concepto se basa en el trabajo compartido, es decir que el procesador cede el control de la *E/S* a un módulo que se encarga de ejecutar este tipo de operaciones y esperar a que se completen, y cuando esto suceda avisa al procesador (que mientras tanto se encuentra ejecutando otras instrucciones) que puede continuar con las operaciones siguientes y que quedaron pendientes por la petición de *E/S*.

18.2 Las funciones de un sistema operativo

En términos generales, podemos definir un sistema operativo como el componente de software encargado de la interrelación de los programas con los componentes del equipo y del equipo con el usuario.

Esta definición es apropiada para ver la importancia del sistema operativo. Sin embargo, es una visión bastante simplificada de las funciones de un sistema operativo, por lo cual es necesario ayudar un poco más a entender para qué sirve y cómo se utiliza un sistema operativo.

Una de las principales funciones del sistema operativo es la de servir de máquina virtual, esta funcionalidad es la que permite a los programadores abstraerse de la capa de hardware y utilizar una serie de llamadas al sistema operativo para que éste sea el encargado de trabajar a bajo nivel con los componentes físicos. Si no tuviese esta funcionalidad, el programador debería escribir el código de programa que permita usar estas funciones de forma individual para cada dispositivo al que se desee dar el soporte adecuado. Es por este motivo que se dice que el sistema operativo es una interfaz entre las aplicaciones y el hardware del equipo.

La otra gran función de un sistema operativo es la de controlar el acceso y la utilización de los recursos del sistema. Además, también controla quién hace utiliza estos recursos y en que momento; sin esta función del sistema operativo, todos los procesos podrían enviar al mismo tiempo sus peticiones de escritura/lectura a un disco duro y, por consiguiente, no se podría garantizar que se completase con éxito ninguna de las peticiones ya que no existiría forma de controlar en que momento se encuentra ocupado ejecutando la orden encomendada por uno de los procesos.

Por los tanto, un sistema operativo debe conocer cuando un recurso está siendo utilizado y en qué momento está libre y, en función de esto, permitir o no su uso.

18.3 Conceptos básicos

Antes de continuar explorando en el mundo de los sistemas operativos es necesario hacer una pausa para hablar sobre algunos conceptos, o términos, que se han mencionado en las secciones anteriores y que debido a su importancia continuaremos viendo.

18.3.1 Los procesos

La gran mayoría de sistemas operativos modernos están basados fundamentalmente en este concepto. Un proceso es la forma en qué un sistema operativo trata a un programa durante su ejecución. Como ya hemos mencionado, para que el procesador pueda trabajar con un programa es necesaria un poco de información adicional (como los registros de operación en el procesador, los datos de entrada del programa, etc) que acompañe al código objeto (o ejecutable). Por los tanto, cuando decimos que un proceso es la forma en qué el sistema trata a todo programa durante su ejecución, estamos haciendo referencia a todo este conjunto.

Al arrancar, el sistema operativo genera algunos procesos principales que serán los encargados de crear los demás procesos, como, a su vez, cualquier proceso puede

crear otros procesos (denominados procesos hijo). Entonces tenemos una estructura jerárquica en forma de árbol donde todos los procesos (excepto los iniciales del sistema operativo) tienen un proceso padre y, a su vez, tienen la capacidad de tener uno o más procesos hijo. A este conjunto formado por un proceso padre y todos sus procesos hijo se le denomina familia de procesos.

La siguiente figura nos muestra un ejemplo real de un árbol de procesos (a través del comando *pstree*) en un servidor *Debian GNU/Linux*. Como podemos observar, todos los procesos tienen un proceso padre común (*init*) y cada proceso tiene o no procesos hijo, que a su vez también pueden tener otros procesos hijo.

```
$ pstree
init-+-atd
     |-3*[automount]
     |-bdflush
     |-cron
  [...]
     |-usb-storage-1
     |-usb-storage-2
     |-10*[xterm---zsh]
     |-xterm---zsh---mutt
     |-2*[xterm---su---zsh]
     |-xterm---zsh---ssh
     |-xterm---zsh---pstree
     |-ypbind---ypbind---2*[ypbind]
     `-zsh---startx---xinit4-+-X
                             `-ctwm-+-xclock
                                    |-xload
                                    `-xosview.bin
```

Un árbol de procesos en un sistema GNU/Linux

En esta introducción a los procesos es necesario reseñar otros temas de vital importancia en relación con los procesos:

- Multiprogramación.
- Multiprocesamiento.
- Procesamiento distribuido.

Multiprogramación

La multiprogramación permite administrar varios procesos en un único procesador, alternando su ejecución a través de controles de tiempo, asignando límites de tiempo de ejecución a cada proceso y alternándolos a medida que se alcanza este límite.

Multiprocesamiento

Este es otro recurso dentro de la administración de los procesos. Sin embargo, y a diferencia de la multiprogramación, este recurso se basa en la distribución de los procesos en múltiples procesadores.

Procesamiento distribuido

En este caso, la administración de los procesos se complica un poco más al tener que distribuir la carga de procesamiento entre diferentes procesadores, pero con el añadido de que éstos se encuentran en eqipos diferentes. La esencia del procesamiento distribuido es manipular todos los recursos de que dispone el conjunto distribuido como si se tratase de una única máquina.

18.3.2 Los archivos

Otro componente de un sistema operativo es el sistema de archivos. Todo proceso necesita para su ejecución que los datos que va a utilizar se encuentren en algún sitio; lo que realiza el sistema de archivos es permitir que esta información se organice de una manera lógica y sencilla, tal y como hemos visto en el capítulo de los discos duros.

Todo sistema operativo debe ofrecer el soporte adecuado (en forma de llamadas al sistema o instrucciones) para que el sistema de archivos funcione de forma adecuada. Entre las principales herramientas para este soporte encontramos las siguientes:

- Creación y destrucción de archivos.
- Apertura y cierre de los archivos.
- Lectura y escritura.

Adicionalmente, la mayoría de sistemas operativos permiten tener cierto control sobre los archivos pudiéndolos asignar distintas propiedades adicionales. Un archivo es una colección de Bytes relacionados bajo un único nombre. Los archivos también se encuentran organizados bajo una estructura que los relaciona lógicamente, denominada directorio o carpetas.

Debido a esto podemos ver observar que el sistema de archivos también es una estructura jerárquica, que tiene un inicio en el directorio raíz (algunos sistemas operativos como los derivados de *MS-DOS* tienen un directorio raíz en cada partición).

18.3.3 Las llamadas

Las llamadas al sistema son unas instrucciones especiales que utiliza el mismo sistema para la comunicación entre los programas y el sistema operativo. Las llamadas están estrechamente relacionadas con las funciones de las librerías estándar de cada sistema operativo, ya que para cada una de las funciones existe su correspondiente llamada al sistema.

18.3.4 El núcleo (Kernel)

El *kernel* del sistema operativo es la base fundamental del mismo, es el encargado de toda la comunicación entre el hardware y el software, así como de la administración del mismo. La mayoría de las características y conceptos que estamos tratando hacen referencia, por lo general, al *kernel* del sistema operativo. En realidad, el *kernel* es un componente más del conjunto de aplicaciones que forman un sistema operativo.

18.3.5 El intérprete de comandos (shell)

Es el componente del sistema que permite introducir las órdenes que debe ejecutar el equipo. El *shell* esconde los detalles internos del *kernel* sobre el que se está ejecutando. El *shell* también es conocido como línea de comandos.

18.4 Tipos de sistemas operativos

Siguiendo la tendencia de todo lo relacionado con la informática, los sistemas operativos también son susceptibles de ser clasificados o divididos en función de algunas de sus características:

- Por su estructura.
- Por los servicios ofrecidos.
- Por el soporte que ofrece a los servicios.

18.4.1 Sistemas por la estructura

Dependiendo de su estructura, los sistemas operativos pueden clasificarse de la forma que veremos a continuación en los subapartados correspondientes.

Monolíticos

Esta es la estructura de los primeros sistemas operativos, que estaban constituidos fundamentalmente por un único programa compuesto de un conjunto de rutinas entrelazadas de tal forma que cada una podía llamar a cualquier otra. Las características fundamentales de este tipo de estructura son:

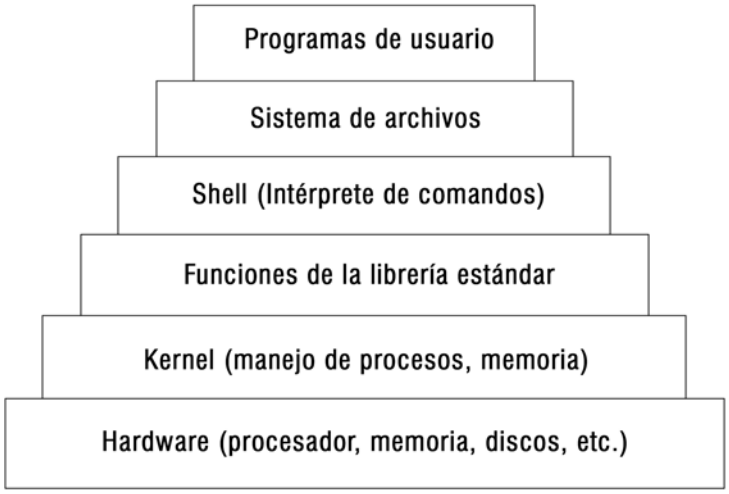
- Construcción del programa final a base de módulos compilados por separado y que se unen a través de un enlazador (*linker*).
- Buena definición de los parámetros de enlace entre las distintas rutinas existentes, lo que provoca un buen acoplamiento.
- Carecen de protecciones y privilegios en las rutinas que manejan diferentes aspectos de los recursos del equipo, como la memoria, los discos, etc.
- Están hechos a medida, por lo que son eficientes y rápidos pero, el mismo motivo, carecen de flexibilidad para soportar diferentes entornos de trabajo o tipos de aplicaciones.

Con capas

A medida que aumentaron las necesidades de los usuarios y se perfeccionaron los sistemas, se hizo necesaria una mayor organización del sistema operativo, donde una parte del mismo contenía subpartes y organizarlo en forma de niveles.

El sistema operativo se dividió en pequeñas partes, de forma que cada una de ellas estuviese perfectamente definida y con una evidente comunicación con el resto de los

elementos. Se constituyó una estructura jerárquica, o de niveles, en los sistemas operativos, el primero de los cuales fue denominado *THE* (*Technische Hogeschool Eindhoven*), y que se utilizó con fines didácticos. Se puede pensar también en estos sistemas como si fueran multicapa. El sistema *UNIX* (y sus derivados) entran en esa categoría de sistemas operativos.



Modelo de capas del sistema operativo UNIX

La mayoría de los sistemas operativos actuales están basados en la estructura anterior. Otra forma de ver este tipo de sistema operativo es la denominada de anillos concéntricos o *rings*. En el sistema de anillos, cada uno tiene una apertura, conocida como puerta o trampa (*trap*), por donde pueden entrar las llamadas de las capas inferiores.

De esta forma, las zonas más internas del sistema operativo, o núcleo del sistema, están más protegidas ante los accesos indeseados desde las capas más externas. Las capas más internas serán, por lo tanto, más privilegiadas que las externas. La figura anterior da un ejemplo real de como se organizan las capas en un sistema operativo.

Con máquina virtual

Se trata de un tipo de sistema operativo que presenta una interfaz para cada proceso, mostrando una máquina que parece idéntica a la máquina real subyacente. Estos sistemas operativos separan dos conceptos que suelen estar unidos en el resto de sistemas:

1. La multiprogramación.
2. La máquina extendida.

El objetivo de los sistemas operativos de máquina virtual es el de conseguir integrar distintos sistemas operativos, dando la sensación de disponer de varias máquinas diferentes en un mismo equipo.

El núcleo de estos sistemas operativos se denomina monitor virtual y tiene la misión de llevar a cabo la multiprogramación, presentando a los niveles superiores tantas máquinas virtuales como se soliciten. Estas máquinas virtuales no son máquinas extendidas, sino una copia de la máquina real, de manera que en cada una de ellas pueda ejecutarse un sistema operativo diferente, que será el encargado de ofrecer la máquina extendida al usuario.

Un ejemplo interesante de este tipo de sistemas operativos es el sistema *vmware* (http://www.vmware.com), que gracias a la tecnología de máquina virtual permite ejecutar diferentes sistemas operativos en el mismo equipo.

Cliente/Servidor

El tipo más reciente de sistemas operativos es el denominado Cliente/Servidor, también conocido como *Microkernel*, que se puede ejecutar en la gran mayoría de los ordenadores, ya sean grandes o pequeños. Este sistema sirve para toda clase de aplicaciones y, por lo tanto, es de propósito general y cumple los mismos requisitos que los sistemas operativos convencionales.

El núcleo del sistema tiene la misión de establecer la comunicación entre los clientes y los servidores. Los procesos pueden ser tanto servidores como clientes. Por ejemplo, un programa normal es un cliente que llama al servidor correspondiente para acceder a un archivo o realizar una operación de entrada/salida en un dispositivo concreto.

A su vez, un proceso cliente puede actuar como servidor de otro. Este paradigma ofrece gran flexibilidad en cuanto a los posibles servicios en el sistema final, ya que el núcleo solamente provee las funciones más básicas de memoria, entrada/salida, archivos y procesos, dejando a los servidores proveer la mayoría de funciones que el usuario final o programador puede necesitar.

Estos servidores deben tener mecanismos de seguridad y protección que, a su vez, serán filtrados por el núcleo del sistema que controla el hardware. Uno de los sistemas precursores de este tipo de sistemas operativos fue el sistema operativo *Mach*. En la actualidad, el proyecto de código libre *GNU* está desarrollando el *kernel Hurd* (http://www.gnu.org/software/hurd) y ya podemos encontrar disponible un sistema operativo completo con basado en este: el siustema *Linux Debian GNU/Hurd* (http://www.debian.org/ports/hurd).

18.4.2 Sistemas por los servicios

Esta clasificación es la más importante, al ser la más usada y conocida desde el punto de vista del usuario final.

Monousuario

Los sistemas operativos monousuario son aquéllos que sólo admiten un usuario a la vez, sin importar el número de procesadores que tenga el ordenador o el número de procesos o tareas que el usuario ejecute al mismo tiempo. Como ejemplo de este tipo de sistemas operativos encontramos, entre otros:

- *MS-DOS*.
- *Windows 9x*, *Windows ME* y *Windows XP Home Edition*.
- *Mac OS* (antes de la versión *OS X*).

Multiusuario

Los sistemas operativos multiusuario son capaces de dar servicio a más de un usuario a la vez, ya sea por medio de varias terminales conectadas al servidor o a través de sesiones remotas en una red de comunicaciones. No tiene importancia el número de procesadores en la máquina ni el número de procesos que cada usuario puede ejecutar de forma simultánea.

Algunos sistemas como *GNU/Linux* ofrecen, además, la posibilidad de trabajar con terminales virtuales. Esta característica permite emular en un único equipo el trabajo de varias terminales remotas, ya que se puede iniciar una sesión como si fuésemos diferentes usuarios (o como el mismo usuario varias veces) utilizando el mismo hardware pero de forma que parezca que cada uno trabaja con un equipo independiente. Ejemplos de sistemas operativos multiusuario son:

- *UNIX* (y sus derivados).
- *Windows NT*, *Windows 2000* y *Windows XP Professional Edition*.
- *Mac OS X*.

Monotarea

Los sistemas monotarea son los que sólo permiten una tarea a la vez por usuario. Puede darse el caso de un sistema multiusuario y monotarea, en el que se admiten varios usuarios al mismo tiempo pero en el que cada uno de ellos sólo puede estar realizar una tarea a la vez. Algunos ejemplos son *MS-DOS*, *Windows 3.x*, *Windows 95* (sistemas que sólo simulaban la multitarea, pero que realmente pertenecen a esta clasificación de monotarea).

Multitarea

Un sistema operativo multitarea es el que permite al usuario estar realizando varias acciones al mismo tiempo. Por ejemplo, puede estar escuchando música con un reproductor de *CDs*, escribiendo un texto, copiando archivos entre directorios y descargando música de *Internet* (además de mantener una interfaz gráfica). Es habitual encontrar en ellos interfaces gráficas orientadas al uso de menús y del ratón, lo que permi-

te un rápido intercambio entre las tareas, mejorando su productividad. Algunos ejemplos son: *Mac OS*, *UNIX* (y sus derivados), *Windows 98*, *Windows ME*, *Windows 2000* y *Windows XP*.

Monoprocesador

Un sistema operativo monoprocesador es aquél que sólo es capaz de manejar un procesador del equipo, de manera que si el ordenador tuviese más de uno sería inútil. El ejemplo más típico de este tipo de sistemas es el *MS-DOS* y *Mac OS* (en sus versiones más antiguas).

Multiprocesador

Un sistema operativo multiprocesador se refiere al número de procesadores del sistema, que es más de uno y es capaz de usarlos todos a la vez para distribuir su carga de trabajo. Generalmente estos sistemas trabajan de dos formas distintas: simétricamente y asimétricamente.

Cuando se trabaja de manera asimétrica, el sistema operativo selecciona uno de los procesadores para que sirva de procesador maestro y se encargue de distribuir la carga a los demás procesadores, que reciben el nombre de esclavos. Cuando se trabaja de manera simétrica, los procesos o partes de ellos (hilos o *threads*) son enviados indistintamente a cualquiera de los procesadores disponibles, teniendo, teóricamente, una mejor distribución y equilibrio de la carga de trabajo. Se dice que un hilo es la parte activa en memoria de un proceso, lo cual puede consistir en un área de memoria, un conjunto de registros con unos valores específicos, la pila u otros valores. Un aspecto importante a considerar en estos sistemas es la forma de desarrollo de las aplicaciones para aprovechar los varios procesadores.

18.4.3 Sistemas por la forma de ofrecer los servicios

Esta clasificación también se refiere a una visión externa, que en este caso se refiere a la del usuario: cómo él accede a los servicios. Bajo esta clasificación podemos detectar dos tipos principales de sistemas operatiovs: los sistemas operativos de red y los sistemas operativos distribuidos.

Sistemas operativos de red

Los sistemas operativos de red se definen como los que tienen la capacidad de interactuar con los sistemas de otros eqeuipos a través de un medio de transmisión con el objeto de intercambiar información, transferir archivos, ejecutar comandos remotos y otras actividades.

El punto crucial de estos sistemas es que el usuario debe conocer la ubicación de los recursos a los que desea acceder (dirección y/o nombre de la máquina remota, ruta o camino al recurso compartido, etc) e, incluso, en ocasiones puede ser necesario cono-

cer la sintaxis de una serie de instrucciones necesarias para la utilización del recurso (vale la pena aclarar que esto cada vez es menos frecuente debido a las interfaces más amigables*f* con el usuario, aunque sin lograr superar la flexibilidad que puede ofrecernos una línea de comandos).

Sistemas operativos distribuidos

Los sistemas operativos distribuidos abarcan los servicios de red, logrando integrar recursos (impresoras, memoria, procesos, unidades centrales de proceso) en una sola máquina virtual a la que el usuario accede de forma transparente. Es decir, ahora el usuario ya no necesita saber la ubicación de los recursos, sino que los conoce por el nombre y simplemente los usa como si todos ellos fuesen partes locales de su lugar de trabajo habitual.

La labor del sistema operativo distribuido es la de permitir a las aplicaciones hacer uso de los diversos procesadores, memorias, discos y, en general, todo dispositivo conectado al sistema, como si se encontraran físicamente en la misma máquina. Las razones para crear o utilizar sistemas distribuidos son dos principalmente: por necesidad (debido a que los problemas a resolver son inherentemente distribuidos) y para tener más confiabilidad y disponibilidad de recursos.

En general, los sistemas distribuidos (no solamente los sistemas operativos) exhiben algunas ventajas sobre los sistemas centralizados, como por ejemplo:

- **Economía**: la relación precio/rendimiento de la suma del poder de los procesadores separados contra el poder de uno centralizado es mejor cuando están distribuidos.
- **Velocidad**: relacionado con el punto anterior, la velocidad de la suma es muy superior.
- **Confiabilidad**: si falla una máquina, el sistema total sigue funcionando.
- **Crecimiento**: el poder total del sistema puede irse incrementando al añadir pequeños sistemas, lo cual es mucho más difícil en un sistema centralizado.
- **Distribución**: algunas aplicaciones requieren una distribución física.

Por otro lado, los sistemas distribuidos también tienen algunas ventajas sobre los sistemas aislados. Estas ventajas son:

- **Compartir datos**: un sistema operativo distribuido permite compartir datos más fácilmente que los sistemas aislados, que los duplican en cada nodo para poder conseguirlo.
- **Compartir dispositivos**: un sistema distribuido permite acceder a los dispositivos desde cualquier nodo de forma transparente, cosa imposible en los sistemas aislados.
- **Comunicaciones**: la comunicación persona a persona es factible en los sistemas distribuidos, en los sistemas aislados no es así.

- **Flexibilidad**: al poder agregar o quitar máquinas del conjunto distribuido, la flexibilidad que se logra es bastante grande aunque sin sacrificar otros factores. Por lo tanto, dependiendo de la tarea (o tareas) a realizar es posible aumentar o disminuir (rara vez) el poder de procesamiento.

18.5 La gestión de los procesos

Proceso es el nombre con el que se denomina la ejecución de un programa individual, formado por una serie de instrucciones que el procesador tiene que ejecutar. La mayoría de los sistemas operativos modernos basan toda su estructura alrededor de este concepto, y se presentan tres requisitos fundamentales que debe seguir todo sistema operativo en relación a los procesos:

- El sistema operativo debe intercalar la ejecución de un conjunto de procesos para maximizar la utilización del procesador ofreciendo, a la vez, un tiempo de respuesta razonable.
- El sistema operativo debe asignar los recursos a los procesos conforme a una política específica (por ejemplo, ciertas funciones o aplicaciones son de prioridad más alta).
- El sistema operativo podría tener que dar soporte a la comunicación entre procesos y la creación de procesos por parte del usuario, labores que pueden ser de ayuda en la estructuración de las aplicaciones.

18.5.1 Los ciclos de vida

Dependiendo de la forma como se ha diseñado el sistema operativo un proceso puede tener dos o más estados. Un estado, como su nombre indica, es una condición especial de la ejecución del proceso en un instante determinado de tiempo. Evidentemente, el modelo más sencillo es en el que sólo existen dos estados:

- Ejecutándose.
- No ejecutándose.

El primer estado siempre es el de no ejecución, ya que este se da en el mismo momento en que se crea el proceso y queda en espera para su próxima ejecución. En cualquier momento cuando el proceso se encuentra en ejecución puede darse un suceso, ya sea interno o externo, (p.e una terminación normal, un error de E/S, una terminación del tiempo de ejecución, etc.) que lo convierta de nuevo al estado de no ejecución. Es necesario destacar que este modelo no permite el uso de interrupciones debido a que no existe un estado de pausa o bloque similar que permita detener unos instantes la ejecución del proceso y reanudarla después desde el punto en que fue interrumpida.

Este modelo como se puede ver es bastante sencillo pero sería demasiado limitado para satisfacer las necesidades de un sistema operativo moderno, por lo cual los científicos de la computación han ideado (y aplicado) modelos con mayor cantidad de

estados, uno de los más completos es el que le otorga la posibilidad al proceso de tener 5 estados (o más).

Los cinco estados de un proceso

En este modelo se han dividido los dos estados anteriores en otros posibles estados para brindar al sistema operativo una mayor flexibilidad. Estos cinco estados son:

- **Nuevo**: este es el primer estado de todo proceso y se da tan pronto es creado por el sistema operativo pero aún no ha pasado a la cola de los procesos listos para ser ejecutados. Un proceso sólo puede encontrarse en este estado una única vez durante toda su existencia.
- **Listo**: se llega a este estado después de pasar por el estado anterior o cuando se reanuda después de un estado de bloqueo o de ejecución. Cuando un proceso se encuentra en este estado le indica al procesador que está disponible para su ejecución.
- **Ejecución**: este estado se presenta cuando el procesador está trabajando con las instrucciones del proceso, es decir, lo está ejecutando. Se llega a este estado desde el estado de listo. Cada sistema operativo asigna una cantidad de tiempo a cada proceso para ejecutarse, cuando este tiempo termina pero el proceso aún no ha terminado de ejecutarse, el sistema guarda la información del estado de ejecución del proceso y lo devuelve a la cola de procesos con el estado de listo, para evitar que un proceso monopolice el procesador.
- **Bloqueado**: cuando ocurre un suceso que asigna algún tipo de espera al proceso para poder continuar con su ejecución, este pasa al estado de bloqueo mientras dura la espera.
- **Terminado**: en el momento en que el proceso ya no necesita continuar con su ejecución, los recursos usados son liberados y el proceso pasa al estado de terminado, con lo que concluye su ciclo de vida.

Es importante aclarar que este modelo es presentado en algunos textos con una variante en la que se suprimen el primer y el último estado, pero la esencia es la misma, las transiciones entre los estados 2 a 4 son exactamente las mismas.

Estados en Linux

Antes de continuar, es importante decir que el modelo que presentamos a continuación está basado en *Linux 2.4.x*. Es posible que existan algunas pequeñas diferencias en los núcleos de versiones anteriores (o posteriores), pero la esencia es la misma en todos los que siguen el estilo del sistema *V* de *Unix*. *Linux 2.4.x* implementa tareas o procesos que pueden existir en uno de 5 estados (7 si tenemos en cuenta los dos últimos que no son realmente estados):

1. **Ejecutándose**: este estado le indica al núcleo que el proceso está en la cola de ejecución.

2. **Proceso con interrupciones**: en este estado el proceso se encuentra esperando (o durmiendo) pero puede ser despertado en cualquier momento por una señal o por la terminación del tiempo.
3. **Proceso sin interrupciones**: en este estado el proceso también se encuentra dormido pero a diferencia del anterior, el proceso no puede ser despertado por eventos externos.
4. **Zombi**: este estado se presenta cuando la tarea ha terminado pero aún faltan algunos recursos por liberar, por lo que el proceso puede considerarse todavía como activo dentro del sistema.
5. **Parado**: el proceso se encuentra parado bien por una señal de control del trabajo o por una señal del sistema.
6. **Exclusivo**: realmente no se trata del estado de algún proceso pero puede ser uno de los estados 2 o 3.
7. **Terminado**: cuando el proceso a concluido de forma satisfactoria puede decirse que ha entrado en este estado.

18.5.2 Concurrencia

En los sistemas operativos, la concurrencia se presenta cuando dos o más procesos deben hacer uso de un mismo recurso del sistema al mismo tiempo. Podríamos pensar que se está presentando una contradicción o que estamos hablando de sistemas con más de un procesador, ya que en un único procesador no se puede ejecutar más de un proceso al mismo tiempo; esto es cierto pero debemos tener presentes otros factores y para poder explicarlo mejor utilizaremos un ejemplo a través de la rutina que presentamos a continuación:

```
void ejemplo (void)
{
printf (“Entre el número que desee: ”);
scanf (“%d”, &pos_comp);
pos_comp += 10;
printf (“El número entrado más 10 es: %d”, pos_comp);
return;
}
```

Supongamos que esta es una rutina del sistema y que la pueden utilizar diferentes procesos (en cualquier momento). La rutina se encarga de leer la entrada estándar de un número a través del teclado, y lo almacena en la variable global *pos_comp*. Esta variable es una posición compartida en la memoria y todo proceso que llame a esta rutina puede leer y escribir en ella. Entonces, si un proceso A llama a la rutina pero, por alguna razón, la *CPU* detiene su ejecución después de haber finalizado la instrucción de recogida del valor introducido, por lo cual la variable ya ha tomado un nuevo valor (supongamos 15) que ha sido el introducido por el usuario cuando el proceso A

aún se encuentra bloqueado, el proceso B es ejecutado y también usa la rutina del ejemplo, lee un nuevo valor (supongamos 11) y continua su ejecución normal hasta el final de la rutina y, por lo tanto, debe mostrar el resultado de 21 y hasta este punto todo es completamente normal. Sin embargo, después de un tiempo se reanuda la ejecución del proceso A desde el punto en que fue interrumpido y como debe usar la variable *pos_comp* se la pide a la memoria y continúa su ejecución normal hasta finalizar la rutina. Sería de esperar que al haber introducido un 15, el resultado sea 25, pero no hemos tenido en cuenta algo: la variable *pos_comp* también fue utilizada por el proceso B y modificó su valor y ahora su valor es 21, por lo que el resultado que nos mostrará el proceso A es 21, ya que este es el valor de 11 + 10.

Este ejemplo nos demuestra como la concurrencia puede presentarse (y de hecho lo hace bastante) en cualquier instante y como puede afectar significativamente al trabajo del equipo. Claro que el ejemplo anterior es una de las tantas formas en que la concurrencia afecta la interacción entre distintos procesos, pero no es la única. Generalmente, podemos distinguir entre tres tipos de interacción entre los procesos:

- Competencia.
- Cooperación por compartición.
- Cooperación por comunicación.

Conocimiento	Relación	Influencia	Problemas control
Los procesos no tienen conocimiento de los demás procesos.	Competencia	Los resultados de un proceso son independientes de las acciones de los otros. Los tiempos de los procesos pueden verse afectados.	Exclusión mutua. Interbloqueo (recursos renovables). Inanición.
Los procesos tienen conocimiento indirecto de los otros (p.e. objetos compartidos).	Cooperación por compartición	Los resultados de un proceso pueden depender de la información obtenida de los otros. Los tiempos de los procesos pueden verse afectados.	Exclusión mutua. Interbloqueo (recursos renovables). Inanición. Coherencia de datos.
Los procesos tienen conocimiento directo de los demás.	Cooperación por comunicación	Los resultados de un proceso pueden depender de la información obtenida de los otros. Los tiempos de los procesos pueden verse afectados.	Interbloqueo (recursos consumibles). Inanición.

Estos tipos de interacción se producen dependiendo del nivel de conocimiento que un proceso tiene de los demás. La tabla anterior nos presenta de forma resumida, aunque bastante clara, los tipos de relaciones entre los procesos dependiendo de los grados de conocimiento. También aparece la forma en que esta relación influye en los otros procesos y, además, algunos de los problemas que se pueden presentar.

Como podemos observar, la competencia entre procesos nos puede plantear tres tipos diferentes de problemas. El primero de ellos se basa en situaciones como la del ejemplo anterior, y en estos casos es necesario que el sistema operativo aporte una forma de exclusión mutua, es decir garantizar que un proceso que utilice un recurso compartido tenga la seguridad que otro proceso no lo ha modificado durante su ejecución.

Desafortunadamente, la exclusión mutua conlleva otro gran problema, el interbloqueo, que se da cuando dos o más procesos requieren acceso a dos o más recursos compartidos.

El problema se puede presentar en el caso de que el sistema operativo asigne a cada proceso un recurso, pero dicho proceso necesite, a su vez, el recurso que tiene otro proceso (y viceversa) por lo que ambos procesos son bloqueados a la espera de que se libere el recurso que necesita cada uno, por lo cual ambos procesos se bloquean permanentemente ya que ninguno de los dos puede terminar hasta no utilizar el recurso que les falta y por este motivo el recurso que ya ha sido asignado no se puede liberar.

El tercer problema es la inanición, que se presenta cuando un proceso permanece en estado de bloqueo permanente a la espera de poder usar un recurso que necesita, pero que por algún motivo externo, nunca logra que le sea asignado (por lo general por que dos o más procesos se pasan mutuamente el control del recurso, por lo que siempre estará ocupado).

Estos tres problemas también se dan cuando los procesos no compiten entre sí por el recurso si no que, por el contrario, deben cooperar entre ellos (sin tener conocimiento exacto de quién es el otro y que hace) ya que son conscientes de que no son los únicos procesos que hacen uso de este recurso. Para el caso de la exclusión mutua debemos aclarar que sólo se debe hacer uso de ella en el caso que los datos compartidos deban ser modificados.

Adicionalmente, en la cooperación por compartición también se debe garantizar la coherencia de los datos, ya que puede darse el caso de que a pesar que cuando se modifica un dato nadie más pueda modificarlo (por la exclusión mutua) hasta que no finalicen las instrucciones que hacen uso del dato, es posible que se requiera de algún otro dato (o datos) con los que se trabaje, en conjunto con el que tiene la garantía de la exclusión mutua, en diversas partes del proceso puede suceder que entre cada uso de la información del dato pueda variar su contenido.

Por lo tanto, puede presentarse el caso de que aún habiendo exclusión mutua, el resultado de la ejecución puede variar su comportamiento normal. Para solucionar esto se

utiliza la atomicidad de las instrucciones, es decir no pueden presentarse cambios mientras un conjunto de instrucciones (declaradas como atómicas) se esté ejecutando. Este tipo de situaciones se presenta mucho en las aplicaciones de bases de datos.

Exclusión mutua

El principio de exclusión mutua es un mecanismo empleado en el diseño de los sistemas operativos para evitar los problemas de competencia por los recursos. Se basa en definir una zona, o región crítica, que está marcada por las instrucciones que hacen uso del recurso, o recursos, por el que se presenta la competencia (recurso crítico). Existen diferentes métodos de aplicación de la exclusión mutua tanto por hardware como por software.

Interbloqueo

Este problema se presenta cuando dos o más procesos se bloquean mutuamente a la espera de poder usar un recurso crítico y se mantiene de manera permanente. No existen, como pasa en la exclusión mutua, grandes soluciones que permitan deshacer un bloqueo entre procesos y, por el contrario, se deben utilizar diversas técnicas para prevenir este tipo de situaciones.

Inanición

La inanición se presenta cuando un proceso nunca logra acceder a un recurso crítico del sistema y, por lo tanto, no puede continuar con su ejecución normal.

18.6 Comparativa de sistemas

Hacer una verdadera comparativa entre un sistema operativo y otro es un tema bastante delicado, en especial si se trata de ser lo más imparcial posible y, a la vez, tratar de abarcar todos los requisitos que la mayoría de usuarios esperan encontrar en su plataforma de trabajo.

Teniendo presente esta reflexión lo primero que es necesario plantearse antes de empezar es saber cuáles son las bases para la comparación.

Hemos divido la respuesta en dos aspectos distintos: el administrativo y el técnico. A nivel administrativo se ha detectado que la mayoría de usuarios sin conocimientos técnicos acerca del funcionamiento de los sistemas operativos evalúan los siguientes aspectos:

1. Licencia y precio.
2. Estabilidad y rendimiento.
3. Facilidad de uso.
4. Soporte.

Generalmente, éstos son los principales aspectos que un administrador de sistemas tiene presentes en el momento de presentar una evaluación a los usuarios no técnicos. A nivel técnico, los aspectos que se evalúan son:

1. Compatibilidad con otras plataformas.
2. Portabilidad.
3. Requerimientos de hardware.

Aunque a nivel técnico podemos encontrar una gran cantidad de factores más que no hemos incluido en esta revisión es necesario aclarar que hemos procurado ser bastante genéricos, ya que cada compañía (e incluso persona) puede tener una perspectiva diferente en cuanto a la evaluación del mejor sistema operativo.

Ahora que hemos visto los componentes de la comparativa, también es indispensable enumerar los sujetos de estudio, es decir los sistemas operativos sobre los que pretendemos hacer el estudio.

Es necesario destacar que al igual que los anteriores aspectos, los siguientes también tienen la característica de la generalidad, por lo cual sólo hemos escogido una pequeña muestra de la gran cantidad de sistemas operativos disponibles en el mercado.

Los sistemas operativos que vamos a comparar son:

1. *FreeBSD*.
2. *GNU/Linux*.
3. *Mac OS X*.
4. *NetBSD*.
5. *OpenBSD*.
6. *Windows 98*.
7. *Windows 2000*.
8. *Windows XP*.

A continuación veremos una breve descripción de cada uno de estos sistemas operativos, con el objetivo de dar una visión general. Después pasaremos a explicar cada uno de los puntos de la comparación.

18.6.1 Los sistemas operativos

FreeBSD

FreeBSD es un sistema operativo derivado del *386 BSD* y se trata de un sistema operativo de distribución libre (y gratuito) creado por cientos de desarrolladores (*http://www.freebsd.org*). Es muy usado como servidor de *Internet* debido a sus altas prestaciones en las comunicaciones.

GNU/Linux

Aunque tenemos mucha más información, en forma de los manuales oficiales en el *CD-ROM* que acompaña al libro, es necesario destacar que se trata de un sistema libre desarrollado por miles de personas a través de *Internet* y que es el sistema operativo con un mayor crecimiento en la actualidad, y el segundo de mayor uso en el mundo.

Mac OS X

Es un sistema operativo desarrollado por la compañía Apple Computer Inc. y se trata, prácticamente, de una reescritura completa de su sistema operativo *Macintosh*. Este nuevo sistema *Mac* está basado en el sistema *Darwin*, un proyecto de código abierto que utiliza características de otros sistemas *UNIX*. La idea es crear un completo sistema operativo con la flexibilidad y robustez de *UNIX* y la facilidad de uso que hasta ahora a caracterizado a los sistemas del entorno *Mac*.

NetBSD

NetBSD (http://www.netbsd.org)es el sistema operativo con una mayor portabilidad del mundo y se trata de otro descendiente de *4.4 BSD* y *386 BSD*, lo que implica que puede ser libremente distribuido en forma binaria o de código fuente. El principal objetivo de *NetBSD* es la portabilidad, obviamente sin descuidar la seguridad y estabilidad como la mayoría de los sistemas operativos derivados de *UNIX*.

OpenBSD

Este sistema operativo es un derivado de *NetBSD* y sus principales objetivos son la seguridad, la estandarización y la portabilidad. Está catalogado como el sistema operativo más seguro del mundo.

Windows 98

Este es uno de los sistemas operativos con una mayor cantidad de usuarios en el mundo, ya que se encuentra a mitad de camino (en términos de temporalidad) entre la aparición de *Windows 95* y de *Windows 2000*. Entre sus principales características podemos encontrar el soporte real a la multitarea.

Windows 2000

Windows 2000 fue la nueva generación del sistema operativo *Windows NT* para redes de Microsoft, constituyéndose en una mejora significativa desde varias perspectivas, entre las que destacaríamos la estabilidad.

Windows XP

Este sistema operativo utiliza el nuevo motor del sistema que Microsoft desarrolló para *Windows 2000* y, por lo tanto, integra unas altas prestaciones gráficas junto a las

características de trabajo corporativo heredadas de *Windows 2000* y que otorgan una alta estabilidad al sistema.

18.7 Comparación a nivel administrativo

18.7.1 Licencia

Cuando hablamos de licencia nos referimos al acuerdo que se firma entre las dos partes: el productor y el usuario del sistema operativo y, que por lo general, el segundo descuida (la mayoría de las veces se limita a aceptar el acuerdo sin leerlo brevemente) pero que constituye un acuerdo de seguridad para el primero. Se encuentra muy ligada al precio (a pesar de tratarse de dos conceptos diferentes) debido a que lo que el usuario compra son licencias.

A continuación daremos un repaso a los esquemas de licencia de cada uno de los sistemas opertativos de nuestro estudio.

FreeBSD

Como ya hemos visto anteriormente, este sistema operativo es libre, su *kernel* está licenciado bajo la licencia *BSD* al igual que una gran cantidad de sus aplicaciones, especialmente las de integración con el sistema o de configuración.

Sin embargo, como la mayoría de los sistemas operativos *UNIX* utiliza algunas de las herramientas generadas por el proyecto *GNU*, por lo que algunas partes del sistema *FreeBSD* se encuentran licenciadas por la *GPL*. En cuanto al precio, el sistema *FreeBSD* se puede conseguir de forma gratuita descargándolo de *Internet*.

GNU/Linux

El sistema operativo *GNU/Linux* se encuentra liberado y protegido, a la vez, por la licencia pública general o *GPL*, que destaca por ofrecer las siguientes libertades:

1. Libertad para ejecutar el programa con cualquier propósito.
2. Libertad para modificar el programa y adaptarlo a nuestras necesidades. Para que esta libertad sea efectiva en la práctica, es necesario disponer del código fuente.
3. Libertad para redistribuir copias, tanto gratis como por dinero.
4. Libertad para distribuir versiones modificadas del programa, de manera que la comunidad *GNU* pueda beneficiarse con las mejoras.

En cuanto al precio, *GNU/Linux* se encuentra disponible en *Internet* sin coste alguno. Sin embargo, incluso los propios fabricantes de las diferentes distribuciones o cualquier persona o empresa puede cobrar por la distribución del sistema.

Mac OS X

El licenciamiento del *Mac OS X* merece una especial atención, ya que este sistema operativo incluye un esquema libre y propio. Este sistema tiene su base fundamental (*core*) en el sistema libre *Darwin*, liberado bajo la licencia *Apple Public Source License* (*APSL*) cuya versión 2.0 ha sido aprobada por la *Free Software Foundation* como una licencia de software libre. Sin embargo, como ya hemos dicho, también posee componentes propios, como la interfaz gráfica *Aqua* (introducida aprovechando la aparición de los iMac), por lo que su esquema de licencia también es propietario. Por ejemplo, el *Mac OS X Phanter* (una de las últimas versiones) dispone de dos alternativas de compra: 1 usuario (es decir sólo se puede instalar en un equipo) y el pack familiar para un máximo de 5 usuarios.

NetBSD

Este sistema operativo se encuentra liberado bajo la licencia *BSD* y esto implica, obviamente, que se garantiza su libre redistribución para cualquier fin.

OpenBSD

Esta es la tercera variante directa de los *BSD* y al igual que los anteriores es libre y se rige por la licencia *BSD*, que garantiza su libre redistribución, aunque sujeta a los siguientes aspectos generales:

- La redistribución debe mantener cualquier nota de copyright del original, preservando la propiedad intelectual.
- El software regido por esta licencia se entrega sin ningún tipo de garantía.

Windows 98

El esquema de licencias de la empresa responsable de este sistema operativo ofrece varias alternativas, desde la licencia *OEM* para los equipos con el sistema preinstalado, hasta las licencias de tipo *campus agreement*, es decir, licencias para cuando es necesario licenciar un número significativo de equipos.

Sin embargo, ya no es posible conseguir de forma oficial este sistema operativo, ya que Microsoft decidió (unilateralmente) dejar de venderlo, y tan sólo es posible conseguirlo a través de terceros que aún tengan remanentes.

Windows 2000

El esquema de licencia es similar al de *Windows 98*, aunque para esta versión es todavía un poco más complicado, ya que no sólo debe tenerse en cuenta el número de equipos que deben poseer una licencia, sino también el número de procesadores en cada equipo, en especial si pasa de 2, ya que la licencia (y el mismo sistema operativo) deben ser versiones especiales. *Windows 2000* fue lanzado al mercado en 4 versiones diferentes:

- *Professional*.
- *Server*.
- *Advanced Server*.
- *Datacenter Server* (únicamente disponible para su distribución con equipos nuevos y sólo para distribuidores autorizados).

Cada una de estas versiones posee unas características que la hacen más adecuada a determinadas necesidades. La versión *Professional* es la versión para equipos domésticos y ofrece las funciones básicas. La versión *Server* ya ofrece las funciones de los servidor de archivos e impresión, así como capacidades de servidor *Web*. La versión *Advanced Server* añade las funciones de alta disponibilidad a las de la versión *Server*.

Windows XP

Al igual que su antecesor, Windows XP apareció en dos versiones diferentes:

- *Home Edition*: es una versión orientada al sector doméstico.
- *Professional*: esta versión incluye unas mayores capacidades de seguridad y está orientada al sector empresarial.

18.7.2 Estabilidad y rendimiento

A pesar de que existen varias pruebas para medir el rendimiento de un sistema, generalmente, éstas se ejecutan en condiciones generadas , es decir, son realizadas en laboratorios y con equipos especialmente puestos a punto para la comparación y, por lo tanto, no son muy fiables cuando tratamos de estudiar la realidad del uso diario.

Cuando hablamos de estabilidad nos referimos a las capacidades del equipo para soportar la carga de trabajo sin tener alteraciones que impliquen bloqueos o reinicios inesperados, que derivan en tiempo perdido e, incluso, en posibles perdidas de datos; el rendimiento hace referencia a la relación trabajo/tiempo, es decir a la mejor administración de los recursos del sistema logrando una ejecución rápida de las aplicaciones sin descuidar la estabilidad.

FreeBSD

Sin duda alguna, el sistema *FreeBSD* es uno de los sistemas operativos más estables del mercado. Evidentemente, esto es así por ser una variante de *UNIX*. Por ejemplo, en las estadísticas de servidores *Web* que realiza la consultora independiente Netcraft, los 50 sitios *Web* que más tiempo han estado sin tener que reiniciarse se encuentran repartidos casi equitativamente entre equipos con el sistema operativo *FreeBSD* y *OpenBSD*.

En cuanto a rendimiento, también podemos observar que una gran cantidad de sitios de *Internet* se encuentran soportados por este sistema operativo.

GNU/Linux

Aunque el rendimiento puede variar entre una distribución y otra, depende en gran medida del hardware en el que se está ejecutando. *GNU/Linux* es un sistema operativo bastante fiable tanto en rendimiento como en estabilidad.

Debido a su gran flexibilidad es posible personalizarlo de múltiples maneras, de forma que cumpla con los requisitos necesarios, es decir si necesitamos una estación de trabajo con una buena interfaz gráfica para el hogar o la oficina, podríamos no ejecutar los servidores u otras aplicaciones y dejar simplemente lo que necesitamos. Si por el contrario lo que deseamos es un servidor para *Internet* o algún otro tipo de servicio como *FTP* o de bases de datos, no es necesario que tenga una interfaz gráfica, ya que va a consumir recursos de memoria y procesador que se pueden necesitar en algún momento para los servidores.

En cuanto a la estabilidad y como buen derivado de *UNIX*, el sistema *GNU/Linux* es altamente estable.

Mac OS X

La estabilidad en los sistemas *Macintosh* ha sido altamente valorada a lo largo de su historia. Con este nuevo sistema, la casa Apple ha logrado una buena combinación entre estabilidad y rendimiento y ahora ya es considerado como competencia incluso de los sistemas *GNU/Linux* o *FreeBSD* en términos de fiabilidad. De la misma forma, el trabajo a nivel de rendimiento en esta plataforma parece superar a las versiones anteriores de este mismo sistema operativo.

NetBSD

El desarrollo de *NetBSD* se caracteriza por su transparencia en la codificación para permitir su protabilidad. Sin embargo, esto conlleva otro beneficio añadido como es la estabilidad y el rendimiento. El principal objetivo de *NetBSD* no es ofrecer cada vez nuevas características que puedan resultar incompatibles con algunas plataformas, sino que, por el contrario, procura escribir y mantener un código sin mayores problemas para que su mantenimiento sea realizable de forma fácil.

OpenBSD

Aunque el principal propósito de *OpenBSD* es la seguridad, esto no implica que su rendimiento y estabilidad no sean los más adecuados. En gran parte, esto se debe a que por más seguro que pueda llegar a ser un sistema operativo sin ser lo suficientemente estable para que esa seguridad sea útil, éste no llegaría ni a ser utilizado.

Windows 98

A diferencia de los anteriores sistemas, precisamente la estabilidad es una de las grandes carencias de este sistema operativo. Cualquier usuario se habrá encontrado con

algún bloqueo del sistema, reinicio, o la tristemente famosa pantalla azul indicando que ha ocurrido algún tipo de error del sistema y que, muy posiblemente, sólo pueda solucionarse reiniciando el sistema.

También es necesario reconocer el adelanto que significó respecto a su antecesor (*Windows 95*) tanto en estabilidad como en rendimiento, especialmente este último al incorporar, por primera vez, la multitarea de forma real.

Windows 2000

Este es uno de los mejores sistemas *Windows* que se han desarrollado, ya que integraba el superior rendimiento y mayor estabilidad de *Windows NT* (en comparación con *Windows 95/98*) con la interfaz unificada y más amigable de *Windows 95/98*. Su estabilidad era especialmente elevada en las versiones *Server* y *Advanced Server* y podía llegar a compararse con la de algunos sistemas *UNIX*, hecho que le permitió incluso conseguir una porción del mercado de los servidores en *Internet*.

Windows XP

En sus primeras versiones realmente tenía varias deficiencias en cuanto a estabilidad. Sin embargo, en la actualidad (después del *Service Pack 4*) ha logrado mejorar ampliamente aunque sin lograr superar a su antecesor *Windows 2000*.

18.7.3 Facilidad de uso

Si el punto que hemos tratado anteriormente podía ser algo subjetivo, el punto que vamos a ver ahora sobre la facilidad de uso lo es aún más, ya que lo que para algunos usuarios puede ser fácil (como una línea de comandos) para otros puede resultar absolutamente difícil de usar. Por esta razón, tan sólo comentaremos algunas alternativas de las interfaces de línea de comandos como de las interfaces gráficas.

Para este apartado hemos agrupado los 3 sistemas *BSD* y *GNU/Linux* en un sólo grupo, ya que las características de las interfaces son similares, de la misma manera hemos agrupado las 3 versiones de *Windows* por la misma razón. Caso a parte es el sistema *Mac OS X*, ya que a pesar de ser también una variante *UNIX*, sus cambios en la interfaz resultan altamente significativos.

FreeBSD, NetBSD, OpenBSD, GNU/Linux

Como todo sistema *UNIX*, en estos sistemas operativos la interfaz primaria es la línea o interprete de comandos, la cual les otorga una gran flexibilidad, pero puede llegar a ser intimidante para algunos usuarios, en especial a aquéllos que están acostumbrados a trabajar exclusivamente con interfaces gráficas como ocurre con otros sistemas operativos .

Sin embargo, esto no es ningún impedimento para que los sistemas *BSD* y *GNU/Linux* puedan tener interfaces gráficas bastante amigables y que permitan realizar

algunas de las operaciones a las que se tiene acceso a través del interprete de comandos. Todo sistema *UNIX* tiene como motor gráfico el sistema *X-Window* y, específicamente, estos tres sistemas operativos utilizan una implementación libre de este servidor, que se denomina *XFree86*. Por esta razón, la mayoría de gestores de ventanas e, incluso, los dos proyectos de entornos de escritorio (*GNOME* y *KDE*) pueden ser utilizados en estas plataformas. Para entender un poco mejor estos conceptos es importante clarificar un poco el manejo de la *GUI* en *UNIX*.

La primera capa del entorno gráfico (obviamente por encima del *kernel* y otros componentes) es el servidor *X* (en nuestro caso el *XFree86*), y es el encargado de proveer al sistema una forma de soportar las aplicaciones que hagan uso de las indispensables interfaces gráficas.

Sin embargo, con el *XFree* lo único que podemos hacer es colocar en la pantalla la aplicación, pero sin bordes de ninguna clase, y mucho menos la posibilidad de poder maximizar, minimizar o mover la ventana, para esto se utiliza una nueva capa sobre el servidor *X*, a la que se denomina *WindowManager* o *Manejador de Ventanas*, y estos ya nos permiten hacer todas las operaciones que regularmente realizamos en cualquier entorno gráfico.

Por encima de los manejadores es posible colocar los entornos de escritorio *GNOME* (*GNUNetwork Object Model Environment*) o *KDE* (*K Desktop Environment*). Estos entornos poseen varias características que los hacen más convenientes que los manejadores de ventanas, entre las que podemos destacar la integración. A continuación veremos algunas capturas de pantalla con algunas versiones de *GNOME*, *KDE* o de *Manejadores de Ventanas*.

En estos sistemas *UNIX* podemos personalizar la interfaz para acomodarla a nuestros gustos o necesidades. De todo esto, podemos concluir que la facilidad de uso es muy relativa, pero que, por lo menos, en estos sistemas es posible contar con varias alternativas que se ajusten a las necesidades particulares de cada usuario.

Mac OS X

Desde sus inicios, los sistemas *Mac* han destacado por su facilidad de uso y por su amigable interfaz gráfica, y esta última versión no es una excepción. Su sistema de ventanas *Aqua* se integra de forma nativa con el sistema de ventanas *XFree86*, lo cual consigue una gran apariencia gráfica.

Windows 98, 2000, XP

A diferencia de los sistemas *UNIX*, la interfaz primaria para *Windows* es la *GUI* y no la línea de comandos mientras que en *UNIX* la mayoría de cosas que se hacen a través del interprete de comandos, en estos sistemas es posible (aunque a veces de forma más compleja) hacerlas mediante la *GUI*.

Por el contrario, en estos sistemas es bastante complicado (y muchas veces imposible) desempeñar la mayoría de acciones de la *GUI* a través de una interfaz de línea de comandos.

Para la mayoría de usuarios la interfaz más sencilla para trabajar y visualmente más atractiva es la es estos sistemas (aunque en mayor medida esta apreciación se debe a la falta de conocimiento de otras alternativas), la organización de la interfaz en los sistemas *Windows* sumada a su amplia difusión hacen que para el usuario sea un poco más rápido encontrar lo que necesita.

Windows XP

18.7.4 Soporte

Por soporte entendemos la ayuda que otra compañía o persona puede prestarnos en caso de fallos o ayuda para otro tipo de problemas. Existen diversos tipos de soporte, siendo los más comunes el telefónico y el del técnico que se traslada hasta donde se encuentra el equipo afectado. Sin embargo, no siempre es necesario que el soporte se preste de una de estas formas.

En muchas ocasiones también se puede conseguir soporte a través de *Internet*, bien sea a través de la documentación que solvente el problema o incluso consultando en foros o grupos de interés, en los que otras personas nos pueden ayudar mucho con posibles soluciones.

FreeBSD, NetBSD, OpenBSD, GNU/Linux

Debido a que estos sistemas operativos no han sido desarrollados por una compañía, el soporte es algo opcional en el sentido de que no existe personal certificado por la empresa para llevar a cabo esta tarea. Aparentemente, esto significaría que no es posible encontrar soporte técnico para estos sistemas operativos pero, sin embargo, esto dista mucho de la realidad, ya que en todo el mundo existen cientos de empresas que se dedican a esta tarea.

En los sitios *Web* de cada uno de estos sistemas operativos, los desarrolladores recomiendan algunas de estas empresas, aunque siempre es posible encontrar otras con igual o mayor calidad en su trabajo; la ventaja de tratarse de sistemas libres permite que cualquiera pueda aprender su funcionamiento hasta en los aspectos más técnicos (cosa que no suele ocurrir con los soportes de otros sistemas donde el código es cerrado) y, por lo tanto, ofrecer la ayuda más adecuada en cualquier situación.

Pero esta no es la única forma de encontrar ayuda para solucionar problemas con estos sistemas, ya que en *Internet* podemos encontrar mucha documentación sobre como hacer las cosas, muy buenos manuales y, en general, muchas ideas que nos pueden servir de mucho.

Mac OS X

Apple sólo ofrece soporte a través de *Internet* a sus usuarios registrados, aunque también mantiene gran cantidad de documentación para que pueda ser visitada libremente. Igualmente, a través de esta página *Web* también es posible encontrar información acerca de grupos de usuarios en todo el mundo.

Además, al tratarse fundamentalmente de un derivado de *UNIX*, existen muchas cosas que se realizan de forma similar, por lo cual una gran parte de la abundante documentación para los sistemas operativos de este tipo también es de utilidad para el *Mac OS X*.

Windows 98, 2000, XP

En esta familia de sistemas operativos aunque existen muchas empresas que brindan soporte de forma independiente, también existen algunas que son contratadas o certificadas directamente por el fabricante.

18.8 Comparación a nivel técnico

En la sección anterior pudimos apreciar algunos detalles que diferencian o relacionan a los distintos sistemas operativos sin adentrarnos en los detalles técnicos. En esta sección abordaremos algunos de estos detalles para poder formarnos una idea mucho más completa de cada sistema y así poder identificar los que mejor se adaptan a nuestras necesidades personales.

18.8.1 Compatibilidad con otras plataformas

Cuando hablamos de compatibilidad con otras plataformas debemos tener en cuenta los siguientes conceptos:

- **Soporte a sistemas de archivos diferentes**: este concepto se asocia a la capacidad que tiene una plataforma de leer y/o escribir en sistemas de archivos de otras plataformas diferentes.
- **Ejecución de aplicaciones compiladas para otras plataformas**: algunos sistemas operativos son capaces de ejecutar aplicaciones que fueron diseñadas para trabajar en una plataforma diferente. La mayoría de las veces esto se lleva a cabo a través de otra aplicación que actúa como una capa intermedia entre el sistema operativo y la aplicación. Esta capa permite emular las llamadas al sistema y a algunos otros procesos del sistema operativo para que el programa piense que se está comunicando con la plataforma para la cual fue compilado.
- **Compartir recursos en red con otras plataformas**: ya sabemos que el mundo actual es un mundo conectado, es decir, las redes y la comunicación entre diferentes equipos es algo cada vez más indispensable y, por lo tanto, el sistema operativo debe ser capaz de comunicarse con los sistemas diferentes de otros equipos y, obviamente, ofrecerles la posibilidad de utilizar sus recursos (generalmente archivos e impresoras, aunque también podría ser cualquier dispositivo).

18.8.2 Portabilidad

Cuando en sistemas operativos se habla de portabilidad se hace referencia a la posibilidad de utilizarlos en diferentes tipos de procesadores que incluso pueden tener arquitecturas o diseños diferentes.

18.8.3 Requerimientos de hardware

Hace referencia al hardware mínimo necesario del equipo sobre el que se podrá ejecutar el sistema operativo sin demasiados problemas.

18.9 El sistema operativo GNU/Linux

En 1991, el joven finlandés Linus Benedict Torvalds comenzó a desarrollar un sistema operativo inspirado en *Minix* (sistema clónico de *UNIX* desarrollado por el profesor Andrew S. Tanenbaum en 1987 como complemento práctico de su libro *Operating Systems: Design and Implementation*). *Minix* era distribuido bajo una licencia libre y, por este motivo, muy utilizado para la enseñanza práctica de los sistemas operativos.

En poco tiempo y gracias a la ayuda de muchas personas, entre otras el propio Tanenbaum, con quién sostuvo acaloradas discusiones sobre varios puntos del desarrollo del sistema, siendo la más comentada la conveniencia de un *kernel* de tipo

modular (opinión de Tanenbaum) o monolítico (opinión de Torvalds), Linus pudo desarrollar un núcleo al que denominó *Linux*.

Cuando *Linux* empezó a ser funcional alrededor de 1992, la decisión fue obvia ya que *GNU* poseía todas las herramientas de un sistema operativo menos el *kernel* y Linus había desarrollado un *kernel*, que sin ser el óptimo, cumplía con las necesidades del proyecto. El resultado evidente fue el *GNU/Linux*.

18.9.1 GNU/Linux

Al comienzo, la utilización de *GNU/Linux* resultaba bastante complicada aún teniendo conocimientos de informática, debido a la necesidad de descargar y compilar cada paquete de forma independiente. Debido a esta complejidad, algunas personas como Patrick Volkerding (creador de *Slackware*, una de las primeras distribuciones) decidieron crear una forma de hacer todo este proceso más sencillo creando el concepto de distribución. Una distribución agrupaba en medios magnético diversos paquetes y ofrecía un sistema de instalación relativamente más sencillo. Con el tiempo, algunas otras personas como Bob Young y Marc Ewing (fundadores de *RedHat*) adoptaron esta idea y pronto comenzó una guerra de distribuciones, en la cual todos ganaban pero el mayor beneficiario fue el usuario final.

18.9.2 Distribuciones GNU/Linux

Una distribución *GNU/Linux* es un conjunto conformado por el *kernel Linux*, aplicaciones del proyecto *GNU* y un sistema de empaquetado e instalación. Esto permite la instalación y la puesta a punto de un equipo con este sistema operativo de una forma bastante más sencilla.

Muchas veces, existen grandes diferencias entre las distribuciones, como la forma de organizar los archivos (aunque poco a poco se está logrando una estandarización) y/o el sistema de paquetes para la instalación de software (o la ausencia de este en algunos casos). En el mercado existen muchas distribuciones y cada una con un énfasis o características especiales que la hacen más recomendable para un usuario u otro. Sin embargo, existen algunas de propósito general y de reconocida trayectoria o popularidad en el mundo. Para un listado actualizado de algunas distribuciones populares disponibles podemos visitar la dirección http://www.linux.org/dist/index.html. En este sitio, es posible buscar a través de distintos parámetros para encontrar una distribución que se ajuste a nuestras necesidades.

Debian (http://www.debian.org)

Es la única distribución del sistema *GNU/Linux* totalmente libre y su desarrollo, a diferencia de las demás distribuciones, no está administrado por una empresa comercial, sino que *Debian* es desarrollado por una comunidad organizada a través de *Internet* (en la que cualquiera que lo desee y demuestre la suficiente capacidad puede participar) y que tiene como objetivo común seguir un contrato social que garantiza,

entre otras cosas, que todos los paquetes de la distribución sean 100% software libre (excepto los que pertenecen al grupo *non-free*).

El proyecto *Debian* nació el 16 de agosto de 1993 como una iniciativa de Ian Murdock de desarrollar una distribución que se basara en el espíritu de libertad de *Linux* y *GNU*, cometido que logró gracias a la desinteresada ayuda de otros hackers y, más adelante, con el apoyo de la *Free Software Foundation* (fundación para el software libre).

La primera versión disponible de *Debian* fue la 0.01 (liberada en agosto de 1993). La distribución Debian se consigue en 3 versiones diferentes según las necesidades de los usuarios:

1. **Estable (*Stable*):** son los paquetes (y las versiones de éstos) que han sido extensamente probados en las otras dos versiones, tratando de garantizar el máximo de estabilidad y seguridad (es la más recomendable para los servidores).
2. **De prueba (*Test*):** en esta versión se prueban todos los paquetes que posiblemente formarán parte de la siguiente versión estable.
3. **Inestable (*Unstable*):** lo de inestable sólo es una precaución para decir que no ha sido suficientemente probada. Todo nuevo paquete que se considera que puede formar parte de *Debian* entra primero en esta versión para ser probado exhaustivamente por la comunidad y después de cierto tiempo continua su paso ascendente hacia la versión estable.

En la actualidad (en el momento de escribir este libro) la última versión oficial (estable) de *Debian* es la 3.1 (conocida como *sarge*).

Mandrake (http://www.mandrakelinux.com)

Mandrake GNU/Linux es desarrollado por la casa francesa Mandrake Soft (*http://www.mandrakesoft.com*), y está basada en la distribución *RedHat* y se trata de una de las más amigables con el usuario. La última versión es la 9.2 (conocida con el nombre de *FiveStar*).

Morphix (http://www.morphix.org)

Morphix es otra de las grandes (en términos de popularidad) distribuciones del tipo *LiveCD*. Tiene como base *Debian GNU/Linux* y la principal característica de *Morphix* es que en realidad se trata de una distribución a partir de la cual es relativamente sencillo crear otra. Como ejemplo de esto en el sitio *Web* de esta distribución es posible encontrar 4 versiones diferentes:

- **Morphix Combined Gamer:** versión orientada al entretenimiento que incluye una gran cantidad de juegos, además de los controladores (*drivers*) para varias de las tarjetas de aceleración gráfica más difundidas.

- **Morphix Combined Gnome (o M. C. HeavyGUI):** esta versión es para equipos con altas capacidades de procesamiento (+128 MB de *RAM*, *Pentium III* o superior) Incluye como el nombre lo indica el gestor de ventanas *GNOME*, así como una gran variedad de aplicaciones para el trabajo diario y el tratamiento de imágenes. La especial característica de la mayoría de estas aplicaciones es su gran uso de los recursos del sistema, especialmente para poder desplegar una interfaz de usuario visiblemente muy atractiva.
- **Morphix Combined KDE:** la versión *KDE* de esta distribución además de incluir este gestor de ventanas contiene una gran cantidad de aplicaciones que están diseñadas para un mejor desempeño sobre *KDE* (algunas incluso sólo se pueden ejecutar bajo este entorno gráfico).
- **Morphix Combined LigthGUI:** la versión de interfaz de bajo consumo ofrece como entorno gráfico *XFCE* y algunos paquetes de productividad, con la característica principal de su bajo consumo de los recursos de hardware.

RedHat (http://www.redhat.com)

Es tal vez la más utilizada a nivel mundial, su desarrollo está a cargo de RedHat Inc., una empresa norteamericana que es la más grande (a nivel de ventas y capital) en el mundo *Open Source*.

RedHat fue desarrollada originalmente por Marc Ewing en 1994 como una alternativa a la distribución *Slackware* (las más popular entonces). A finales de ese mismo año se asocia con Bob Young y se empieza a consolidar lo que más adelante sería RedHat Inc.

La principal característica a resaltar de *RedHat Linux* es su sistema de paquetes *RPM* (*RedHat Package Manager*, manejador de paquetes de *RedHat*) que constituyó una verdadera revolución al facilitar la instalación de software de una manera sorprendente (en esa época para instalar cualquier paquete tocaba hacerlo desde sus fuentes, compilando cada paquete de forma independiente).

Slackware (http://www.slackware.com)

Es una de las distribuciones más antiguas en el mercado (su primera versión salió en abril de 1993) y fue desarrollada, originalmente, por Patrick Volkerding y en la actualidad es mantenida por la empresa Slackware Inc. en la que participan varias personas más. La filosofía de *Slackware* es ser la distribución más parecida a *UNIX*, sin dejar de lado obviamente los estándares *Linux*.

SUSE (http://www.suse.com)

Es una distribución alemana basada en *RedHat* y es una de las más amigables con el usuario, destacando especialmente su sistema de instalación/configuración, que permite al usuario configurar tanto el hardware como el software desde un único lugar, además de la administración general del sistema.

18.10 Windows XP

18.10.1 Introducción

La nueva versión de *Windows*, llamada *Windows XP*, supone un cambio importante respecto a la versión anterior, *Windows Millennium*. Desde la aparición de *Windows 95*, las nuevas versiones siempre han sido una evolución de la versión original. Sin embargo, en esta nueva versión del sistema operativo se ha producido un cambio de mayor trascendencia, ya que se ha cambiado el núcleo (*kernel*) del sistema. De cara al usuario final estos cambios pasan casi desapercibidos, pero podemos afirmar que *Windows XP* no es sólo una versión más de la familia de sistemas operativos *Windows*, sino que supone un nuevo concepto de sistema.

Hasta la aparición de *Windows XP*, la casa Microsoft disponía de dos sistemas operativos diferentes, uno para el entorno personal, *Windows 98* o *Windows Millennium*, y el otro para el entorno profesional, *Windows NT/2000*. Con la aparición de *Windows XP* se produjo una convergencia de ambas versiones, ya que se ha partido del núcleo del sistema de *Windows 2000*. A partir de este punto se han realizado algunos cambios para diferenciar las dos versiones existentes de *Windows XP*, una para el ámbito personal, conocida como *Windows XP Home Edition*, y otra para el ámbito profesional, conocida como *Windows XP Professional*.

Para los usuarios domésticos, la gran ventaja de este nuevo sistema es que *Windows XP* ha adquirido la robustez y estabilidad de Windows NT/2000. Esto es que *Windows XP* se queda menos veces bloqueado. Para los usuarios profesionales, la gran ventaja se debe a que *Windows XP* tiene una mayor compatibilidad con el hardware de la que gozaba *Windows NT/2000*.

Otra de las novedades es que *Windows XP* dispone de un nuevo sistema de usuarios completamente diferente al de las versiones anteriores de los sistemas operativos

domésticos de la familia Windows. Este nuevo sistema ha sido heredado del sistema operativo de uso profesional *Windows NT/2000*.

Ahora, en esta nueva versión, podemos definir varios usuarios con distintos perfiles independientes. Esto quiere decir que cada usuario puede tener distintos permisos, cosa que le permite realizar unas determinadas tareas. Cada usuario, por ejemplo, tendrá una carpeta *Mis documentos* propia, que a la vez podrá estar protegida con una contraseña. También dispondremos de una carpeta general, a la que tendrán acceso todos los usuarios del ordenador y donde podremos colocar los documentos que queramos compartir.

La gran ventaja es que podemos pasar de un usuario a otro sin la necesidad de apagar el ordenador, simplemente deberemos iniciar una nueva sesión con otro nombre de usuario. Más tarde, podremos volver a la sesión del primer usuario que deberá permanecer en el mismo estado que se dejó. El sistema operativo es el encargado de manejar los distintos usuarios activos.

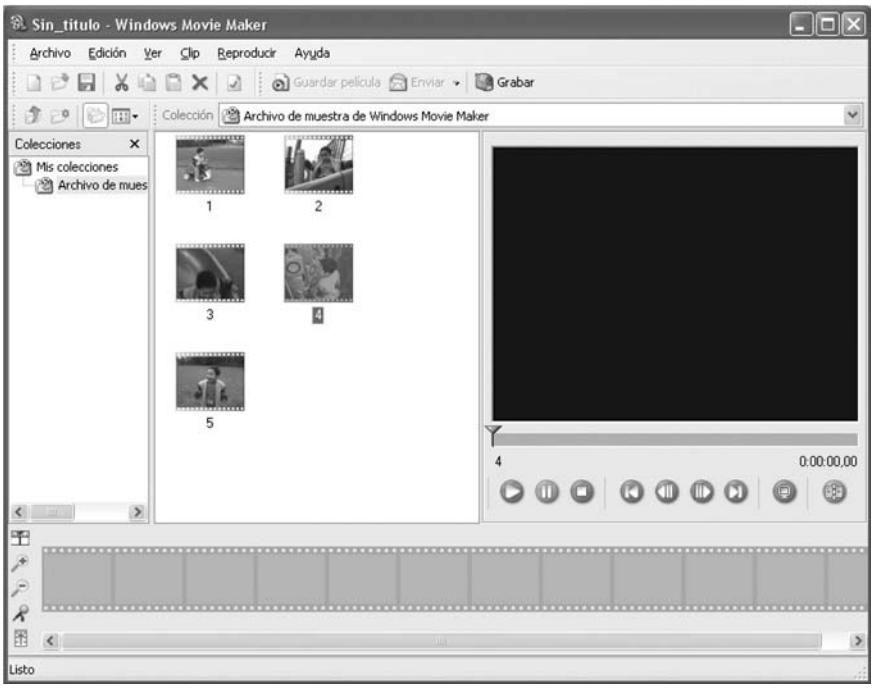
El nuevo *Explorador de Windows* ha resultado bastante mejorado. La gran novedad es que se pueden grabar *CDs* con sólo arrastrar y soltar los archivos desde el mismo *Explorador de Windows* a la unidad correspondiente a la grabadora. Otra gran novedad es que el *Explorador de Windows* ofrece un menú lateral con las opciones más útiles, siempre en función del objeto que tengamos seleccionado en cada instante. El nuevo *Explorador de Windows* ofrece un nuevo modo muy útil para ver las distintas imágenes, llamado *Tira de imágenes*. Con este modo, la imagen seleccionada se muestra en formato amplio y el resto de imágenes en formato reducido. Además, este modo de vista ofrece unos botones para avanzar y retroceder a lo largo de las distintas imágenes de la carpeta.

La nueva vista Tira de imágenes de Windows XP

Entre las otras mejoras, destacaremos que ahora es más fácil configurar una red doméstica y que *Windows XP* se recupera mejor de los errores.

> **Nota:** Cuando se produce un error grave, el mismo sistema operativo *Windows XP* genera un registro del mismo, con toda la información referente al error y nos ofrece la posibilidad de enviarlo a Microsoft.

Además, *Windows XP* arranca más rápido que sus predecesores y en la parte derecha del menú *Inicio* aparecen las aplicaciones que más se usan para poder acceder a ellas de forma más rápida. Existe un nuevo componente del sistema, llamado *Windows Movie Maker* que sirve para editar películas.



Una ventana de Windows Movie Maker

La novedad que ha suscitado más polémica ha sido la necesidad de activar el producto durante los 30 días siguientes a la instalación del mismo. La activación consiste en comunicar a Microsoft que se ha instalado el sistema operativo en un determinado ordenador y así no podrá instalarse en otros ordenadores.

18.10.2 Requisitos de hardware

Los requisitos de hardware mínimos para ejecutar *Windows XP* son los siguientes:

- **Procesador**: *Pentium 233* (o equivalente) o superior (se recomienda un *Pentium III*, *Pentium 4* o equivalente, como *AMD Duron* o *Athlon*).

- **Memoria**: 64 MBytes de memoria *RAM*. Se recomiendan 128 MBytes como mínimo. El máximo de memoria son 4 GBytes.
- **Disco duro**: al menos 1.5 GBytes de espacio libre en el disco duro que albergará el sistema.
- **Vídeo**: tarjeta gráfica super *VGA* (resolución de 800 x 600) o superior.
- **Monitor**: color con una profundidad de 16 bits o superior.
- **Teclado y ratón**: compatibles.
- **CD-ROM**: Velocidad de lectura mínima de 12x.

Estos son los requisitos mínimos y no necesariamente proporcionarán un rendimiento satisfactorio. Microsoft expone los requisitos mínimos que proporcionarán el rendimiento con el que un usuario medio puede trabajar sin problemas.

> **Nota:** El tipo de hardware que funciona perfectamente con los sistemas operativos *Windows 9x* podría no hacerlo necesariamente con *Windows XP*.

18.10.3 La compatibilidad de hardware

Existen tres métodos fundamentales para asegurar la compatibilidad del hardware. A continuación vamos a tratarlas por separado.

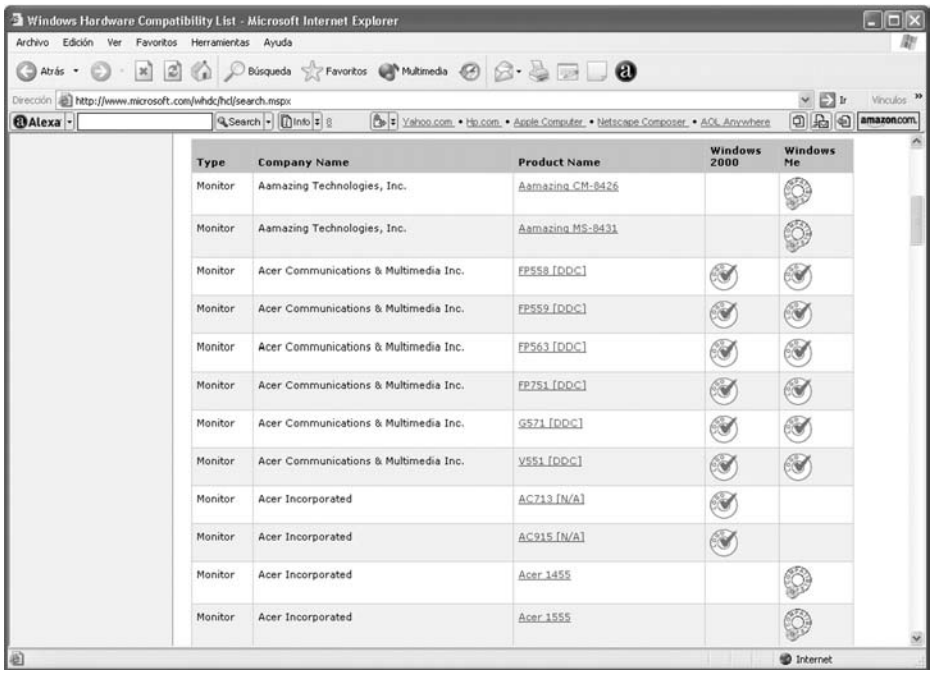
La lista de compatibilidad

El primer método es relativamente sencillo y puede ayudarnos en la adquisición de hardware nuevo. La empresa Microsoft ha llevado a cabo la mayor parte de las pruebas de compatibilidad y ha reunido la información en una lista de compatibilidad de hardware.

> **Nota:** Podemos encontrar la lista de compatibilidad de hardware en la carpeta Support del *CD* de *Windows XP*, aunque la lista estará mucho más actualizada en la *Web* de Microsoft. Si el hardware que deseamos instalar no aparece en la lista, el proceso de instalación podría no realizarse correctamente. Para encontrar la versión más reciente de la lista insertaremos el *CD* de *Windows XP* y pulsaremos en la opción *Visitar el sitio Web de compatibilidad*.

Si no sabemos el hardware que contiene el equipo no debemos preocuparnos en exceso. Podemos ejecutar la herramienta de compatibilidad de *Windows XP* desde el mismo *CD* de instalación. Esta utilidad detecta el hardware y verifica su compatibilidad con *Windows XP*. Para ello:

1. Insertaremos el *CD* y elegiremos la opción *Realizar tareas adicionales* en la ventana principal de instalación.
2. A continuación, elegiremos *Comprobar compatibilidad del sistema*.
3. Durante la ejecución del programa se creará un informe que nos comunicará si el equipo está preparado para la instalación de *Windows XP*.



Type	Company Name	Product Name	Windows 2000	Windows Me
Monitor	Aamazing Technologies, Inc.	Aamazing CM-8426		
Monitor	Aamazing Technologies, Inc.	Aamazing MS-8431		
Monitor	Acer Communications & Multimedia Inc.	FP558 [DDC]	✓	✓
Monitor	Acer Communications & Multimedia Inc.	FP559 [DDC]	✓	✓
Monitor	Acer Communications & Multimedia Inc.	FP563 [DDC]	✓	✓
Monitor	Acer Communications & Multimedia Inc.	FP751 [DDC]	✓	✓
Monitor	Acer Communications & Multimedia Inc.	G571 [DDC]	✓	✓
Monitor	Acer Communications & Multimedia Inc.	V551 [DDC]	✓	✓
Monitor	Acer Incorporated	AC713 [N/A]	✓	
Monitor	Acer Incorporated	AC915 [N/A]	✓	
Monitor	Acer Incorporated	Acer 1455		
Monitor	Acer Incorporated	Acer 1555		

La lista en el sitio Web de la casa Microsoft

Nota: Si hemos olvidado ejecutar la prueba de compatibilidad antes de la instalación, no debemos preocuparnos, ya que se ejecuta automáticamente al activar el programa de instalación. Es recomendable ejecutarla previamente para conocer, e intentar evitar, los posibles problemas.

Si algún componente no aparece en la lista de compatibilidad deberíamos instalar *Windows XP* en una partición distinta del disco duro. A continuación observamos el resultado de la instalación. Si así no funciona correctamente, deberíamos consultar al fabricante del hardware e intentar conseguir un controlador para *Windows XP*.

Un equipo preparado para Windows XP

Cada vez que se actualiza un nuevo equipo, se observan un número importante de detalles en el rendimiento. Por ejemplo, una respuesta más rápida, más administración global de electricidad, de forma que el sistema usa menos electricidad cuando se encuentra inactivo, consumo reducido de energía debido a un número menor de chips, más opciones de configuración de hardware, una unidad *DVD* más rápida compatible con medios *CD-ROM*, puertos de alta velocidad *USB* y *FireWire*, etc.

Si hemos decidido comenzar desde el principio y adquirir un nuevo equipo, existe un modo muy sencillo para elegirlo antes de comprobar cada una de los elementos en la lista de compatibilidad. Como en el caso de las listas de hardware individuales de la *HCL*, el laboratorio de pruebas de Microsoft concede la insignia de preparado para

Windows a los equipos que cumplen sus requisitos. Para ver una lista de los equipos preparados para ejecutar sin problemas *Windows XP*, podemos acceder a la página de *Internet* http://www.microsoft.com/windowsxp/ready.

18.10.4 Actualización del equipo

La operación de actualizar un equipo está de moda y es la más económica.

La preparación del hardware

La actualización de hardware necesaria para preparar el sistema para *Windows XP* depende, en gran medida, de lo cerca que esté el sistema de los requisitos mínimos de hardware que hemos visto antes.

> **Nota:** cuanto más rápido sea el sistema y más grande el disco duro, mejor para la ejecución de *Windows XP*.

La placa base y el procesador

La forma de actualización más económica es agregar más memoria *RAM*. Sin embargo, si deseamos llevar a cabo una actualización completa del sistema, deberíamos comenzar por la placa base. Los recientes cambios en la velocidad de los procesadores, la miniaturización y los requisitos eléctricos exigen actualizar tanto la placa base como la *CPU*. Incluso si estamos pensando en una actualización de la memoria *RAM*, es recomendable actualizar en primer lugar la placa base, ya que una nueva placa base utiliza, con frecuencia, un tipo de memoria *RAM* más rápida.

Seguramente, desearemos una placa base moderna con un procesador rápido. Por lo tanto, elegiremos una placa base compatible con el esquema de administración de energía *ACPI* (no simplemente *APM*) y un procesador razonablemente rápido (a partir de 2 GBytes, o incluso más rápido), como un *Pentium 4* o AMD *Athlon* (con el núcleo *Thunderbird*).

Prácticamente, todas las nuevas placas base tienen el formato *ATX*. Los suministros de energía *ATX* funcionan junto con el sistema operativo y el conjunto de *chips ACPI*, permitiendo al software controlar el estado de energía del equipo, incluyendo el modo de reposo, suspendido o de ahorro de energía, entre otros.

> **Nota:** podemos consultar el sitio de Internet www.motherboards.org para obtener más información acerca de las placas base, conjuntos de *chips* o tipos de memoria *RAM* más recientes y veloces.

Las ranuras de expansión

También debemos tener en cuenta el número necesario de ranuras (*slots*) para las tarjetas complementarias (como pueden ser tarjetas de sonido y de vídeo, entre otras). Actualmente, existe más hardware integrado en las tarjetas principales, ya que los *chips* de integración a gran escala (*VLSI, Very Large Scale Integration*) lo permiten.

De este modo, es posible que necesitemos menos ranuras que antes. Si deseamos conservar más ranuras disponibles para el resto de las tarjetas, adquiriremos una placa base con todo el sistema integrado *onboard* (como la tarjeta de sonido).

> **Nota:** si decidimos que, posteriormente, agregaremos nuestras propias tarjetas para las funciones incluidas en la placa base, deberemos asegurarnos que podemos desactivarlas mediante la configuración del *BIOS*.

En este momento, la mayor parte de las placas base disponen de ranuras *PCI* y han dejado a un lado las ranuras *ISA*.

La memoria RAM

Como en el caso de otras versiones de *Windows NT*, *Windows XP* utiliza de forma inteligente la memoria que encuentra en el sistema. Si el equipo parece que rasque el disco duro (es decir, observa un retraso, escucha ciertos sonidos y la luz de acceso al disco de la parte frontal de la torre se enciende y se apaga constantemente) es que *Windows XP* está realizando demasiados intercambios de disco. Si es así, adquiriremos e instalaremos más memoria *RAM*.

Las placas base actuales detectan automáticamente la memoria instalada (no es necesario cambiar la configuración) y *Windows XP* lee su configuración y la utiliza cuando sea necesario. En general, cuanta más memoria instalemos, mucho mejor.

El disco duro

Aproximadamente, se necesitan 1.5 GBytes de espacio libre en el disco para instalar *Windows XP*. Esta cantidad es simplemente una pequeña indicación del espacio de almacenamiento de *Windows XP*. Con el aumento del código de programación, deberemos disponer de una gran cantidad de espacio libre.

Como en el caso de la actualización de la memoria *RAM*, la actualización del disco duro es, actualmente, un proceso barato. Sin hacer un gran desembolso, podemos adquirir un disco duro de hasta 60 GBytes o, incluso, mayor.

Como en el caso de la memoria *RAM*, las placas base actuales detectan automáticamente y configuran el disco duro al instalarlo. La instalación de discos duros *EIDE* actuales (llamadas también *ATA*) se ha convertido en un proceso muy sencillo.

Los dispositivos Plug & Play

La tecnología *Plug & Play* (o *PnP*) ha incorporado un nuevo nivel de sofisticación al equipo. Muchos de los problemas en la actualización de equipos tienen su origen en conflictos internos, existentes entre los complementos y los dispositivos periféricos que no son detectados por el sistema operativo y que resultan demasiado complicados de configurar para el usuario. Con la tecnología *Plug & Play*, simplemente se conecta un dispositivo y se reinicia el sistema.

El *Plug & Play* no siempre funciona de la manera esperada, aunque la mayoría de las veces lo hace. Las plataformas basadas en *Windows NT* comenzaron a ser compatibles con *PnP* en *Windows 2000*. Con *Windows XP*, la instalación de dispositivos *PnP* es más sencilla que nunca.

Si *Windows XP* no encuentra un controlador al instalar el dispositivo, se consultará el sitio de *Windows Update*. Actualmente existen más de 5.000 controladores en el sitio de actualización de *Windows XP*. De esta manera, existen probabilidades de que el sistema encuentre automáticamente un controlador adecuado para el dispositivo conectado.

18.10.5 La instalación de Windows XP

Existen tres motivos por los que podemos necesitar instalar una nueva copia de *Windows XP*:

- El sistema operativo actual no admite una actualización a *Windows XP*.
- El sistema operativo actual admite una actualización a *Windows XP* pero no deseamos conservar los archivos y la configuración actual.
- El equipo no tiene un sistema operativo instalado.

El proceso de configuración es similar para las nuevas instalaciones y las actualizaciones, con algunas excepciones importantes. Por ejemplo, durante la nueva instalación podemos cambiar el sistema de archivos y crear una nueva partición para la instalación de *Windows XP*.

Entre las distintas opciones que podemos configurar durante la instalación, podemos elegir la configuración del *Idioma*, configurar las *Opciones avanzadas* y las opciones de *Accesibilidad*. En *Opciones regionales y de idioma* seleccionaremos la región en la que nos encontramos, para que *Windows XP* configure de manera predeterminada la fecha, hora, moneda, números, juegos de caracteres y distribución del teclado.

En las *Opciones avanzadas* podemos cambiar la ubicación predeterminada de los archivos de instalación. También podremos almacenar los archivos del sistema en una carpeta distinta a la carpeta predeterminada (*Windows*) y copiar los archivos de instalación del *CD* al disco duro.

Otro paso importante durante la nueva instalación de *Windows XP* es la elección del sistema de archivos que debe utilizar el sistema. *Windows XP* admite los siguientes sistemas de archivo:

- **FAT32**: se trata de una versión mejorada del sistema de la tabla de asignación de archivos (*FAT*) estándar en los anteriores sistemas operativos de la familia *Windows*, a partir de las versiones posteriores (32 bits) de *Windows 95*. El sistema *FAT32* puede utilizarse en los discos duros grandes, desde 512 MBytes a 32 GBytes.

- NTFS: el sistema de archivos de *Windows NT* (*NTFS*) es utilizado en los sistemas operativos *Windows NT* y *Windows 2000*. El sistema de archivos *NTFS* proporciona una estabilidad y seguridad mejoradas, y admite discos duros grandes de hasta 2 TBytes.

El cambio de *FAT32* a *NTFS* es de un sólo sentido, es decir que si cambiamos el sistema de archivo *FAT*, o *FAT32*, a *NTFS* no podremos cambiar, más adelante, el sistema de archivos a *FAT32*.

Si no estamos seguros del sistema de archivos que debemos utilizar, es útil elegir aquél que el sistema nos ofrece como predeterminado durante la instalación. Si deseamos cambiarlo, deberemos tener en cuenta las siguientes indicaciones:

- Utilizaremos *FAT32* si el disco duro es menor de 32 GBytes.
- Utilizaremos *FAT32* si deseamos instalar más de un sistema operativo en el mismo equipo.
- Utilizaremos *NTFS* si la unidad de disco duro es mayor de 32 GBytes y sólo ejecutaremos un sistema operativo en el equipo.
- Utilizaremos *NTFS* si deseamos obtener una seguridad mejorada.
- Utilizaremos NTFS si necesitamos una mejor compresión de disco.

Durante el proceso de instalación, también podemos crear distintas particiones de disco para organizar la información, por ejemplo para realizar una copia de seguridad de los datos, o instalar más de un sistema operativo en el equipo. Un mismo disco duro puede contener un máximo de cuatro particiones.

18.10.6 El Panel de control

El *Panel de control* es la ventana principal para realizar las modificaciones en todo el sistema, desde las opciones de accesibilidad a los distintos perfiles de usuario.

En *Windows XP*, la casa Microsoft ha cambiado la ubicación algunas características, gracias a la experiencia previa de *Windows 2000*. La mayoría de estos cambios han dado lugar a nuevos subprogramas en el *Panel de control*.

> Nota: el sistema de ayuda del *Panel de control* nos ayudará a encontrar elementos aparentemente desaparecidos de los antiguos modelos del *Panel de control*.

Una gran parte de las opciones de configuración que podemos controlar desde el *Panel de control* no son esenciales para un funcionamiento seguro del sistema. Muchos de los cambios que se pueden realizar a través de los subprogramas del *Panel de control* son mejoras de la interfaz y no están relacionados con la funcionalidad del sistema operativo.

Las distintas opciones de configuración establecidas a través de los subprogramas del *Panel de contro*l serán guardadas en el *Registro*. Algunas de estas opciones hacen refe-

rencia a todo el sistema, mientras que otras se establecen para cada usuario y se activan cuando éste inicia una sesión con su nombre.

Algunas opciones de configuración del *Panel de control* pueden modificarse desde otras ubicaciones de *Windows XP*. Aunque las rutas de acceso a los distintos subprogramas puedan ser diferentes, los resultados de la configuración son los mismos.

> **Nota:** independientemente de la ruta de acceso, normalmente se termina ejecutando una de las extensiones del *Panel de control* (archivos *CPL*) para realizar la ejecución de su propia orden.

Las opciones de configuración que afectan a todo el sistema, como agregar o eliminar dispositivos de hardware, se controlan desde el monitor de seguridad, y requieren los permisos de usuario adecuados para poder realizar estas modificaciones.

Abrir el Panel de control

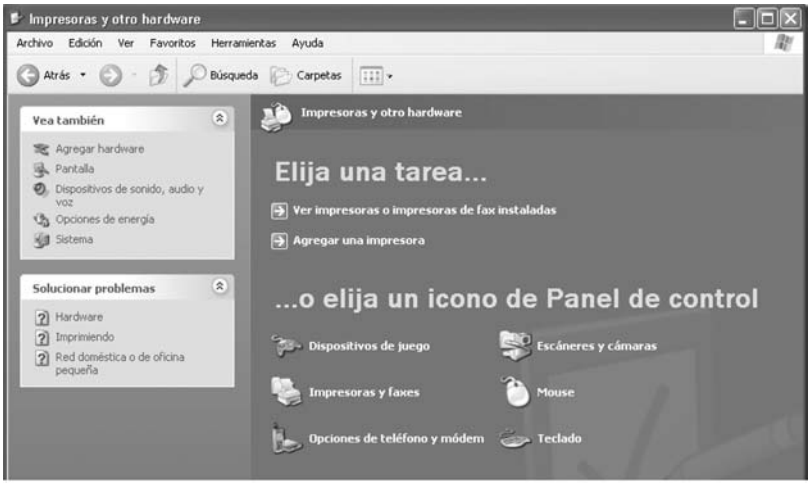
La forma más habitual de acceder a la ventana del *Panel de control* es a través de la opción *Panel de control* del menú *Inicio.* También podemos usar el vínculo de *Panel de contro*l en el menú rápido de acceso *Otros sitios* de la vista de carpetas del *Explorador de Windows*.

El nuevo Panel de control

El *Panel de control* en *Windows XP* se muestra de forma predeterminada en una vista ordenada por categorías. La vista por categorías organiza, en grupos, las funciones más habituales de los distintos subprogramas del Panel de control a las que tenemos acceso. Si nunca hemos configurado un sistema operativo de la familia Windows, la vista ordenada por categorías ofrece una guía lógica para encontrar la ubicación correcta de las opciones para la realización de los cambios previstos.

La vista por categorías del Panel de control

Si pulsamos en un icono en la vista por categorías, se abre otra ventana del *Panel de control* que contiene una clasificación más específica de las distintas tareas que podemos seleccionar, o un cuadro de diálogo correspondiente a la categoría seleccionada en el que podremos establecer las configuraciones deseadas.



Las distintas tareas de la opción Impresoras y otro hardware del Panel de control

En la vista por categorías del *Panel de control* encontramos nueve opciones distintas entre las que escoger. Como podemos ver, la vista por categorías nos ofrece una ruta alternativa a los distintos subprogramas del *Panel de control*. Por defecto, *Windows XP* escoge la vista por categorías para la presentación del *Panel de control*.

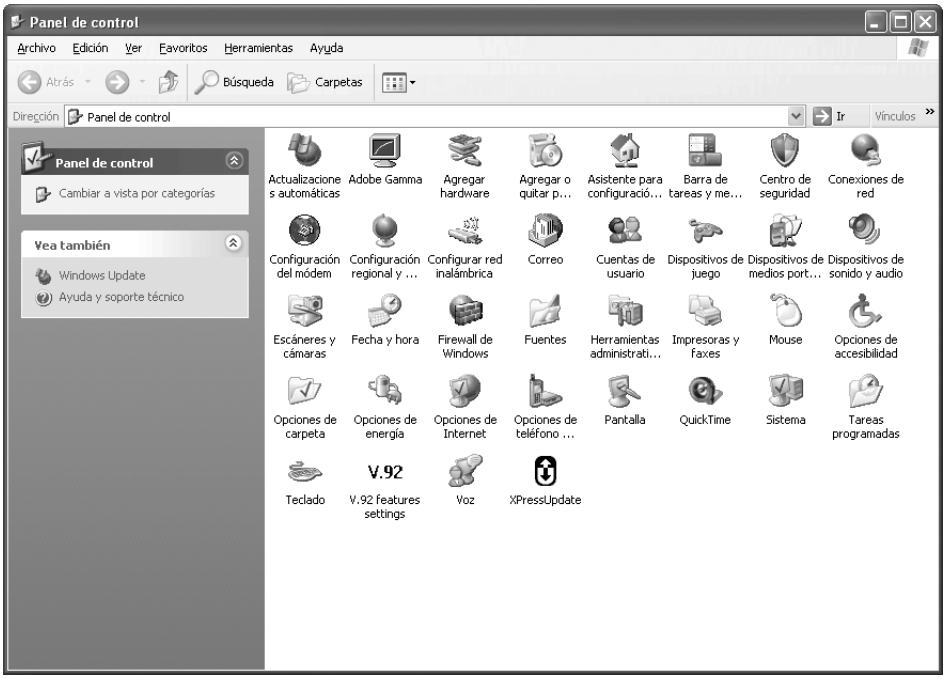
Para los usuarios que prefieran la antigua apariencia del *Panel de control*, *Windows XP* dispone de una vista clásica donde aparecen todos los subprogramas del *Panel de control*, tal y como aparecían en las versiones anteriores de *Windows*.

Para tener acceso a la vista clásica, deberemos pulsar en la opción *Cambiar a Vista clásica* del panel izquierdo de la ventana, en la sección *Panel de control*. Para volver a la vista por categorías, deberemos pulsar en la opción *Cambiar a vista por categorías*.

¿Qué vista debemos utilizar?

El trabajo con la vista por categorías del *Panel de control* puede simplificarnos el acceso a las opciones de configuración más habituales. Sin embargo, no todas las opciones para el manejo de *Windows XP* se pueden encontrar directamente en la vista por categorías. Es por ello, que para disponer de una descripción de los subprogramas, describiremos cada subprograma según la presentación en la vista clásica.

A continuación, enumeramos una lista de todos los subprogramas estándares del *Panel de control*, y para qué se utilizan. El *Panel de control* puede incluir otros subprogramas que se instalan junto con otros productos.



La vista clásica del Panel de control

Opciones de accesibilidad: este subprograma permite configurar el teclado, el ratón, el sonido, la pantalla y otras opciones para facilitar la utilización de los componentes del sistema a los usuarios con problemas visuales, auditivos o motores.

Agregar hardware: instala o soluciona los distintos problemas de una gran variedad de dispositivos de hardware, tales como el *CD-ROM*, los controladores del disco duro, controladores *SCSI*, adaptadores de pantalla, teclado, ratón, etc.

Agregar o quitar programas: permite instalar, desinstalar y modificar aplicaciones o componentes del sistema *Windows XP*.

Herramientas administrativas: proporciona distintos métodos de acceso directo a las herramientas administrativas como, por ejemplo, *Servicios de componentes*, *Administración de equipos*, *Orígenes de datos (ODBC)*, *Visor de sucesos*, *Rendimiento* y *Servicios*.

Fecha y hora: permite definir la fecha, la hora y la zona horaria para el equipo. También puede sincronizar la hora del sistema con un servidor horario de *Internet*.

Pantalla: permite definir las distintas opciones de configuración de varias partes de los elementos de presentación de *Windows XP*, así como otros ajustes de pantalla, como pueden ser el fondo del escritorio, el protector de pantalla, el controlador de video, la calidad del color y la resolución, la frecuencia de actualización, el ahorro de energía, etc.

Opciones de carpeta: permite definir las opciones de carpeta globales del sistema y las asociaciones de archivos.

Fuentes: permite agregar y eliminar los tipos de letra del sistema. También muestra ejemplos de los tipos de letra para pantalla e impresora.

Dispositivos de juego: permite agregar, eliminar y configurar los distintos dispositivos para jugar en el ordenador, tales como joysticks y controladores para juegos.

Opciones de Internet: permite establecer las opciones de *Internet Explorer*.

Teclado: permite establecer la velocidad de repetición de las teclas, la velocidad de intermitencia del cursor, el idioma del teclado y los controladotes. También incluye asistentes para solucionar posibles problemas con el teclado.

Mouse: permite configurar las propiedades del ratón, tales como la velocidad del puntero, la velocidad, orientación y rastro del mismo, la velocidad de la doble pulsación y otras opciones de configuración que dependen del controlador del ratón instalado.

Conexiones de red: permite administrar todas las conexiones de red del sistema, incluyendo la red *LAN*, el acceso telefónico, *WAN* y *VPN*. En este apartado también se configuran los componentes de red (clientes, servicios y protocolos).

Opciones de teléfono y módem: permite agregar, eliminar y configurar las propiedades de los módems instalados en el sistema. A través de este subprograma podemos establecer las diferentes reglas de marcado (números de larga distancia, llamadas en espera, etc.).

Opciones de energía: dispone de las distintas opciones para instalar las funciones para la administración avanzada de energía (*APM, Advanced Power Management*) y para la configuración avanzada y administración de energía (*ACPM, Advanced Configuration and Power Management*). Este subprograma permite establecer la espera antes de apagar el monitor, discos, los recursos del sistema y la hibernación.

Impresoras y faxes: permite agregar, modificar, desinstalar y administrar los distintos dispositivos de impresora y de fax. El uso de este subprograma permite administrar la lista de los trabajos de impresión para cada impresora.

Configuración regional y de idioma: permite establecer cómo *Windows XP* debe mostrar la hora, la fecha, los números y las monedas, siguiendo la configuración regional y las preferencias del idioma.

Escáneres y cámaras: permite agregar, eliminar y definir las configuraciones de los escáneres y cámaras digitales instalados en el sistema.

Tareas programadas: permite programar la ejecución automática de distintas aplicaciones y utilidades del sistema.

Dispositivos de sonido y audio: permite asignar distintos sonidos a los sucesos del sistema. Además, permite la administración de los dispositivos de sonido.

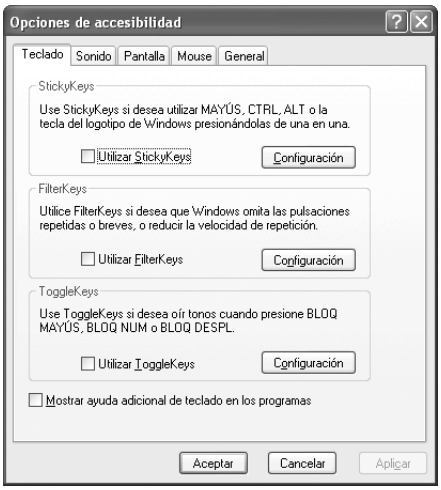
Voz: permite configurar las opciones de voz para la conversión de texto a voz.

Sistema: permite examinar y cambiar la identificación de los usuarios del sistema (grupo de trabajo, nombre de dominio, nombre del equipo), y examinar la configuración de los dispositivos instalados, la cantidad de memoria *RAM*, el procesador, etc. A través de este subprograma podemos agregar, anular y eliminar distintos dispositivos especiales con la ayuda del *Administrador de dispositivos*. También podemos optimizar algunos parámetros del sistema, tales como las variables de entorno y establecer las opciones de inicio en caso de emergencia.

> **Nota:** muchos de los distintos cuadros de dialogo de las distintas opciones del *Panel de control* disponen de un botón con un interrogante en la esquina superior derecha. Al pulsar este botón y, a continuación, pulsar en algún elemento del cuadro de diálogo, obtendremos más información sobre el elemento en cuestión.

Barra de tareas y menú Inicio: permite establecer las distintas propiedades para la barra de tareas y el menú *Inicio*.

Cuentas de usuario: permite agregar, eliminar o modificar las distintas cuentas de usuario instaladas en el sistema. A través de este subprograma podemos asignar grupos, administrar contraseñas y establecer el modo de inicio de sesión.



El cuadro de diálogo Opciones de accesibilidad y sus correspondientes fichas

Las opciones de accesibilidad

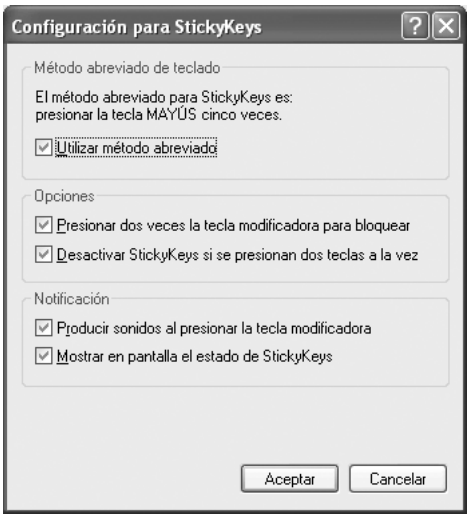
En los últimos años, Microsoft ha incluido más opciones de accesibilidad en el sistema operativo, con características que permiten utilizar *Windows* a los discapacitados. Muchos usuarios tienen dificultades para ver los caracteres en la pantalla, para escri-

bir en el teclado o para controlar el movimiento del ratón. Durante muchos años, los usuarios con parálisis parcial o que tienen problemas de coordinación muscular han tenido serios problemas para utilizar los ordenadores.

Ahora, con la ayuda de estas opciones de accesibilidad, las posibilidades de usar el equipo se han equilibrado entre los usuarios con y sin discapacidades. Las opciones de accesibilidad que ofrece *Windows XP* se pueden desglosar en distintas categorías, correspondientes con sus respectivas fichas: *Teclado*, *Sonido*, *Pantalla*, Mouse y *General*.

Configuración de la accesibilidad del teclado

Las distintas opciones de configuración del teclado controlan problemas como la repetición accidental de teclas o la pulsación de una combinación de teclas. Estas opciones se dividen en tres categorías: *StickyKeys*, *FilterKeys* y *ToggleKeys*. Las opciones de configuración del área *StickyKeys* son útiles para controlar la función de las teclas <Alt>, <CtrI> y <Mayús> si se tienen problemas para pulsar dos teclas a la vez. Para usar esta función, activaremos la casilla de verificación *Utilizar StickyKeys*. A continuación, pulsaremos en el botón *Configuración* para definir las distintas opciones que consideremos necesarias.



Las distintas opciones de configuración de la función StickyKeys

Nota: para la mayoría de usuarios, el método abreviado de pulsar la tecla <Mayús> cinco veces es un buen método para activar la función *StickyKeys*. Si habilitamos este método de activación, podemos comprobar cómo la pulsación de la tecla <Mayús> otras cinco veces desactiva la función, y así sucesivamente.

El área *FilterKeys* permite eliminar las pulsaciones repetitivas de una misma tecla por si tenemos cualquier problema para pulsar una tecla una sola vez y liberarla. Esta función evita tener que realizar muchas pulsaciones de tecla. El método abreviado para

activar y desactivar esta función consiste en mantener pulsada la tecla <Mayús> derecha durante ocho segundos. La función *ToggleKeys*, cuando está activada, hace sonar un tono agudo al realizar una pulsación de las teclas <Bloq Mayús>, <Bloq Despl> y <Bloq Num> y un tono grave al desactivar nuevamente esta función.

Las opciones de accesibilidad del sonido

Las dos opciones de accesibilidad de la ficha de sonido, *SoundSentry* y *ShowSounds*, están indicadas para usuarios con problemas auditivos. En lugar de reproducir un sonido cuando se produce un suceso sonoro, en pantalla aparecerá algún tipo de advertencia visual.

A través de la opción *SoundSentry*, una sección de la pantalla se vuelve intermitente, normalmente la ventana, o aplicación, que provoca el error. Con la opción *ShowSounds* activada aparece un subtitulo, o un icono especial, cuando se reproduce un sonido. La información de la ventana emergente informará del sonido reproducido, error, etcétera.

Las opciones de accesibilidad de la pantalla

El uso de la función de uso de un alto contraste es un modo sencillo de instalar la combinación de color de la pantalla y la selección de fuentes para leer la información de forma más cómoda. Esta opción resulta muy útil para los ojos cansados o en situaciones de iluminación poco favorables.

Las opciones de accesibilidad del ratón

A través de la ficha *Mouse* podemos controlar el ratón con el teclado numérico si tenemos problemas con el control de los movimientos del ratón. Esta opción nos puede ser muy útil cuando el ratón se bloquea por algún motivo. Cuando utilicemos esta opción de accesibilidad, las teclas del teclado numérico con flechas tienen una doble función, al servir también como teclas de control del puntero. Pulsaremos en el botón *Configuración* si necesitamos definir las opciones de configuración de velocidad para las teclas con flechas.

> **Nota:** si personalizamos la configuración de la opción *MouseKeys*, en el cuadro de diálogo *Configuración* deberemos pulsar *Aceptar* y, a continuación, pulsar el botón Aplicar antes de que los cambios surtan efecto.

Las otras opciones de accesibilidad

La ficha *General*, se divide en las siguientes secciones:

Reinicio automático: si varios usuarios utilizan el mismo equipo, es muy recomendable desactivar las características de accesibilidad si no se utilizan durante un espacio de tiempo. Si desactiva la opción correspondiente, las opciones de accesibilidad permanecen operativas hasta que las desactivemos manualmente.

Notificación: estas opciones determinan cómo se nos avisa de que una característica de accesibilidad se ha activado o desactivado. De forma predeterminada, aparece un pequeño cuadro de diálogo, pero podemos cambiarlo por un sonido.

Dispositivos SerialKey: también podemos optar por utilizar dispositivos especialmente diseñados para discapacitados. Simplemente debemos conectar el dispositivo a un puerto serie libre y elegir el puerto (del *COM1* al *COM4*) y la velocidad en baudios.

Opciones administrativas: la primera opción de esta área hace referencia a las distintas opciones de configuración de accesibilidad para el escritorio del inicio de la sesión. Si activamos la segunda opción, las opciones de configuración actuales se copian para cada nuevo usuario que el administrador agregue al equipo.

La instalación de hardware

El subprograma *Agregar hardware* lo utilizaremos para instalar nuevos dispositivos de hardware, así como para solucionar los problemas relacionados con el hardware del equipo. La capacidad para desactivar o desinstalar dispositivos de hardware ha sido dejada para el *Administrador de dispositivos*.

Generalmente, *Windows XP* detecta el nuevo hardware durante el inicio del sistema. Si no se encuentran los controladores de forma automática (lo que ocurre sólo con los dispositivos que no son *Plug & Play*), se nos pedirá la ubicación del controlador necesario. Una vez encontrados los controladores, se instalan y se activa el dispositivo. En bastantes casos, es necesario reiniciar el sistema para que el nuevo dispositivo de hardware funcione correctamente.

> **Nota:** desgraciadamente, los controladores de la mayoría de los dispositivos de hardware requieren que se reinicie *Windows XP* después de su instalación.

Deberemos utilizar el subprograma *Agregar hardware* cuando el fabricante no ofrece un programa de instalación o cuando el dispositivo instalado no es detectado automáticamente durante el inicio del sistema. Si se dan estas condiciones, podemos utilizar el subprograma *Agregar hardware* para instalar los controladores para los nuevos dispositivos.

Cuando se ejecuta el subprograma por primera vez, éste intenta encontrar el nuevo hardware realizando una exploración del sistema, en busca de dispositivos desconocidos. Si durante la exploración no se encuentra ningún dispositivo, se nos pregunta si el hardware ya está conectado o todavía no. Si seleccionamos que el hardware todavía no está conectado, el asistente nos informa que debemos conectar el hardware para continuar con la instalación.

> **Nota:** es muy recomendable leer siempre el manual y las instrucciones que incluye el fabricante para la instalación del dispositivo. Si no se proporciona nada de esto, es recomendable visitar el sitio *Web* del fabricante.

También es recomendable cerrar cualquier aplicación en el momento de realizar instalaciones o actualizaciones de controladores. Es posible que un controlador nuevo bloquee el sistema, aunque no es muy normal en *Windows XP*.

Para el hardware que no es *Plug & Play* o para dispositivos Plug & Play que, por alguna razón, no sean detectados automáticamente, es necesario ejecutar el subprograma *Agregar hardware*. La forma más habitual de llevarlo a cabo es la siguiente:

1. Ejecutamos el subprograma *Agregar hardware*.
2. En el cuadro de diálogo de bienvenida debemos pulsar en el botón *Siguiente*. De forma automática se realiza una búsqueda de hardware nuevo en el sistema.
3. Si el asistente no encuentra ningún dispositivo nuevo, se nos preguntará si el dispositivo de hardware está conectado al equipo. Si no está conectado, se nos pedirá que conectemos el hardware antes de continuar con la instalación de los controladores.
4. A continuación aparece una lista del hardware instalado. Para instalar algún dispositivo nuevo, nos desplazaremos hacia abajo y seleccionaremos la opción *Agregar un nuevo dispositivo de hardware*. En caso contrario, para solucionar problemas seleccionaremos un dispositivo ya instalado. Para continuar, pulsaremos en el botón *Siguiente*.
5. Si elegimos solucionar los problemas de un dispositivo existente, *Windows XP* nos muestra un informe del estado del dispositivo y un mensaje para pulsar el botón *Finalizar* para poder ejecutar el *Solucionador de problemas*.
6. Si seleccionamos la opción Agregar un nuevo dispositivo de hardware, se nos preguntará dónde debe buscar el controlador del dispositivo.
7. Es muy poco probable que la segunda búsqueda automática detecte el nuevo dispositivo. En la mayoría de los casos, deberemos de proceder con el método manual.
8. En el método manual, deberemos seleccionar un tipo general de hardware o el elemento *Mostrar todos los dispositivos*. A continuación deberemos pulsar en *Siguiente*.
9. Acto seguido aparece una lista de fabricantes y de modelos de dispositivos. Si aparece el producto, lo seleccionamos y después, pulsamos en *Siguiente*. Si no aparece en la lista pulsaremos en el botón *Utilizar disco*.
10. El botón *Utilizar disco* abre un cuadro del diálogo donde deberemos escribir la ruta de acceso a los controladores del nuevo dispositivo.
11. A continuación, es recomendable ir aceptando las indicaciones mientras aparecen, porque cada tipo de dispositivo tiene unos requisitos distintos. Algunos no necesitan ninguna configuración adicional, mientras que otros requieren definir los puertos, las interrupciones u otros parámetros. También es posible que tengamos que usar el *CD-ROM* de *Windows XP*.

Nota: si el asistente detecta un conflicto en la configuración del dispositivo, nos avisará cuando éste finalice. Una solución puede ser retroceder y elegir un modelo de hardware distinto.

La instalación y desinstalación de programas

Cada vez son más las aplicaciones que proporcionan su propio programa de desinstalación, que acostumbra a aparecer como un icono dentro de su carpeta en el menú Inicio. En muy pocas ocasiones instalaremos programas as través del subprograma *Agregar o quitar programas*. Lo más habitual es usar este subprograma para desinstalar aplicaciones cuando el fabricante no proporcione una herramienta para ello.

La desinstalación de programas

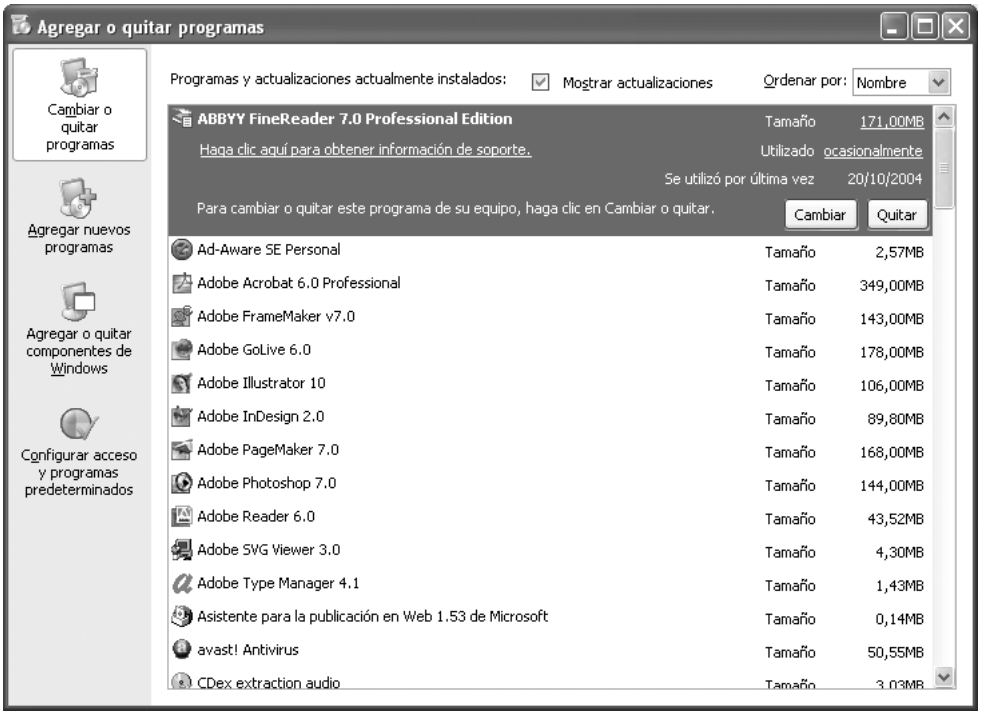
No todos los programas instalados en el sistema aparecen en la ventana del subprograma *Agregar o quitar programas*. No aparecen porque en la base de datos del sistema sólo se registran los nombres de los archivos y las ubicaciones de los programas que cumplen con el estándar *API* de *Windows* para la instalación, lo que permite borrarlos sin afectar de forma negativa al funcionamiento de *Windows XP*. Muchas aplicaciones antiguas, o las menos sofisticadas, simplemente se instalan por si mininas y no se registran en el sistema operativo.

La mayoría de las aplicaciones actuales están escritas según los estándares de *Microsoft Windows* para facilitar su instalación y eliminación. La lista de programas que aparecen en la ventana *Agregar o quitar programas* es la respuesta de la industria de software a las quejas y críticas de los usuarios acerca de los programas que, una vez instalados, son difíciles de eliminar del disco duro.

Esta problemática fue la causa de la aparición en el mercado de programas tales como *Uninstaller*, *CleanSweep* y otras utilidades que supervisan y guardan una base de datos de los archivos que instala un programa. Estas utilidades borran estos archivos de forma eficaz cuando decidimos eliminar un programa, dejando el sistema con las opciones de configuración establecidas antes de la instalación.

Incluso si una aplicación no se instala a través del subprograma *Agregar o quitar programas* debería darse a conocer al sistema y registrar los cambios que realice en el msimo, permitiendo la desinstalación desde este subprograma del *Panel de control*. Y no sólo eso, el desinstalador integrado en *Windows XP* permite realizar cambios en las aplicaciones instaladas, tales como agregar o eliminar componentes. El uso de la característica de desinstalación del subprograma es sencilla:

1. Ejecutaremos el subprograma *Agregar o quitar programas*.
2. Examinaremos la lista de aplicaciones instaladas. Podemos ordenar las aplicaciones siguiendo algún criterio interesante en el cuadro de ordenación.
3. Seleccionamos el programa que deseemos desinstalar.
4. Pulsamos en el botón *Cambiar o quitar programas*.



La ventana Agregar o quitar programas

Nota: Algunas aplicaciones (como *Microsoft Office*) pueden pedirnos que insertemos el *CD-ROM* del programa cuando intentemos cambiar la configuración o eliminar las aplicaciones.

La instalación de nuevos programas

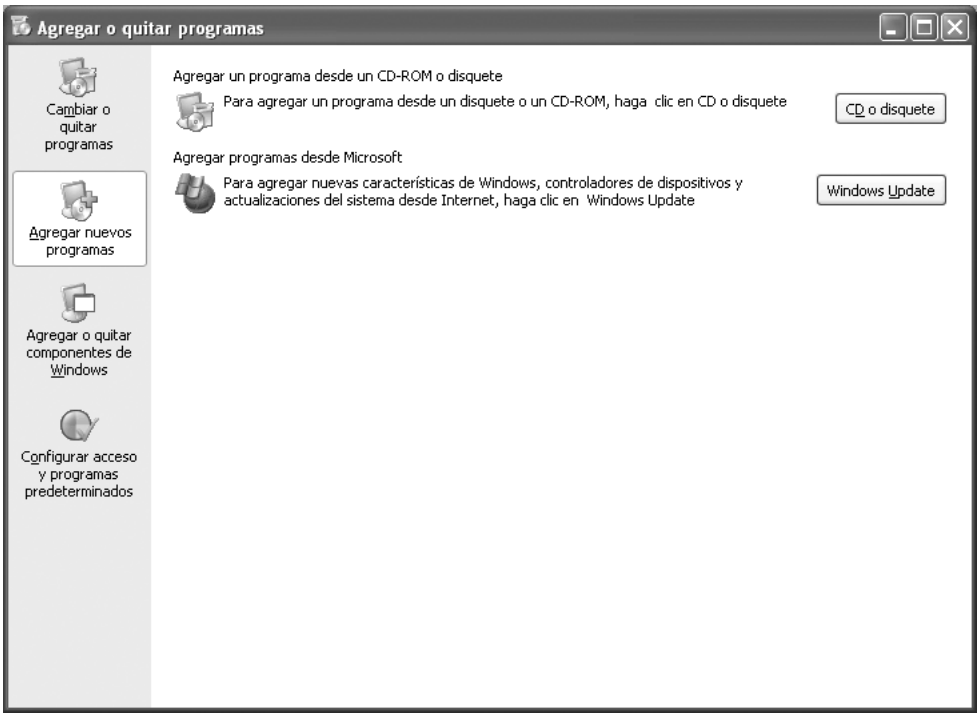
Normalmente, la instalación de nuevos programas es tan sencilla como insertar un *CD-ROM* en la unidad correspondiente. El programa de reproducción automática de la mayoría de los *CD-ROM* ejecuta el programa de instalación. Lo mismo pasa con los programas descargados de *Internet*.

Si lo deseamos, también podemos instalar los programas desde el subprograma *Agregar o quitar programas*. Esta parte del subprograma ofrece una presentación para ejecutar un programa de instalación. Esto es así para recabar datos, para su posterior desinstalación, de las aplicaciones que de otra manera no se registran correctamente en el sistema. Para ello:

1. Pulsamos en el botón *Agregar nuevos programas* del panel izquierdo de la ventana *Agregar o quitar programas*.
2. En este punto debemos seleccionar una de las dos opciones:

   CD o disquete: elegiremos esta opción para abrir la carpeta que contiene la rutina de instalación.

Windows Update: esta opción ejecuta *Internet Explorer*, conecta con el sitio de Microsoft y ejecuta *Windows Update*. Esta opción sirve par actualizar la instalación de *Windows XP*.



Las dos opciones para la instalación de un programa

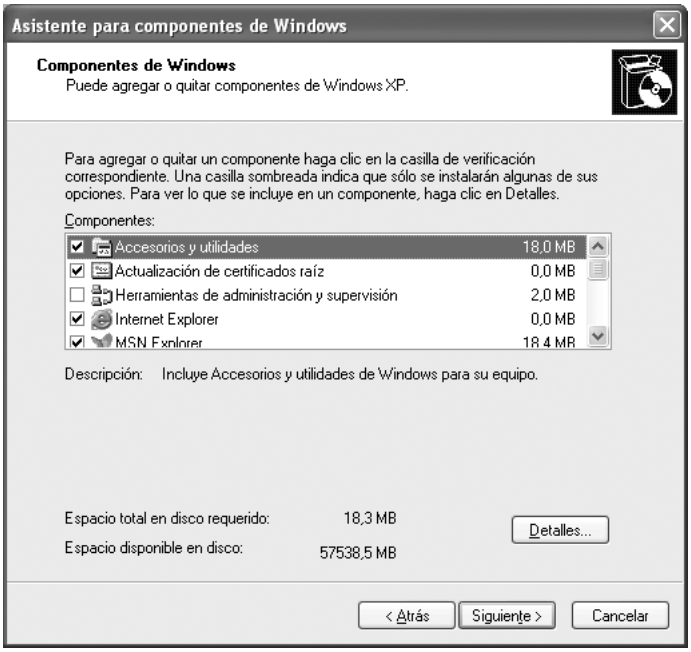
3. El asistente busca cualquier archivo denominado setup.exe o instalar.exe. Síiel programa de instalación que está buscando no se denomina así, es necesario conocer su nombre. Es muy importante elegir un archivo que sea de instalación.
4. Para terminar deberemos pulsar en *Finalizar* para completar la tarea y que se ejecute la rutina de instalación. Las instrucciones del proceso de instalación varían en función del programa. Después de la correcta instalación, el nuevo programa aparece en la lista de programas que se pueden desinstalar sólo si es compatible con la utilidad *Agregar o quitar programas de Windows XP*.

Los componentes de Windows XP

Además de poder gestionar aplicaciones y productos complementarios a través del subprograma *Agregar o quitar programas*, los componentes de *Windows XP* también se instalan y se quitan aquí. Para ello, deberemos pulsar en el botón *Agregar o quitar componentes de Windows* para abrir el cuadro de diálogo *Asistente para componentes de Windows*.

Desde este cuadro de diálogo podemos instalar los componentes adicionales de *Windows XP* activando sus correspondientes casillas de verificación. Podemos desinstalar componentes ya existentes desactivando sus correspondientes casillas de verificación.

> **Nota:** en este cuadro de diálogo debemos tener cuidado, pues hay muchos subcategorías de selección. Pulsando en el nombre de un elemento, no en su casilla de verificación, para su selección y pulsando el botón *Detalles* podemos ver los componentes secundarios de la categoría seleccionada. Activando o desactivando una categoría del nivel superior, instalamos o quitamos todos sus componentes secundarios.



El Asistente para componentes de Windows

Una vez seleccionadas las opciones correspondientes y tras pulsar en *Siguiente*, el sistema instalará o quitará los componentes deseados. Es muy probable que tengamos que insertar el *CD-ROM* de *Windows XP* en la unidad de *CDs*.

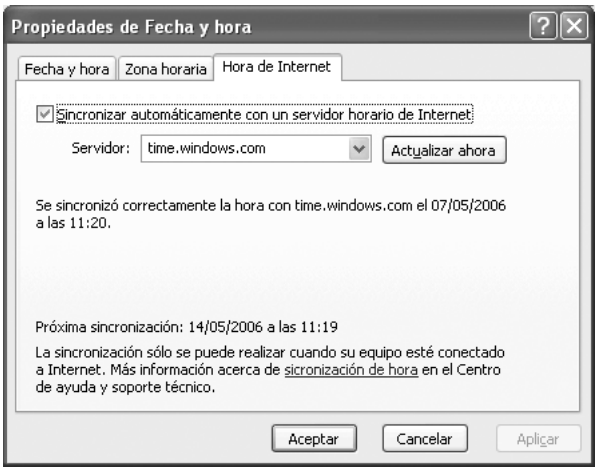
La fecha y la hora del sistema

La opción *Fecha y hora*, del *Panel de control*, es un subprograma sencillo que sirve para ajustar la fecha y la hora del sistema.

> **Nota**: al ajustar el reloj del sistema, lo que se hace es ajustar el reloj del hardware del equipo, mantenido por una pila en la placa base.

Si somos miembros de un dominio de red en *Windows XP*, no será necesario ajustar el reloj, ya que este se mantiene sincronizado con el administrador del dominio (un servidor de *Windows 2000, Windows NT* o *Windows .NET*). Esto es debido a que muchos servicios de red, incluyendo los protocolos de autenticación y de replicación, requieren la sincronización exacta, o cercana, de todos los miembros de la red.

Si el sistema es independiente, podemos sincronizar los relojes con un servidor horario de Internet. El subprograma *Fecha y hora* incluye una ficha nueva para ello. La opción de sincronización horaria con un servidor horario de *Internet*, a través del subprograma *Fecha y hora*, está reservada a los miembros del mismo grupo de trabajo, los sistemas independientes y los administradores de dominio.

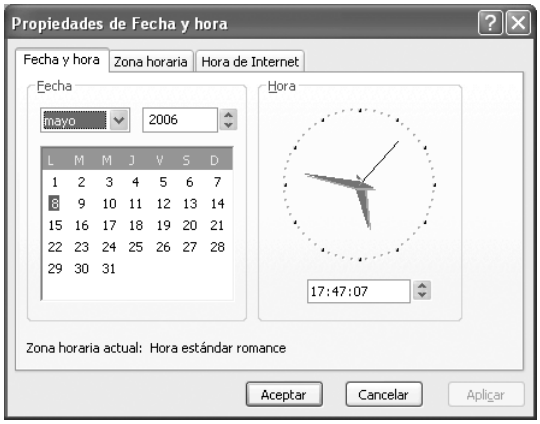


La ficha del la ventana Fecha y hora para la sincronización horaria

Para configurar la fecha y la hora del sistema, seguiremos los pasos siguientes:

1. Abriremos el subprograma *Fecha y hora*.
2. Modificaremos la hora y la fecha escribiendo lo que deseemos en los cuadro de texto o pulsando en las flechas. Pulsaremos directamente sobre las horas, minutos o segundos antes de que las flechas pequeñas situadas a la derecha modifiquen el valor. Después de establecer el mes y el año, podemos pulsar en el día correcto en el calendario que aparece.
3. En la ficha *Zona horaria* deberemos definir la zona. Si queremos que el reloj del equipo se ajuste de forma automática con la luz solar evitando tener que introducir los cambios horarios, activaremos la casilla de verificación de la opción *Cambiar la hora automáticamente según el horario de verano*.
4. En la ficha *Hora de Internet* podemos activar la sincronización del reloj con un servidor horario de *Internet*. La lista desplegable incluye dos servidores horarios conocidos, pero podemos agregar otros. Para realizar una sincronización, pulsaremos en el botón *Actualizar ahora*.

5. Pulsaremos en *Aceptar* para guardar los cambios y cerrar el subprograma *Fecha y hora*.

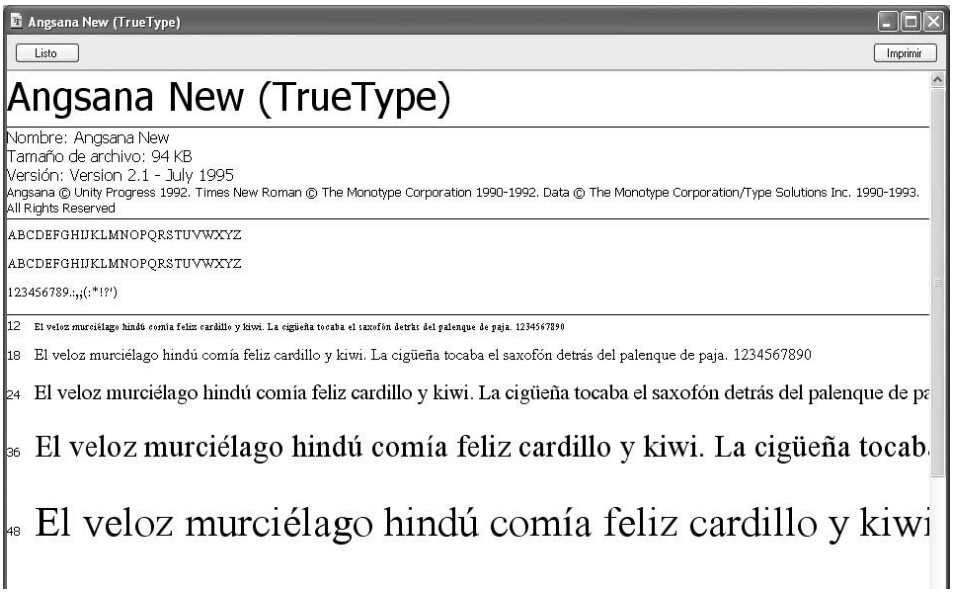


La hora y la fecha del sistema

Al activar la sincronización con *Internet*, el reloj se adapta a los servidores horarios una vez cada semana. La sincronización con *Internet* debe configurarse en sistemas con una conexión permanente con *Internet*.

Las fuentes

Con el subprograma *Fuentes* del *Panel de control* podemos ver, agregar y eliminar fuentes del sistema. Para ver una fuente, simplemente pulsaremos dos veces en el nombre de la fuente en la ventan del subprograma *Fuentes*.



El Visor de fuentes

El *Visor de fuentes* nos mostrará los detalles de la fuente, su juego de caracteres y varios tamaños de los mismos. La mejor forma de agregar una fuente al sistema es a través de la opción de menú *Archivo/Instalar nueva fuente*. Quitar una fuente es más sencillo. Basta con seleccionar una, o varias fuentes, en la ventana del subprograma *Fuentes* y pulsar la tecla <Supr>.

> **Nota**: existen varias fuentes del sistema importantes y necesarias para los cuadros de diálogo del sistema y otras funciones, que nunca deben eliminarse. Éstas son *Courier*, *MS Sans*, *MS Sans Serif*, *Small Fonts* y *Symbol*.

Los dispositivos de juego

Si utilizamos el equipo para divertirnos jugando, es posible que necesitemos un dispositivo de juego. Los dispositivos más habituales son los joystick, los controladores de vuelo, volantes u otros dispositivos de hardware diseñados específicamente para los juegos. Por ejemplo, los fanáticos de los juegos de carreras no disfrutan lo suficiente hasta que no consiguen un volante simulador de fuerzas con su propio conjunto de pedales para el acelerador y el freno.

Si compramos un dispositivo de juego, probablemente incluirá un programa de instalación. De lo contrario, *Windows XP* puede detectarlo automáticamente o necesitaremos ejecutar el subprograma *Agregar hardware*. Si estos pasos no funcionan, intentaremos agregarlo con la ayuda del subprograma *Dispositivos de juego*.

> **Nota**: en la mayoría de los casos, los dispositivos de juego *USB* se instalan de forma muy sencilla. Basta con conectar el dispositivo para que todo vaya sobre ruedas.

Durante los últimos años ha habido una cierta migración a la plataforma de *Windows 2000* para juegos, pero *Windows XP* está mejor diseñado para jugar y, seguramente, tendrá más suerte. *Windows XP* es compatible con *DirectX 8.0*, incluidos los controladores de la tarjeta aceleradora de video y de la tarjeta de sonido que proporcionan una mejor reproducción para el video, los gráficos a todo color y para las animaciones en tres dimensiones.

DirectX determina de forma automática las características del hardware del equipo y, a continuación, define los parámetros de los programas para que éstos coincidan. Esto permite que las aplicaciones multimedia se ejecuten en cualquier equipo basado en *Windows* y, al mismo tiempo, asegura que estas aplicaciones multimedia se beneficien completamente del hardware de alto rendimiento. Las funciones de bajo nivel de *DireetX 8.0* son compatibles con los componentes que forman la capa *DirectX Foundation*, que son los siguientes:

- *DirectDraw*.
- *Direct3D*.
- *DirectSound*.

- *DirectMusic*.
- *DirectInput*.
- *DirectPIay*.
- *DirectShow*.

Para los amantes de los juegos resultan especialmente interesantes *DirectDraw* (que proporciona un acceso rápido a las características de aceleración de hardware de una tarjeta de vídeo), *DirectInput* (para procesar rápidamente la entrada de los dispositivos de juego), *Direct3D* (compatible con gráficos tridimensionales en tiempo real) y *DirectPIay* (que admite conexiones de juegos a través del módem, de *Internet* o de una red de área local).

Después de instalar un dispositivo de juego, podemos pulsar en el botón *Opciones avanzadas* para modificar el identificador del dispositivo y el puerto al que se conecta. A cada dispositivo de juegos le deberemos asignar un identificador diferente.

> **Nota**: para instalar un dispositivo personalizado deberemos pulsar en *Agregar* y, a continuación, una vez seleccionado el modelo pulsaremos en *Personalizar*. En este punto deberemos seguir las opciones de configuración para el tipo de dispositivo, los ejes y el número de botones. A continuación, le asignamos un nombre al dispositivo.

El teclado

El subprograma *Teclado* permite configurar el comportamiento del teclado, comprobar el controlador del mismo y solucionar los posibles problemas.

Las opciones del subprograma Teclado

Las principales opciones de configuración de este cuadro de diálogo son la velocidad de repetición, el retraso de la misma y la velocidad de intermitencia del cursor. También podemos modificar la velocidad de intermitencia del cursor si el cursor intermitente nos molesta por alguna razón. Los valores predeterminados para estas opciones del teclado son adecuados para la mayoría de usuarios y teclados.

> Nota: la ficha Idioma ha sido eliminada de este subprograma. Para cambiar la configuración de idioma, deberemos utilizar el subprograma *Configuración regional y de idioma*.

El ratón

Al ser el ratón el medio principal de interacción con el equipo, es necesario optimizar su funcionamiento. El subprograma *Mouse* del Panel de control nos permite ajustar muchos de los aspectos del funcionamiento del ratón, tales como:

La configuración de las propiedades del ratón

- La configuración de los botones derecho e izquierdo.
- La velocidad de la doble pulsación.
- El bloqueo de la pulsación.
- El aspecto de los punteros.
- El esquema del puntero.
- La velocidad del puntero.
- Mejorar la precisión del puntero.
- Mover el puntero al botón predeterminado en un cuadro de diálogo.

- Mostrar el rastro del puntero.
- Ocultar el puntero mientras se escribe.
- Mostrar la ubicación del puntero al pulsar la tecla <Ctrl>.
- Hacer que el deslizamiento de la rueda avance o retroceda un número determinado de líneas o de pantallas.
- Solucionar posibles problemas.

Muchas de las opciones varían dependiendo del tipo de dispositivo de señalización instalado y, a veces, se proporcionan incluso opciones más específicas si el dispositivo de señalización incluye un controlador personalizado.

Si somos zurdos, ahora podemos mover el ratón a la parte izquierda del teclado y, a continuación, invertir la función de los botones usando el subprograma *Mouse*. Las pulsaciones del botón secundario se convierten entonces en pulsaciones del botón primario.

En la misma ficha *Botones*, podemos definir la velocidad de la doble pulsación. Una configuración media suele ser la adecuada para la mayoría de los usuarios. Para probar la velocidad de la doble pulsación pulsaremos en la caja de sorpresas. La caja se abre o se cierra si la doble pulsación fue correcta.

El cursor del puntero cambia según la tarea que se está realizando. Podemos modificar los cursores para mejorar la visibilidad. Podemos instalar cursores animados para entretenernos mientras se espera a que finalice algún proceso.

> **Nota**: para comprobar las opciones de configuración del control deslizante de velocidad no tenemos que pulsar en *Aplicar*. Con el movimiento del control deslizante se ve afectada la velocidad de la doble pulsación del ratón.

Windows XP incluye un número suficiente de cursores para satisfacer al usuario, organizados en esquemas. Los esquemas de cursores son colecciones de formas de cursores. Cuando seleccionamos un esquema, todos los cursores del esquema se activan a la vez.

Las opciones de marcado y módem

El subprograma *Opciones de teléfono y módem* nos ofrece la posibilidad de agregar, eliminar y configurar las propiedades de los módems instalados en el sistema. También podemos definir las normas de marcación, tales como los números de larga distancia, el intervalo de espera entre llamadas, las llamadas con tarjeta de crédito, etc.

La interfaz de telefonía

El subprograma *Opciones de teléfono y módem* del *Panel de control* pone a nuestra disposición una ubicación central para modificar algunas de las opciones de configuración, así como instalar y configurar los dispositivos de teléfono y módem. En gene-

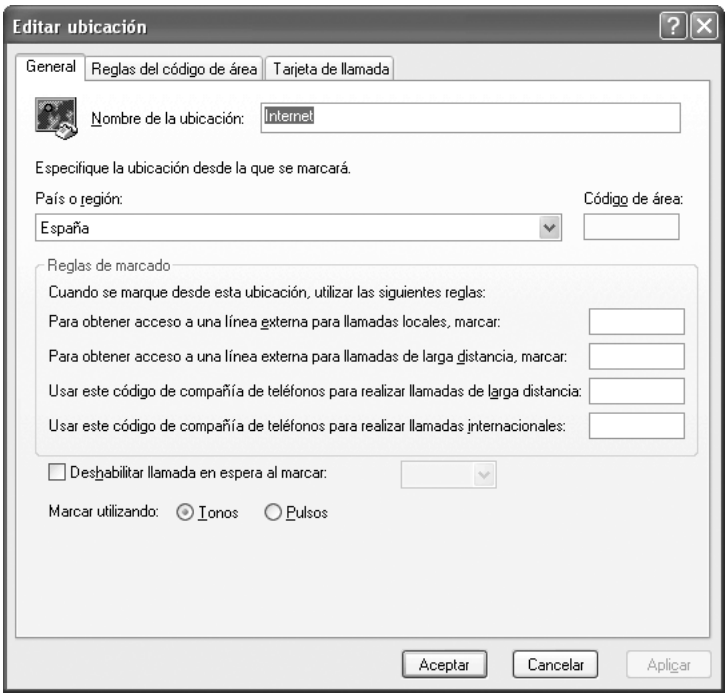
ral, las propiedades predeterminadas del módem funcionan correctamente y no será necesario modificarlas. Si necesitamos cambiarlas, deberemos recordar que los cambios afectan a todas las aplicaciones que utilizan el módem.

Para cambiar las propiedades del módem una vez finalizada la instalación del mismo, abriremos el *Panel de control* y realizaremos una doble pulsación en el icono *Opciones de teléfono y módem*.

> **Nota**: *Windows XP* es compatible con la marcación *PPP* multivínculo. Ésta marcación permite agrupar varios módems para aumentar el ancho de banda de la conexión. Este proceso es complicado y requiere varias líneas de teléfono y varias cuentas, así como un *ISP* que admita la sincronización de varios módems.

Las reglas de marcado

En la ficha *Reglas de marcado*, del cuadro de diálogo *Opciones de teléfono y módem*, podemos configurar las ubicaciones de marcado y las normas que afectan a estas ubicaciones, como los prefijos para las líneas externas, los códigos de acceso a las tarjetas de llamada, etc. Podemos modificar o agregar una ubicación desde esta ficha. Las opciones son las siguientes:



Las distintas opciones de configuración

Nombre de la ubicación: en este campo especificamos el nombre de la ubicación seleccionada para realizar la marcación.

País o región: este campo contiene un menú desplegable que muestra la lista de los códigos internacionales de marcación para la mayoría de los países del mundo.

Código de área: en este campo introducimos nuestro propio código de área.

Para tener acceso a una línea externa para llamadas locales: si la línea de módem se encuentra en una oficina donde es necesario marcar el 9 para acceder a una línea exterior, aquí escribiremos ese número.

Para tener acceso a una línea externa para llamadas de larga distancia: Si tenemos que marcar el 8 o el 9 para las llamadas de larga distancia, escribiremos aquí ese número.

Usar este código de compañía de teléfonos para realizar llamadas de larga distancia: si utilizamos uno de los servicios interprovinciales que requieren el marcado de un código de compañía de teléfonos, utilizaremos este campo para introducirlo.

Usar este código de compañía de teléfonos para realizar llamadas internacionales: si utilizamos uno de los servicios internacionales de llamada que requieren un código de compañía de teléfonos, utilizaremos este campo para su introducción.

Deshabilitar llamada en espera al marcar: si el servicio telefónico consta de la llamada en espera, puede ser molesto, y afectar negativamente a la conexión de datos. La mayoría de los servicios de llamada en espera permiten desconectar el servicio para la llamada actual marcando un código numérico antes de llamar. Si disponemos del servicio de llamada en espera, debemos activar esta opción y escribir el código de su compañía de teléfonos.

Marcar utilizando Tonos o Pulsos: la mayoría de los teléfonos de botones utilizan la marcación por tonos. Sin embargo, los antiguos teléfonos de disco y algunos teléfonos de botones utilizan la marcación por pulsos. Cambiaremos la opción a *Pulsos* si no se establece una conexión.

Las reglas de código de área

En la ficha *Reglas de código de área* podemos definir los detalles sobre el uso de un código de área, específicamente el uso del prefijo 1 para ciertos intercambios. Si tenemos que marcar 1 para ciertas áreas, podemos agregar ese prefijo aquí.

Las propiedades del módem

La segunda ficha del cuadro de diálogo *Opciones de teléfono y módem* incluye las propiedades del módem. Al acceder a las propiedades de un módem se abre el mismo cuadro del diálogo que al tener acceso a las propiedades de un dispositivo a través del *Administrador de dispositivos*.

Normalmente, no deberemos cambiar las propiedades del módem. Para acceder a las propiedades del módem seguiremos los pasos siguientes:

1. Pulsaremos en la ficha *Módems*.
2. Seleccionaremos, con una pulsación del ratón, el módem en la lista de módems instalados en el equipo.
3. Pulsaremos el botón *Propiedades*.

A continuación enumeramos algunas de las opciones más importantes de configuración del módem:

Puerto: utilizaremos el menú desplegable *Puerto* para especificar el puerto *COM* o *USB* al que está conectado el módem.

Volumen del altavoz: el control del volumen del altavoz se configura mediante un control deslizante que define la intensidad del altavoz del módem.

Velocidad máxima del puerto: cuando el módem establece una conexión, intenta utilizar la máxima velocidad para el intercambio de datos con el módem del otro lado de la línea. La velocidad máxima debería ser de tres o cuatro veces la velocidad estimada del módem.

Control de marcado: elegimos si el marcador debe esperar a detectar un tono de marcado antes de continuar.

Comandos de inicialización adicionales: la sección *Configuraciones adicionales* es un lugar para enviar comandos *AT* adicionales al módem. En la mayoría de los casos no es necesario agregar ningún comando especial.

Protocolos de datos: si utilizamos un teléfono celular con el módem, seleccionaremos la opción *Protocolo celular* en la lista desplegable *Protocolo de datos*. Los teléfonos celulares utilizan protocolos especiales de compresión y corrección de errores para aumentar la velocidad de transmisión.

Cambiar las preferencias predeterminadas: los bits de datos, de paridad y de parada deben ser iguales en ambos extremos de una conexión. La configuración más habitual es de 8 bits de datos y ningún bit de parada. Esto se define en la ficha *Opciones avanzadas*. En la ficha *General* encontramos las preferencias de desconexión de la línea cuando está inactiva y el tiempo que transcurre sin respuesta antes de que se cancele la llamada.

Configuración de puerto avanzada: pulsaremos este botón para abrir el cuadro de diálogo del mismo nombre. Estas opciones determinan cómo se almacenan las llamadas entrantes y salientes en el búfer, mediante *UARTS* del puerto *COM*.

El diagnóstico

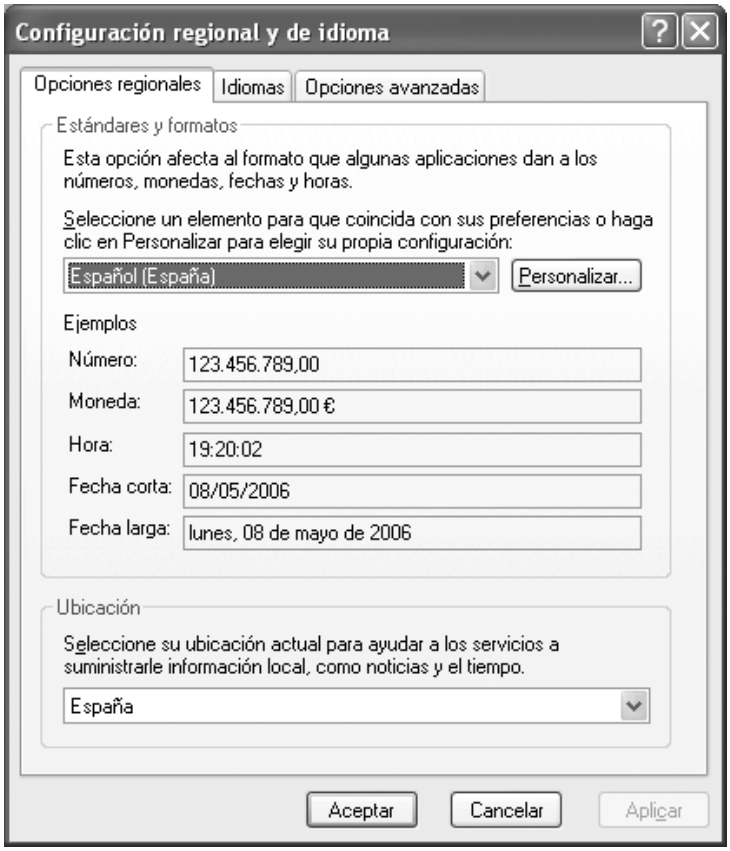
Al activar el diagnóstico desde el cuadro de diálogo *Propiedades de Módem*, la ficha *Diagnóstico* le solicita al módem que se identifique. Después prueba la capacidad del módem para responder al grupo estándar de comandos *AT*, para mostrar el contenido del registro interno y mostrar las opciones de configuración.

Los resultados de la pregunta del diagnóstico tienen sentido si se comparan con los resultados previstos del manual del módem. Si no coinciden, deberemos buscar la información para la solución de problemas en el manual, o ponernos en contacto con el proveedor para que nos proporcione las opciones específicas de reparación.

La configuración regional

Las opciones de *Configuración regional y de idioma* afectan a la presentación de la hora, la fecha, los números y la moneda en *Windows XP*. En el momento de la instalación de *Windows XP* ya hemos seleccionado las opciones regionales. La ejecución de este subprograma del *Panel de control* abre el cuadro de diálogo *Configuración regional y de idioma*.

Para cambiar la configuración, pulsaremos en la ficha correcta y, después, seleccionaremos el cuadro de lista desplegable de la opción deseada. En cada sección aparecen ejemplos de la configuración actual. Los estándares predefinidos están organizados por idioma y, después, por país. Si no logramos encontrar un estándar adecuado a nuestras preferencias, siempre podremos crear un formato personalizado.



El cuadro de diálogo Configuración regional y de idiomas

Los escáneres y cámaras digitales

A través del subprograma *Escáneres y cámaras* podemos agregar, quitar, configurar y solucionar problemas para los escáneres y cámaras digitales. Dado que últimamente se van equiparando a las impresoras de toda la vida, es necesario estar preparado para facilitar la transferencia de documentos e imágenes desde estos dispositivos al equipo. Con la llegada de las cámaras digitales surgen muchos métodos apropiados para facilitar la transferencia de imágenes capturadas.

El subprograma *Escáneres y cámaras* también se usa para instalar cámaras de vídeo digitales y otros dispositivos de captura de imágenes. Una vez instalado un dispositivo, el subprograma *Escáneres y cámaras* puede vincularlo a un programa instalado en el equipo. La detección del escáner, o de la cámara digital, se producirá automáticamente cuando la función *Plug & Play* de *Windows XP* detecte el dispositivo.

Sin embargo, como ya es habitual, algunas veces es necesario utilizar el *Asistente para agregar hardware*. Para ello, utilizaremos este subprograma para instalar un escáner o una cámara digital sin problemas:

1. Ejecutamos el subprograma *Escáneres y cámaras* y pulsamos en la opción *Agregar un dispositivo de imagen*. El dispositivo debería estar conectado antes de este paso. Seguimos las indicaciones del asistente. Seleccionaremos la marca y el modelo si fuese necesario.
2. Seleccionamos el puerto al que está conectado el dispositivo. Es posible usar la opción *Detección de puerto automática* si deseamos no equivocarnos de puerto.
3. Al instalarse un nuevo elemento, el controlador se agrega a la lista de arranque y se modifican las funciones necesarias del sistema para permitir el acceso al dispositivo.

Podemos comprobar el estado de un escáner o de una cámara digital a través del botón *Propiedades*. En el cuadro de diálogo *Propiedades* del dispositivo podemos modificar el número de puerto si es necesario, así como decidir acerca de otras opciones de configuración. Si existen perfiles de colores disponibles para el dispositivo, se pueden agregar o quitar a través de la ficha *Administración del color*.

Los dispositivos de sonido y audio

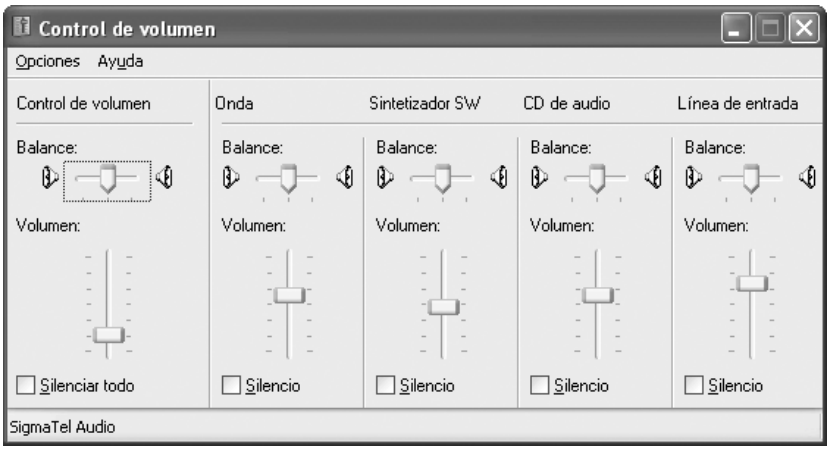
A través del subprograma *Dispositivos de sonido y audio* podemos configurar el sistema de sonido y los distintos sonidos del equipo. Podemos aprovechar mucho las opciones de este subprograma si se es un amante del sonido. Casi todos los usuarios utilizan este subprograma para cambiar los sonidos que reproduce el equipo como respuesta a ciertos eventos del sistema.

> Nota: los sonidos que utiliza el sistema se almacenan en formato *WAV*. Es posible crear o descargar cualquier sonido que se pueda imaginar desde *Internet*.

Las opciones para los sonidos del equipo

La ficha Volumen

La ficha *Volumen* se utiliza para establecer el volumen principal del equipo. Éste es el mismo control de volumen que aparece cuando se pulsa el altavoz de la bandeja del sistema. Pulsaremos el botón *Propiedades avanzadas* para abrir el cuadro de diálogo *Control de volumen,* desde donde tenemos acceso a todos los controles de entrada y salida de sonido.



El cuadro de diálogo Control de volumen

La ficha *Volumen* también incluye los controles del altavoz. Si disponemos de un sistema especial de altavoces, como sonido envolvente o en 3D, estos controles ayudan a definir la configuración.

La ficha Sonidos

La ficha *Sonidos* sirve para asociar los eventos de *Windows* a los distintos sonidos. *Windows XP* incluye una gran cantidad de archivos de sonido. En esta ficha podemos crear y guardar combinaciones personalizadas de sonido. A pesar de la diversidad de sonidos de la colección, podemos utilizar los sonidos grabados con la *Grabadora de sonidos*. Para asignar sonidos a los eventos específicos de *Windows* seguiremos los pasos siguientes:

1. Ejecutamos el subprograma *Dispositivos de sonido y audio* y seleccionamos la ficha *Sonidos*.
2. La sección *Sucesos de programa* nos muestra una lista con los eventos del sistema que pueden llevar sonidos asociados. Un evento con un icono de altavoz junto a él ya tiene un sonido asociado. Pulsaremos el botón reproducir (el que tiene un triángulo) para escuchar el sonido. El archivo de sonido asociado a este suceso aparece en una lista, en el cuadro de opciones *Sonidos*.

Nota: los archivos *WAV* que utilicemos para los sonidos del sistema no deberían ser demasiado grandes. Los archivos de sonidos pueden ser muy grandes, especialmente si se graba en estéreo de 16 bits. Como norma general, deberíamos mantener el tamaño mínimo para los sonidos del sistema, dado que un archivo muy grande requiere unos segundos para descargarse y reproducirse.

3. Pulsaremos en el suceso al que queremos asignar un sonido o cambiar el sonido ya asignado.
4. Abrimos la lista desplegable *Sonidos* y elegimos el archivo *WAV* que deseamos utilizar para el suceso.

El subprograma *Dispositivos de sonido y audio* también permite guardar combinaciones de sonidos. Es normal cansarse de una combinación de sonidos, de modo que cambiarlos es algo comprensible. Las combinaciones que ofrece *Windows XP* son bastante buenas y considerando el esfuerzo necesario para crear combinaciones propias, lo mejor es probar los esquemas predefinidos. Para seleccionar un esquema de sonido existente, utilizaremos la lista desplegable *Combinación de sonidos*.

Nota: puede parecer que algunos de los nombres de sucesos no tienen sentido, como *Asterisco*, *Parada crítica* o *Exclamación*. Estos nombres son para las distintas clases de cuadros de diálogo que *Windows XP* muestra de vez en cuando. Los sonidos que escucharemos más frecuentemente son *Abrir programa*, *Cerrar programa*, *Minimizar* y *Maximizar*.

Las fichas Voz y Audio

En las fichas *Audio* y *Voz* podemos declarar el hardware predeterminado que deseamos utilizar para la reproducción del sonido, grabación del sonido, reproducción de música *MIDI*, reproducción de voz y grabación de voz.

La ficha Audio

La mayoría de los sistemas ofrecen opciones mínimas en estos aspectos porque los equipos normales tienen un único sistema de sonido. Los botones *Opciones avanzadas* pueden ser útiles en función del conjunto de chips del sistema de sonido.

La ficha Voz

Algunos ofrecen opciones para el ajuste de bajos y agudos; el estéreo envolvente (un tipo de sonido de mayor amplitud basado en el ajuste de la fase de la señal que llega

al amplificador); las opciones de conversión o la velocidad de muestreo; la optimización de la ecualización en función del tipo de altavoces y la aceleración del hardware.

La casilla de verificación *Usar sólo dispositivos predeterminados* determina la tarjeta de sonido que utilizarán los programas. Si utiliza programas que requieren un tipo específico de tarjeta de sonido y esa tarjeta de sonido está seleccionada en esta ficha, activamos esta casilla de verificación. De esta manera, si por alguna razón los dispositivos preferidos no están disponibles, el programa no insiste intentando utilizar una tarjeta de sonido que *Windows XP* considera como una sustitución razonable.

La ficha Hardware

La ficha *Hardware* del cuadro de diálogo *Propiedades de Dispositivos de sonido y audio* nos muestra una lista de todo el hardware de sonido, vídeo, *DVD* y elementos multimedia instalados en el sistema. Podemos comprobar las propiedades, el estado actual y solucionar los posibles problemas.

El Administrador de dispositivos

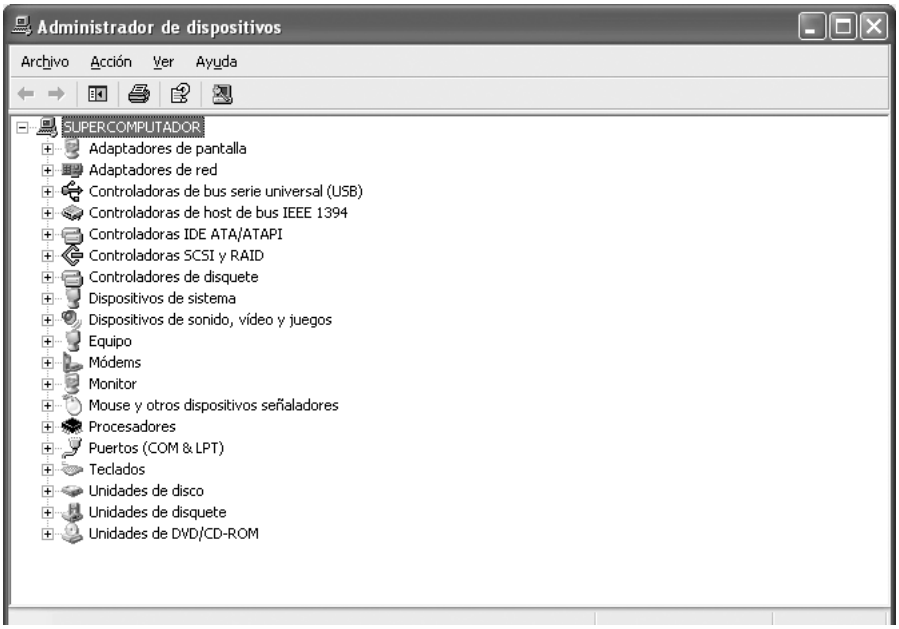
El subprograma *Sistema* contiene la utilidad Administrador de dispositivos. El *Administrador de dispositivos* se abre desde la ficha *Hardware* del subprograma *Sistema*. Al iniciarse, aparece una lista ordenada por categorías de los distintos dispositivos instalados en el sistema. Para ver los dispositivos de cada categoría, debemos ampliar las opciones de muestreo a través del signo <+>. A continuación y para abrir el cuadro del diálogo *Propiedades* de un dispositivo, basta con realizar una doble pulsación.

El *Administrador de dispositivos* sirve para varias funciones. La principal es ayudar en la resolución de los problemas con el hardware. Cuando cualquier dispositivo deja de funcionar según lo esperado, aparece resaltado con un triángulo amarillo o una señal roja de stop. El triángulo amarillo indica una advertencia o un posible problema. Una señal roja de stop indica un conflicto del dispositivo. Cuando ejecutamos el *Administrador de dispositivos* y un dispositivo tiene un problema especial, su categoría se amplía para poder ver fácilmente el icono de advertencia o de error.

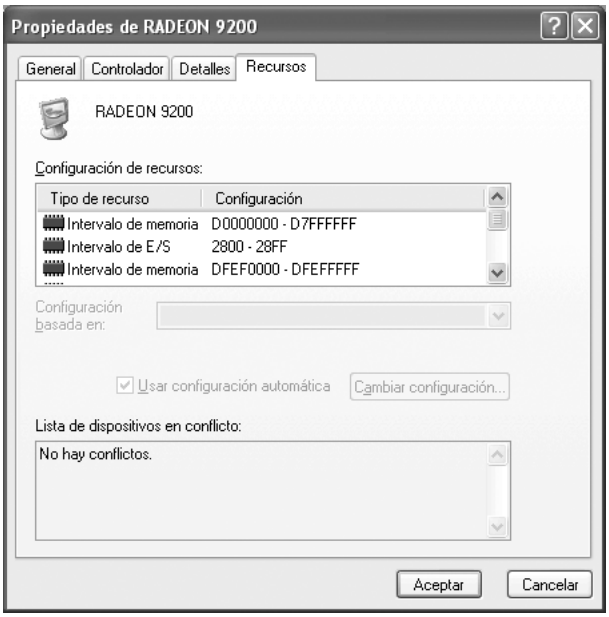
Cuando se abre el cuadro de diálogo *Propiedades* de un dispositivo, la ficha *General* muestra la información básica sobre el dispositivo y detalles sobre su estado actual. Para solucionar un posible error en ese momento pulsaremos en el botón *Solucionador de problemas* para utilizar un asistente.

Dependiendo del dispositivo, pueden aparecer otras fichas en el cuadro de diálogo *Propiedades*. Casi siempre aparece la ficha Controlador, que suele ir acompañada de una ficha *Recursos*. La ficha *Controlador* ofrece detalles sobre el controlador instalado actualmente para el dispositivo y permite actualizar, volver al controlador anterior o desinstalarlo. La opción *Volver al controlador anterior* es una característica nueva de *Windows XP*.

La capacidad de quitar el controlador actual para volver al controlador anterior puede ser una bendición para solucionar posibles problemas de compatibilidad. Muchas veces es necesario quitar un controlador nuevo y, después, volver a instalar el hardware para recuperar el controlador anterior.



El Administrador de dispositivos



La ficha Recursos

La ficha *Recursos* especifica los recursos del sistema a los cuales está asignado el dispositivo. Estos incluyen las interrupciones, los intervalos de *E/S*, *DMA*, etc. Para muchas tarjetas *Plug & Play*, la configuración de la ficha *Recursos* es de sólo lectura.

A partir de la vista principal del *Administrador de dispositivos* podemos realizar las siguientes acciones:

- Cambiar la vista entre dispositivos por tipo, dispositivos por conexión, recursos por tipo y recursos por conexión.
- Forzar una exploración para buscar cambios de hardware.
- Actualizar el controlador para el dispositivo seleccionado.
- Desactivar el dispositivo en el perfil actual de hardware.
- Desinstalar el dispositivo del sistema.
- Expulsar o desconectar un dispositivo.

18.10.7 La actualización de hardware en Windows XP

En este apartado trataremos los aspectos relacionados con la instalación y actualización del hardware en *Windows XP*.

> **Nota**: lo más útil que podemos hacer para mejorar el funcionamiento de un equipo con *Windows XP* es instalar mucha memoria *RAM*. *Windows XP* necesita más memoria *RAM* que cualquier versión anterior de *Windows*. Una ampliación de la memoria *RAM* hasta, por lo menos, 256 MBytes supone una gran diferencia de rendimiento.

La configuración del BIOS

El buen funcionamiento de *Windows XP* también depende de una configuración apropiada del *BIOS* para detectar y utilizar el hardware de forma correcta. *Windows XP* se inicia de forma mucho más rápida que las otras versiones de *Windows*. Para mejorar aún más la velocidad de inicio podemos aplicar la siguiente sugerencia.

> **Nota**: configurar el orden de inicio del *BIOS* con la unidad de disco duro que contenga el sistema operativo, omitiendo el arranque desde un disquete o un *CD-ROM*.

La actualización del disco duro

Una de las mejoras más eficaces para un equipo es conseguir un disco duro más rápido y de mayor tamaño. Los discos duros *SCSI* han sido siempre mejores que los *IDE*, pero la nueva generación de discos duros *EIDE UDMA* son más rápidos y más económicos que los *SCSI*.

Un bus *IDE* admite cuatro unidades (dos por canal) y suele estar integrado en la placa base. La instalación de una unidad *CD-RW* o *DVD-ROM* ocupa una de las ranuras, permitiendo un máximo de tres unidades de disco duro *IDE*.

A continuación, enumeramos las consideraciones más fundamentales para realizar la actualización del disco duro del sistema:

- No instalar una unidad de disco duro y una unidad de *CD-ROM* en el mismo canal, a menos que sea necesario. En algunos equipos, el canal *IDE* negocia la velocidad de transmisión con el dispositivo más lento del canal, ralentizando la velocidad de transferencia efectiva del disco duro. Designaremos la unidad de disco duro que contiene *Windows XP* como la unidad maestra principal.
- Desfragmentar el disco duro con la utilidad *Desfragmentador de disco*. Esto permitirá reorganizar los elementos de su disco duro para que los programas se ejecuten de forma más rápida.
- Actualizar el controlador del disco duro (mejor dicho, del adaptador *host*). Debemos comprobar que disponemos de un controlador que pueda aprovechar al máximo el hardware. Si estamos utilizando un disco duro *IDE*, el adaptador host esta integrado en la placa base (siempre que usemos un sistema basado en un procesador *Pentium* o superior).
- Instalar una unidad de disco más rápida: *UDMA/33* o más rápida para *IDE*; *Ultra2Wide* o más rápida para SCSI. Las unidades *IDE* que giran a 7.200 rpm son ya bastante habituales, y su mayor velocidad permite que los datos se transfieran a una mayor velocidad.

Nota: si los discos duros admiten *UDMA/33* (*Ultra DMA*) o modos más rápidos, pero la placa base sólo admite los modos más lentos *PIO*, deberemos instalar un adaptador *host UDMA* o cambiar la placa base. La mayoría de las placas base modernas admiten *UDMA*.

La ampliación de la memoria RAM

La actualización más asequible para el equipo es la ampliación de la memoria *RAM*. Aunque Microsoft indica que *Windows XP* puede ejecutarse con tan sólo 64 MBytes de memoria *RAM*, lo cierto es que sólo funciona bien con 128 MBytes, aunque 256 MBytes son mucho mejor para el rendimiento general del sistema.

Windows XP reconoce, de forma automática, la memoria *RAM* agregada y adapta la configuración interna para aprovechar al máximo la ampliación de la memoria.

Nota: deberemos comprar el máximo de memoria *RAM*, especialmente si el sistema utiliza módulos *DIMM* de *SDRAM* o *DDR SDRAM*. Los precios de la memoria fluctúan constantemente, pero actualmente los módulos no son nada caros. Por lo tanto, es una actualización muy rentable y adecuada para el buen funcionamiento del sistema.

Si ejecutamos aplicaciones que requieren una gran cantidad de memoria y mucho espacio en disco, como pueden ser la digitalización de imágenes, el tratamiento de las mismas, la edición de vídeo o de bases de datos de gran tamaño, deberíamos ampliar

la memoria a partir de 256 MBytes. *Windows XP* no sufre la limitación de 512 MBytes de *Windows 9x* y *Windows ME*, y puede llegar hasta los 1.5 GB de memoria *RAM*.

Agregar nuevo hardware

Una de las tareas más comunes es agregar nuevo hardware y quitar el viejo. Para ello, el *Panel de control* dispone de la utilidad llamada *Agregar hardware*. Deberemos utilizarla si el sistema operativo no reconoce, automáticamente, que hemos eliminado o agregado algún componente nuevo, ya sea un elemento periférico como una impresora, o un dispositivo interno como un *DVD-ROM*, un disco duro adicional, etc.

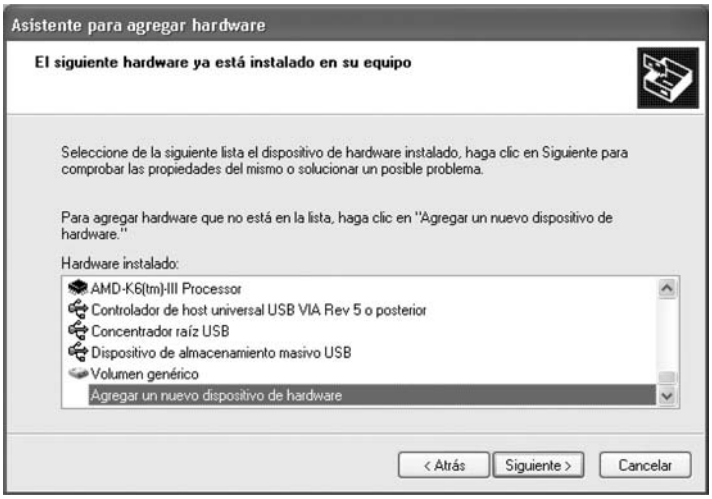
Utilizaremos frecuentemente este subprograma si trabajamos con hardware que no es *Plug & Play*. La instalación del hardware *Plug & Play* no es muy complicada, ya que *Windows XP* lo detecta sin demasiadas complicaciones e instala los controladores necesarios que le indican cómo tener acceso al nuevo hardware.

> **Nota**: utilizaremos el *Administrador de dispositivos*, y no la utilidad *Agregar hardware*, para ajustar la configuración de los dispositivos ya instalados en el equipo. Utilizaremos la opción *Agregar hardware* sólo para solucionar problemas con el hardware ya instalado. La manera más rápida de abrir cualquiera de estas dos utilidades pulsar la opción *Mi PC* del menú *Inicio* con el botón secundario del ratón y seleccionar la opción *Propiedades* del menú contextual. Desde la ficha *Hardware* podremos abrir el *Administrador de dispositivos*.

Si hemos adquirido una tarjeta u otro dispositivo de hardware, primero deberemos leer el manual que se incluye en la caja para obtener los detalles acerca de los procedimientos de instalación. En la mayoría de los casos, el fabricante de hardware ya nos proporciona un programa de instalación. No obstante, si no se incluyen las instrucciones, podemos instalar el hardware físicamente y continuar leyendo los siguientes apartados del capítulo. Para proceder a la instalación, deberemos cerrar los programas que se estén ejecutando, como precaución para que el proceso de instalación no bloquee el equipo.

En el caso del hardware que no es *Plug & Play*, o para los dispositivos *Plug & Play* que no se detectan automáticamente, deberemos ejecutar el subprograma *Agregar hardware*. En un primer paso, el asistente comienza buscando nuevo hardware P*lug & Play*. Si no encuentra nada, *Windows XP* pregunta si el dispositivo está conectado. Si seleccionamos la opción *Sí, ya he conectado el hardware*. En este punto, *Windows XP* interpreta que existe algún problema con un dispositivo o que necesitamos instalarlo de forma manual. Los dispositivos ya instalados aparecen con una opción para solucionar posibles problemas o para agregar algo nuevo.

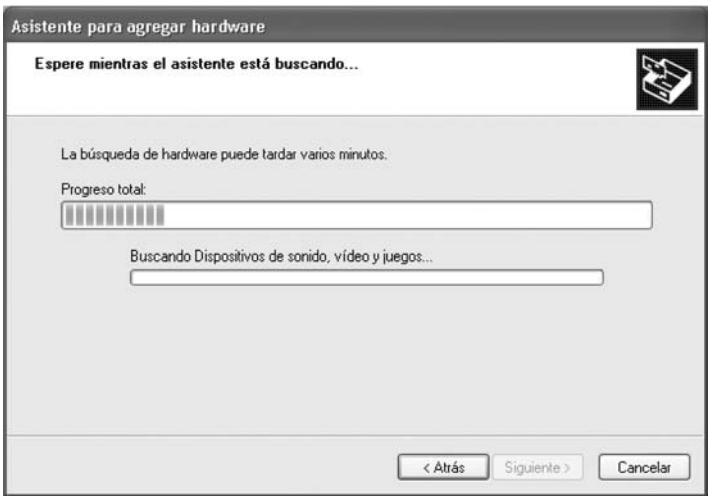
La opción que combina la posibilidad de agregar hardware nuevo o de solucionar los problemas del hardware instalado es algo extraña hasta que nos acostumbramos a ella. Si seleccionamos la opción *No, no he agregado el hardware todavía*, el asistente para la instalación del nuevo hardware se cierra.



La opción Agregar un nuevo dispositivo de hardware para los dispositivos que no son Plug & Play

En el siguiente paso, el asistente nos preguntará si deseamos que intente una detección e instalación automática del hardware, o si queremos especificar el dispositivo manualmente.

Si seleccionamos que intente una búsqueda automática, aparecerá un indicador que nos informa del progreso para cada categoría de hardware y del progreso total.



El progreso en la búsqueda automática de nuevos dispositivos

Cuando el asistente detecta un problema, nos avisará al finalizar. A continuación, podemos abandonar la instalación o seguir a pesar del problema. También podemos hacer una copia de seguridad y seleccionar un modelo de hardware diferente que consideremos compatible con el dispositivo que intentamos instalar.

Al detectar un problema de hardware, el asistente nos informa de ello al final del asistente

A partir de este punto ya podemos configurar manualmente los recursos de hardware de este dispositivo. Para ello, pulsaremos en la opción *Ver o cambiar recursos para este hardware (avanzado)* para cambiar la configuración establecida. n la siguiente ventana pulsaremos en el botón *Configurar manual*. Acto seguido aparece un cuadro de diálogo mostrando la configuración conflictiva y el otro dispositivo que utiliza la misma configuración.

Para solucionar el problema pulsaremos el botón *Cambiar configuración* para cambiar la configuración; si el sistema muestra un mensaje de error, utilizaremos el cuadro *Configuración basada en* para intentar otra configuración básica, o cambie los valores de la configuración manual que hemos definido. Una vez solucionado el conflicto pulsaremos el botón *Aceptar* y reiniciaremos el equipo.

Otras posibilidades

Cuando el hardware que intentamos instalar no aparece en la lista de dispositivos del asistente, el problema puede ser debido a uno de las siguientes posibilidades:

- El hardware es más nuevo que *Windows XP*.
- El hardware es realmente antiguo y Microsoft ha decidido no implementar compatibilidad para él en *Windows XP*.
- El hardware debe configurarse con la ayuda de un programa especial de instalación, como ocurre con algunas unidades extraíbles, como las unidades *ZIP* de la casa Iomega.

En estos casos, deberemos obtener el controlador del fabricante y guardarlo en un disco extraíble o en el disco duro. Si el fabricante provee un disco de instalación, olvídese de este paso, y siga las instrucciones del fabricante. Sin embargo, si el fabricante proporciona un disco de instalación sin instrucciones, haremos lo siguiente:

1. Ejecutaremos la utilidad *Agregar hardware* y pulsaremos en el correspondiente botón *Siguiente*.
2. Seleccionaremos la opción *Sí, ya he conectado el hardware* y volveremos a pulsar en *Siguiente*.
3. Nos desplazaremos hasta la parte inferior de la lista y seleccionaremos la opción *Agregar un nuevo dispositivo de hardware*. Volveremos a pulsar en el botón *Siguiente*.
4. Seleccionaremos la opción *Instalar el hardware seleccionándolo manualmente de una lista (avanzado)* y pulsaremos en *Siguiente*.
5. Seleccionaremos la categoría de dispositivo apropiada y pulsaremos en el botón *Siguiente*.
6. Pulsaremos en el botón *Utilizar disco*. Escribiremos la ubicación del controlador necesario para el dispositivo. Si utilizamos la opción *Examinar* para buscar la ubicación del controlador, buscaremos el directorio donde aparezca el archivo *INF* correspondiente.
7. Una vez tenemos el controlador, pulsaremos el botón *Aceptar* para finalizar la instalación del dispositivo.

La desinstalación de hardware

Uno de los cambios importante de *Windows 2000* a *Windows XP* es que el asistente para agregar nuevo hardware de *Windows XP* no se duplica en un asistente para quitar hardware, como pasa con el asistente para agregar o quitar hardware de *Windows 2000*. Para desinstalar un dispositivo de hardware, es necesario utilizar la sección *Administrador de dispositivos* del subprograma *Sistema*.

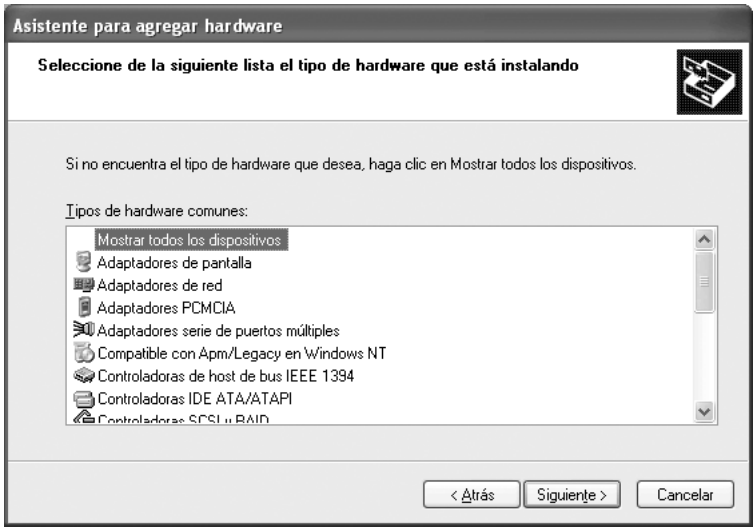
El subprograma Sistema

Una de las herramientas que, probablemente, utilizará con más frecuencia para administrar los distintos dispositivos de hardware del equipo es el subprograma *Sistema* del *Panel de control*.

Para abrir el subprograma *Sistema* seleccionaremos la opción *Rendimiento y mantenimiento* en el *Panel de control* y pulsaremos en la opción *Sistema*.

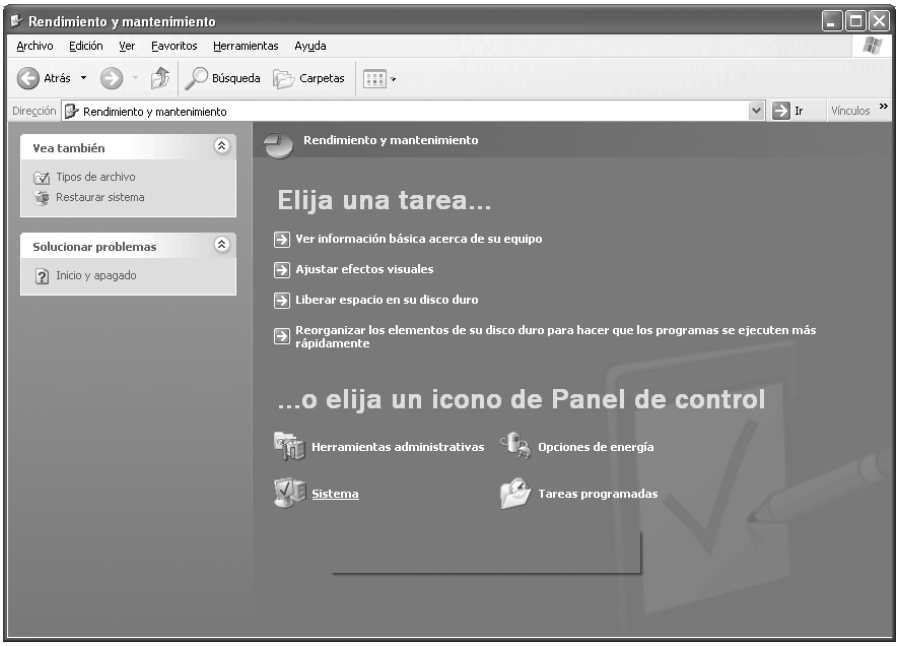
> Nota: si el asistente encuentra un dispositivo nuevo que no requiere ninguna configuración por nuestra parte, aparecerá una viñeta de ayuda cerca de la bandeja del sistema, ofreciéndonos la información adecuada del dispositivo que estamos instalando y el resultado de la instalación.

Si el dispositivo no se detecta correctamente, deberemos pulsar en el botón *Siguiente* y aparecerá una lista con los distintos tipos de dispositivos más comunes. Si no encontramos la categoría que incluya nuestro hardware, seleccionaremos la opción *Mostrar todos los dispositivos*. Para continuar con el proceso deberemos pulsar en *Siguiente*.



Los distintos tipos de dispositivos de hardware más comunes se enumeran por orden alfabético

Debemos comprobar que seleccionamos el nombre de la marca y el nombre del modelo exacto del dispositivo que estamos instalando. Es muy probable que debamos insertar el *CD-ROM* de *Windows XP* para que puedan cargarse los archivos del controlador apropiado. Si el hardware disponía un disco de instalación, utilizaremos el botón *Utilizar disco* para instalar directamente el controlador desde el disco de instalación del fabricante.

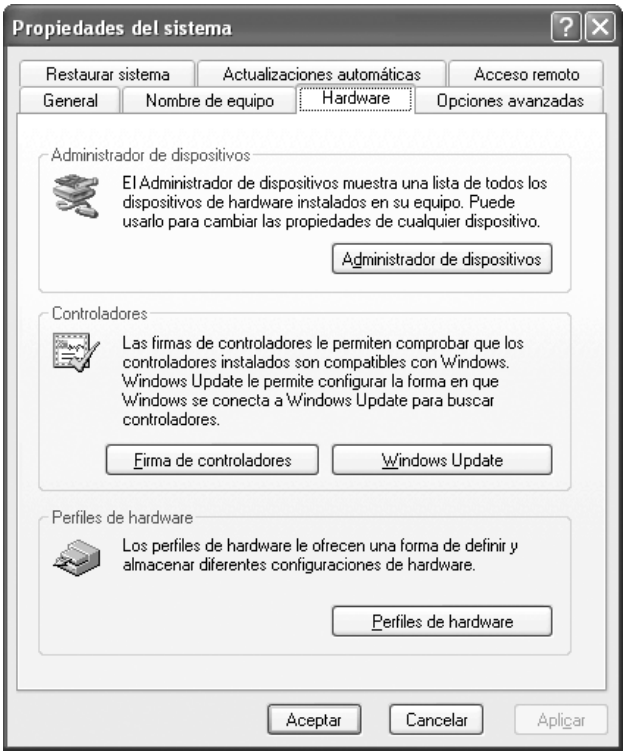


La opción Sistema del Panel de control

Nota: si el dispositivo debe conectarse en un puerto serie, paralelo o *SCSI*, es posible que debamos conectarlo y reiniciar el sistema para instalarlo. Algunos de estos dispositivos no se pueden instalar mediante el asistente para agregar nuevo hardware si no están conectados cuando se inicia el sistema.

En la ficha *General* encontraremos la información del sistema operativo que ejecuta el equipo, el nombre de usuario registrado, el tipo y la velocidad del equipo y la cantidad de memoria *RAM* que tenemos instalada. Examinar esta ficha es una manera rápida de descubrir la cantidad de memoria *RAM* que detecta *Windows XP*, lo cual es especialmente útil si acabamos de ampliarla.

Aparte de la ficha *Hardware*, las otras fichas y opciones de configuración de este subprograma están más relacionadas con aspectos de reparación de red y de sistema.



La ficha Hardware del subprograma Sistema

La primera opción de la ficha *Hardware* es el botón *Administrador de dispositivos*. Este botón accede a la ventana corrspondiente a los dispositivos instalados en el sistema, que veremos a continuación. Después de tener instalado el hardware, los otros botones de la ficha *Hardware*, *Firma de controladores*, *Windows Update* y *Perfiles de hardware* se utilizan para administrar el hardware y para solucionar los distintos problemas de configuración.

Los dos botones que se describen en las siguientes secciones son *Firma de controladores* y *Administrador de dispositivos*. El hecho de que estos dos botones se encuentren en la ficha *Hardware* no es una casualidad. Los fabricantes de controladores han trabajado muy de cerca con Microsoft para asegurarse que los usuarios de *Windows XP* dispongan de controladores fiables para el hardware.

Aunque el manejo del hardware instalado y sus controladores se realiza a través del *Administrador de dispositivos*, el funcionamiento del hardware con *Windows XP* depende de la configuración en la opción de *Firma de controladores*.

Las firmas de los controladores

La opción *Firma de controladores* es una característica de seguridad desarrollada por Microsoft para *Windows 2000* y *Windows ME*. Los controladores firmados son los controladores que han pasado los estándares de pruebas establecidas por los laboratorios de calidad hardware para *Windows* (*WHQL*), y son los controladores ideales para ser utilizados en *Windows XP*.

Todos los controladores incluidos en el *CD-ROM* de *Windows XP*, así como los que puede encontrar en *Windows Update* incluyen una firma digital que indica que han pasado la prueba de los laboratorios. Se pueden utilizar controladores no firmados, pero la configuración predeterminada de la opción *Firma de controladores* nos lo advertirá. Utilizaremos los controladores no firmados sólo si no disponemos de controladores firmados para el hardware.

> Nota: si visitamos el sitio *Web* del fabricante del dispositivo de hardware para obtener los controladores, quizá veamos que existen controladores firmados y no firmados. Si deseamos bloquear el uso de cualquier controlador no firmado, o desactivar la advertencia ante un intento de instalación de controladores no firmados, deberemos pulsar en el botón *Firma de controladores* y seleccionar la opción correspondiente.

El Administrador de dispositivos

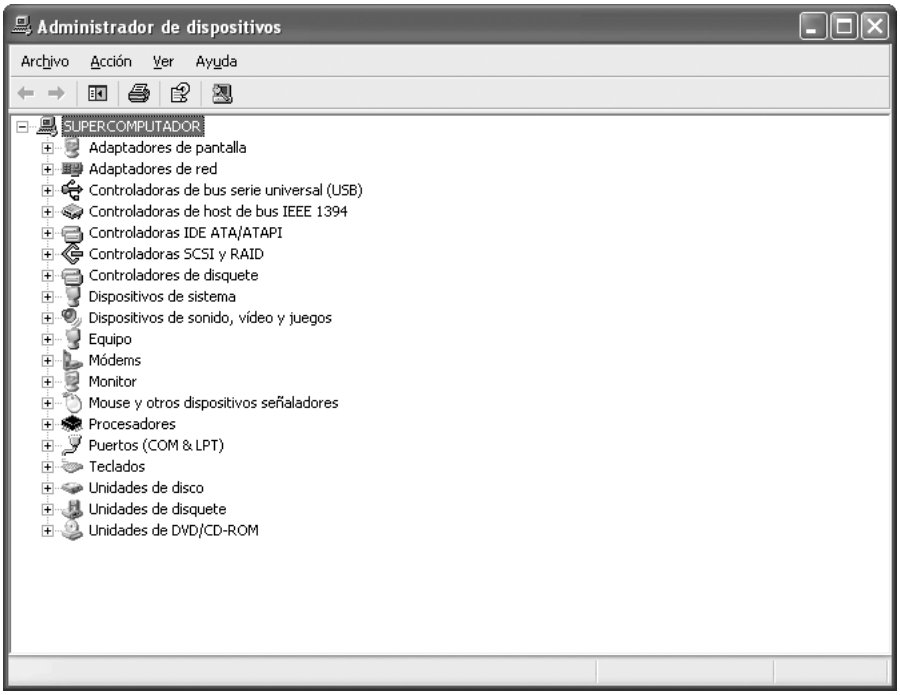
La opción *Administrador de dispositivos* se usa para comprobar la configuración de los distintos dispositivos, reinstalar y actualizar los controladores existentes, y desinstalar dispositivos innecesarios.

> Nota: debemos tener en cuenta que los elementos problemáticos están marcados con un signo de exclamación o con una señal de stop.

En el listado que aparece en el *Administrador de dispositivo*s pulsaremos en el signo + (más) para expandir un tipo de dispositivo y, así, examinar minuciosamente los distintos elementos instalados en ese tipo.

Podemos comprobar el estado de cualquier elemento instalado en el equipo a través de una pulsación con el botón secundario y seleccionar la opción *Propiedades* del

menú contextual. Si elegimos esta opción se abre el cuadro de diálogo *Propiedades* del elemento seleccionado. En este cuadro de diálogo podemos hacer algunos de los ajustes a los distintos dispositivos.



La ventana del Administrador de dispositivos

Las opciones del cuadro del diálogo *Propiedades* varían de un tipo de dispositivo a otro, pero casi todos utilizan las fichas *General, Controlador* y *Recursos*. Los elementos más importantes se enumeran a continuación:

General: en esta ficha encontramos el *Estado del dispositivo*, que nos indica si el dispositivo funciona correctamente o ha aparecido algún problema. Si el dispositivo no funciona de forma correcta, podemos pulsar el botón *Solucionador de problemas* para iniciar el asistente que nos guiará para buscar una solución. Si aparece el botón *Reinstalar el controlador* podemos pulsarlo para reinstalar el controlador del dispositivo si este no funciona correctamente. En el apartado *Uso del dispositivo* podemos comprobar las opciones de configuración del hardware del equipo. En este apartado podemos desactivar un dispositivo problemático seleccionando la opción *No utilizar este dispositivo (deshabilitar*).

Controlador: en esta ficha encontramos los siguientes botones:

Detalles del controlador: a través de este botón accederemos a una ventana de información que nos muestra al creador del controlador, la versión y su ubicación en el disco duro.

- *Actualizar controlador*: este botón inicia el *Asistente para actualización de hardware*, que nos guiará en el proceso de agregar un controlador.
- *Volver al controlador anterior*: esta opción restaura el controlador instalado anteriormente. Esta opción resulta muy interesante cuando un nuevo controlador hace fallar al sistema.
- *Desinstalar*: pulsando en este botón eliminamos los controladores de un dispositivo. La eliminación de los mismos, desinstala el dispositivo del sistema. A veces, la eliminación y posterior reinstalación de un dispositivo puede solucionar los problemas relacionados con dicho dispositivo.

Nota: en los apartados siguientes se explican más detalladamente algunas de las funciones incluidas en las fichas que estamos describiendo.

El cuadro de diálogo Propiedades de la tarjeta de red

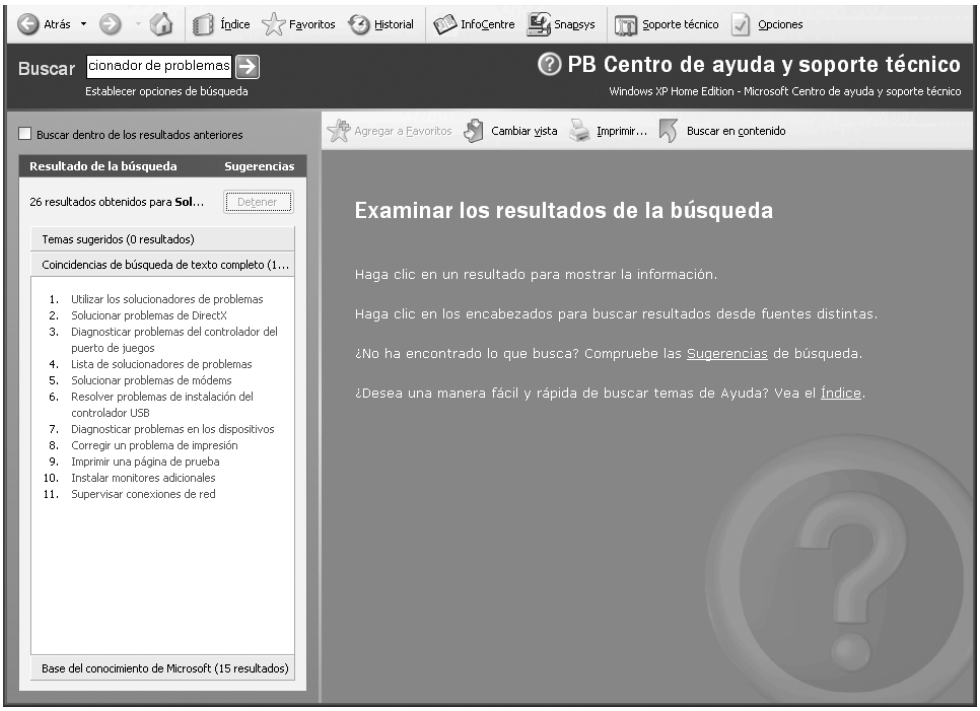
Recursos: si sospechamos que el controlador tiene problemas con otros dispositivos del equipo y por ello no funciona, tal vez los podamos configurar de forma manual en esta ficha.

El solucionador de problemas

Existen nueve solucionadores de problemas de hardware incluidos en *Windows XP*. Aunque no siempre solucionan los problemas, nos ayudarán a investigarlos y, muy posiblemente, lleguemos a una conclusión que no habíamos considerado. Podemos iniciar los solucionadores de problemas desde varios puntos distintos de *Windows*

XP, como los cuados de diálogo *Propiedades de Pantalla* y *Administrador de dispositivos*. También podemos tener acceso a la mayoría de ellos desde el *Centro de ayuda y soporte técnico*. Para ello:

1. Pulsaremos en la opción *Ayuda y soporte técnico* del menú *Inicio*.
2. Escribimos *Solucionador de problemas* en el cuadro *Buscar* y pulsamos en la flecha para iniciar el proceso de búsqueda.



Los resultados de la búsqueda

3. Aparecerá una lista de los distintos asistentes para solucionar problemas.
4. En pantalla aparecen los distintos resultados de la búsqueda. A través de estos vínculos se activan los distintos solucionadotes de problemas.

Nota: si un controlador *USB* no se instala cde la manera orrecta, especialmente si el dispositivo correspondiente no aparece en el *Administrador de dispositivos*, el problema acostumbra a deberse al *BIOS* del sistema. La mayoría de *BIOS* disponen de una opción de configuración que puede habilitar o deshabilitar los puertos *USB*. En este caso deberemos activar la compatibilidad *USB*.

También podemos intentar solucionar un problema de un determinado dispositivo instalado ejecutando el subprograma Agregar hardware del *Panel de control*. Por otro lado, el cuadro de diálogo de propiedades para algunos dispositivos también dispone de su propio *Solucionador de problemas*.

Actualizar los controladores

Después de haber descargado el nuevo controlador, abriremos el *Administrador de dispositivos*. Instalaremos el nuevo controlador siguiendo los siguientes pasos:

1. Abrimos la ventana del *Administrador de dispositivos*.
2. Pulsamos en el dispositivo en cuestión, y abrimos su cuadro de diálogo *Propiedades*.
3. Seleccione la ficha *Controlador* y pulsamos en el botón *Actualizar controlador* para iniciar la actualización.

Activamos la opción adecuada en el Asistente para actualización de hardware

4. Cuando se inicie el *Asistente para actualización de hardware* activaremos la opción *Instalar desde una lista o ubicación específica (avanzado)* y pulsaremos en *Siguiente*.
5. En el siguiente paso, activamos la opción *No buscar. Seleccionaré el controlador que se va a instalar* y pulsamos en *Siguiente*.
6. En este punto, pulsaremos el botón *Utilizar disco*. Buscamos la ubicación del controlador y pulsamos en *Aceptar*.

Volver al controlador anterior

Cuando *Windows XP* instala un controlador nuevo para un dispositivo mantiene el controlador antiguo en el disco, en lugar de sustituirlo, como ocurre en las versiones anteriores de *Windows*.

Para iniciar el proceso de restauración del controlador, pulsamos el botón *Volver al controlador anterior* de la ficha *Controlador*. Si no existen versiones anteriores del

controlador podremos ejecutar el *Solucionador de problemas* para ayudar a solucionar los problemas del dispositivo. Si se encuentra disponible una versión anterior del controlador, pulsaremos en *Sí* para volver a utilizar ese controlador.

La ficha Recursos

La configuración manual de los recursos de un dispositivo puede crear problemas con otros dispositivos ya instalados en el sistema. Realizar la configuración de forma manual impone ciertas restricciones en la capacidad del sistema *Plug & Play* para asignar, dinámicamente, los recursos en futuras instalaciones de dispositivos.

> **Nota**: debemos prestar cierta atención la lista desplegable *Configuración basada* en de la ficha *Recursos*. Esta lista muestra las opciones de configuración de hardware en las que está habilitado el dispositivo en cuestión. Si seleccionamos una configuración de hardware que no es la predeterminada, y cambiamos cualquier configuración de los recursos, pueden surgir problemas cuando utilicemos la configuración de hardware predeterminada. Los problemas derivados de los conflictos con los recursos pueden deshabilitar el hardware y hacer que el equipo no funcione correctamente.

Para asignar un recurso, seleccionamos el recurso en cuestión de la ficha *Recursos* y pulsamos el botón *Cambiar configuración*. Para evitar que cualquier persona pueda entrar y dañar algo, el cuadro de diálogo de asignación manual de recursos supervisa lo que hacemos. Si intentamos asignar un recurso que se está utilizando, se nos advertirá sobre el conflicto.

La ficha Recursos con los distintos recursos usados por el dispositivo

Los conflictos de hardware

Windows, junto con su tecnología *Plug & Play*, ha mejorado mucho en los últimos años en la detección y prevención de problemas de hardware. Sin embargo, siguen apareciendo problemas del sistema, especialmente cuando se utilizan tarjetas *ISA* y otro hardware heredado. Frecuentemente, los problemas de configuración e instalación sólo se deben a una configuración incorrecta.

Estos problemas se pueden deber a una tarjeta que se disputa la petición de interrupción con otra tarjeta, la dirección del puerto de *E/S*, el acceso directo a memoria o la dirección de la memoria. Como y hemos visto, podemos obligar a *Windows XP* para que utilice recursos del sistema seleccionados de forma manual. Los recursos principales que utiliza el hardware son los siguientes:

- Las interrupciones.
- *DMA* o acceso directo a memoria.
- Las direcciones de puerto de *E/S*.
- El canal *DMA*.

Si instalamos tarjetas *Plug & Play*, muy posiblemente no será necesario volver a preocuparnos de esta configuración, especialmente si las tarjetas utilizan ranuras de expansión *PCI* (prácticamente todas). Si el sistema funcionaba originalmente con *Windows 98* o versiones más modernas de *Windows*, seguramente admite una característica conocida como manejo de interrupciones, que elimina la causa principal de los conflictos de hardware al usar tarjetas *PCI*.

Sin embargo, si el sistema todavía utiliza tarjetas *ISA*, o todavía utiliza los puertos heredados, como el puerto serie (*COM*), paralelo (*LPT*), y los puertos *PS/2*, las interrupciones y otros conflictos pueden seguir siendo uno de los diferentes problemas a resolver.

Los recursos de hardware

El *Administrador de dispositivos* de *Windows XP* puede ayudarnos a determinar los recursos que utiliza el hardware, y *Windows XP* puede administrar la instalación de hardware *Plug & Play* casi sin ayuda.

Sin embargo, como acabamos de mencionar, las tarjetas *ISA* y los conflictos entre los puertos *ISA* integrados (serie, paralelo y *PS/2*) y las tarjetas *PCI* pueden hacer que el sistema no funcione del todo bien.

Las interrupciones

La arquitectura del PC incluye un método para que una parte del hardware obtenga la atención del procesador rápidamente. Esto se hace a través de un mensaje llamado petición de interrupción, o *IRQ*.

Las interrupciones se envían en uno de los 15 cables *IRQ* del bus del equipo. Este tipo de petición es una línea directa al procesador. En realidad, la parte del sistema operativo denominada controlador de interrupciones es la responsable de hacerlo.

A continuación enumeraremos las interrupciones habituales en un sistema equipado con componentes de la casa Intel:

0 Cronómetro del sistema (la utiliza el sistema; no disponible).
1 Teclado (la utiliza el sistema; no disponible).
2 Redirigida a IRQ 9; no disponible.
3 COM2, COM4 (se puede compartir si no se utilizan COM2 y COM4 al mismo tiempo).
4 COM1, COM3 (se puede compartir si no se utilizan COM1 y COM3 al mismo tiempo).
5 Puerto LPT2 o tarjeta de sonido compatible con Sound Blaster.
6 Controlador de disquetes (la utiliza el sistema; nunca disponible).
7 LPT1 (puerto de impresora).
8 Reloj del sistema (la utiliza el sistema; nunca disponible).
9 Controlador de host universal PCI a USB.
10 Suele estar disponible.
11 Suele estar disponible.
12 Ratón PS/2.
13 Procesador de datos numéricos (no disponible; todos los procesadores modernos incluyen un coprocesador matemático integrado).
14 Canal IDE principal (nunca disponible).
15 Canal IDE secundario.

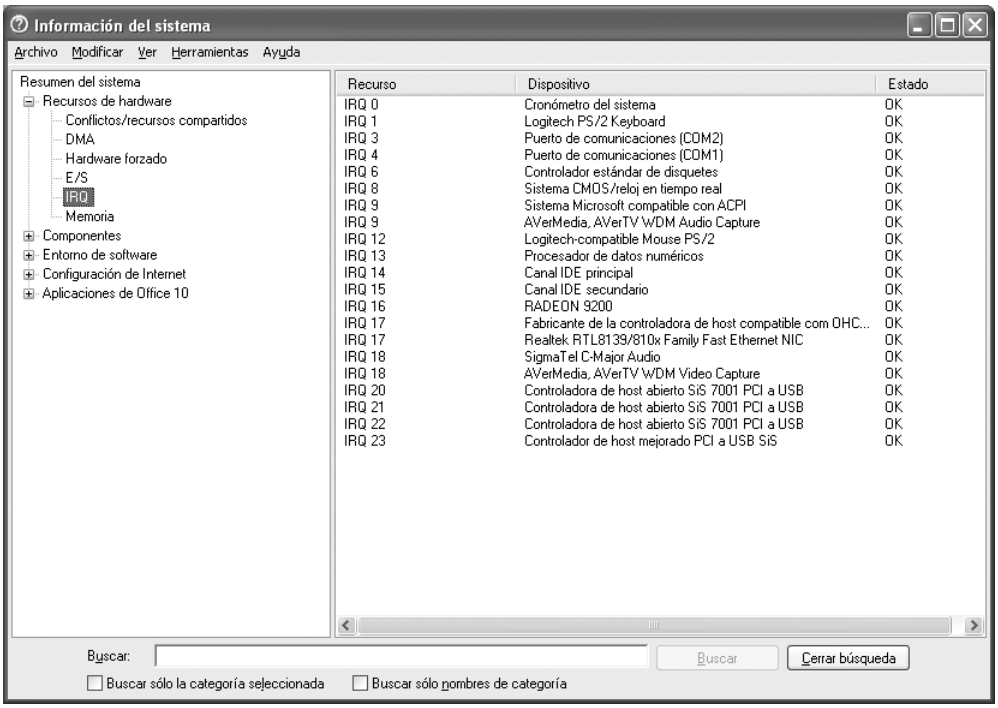
Otros dispositivos complementarios que suelen utilizar interrupciones son:

- Las tarjetas gráficas modernas *PCI* y *AGP*.
- Las tarjetas de adaptador *host SCSI*.
- Las tarjetas de adaptador *host IDE*.
- Las tarjetas módem/fax.
- Las tarjetas de red.

Sólo con las *IRQs* disponibles, sería imposible instalar todas estas tarjetas en muchos sistemas, a menos que:

- Desactivemos un dispositivo existente.
- Podamos compartir una *IRQ*.

Si dos dispositivos *ISA* (o uno *ISA* y otro *PCI*) intentan compartir una misma *IRQ*, muy probablemente se producirá un bloqueo del sistema, o en el mejor de los casos, ninguno de los dos dispositivos funcionará correctamente.



Las interrupciones en mi ordenador Pentium 4

Nota: hace algunos años, la causa habitual de esto, era la conexión de un ratón serie al puerto *COM1* y de un módem/fax al puerto *COM2*. Como hemos visto en las interrupciones, ahora estos dos puertos utilizan la *IRQ* 4.

Windows XP sólo permite compartir las *IRQ* en las siguientes circunstancias:

- Los dos son dispositivos *PCI*.
- Cuando dos tarjetas *ISA* (o una *ISA* y una *PCI*) utilizan la misma *IRQ*, *Windows XP* cambia a modo de prueba, en el que el procesador comprueba regularmente y atiende las peticiones de *E/S* en lugar de esperar a que las líneas de *IRQ* estén activadas. Obviamente, este proceso ralentiza el rendimiento del equipo porque crea otro bucle al que el sistema operativo debe dar servicio.

Si escasean las peticiones de interrupción disponibles, o tenemos dos elementos que intentan utilizar la misma *IRQ* y generan problemas, deberíamos probar una de las siguientes soluciones:

- Una solución con tarjetas *PCI* que crean problemas consiste en mover la tarjeta *PCI* a otra ranura. En algunos equipos, se asigna la interrupción *PCI* de cada ranura *PCI* (desde A hasta D) a una *IRQ* de tipo *ISA*. A veces, podemos conseguir que el hardware funcione moviendo la tarjeta a otra ranura.
- Otra solución es configurar varios perfiles de hardware. Es muy probable que no consigamos que todos los elementos funcionen en un mismo perfil, pero podemos disponer de un par de perfiles y así, al menos, no tendremos que renunciar a nada. Reiniciaremos el equipo con otro perfil cuando necesitemos tener acceso a una parte determinada del hardware.
- Otra solución es utilizar los dispositivos *USB*, *IEEE-1394* y *SCSI*. Como ya debemos saber, todos estos tipos de puerto admiten varios dispositivos en el mismo cable. No se requiere ninguna interrupción a excepción del controlador, que suele necesitar una, y la mayoría de los sistemas recientes ya disponen de puertos *USB* integrados y activados.

Nota: el bus *USB* admite hasta 127 dispositivos, el bus *IEEE-1394* puede admitir hasta 63 dispositivos y el sistema SCSI suele admitir 7 (o 15 si la controladora es compatible con *Wide SCSI*).

Los canales DMA

Los canales *DMA* se utilizan para transferir, de forma rápida, datos entre la memoria y los periféricos, sin la ayuda del procesador. Un PC normal dispone de ocho canales *DMA*, etiquetados del 0 al 7. Algunas tarjetas utilizan varios de estos canales a la vez.

Los dispositivos más comunes que utilizan los canales *DMA* son los siguientes:

- Los controladores de acceso a memoria.
- Los puertos de impresora *ECP*.
- Los controladores de disquetes.
- Las tarjetas de red *ISA*.
- Las tarjetas de escáner *ISA*.
- Las tarjetas de sonido *ISA*.

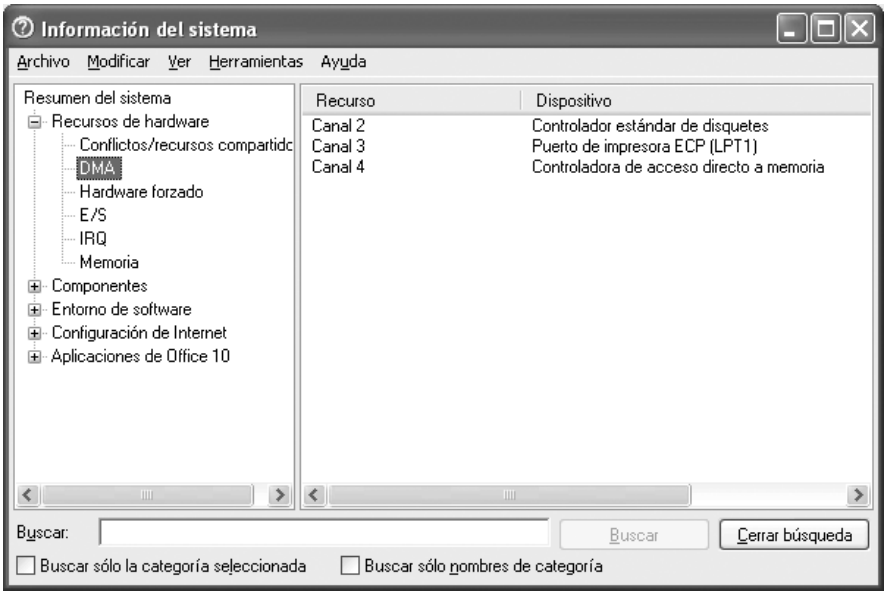
La única ocasión en que un dispositivo *PCI* utiliza siempre un canal de acceso directo a memoria es si está emulando un dispositivo *ISA* que utilice uno.

Compartir canales *DMA* es menos aconsejable que compartir una *IRQ*. Dado que los canales *DMA* se utilizan para transferir datos, no simplemente para activar dispositivos, nunca deberían compartirse los canales de acceso directo a memoria porque un conflicto del acceso directo a la memoria podría provocar la pérdida de los datos.

Habiendo pocos dispositivos que necesiten los canales de acceso directo a memoria es bastante fácil evitar compartirlos.

La siguiente lista muestra las asignaciones típicas de los canales DMA:

0 Se suele utilizar para actualizar las *DMA*.
1 Disponible. Puede utilizarse por las tarjetas de sonido *ISA*.
2 Controlador de disquetes.
3 Puertos de impresora *ECP*.
4 Controladora de acceso directo a memoria.
5 Disponible.
6 Disponible.
7 Disponible.
8 Disponible.



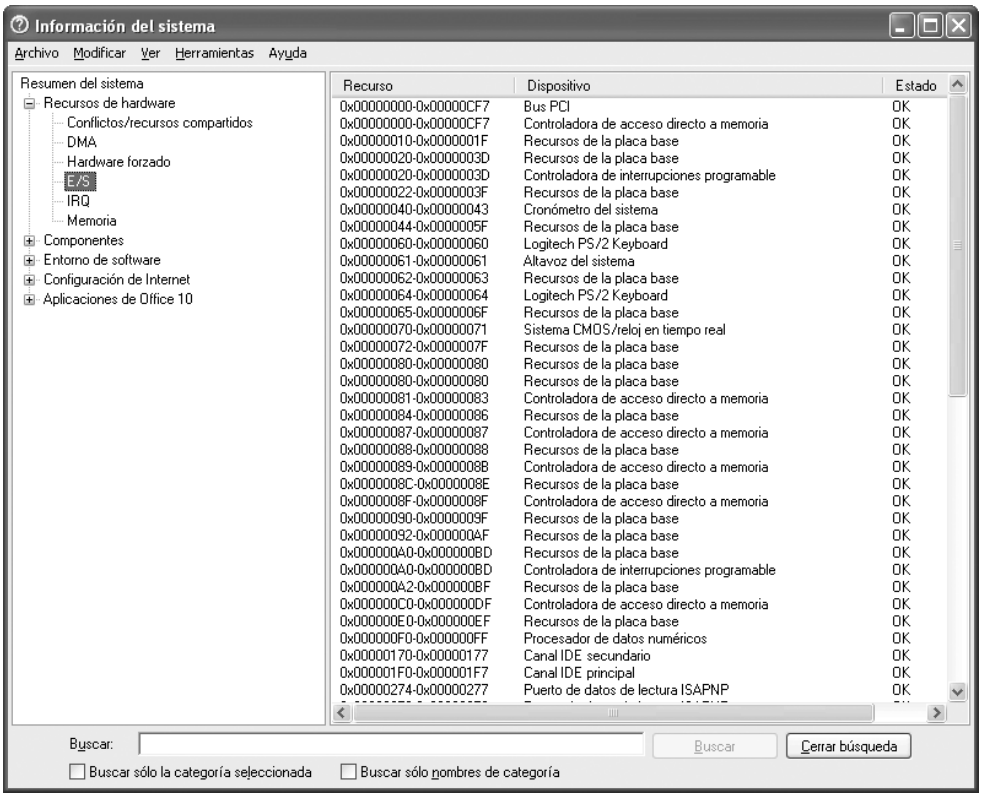
La configuración de los canales DMA

Los puertos de E/S

El uso de canales *DMA* es la forma más rápida para transferir datos entre los componentes en un PC. Sin embargo, todavía se utiliza actualmente una antigua tecnología denominada *E/S*, asignada a la memoria (*E/S* significa entrada/salida). En la arquitectura de los PC, los puertos de *E/S* se asignan a la memoria del sistema y, por lo tanto, el procesador tiene acceso a los mismos mediante las direcciones de memoria. Como ya podemos suponer, cada dispositivo que utilice un puerto de *E/S* debe tener una dirección de puerto diferente.

En la opción *E/S* del nodo *Recursos de hardware* en la ventana del programa *Información del sistema* podemos ver una lista de las direcciones y asignaciones de

E/S. Debemos tener en cuenta que el lenguaje de la asignación de direcciones de memoria se encuentra en formato hexadecimal.



La asignación de E/S

Las direcciones de memoria

De forma parecida a la dirección del puerto de *E/S*, la dirección de memoria es el área de memoria que algún componente de hardware de la placa base utiliza para comunicarse con el procesador. A veces, esta configuración se denomina dirección de inicio. La apariencia de una dirección de inicio de memoria típica es *0xA0000* o simplemente *A0000*. A veces también se elimina el último dígito.

La instalación de un SAI (UPS)

Los apagones e interrupciones de suministro de energía (y la pérdida de datos que pueden provocar) suceden en el momento menos pensado. Aunque *Windows XP* incluye una utilidad de copia de seguridad que se puede utilizar para protegerse ante la pérdida de datos, tampoco debe descuidarse la salvaguarda de la energía en el PC mientras se realizan esas sesiones de copia de seguridad que se deben realizar de forma periódica.

Una unidad de reserva de energía, también denominada *SAI (UPS)*, puede proporcionarle al sistema energía de reserva durante un periodo de diez a quince minutos, que es suficiente para guardar los datos y cerrar el sistema. El *SAI* se conecta a la red general (también puede actuar como estabilizador ante picos eléctricos). El equipo y el monitor se conectan a los enchufes de la parte posterior del *SAI*. Los circuitos electrónicos del interior del *SAI* supervisan continuamente el voltaje de la corriente alterna, y si se produce una subida o bajada de la tensión eléctrica, o si se produce un corte del suministro eléctrico, el *SAI* asume el control y proporciona energía eléctrica al equipo y lo desconecta totalmente de la toma de corriente alterna. Un *SAI* inteligente también se puede conectar a la entrada de datos del equipo (mediante un puerto serie) para enviar un mensaje de datos al sistema operativo del equipo advirtiéndole de la naturaleza del problema en el suministro eléctrico. A continuación, el software del equipo puede decidir las pautas a seguir.

La ficha SAI (UPS) del subprograma Opciones de energía

Como puede imaginar, para evitar la pérdida de datos, el tiempo de respuesta del sistema debe ser muy rápido. Tan pronto como el suministro eléctrico empiece a fallar, el *SAI* debe tomar el control en cuestión de milisegundos. Muchos *SAI* incluyen una conexión de cable serie o de cable *USB*. El cable envía señales al equipo para informarle cuando la unidad de reserva de batería ha asumido el control y para comenzar el proceso de apagado.

Encontramos el subprograma *Opciones de energía* en el *Panel de control*. En la ficha *SAI (UPS)* podemos seleccionar el modelo de *UPS* que está conectado al sistema, configurar las señales en los modelos compatibles con apagado automático y seleccionar cuánto tiempo puede esperar a que el sistema funcione con la batería (factor que se

conoce como tiempo de ejecución del *SAI*). Si el *SAI* no tiene previsto nada para un apagado automático, su alarma nos notificará cuando falle el suministro de la energía. En ese caso, nosotros mismos deberemos apagar el equipo después de guardar los archivos abiertos.

Los perfiles de hardware

Windows XP dispone de la característica denominada *Perfiles de hardware* que permite iniciar *Windows XP* con distintos conjuntos de dispositivos hardware activados. Estos perfiles surgen de la necesidad de acoplar equipos portátiles para que se puedan iniciar con un conjunto distinto de opciones dependiendo de la situación, es decir, si el usuario está de viaje o tiene el equipo conectado a una estación de trabajo, con su propio monitor externo, una unidad de *CD-ROM* adicional, etc.

Cuando trabajamos con un equipo con una gran cantidad de dispositivos, a veces es necesario deshabilitar algunos dispositivos para que otros puedan tener acceso a un número restringido de recursos. Los puertos de conexión en caliente, como *USB*, han reducido la necesidad de utilizar perfiles de hardware, porque su hardware se puede conectar y quitar sobre la marcha.

> **Nota**: además de permitir distintas combinaciones de hardware, los perfiles de hardware permiten que el mismo hardware sea configurado con una distinta asignación de recursos.

Los perfiles de hardware se configuran en el subprograma Sistema del *Panel de control*. En la ficha *Hardware* pulsamos en el botón *Perfiles de hardware* para abrir el cuadro de diálogo del mismo nombre. *Windows XP* crea, de forma automática, un perfil de hardware denominado *Perfil 1*. Los dispositivos adicionales que se van instalando, se agregan automáticamente al perfil.

El cuadro de diálogo Perfiles de hardware

Para crear un perfil de hardware adicional pulsamos en el botón *Copiar*. Para cambiar el hardware habilitado en cualquier perfil, seleccionamos ese perfil al iniciar el equipo y desactivamos la configuración para los dispositivos a través del *Administrador de dispositivos*.