

RESUMEN DE ARQUITECTURA DE COMPUTADORES

MEMORIA EXTERNA

CAPÍTULO 5 (Stalling)

- ✓ Los discos magnéticos son el componente más importante de una memoria externa
- ✓ RAID (Redundant Array of Independent Disks) es una familia de técnicas para utilizar varios discos como un conjunto de dispositivos de almacenamiento de datos en paralelo, con redundancia para compensar los fallos de disco.

DISCO MAGNÉTICO

- ✓ Es un plato circular de metal o plástico, cubierto por una material magnetizable.
- ✓ Los datos son leídos por una cabeza.

ORGANIZACIÓN Y FORMATO DE LOS DATOS

- ✓ Los discos se organizan en un conjunto de anillos concéntricos en el plato: pistas
- ✓ Las pistas adyacentes están separadas por bandas vacías.
- ✓ Los datos se transfieren hacia y desde el disco en bloques, por lo tanto los datos se almacenan en regiones del tamaño de un bloque: sectores.
- ✓ Los sectores adyacentes se separan con intrapistas vacías.
- ✓ El disco se graba con un formato que contiene algunos datos extras, usado sólo por el controlador del disco y no accesibles al usuario.
 - El campo ID es un identificador usado para localizar un sector particular
 - El byte SINCRO delimita el comienzo del campo

CARACTERÍSTICAS FÍSICAS

- ✓ Las cabezas pueden ser fijas o móviles (sobre un brazo en ambos casos):
 - Disco de cabeza fija: hay una cabeza de lectura/escritura por pista
 - Disco de cabeza móvil: contiene una sola cabeza
- ✓ El disco se monta a una unidad de disco que consta del brazo, un eje de rotación y la eléctrica necesaria.
 - Un disco no extraíble está permanentemente montado en la unidad de disco
 - El disco extraíble puede ser quitado y sustituido.
- ✓ Según el tipo de cubierta puede ser:
 - De doble superficie
 - De una sola superficie
- ✓ Según la cantidad de platos
 - Plato único
 - Varios platos que disponen de varios brazo y constituyen un paquete de disco
- ✓ Según el mecanismos de la cabeza:
 - Contacto: hay un contacto físico
 - Separación fija
 - Separación aerodinámica (Winchester): la cabeza está montada en una unidad hermética cerrada así pueden operar más cerca de la superficie del disco, permitiendo que la cabeza sea más pequeña por lo que permite una densidad de datos mayor.

PARÁMETROS PARA MEDIR LAS PRESTACIONES

- ✓ Cuando la unidad de disco está funcionando, el disco está rotando a una velocidad constante. La selección de pista implica un movimiento de la cabeza o una selección electrónica.
- ✓ En un sistema de cabeza móvil, el tiempo que tarda en posicionarse en la pista es el tiempo de búsqueda y el tiempo que tarda el sector en alcanzar la cabeza es el retardo rotacional. La suma de ambos es el tiempo de acceso o tiempo de transferencia de datos.
- ✓ Cuando el proceso hace una petición de E/S, primero debe esperar en cola hasta que el dispositivo esté disponible.
- ✓ Cuando un fichero ocupa todos los sectores de pistas adyacentes se lo llama organización secuencial y reduce los tiempos de acceso ya que el tiempo de búsqueda se hace sólo una vez.

Tiempo de búsqueda

- ✓ Se puede aproximar por la siguiente fórmula lineal:

$$T_s = m \times n + s$$

T_s = tiempo de búsqueda estimado
 n = número de pistas atravesadas
 m = constante que depende del disco
 s = tiempo de comienzo

Tiempo de transferencia

- ✓ Depende de la velocidad de rotación del disco

$$T = b/rN$$

T = tiempo de transferencia
 b = número de bytes a transferencia
 N = número de bytes de una pista
 r = velocidad de rotación en revoluciones por segundo

- ✓ El tiempo de acceso medio total es:

$$T_a = T_s + 1/2r + T$$

RAID

- ✓ Esta técnica desarrolla un conjunto de discos que operen independientemente y en paralelo.
- ✓ Con varios discos las peticiones separadas de E/S se pueden gestionar en paralelo, siempre que los datos estén en discos separados. O si los datos están distribuidos en varios discos también se puede procesar en paralelo una única petición de E/S.
- ✓ El esquema RAID consta de seis niveles independientes que poseen tres características comunes:
 - RAID es un conjunto de unidades físicas de discos vistas por el SO como una única unidad lógica.
 - Los datos se distribuyen a través de las unidades físicas del conjunto
 - La capacidad de los discos redundantes se usa para almacenar información de paridad que garantice la recuperación de los datos en caso de fallo de disco.

NIVEL 0 DE RAID

- ✓ No incluye redundancia para mejorar las prestaciones
- ✓ Los datos del usuario y del sistema están distribuidos a lo largo de todos los discos del conjunto.
- ✓ Esto trae la ventaja de que si hay pendientes dos peticiones diferentes de E/S para dos bloques de datos diferentes, entonces es probable que los bloques pedidos estén en diferentes discos.
- ✓ Los datos son organizados en forma de tiras de datos a través de los discos disponibles.
- ✓ Un conjunto de tiras lógicamente consecutivas se llama franja.
- ✓ En un conjunto de n discos, las primeras n tiras lógicas se almacenan físicamente en la primera tira de cada uno de los n discos.
- ✓ Si una única petición de E/S implica a varias tiras lógicas contiguas, entonces las n tiras se pueden gestionar en paralelo, reduciendo el tiempo de transferencia.

RAID 0 para alta capacidad de transferencia de datos

- ✓ Se puede dar mientras se cumplan dos requisitos:
 - Debe existir una capacidad de transferencia alta en todo el camino
 - La aplicación debe hacer peticiones de E/S que se distribuyan eficientemente sobre el conjunto de discos.

RAID 0 para alta frecuencia de peticiones de E/S

- ✓ Un conjunto de discos puede proporcionar velocidades altas de ejecución de E/S, balanceando la carga de E/S a través de los distintos discos.

NIVEL 1 DE RAID

- ✓ La redundancia se logra por duplicación de todos los datos. Cada franja lógica se proyecta en dos discos físicos separados.
- ✓ Aspectos positivos:
 - Una petición de lectura puede ser servida por cualquiera de los discos que contienen los datos pedidos.
 - Una petición de escritura requiere que las dos tiras correspondientes se actualicen y esto puede hacerse en paralelo (no hay penalización).
 - Cuando una unidad falla se puede acceder a los datos desde la otra unidad.
- ✓ La principal desventaja es el coste
- ✓ RAID 1 puede conseguir altas velocidades de petición de E/S si la mayor parte de las peticiones son lectura.
- ✓ Puede proporcionar una mejora de la transferencia intensiva de datos con un alto porcentaje de lectura.

NIVEL 2 DE RAID

- ✓ Todos los discos miembros participan en la ejecución de cada petición de E/S
- ✓ Se usa la descomposición de datos en tiras muy pequeñas.
- ✓ El código de corrección de errores se calcula a partir de los bits de cada disco y los bits del código que se almacenan en las correspondientes posiciones de bit en varios discos de paridad.

NIVEL 3 DE RAID

- ✓ Se organiza de manera similar a RAID 2, la diferencia es que requiere sólo un disco redundante.
- ✓ Para la corrección de error se calcula un bit de paridad para el conjunto de bits individuales que están en la misma posición de todos los discos de datos.

Redundancia

- ✓ En el caso de un fallo en una unidad, se accede a la unidad de paridad y se reconstruyen los datos desde el resto de los dispositivos. Una vez que se sustituye la unidad que ha fallado, los datos que faltan restauran a la nueva unidad y se reanuda la operación.
- ✓ Caso de que un disco falle, todos los datos estarán todavía disponibles en modo reducido.

Prestaciones

- ✓ Puede conseguir velocidades de transferencia de datos muy altas.
- ✓ Sólo se puede ejecutar a la vez una petición de E/S

NIVEL 4 DE RAID

- ✓ Cada disco opera independientemente, de forma que peticiones de E/S separadas se atienden en paralelo.
- ✓ Son adecuados para velocidades de peticiones de E/S altas y no para aplicaciones que requieren velocidades altas transferencias de datos
- ✓ Utiliza tiras grandes
- ✓ Se calcula una tira de paridad, bit a bit, a partir de las correspondientes tiras de cada disco de datos, y los bits de paridad se almacena en la correspondiente tira del disco de paridad.
- ✓ Hay penalización en la escritura de E/S pequeña por la actualización de los bits de paridad (cada escritura de una tira implica dos lecturas y dos escrituras)

NIVEL 5 DE RAID

- ✓ Distribuye las tiras de paridad a lo largo de todos los discos
- ✓ Para un conjunto de n discos, la tira de paridad está en diferentes discos para las primeras n tiras, y este patrón se repite.

NIVEL 6 DE RAID

- ✓ Se hacen dos cálculos de paridad distintos, que se almacenan en bloques separados en distintos discos.
- ✓ Si los datos requieren N discos consta de N+2 discos
- ✓ La disponibilidad de los datos es alta, pero incluye una penalidad de escritura

Categoría	Nivel	Descripción	Grado de E/S solicitado (lectura/escritura)	Grado de transferencia de datos (lectura/escritura)	Aplicación típica
Estructura en tiras	0	No redundante	Tiras largas: excelente	Pequeñas tiras: excelente	Aplicaciones que requieren altas prestaciones con datos no críticos
Estructura en espejo	1	Espejo	Bueno/regular	Regular/regular	Controladores de sistemas; ficheros críticos
Acceso paralelo	2	Redundancia con código Hamming	Pobre	Excelente	
	3	Bit de paridad	Pobre	Excelente	Aplicaciones con muchas E/S

		intercalado			
Acceso independiente	4	Bloque de paridad intercalado	Excelente/regular	Excelente/pobre	
	5	Paridad distribuida en bloques intercalados	Excelente/regular	Excelente/pobre	Grado de petición alto, lectura intensiva, consulta de datos
	6	Paridad distribuida dual en bloques intercalados	Excelente/regular	Excelente/pobre	Aplicaciones que requieren alta disponibilidad

MEMORIA ÓPTICA

CD-ROM

- ✓ Compact disk read-only memory
- ✓ La información es grabada digitalmente como una serie de hoyos microscópicos en la superficie reflectante.
- ✓ Un hoyo cerca del centro del disco que rota, pasa por delante de un punto fijo más despacio que un hoyo en el exterior y hay que compensar la variación de velocidad. Esto se puede hacer incrementando el espaciado lineal entre bits de información grabados en los segmentos más externos del disco, por lo que el disco gira a una velocidad angular constante (CAV)
- ✓ La ventaja de la tecnología CAV es que los bloques individuales se pueden direccionar a partir de la pista y el sector.
- ✓ La desventaja es que se pierde capacidad.
- ✓ Otra opción es que la información se empaqueta con densidad uniforme a lo largo del disco en segmentos del mismo tamaño, y se explora a la misma velocidad, rotando el disco a una velocidad variable dando la posibilidad de que el láser lea a una velocidad lineal constante (CLV).
- ✓ Cada bloque consta de los siguientes campos:
 - Sincronización: identifica el principio de un bloque.
 - Cabecera: contiene la dirección del bloque y el byte de modo.
 - Datos
 - Auxiliar: datos del usuario adicionales
- ✓ El CD-ROM tiene tres ventajas:
 - Gran capacidad de almacenamiento
 - Se puede replicar en grandes formas baratas
 - Es extraíble, permitiendo usar el disco como memoria de archivo
- ✓ Las desventajas del CD-ROM son:
 - Es sólo lectura y no se puede actualizar
 - Tiene un tiempo de acceso mayor que las unidades de disco magnético.

WORM

- ✓ CD de una-escritura-varias-lecturas
- ✓ Para un acceso más rápido usa CAV
- ✓ Proporciona una grabación permanente de grandes cantidades de datos

DISCO ÓPTICO BORRABLE

- ✓ Se puede escribir y reescribir repetidamente con un sistema magnético-óptico
- ✓ Ventajas:
 - Puede ser usado como memoria secundaria
 - Alta velocidad

- Son extraíbles
- Seguridad

DISCO VÍDEO DIGITAL

- ✓ graba un gran volumen de datos con una calidad alta

DISCO MAGNÉTICO-ÓPTICOS

- ✓ usa un láser óptico para aumentar la capacidad.
 - ✓ La lectura es puramente óptica
 - ✓ Tiene la ventaja de longevidad y el costo por Mbyte es inferior.
-

ENTRADA/SALIDA

CAPÍTULO 6 (Stalling)

- ✓ Hay tres técnicas de E/S: E/S programada, en la que la E/S se produce bajo el control directo y continuo del programa que solicita la operación de E/S; E/S mediante interrupciones, en la que el programa genera una orden de E/S y después continúa ejecutándose hasta que el hardware de E/S lo interrumpe para indicar que la operación ha concluido; y acceso directo a memoria (DMA), en lo que un procesador de E/S específico toma el control de la operación para transferir un gran bloque de datos.
- ✓ El módulo de E/S se conecta al bus de sistema y controla a uno o más dispositivos periféricos. Estos cumplen dos funciones principales:
 - Realizar la interfaz entre el procesador y la memoria a través del bus del sistema o un conmutador central.
 - Realiza la interfaz entre uno o más dispositivos periféricos mediante enlaces de datos específicos.

DISPOSITIVOS EXTERNOS

- ✓ Un dispositivo externo se conecta a la computadora mediante un enlace a un módulo de E/S, que se utiliza para intercambiar señales de control, estado y datos.
- ✓ A este tipo se lo denomina dispositivo periférico
- ✓ Se pueden clasificar en tres categorías:
 - De interacción con los humanos
 - De comunicación con máquinas
 - De comunicación (con dispositivos remotos)
- ✓ Los datos se intercambian en forma de un conjunto de bits que son enviados a, o recibidos desde el módulo de E/S.
- ✓ Las señales de control determinan la función que debe realizar el dispositivo (enviar datos al módulo – INPUT o READ; aceptar datos desde el módulo- OUTPUT o WRITE; indicar el estado a realizar algún control)
- ✓ Las señales de estado indican el estado del dispositivo (READY/NOT-READY)
- ✓ La lógica de control controla su operación en respuesta a las indicaciones del módulo de E/S
- ✓ El transductor convierte las señales eléctricas asociadas al dato en otra forma de energía.

TECLADO/MONITOR

- ✓ De interacción computador/usuario.
- ✓ La unidad básica de intercambio es el carácter y cada carácter está asociado a un código de un 7 bits- ASCII
- ✓ Los caracteres son de dos tipos: imprimibles y de control.
- ✓ Cuando el usuario pulsa una tecla se genera una señal electrónica, interpretada por el transductor, que lo traduce a un código de la tabla ASCII.

CONTROLADO DE DISCO (DISK DRIVE)

- ✓ Intercambia señales de dato, control y estado y controla la lectura/escritura del disco.

MÓDULOS DE E/S

FUNCIONES DE UN MÓDULO

- ✓ Control y temporización: para coordinar el tráfico entre los recursos internos y los dispositivos externos. El sistema utiliza un bus, entonces cada una de las interacciones entre el procesador y el módulo de E/S implica uno o más arbitrajes del bus.
- ✓ La comunicación con el procesador implica:
 - Decodificación de órdenes: el módulo acepta las órdenes que se envían por el bus de control.
 - Datos: intercambio de datos a través del bus de datos.
 - Información de estado
 - Reconocimiento de dirección: un módulo de E/S puede reconocer una única dirección por cada uno de los periféricos que controla.
- ✓ Comunicación con el dispositivo: implica intercambiar órdenes, información de estado y datos.
- ✓ Almacenamiento temporal de datos (data buffering): los datos se almacenan temporalmente en el módulo de E/S y después se envían al periférico a la velocidad de éste o en sentido contrario. Así los módulos deben ser capaces de operar a velocidades variantes.
- ✓ Detección de errores e informarlos al procesador: una clase de errores son los defectos mecánicos y electrónicos en el funcionamiento del dispositivo, otra clase son los cambios accidentales en los bits de transferencia

ESTRUCTURA DE UN MÓDULO DE E/S

- ✓ El módulo se conecta al resto del computador a través de un conjunto de líneas.
- ✓ Los datos se transfieren a y desde el módulo se almacenan temporalmente en uno o más registros de datos.
- ✓ Puede haber uno o más registros de estado que proporcionan el estado presente.
- ✓ Cada módulo tiene una dirección única por cada dispositivo que controla. Y para cada dispositivo posee una lógica específica
- ✓ Un módulo de E/S que se encarga de la mayoría de los detalles del procesamiento presentado al procesador una interfaz de alto nivel se llama canal de E/S o procesador de E/S
- ✓ Uno simple y que requiere un control detallado se denomina controlador de E/S

E/S PROGRAMADA

- ✓ Los datos se intercambian entre el procesador y el módulo de E/S.

- ✓ Cuando el procesador envía una orden debe esperar hasta que la operación de E/S concluya.

ÓRDENES DE E/S

- ✓ El ejecutar una instrucción de E/S, el procesado proporciona una dirección especificando el módulo de E/S particular y el dispositivo externo y una orden
- ✓ Hay cuatro tipos de órdenes:
 - Control: se utilizan para activar periféricos e indicarle que hacer
 - Test: se utiliza para comprobar diversas condiciones de estado asociadas con el módulo de E/S y su periférico
 - Lectura: hace que el módulo de E/S capte un dato de un periférico y lo sitúe en un buffer interno para que después lo capte el procesador
 - Escritura: hace que el módulo de E/S capte un dato del bus de datos y lo transmita al periférico.

INSTRUCCIONES DE E/S

- ✓ Cuando el procesador, la memoria y las E/S comparte un bus común, son posibles dos modos de direccionamiento: asignado en memoria (memory-mapped) y aislado.
- ✓ Con las E/S asignadas en memoria existe un único espacio de direcciones para las posiciones de memoria y los dispositivos de E/S. Se necesita una sola línea de lectura y otra de escritura en el bus. El rango completo de direcciones está disponible para ambos. La ventaja es que puede utilizar un amplio repertorio de instrucciones permitiendo una programación más eficiente, pero utiliza se utiliza el espacio de direcciones de memoria.
- ✓ Se conoce como E/S aislada cuando el espacio de direcciones de E/S está aislado de la memoria.

E/S MEDIANTE INTERRUPCIONES

- ✓ Las E/S con interrupciones consumen gran cantidad del tiempo del procesador puesto que cada palabra de datos que va desde la memoria al módulo de E/S, o viceversa, debe pasar a través del procesador.

PROCESAMIENTO DE LA INTERRUPCIÓN

- ✓ Cuando el dispositivo de E/S termina una operación de E/S, se produce la siguiente secuencia de datos en el hardware.
 1. el dispositivo envía una señal de interrupción al procesador.
 2. el procesador termina de ejecutar la instrucción en curso antes de responder a la interrupción
 3. el procesador comprueba si hay interrupciones, determina que hay, envía una señal de reconocimiento al dispositivo que originó la interrupción, haciendo que el dispositivo desactive su señal de interrupción
 4. el procesador guarda la información necesaria para continuar el programa en curso en el punto en que se interrumpió. Almacena:
 - el estado del procesador en la palabra de estado programa (PSW)
 - las posición de la siguiente instrucción a ejecutar
 5. el procesador carga el contador de programa en la posición de inicio del programa de gestión de la interrupción solicitada. (una vez que el contado de

- programa se ha cargado, la captación de instrucción el control se transfiere al programa de gestión de interrupciones).
6. el puntero de la pila se actualiza para que apunte a la nueva cabecera de la pila y el contador de programa se actualiza para que apunte al comienzo de la rutina de servicio de interrupción.
 7. se procesa la interrupción
 8. cuando se termina el proceso de la interrupción, los valores del registro de estado se recuperan de la pila
 9. se recuperan los valores de PSW y el contado de programas

GESTIONES DE DISEÑO

- ✓ Técnicas de identificación del dispositivo:
 - Múltiples línea de interrupción
 - Consulta a software (software polling): cuando el procesador detecta una interrupción, se produce una bifurcación a una rutina de servicio de interrupciones que se encarga de consultar a cada módulo para determinar cuál provocó la interrupción. Desventaja: consume tiempo.
 - Conexión en cadena (Daisy Chain) (consulta a hardware, vectorizada): la línea de reconocimiento de interrupción se conecta encadenando los módulos uno tras otro. Cuando el procesador recibe una interrupción, activa el reconocimiento de interrupción. Esta señal se propaga a través de la secuencia de módulos de E/S hasta que alcanza el módulo que solicitó interrupción. Normalmente este módulo responde colocando una palabra en las líneas de datos – vector. (interrupción vectorizada)
 - Arbitraje de bus (vectorizada): un módulo de E/S debe disponer del control del bus antes de poder activar la línea de petición de interrupción. Cuando el procesador detecta la interrupción, responde mediante la línea de reconocimiento de interrupción. Después el módulo que solicitó la interrupción sitúa un vector en las líneas de datos.
- ✓ Con líneas múltiples se responde a la de más prioridad. Con consulta software y conexión en cadena es por orden de consulta. El arbitraje de bus emplea un esquema de prioridad.

CONTROLADOR DE INTERRUPCIONES INTEL 82C59A

- ✓ El 80386 posee una sola línea de petición de interrupción – INTR – y una sola línea de reconocimiento de interrupción – INTA.
- ✓ Se configura con un árbitro de interrupciones externos, el 82C59A que puede manejar hasta 8 módulos (pero se pueden poner en cascada para 64 módulos). Su única responsabilidad es la gestión de interrupciones. Es programable.
- ✓ Son posibles los siguientes modos de interrupción:
 - Completamente anidado: se ordena según nivel de prioridad
 - Rotatorio: cuando hay varios con el mismo nivel de prioridad. Un dispositivo pasa a tener la menor prioridad del grupo después de ser servido.
 - Con máscara especial: se permite que el procesador puede inhibir selectivamente las interrupciones desde cierto dispositivo

LA INTERFAZ PROGRAMABLE DE PERIFÉRICOS CON INTEL 82C59A

- ✓ Las 24 líneas de E/S son programables por el 80386 mediante un registro de control, que lo puede variar para especificar los diversos modos de operación
- ✓ Las 24 líneas se dividen en tres grupos de 8 bits que pueden funcionar como un puerto de E/S de 8 bits. (El grupo C se subdivide en grupos de 4 bits)
- ✓ El bus de datos bidireccional de 8 bits se usa para transferir datos al registro de control.
- ✓ El procesador escribe el registro de control para seleccionar el modo de operación y para definir las señales. (modo 0 funcionan como tres puertos. Modo 1 funciona el C como controlador)
- ✓ Las líneas de control tiene la función de sincronización mediante conformidad de señales y la petición de interrupciones.

ACCESO DIRECTO A MEMORIA

INCONVENIENTES DE LA E/S PROGRAMADA Y CON INTERRUPTIONES

- ✓ La velocidad de transferencia de E/S está limitada por la velocidad a la cual el procesador puede comprobar y dar servicio al dispositivo
- ✓ El procesador debe dedicarse a la gestión de las transferencias de E/S; se debe ejecutar cierto número de instrucciones por cada transferencia de E/S

FUNCIONAMIENTO DEL DMA

- ✓ Requiere un módulo adicional en el bus del sistema
 - ✓ El módulo de DMA es capaz de imitar al procesador y de recibir el control de sistema por el procesador.
 - ✓ El módulo de DMA debe utilizar el bus sólo cuando el procesador no lo necesita, o debe forzar al procesador a que suspenda temporalmente su funcionamiento – robo de ciclo.
 - ✓ Cuando el procesador desea leer o escribir un bloque de datos, envía una orden al módulo de DMA, incluyendo:
 - Si solicita lectura o escritura (línea e control de lectura o escritura)
 - La dirección del dispositivo (línea de datos)
 - La posición inicial (línea de datos y almacenada por el DMA)
 - El número de palabras a leer o escribir (línea de datos y almacenada por el DMA)
 - ✓ El módulo de DMA transfiere el bloque completo de datos directamente desde o hacia la memoria Cuando la transferencia se ha terminado, el módulo de DMA envía una señal de interrupción al procesador por lo que el procesador sólo interviene al principio y al final.
 - ✓ El mecanismo de DMA puede configurarse como un procesador suplementario, utiliza E/S programada para intercambiar datos entre la memoria y un módulo E/S a través del módulo de DMA, pero esto es ineficiente.
 - ✓ El intercambio de datos entre los módulos de DMA y E/S se produce fuere del bus de sistema.
-

LA MEMORIA CACHE

CAPÍTULO 5 (Angulo)

NECESIDAD DE LA CACHE

- ✓ Los nuevos microprocesadores se apoyan en tres recursos para mejorar el rendimiento:
 - Arquitectura superescalar: paralelismo explícito y repertorio de instrucciones sencillas
 - Supersegmentación: segmentación con elevado número de etapas
 - Potenciación del subsistema de la memoria caché, para aumentar la velocidad de la memoria.
- ✓ Un procesador segmentado básico ejecuta cada instrucción en cinco etapas:
 - Búsqueda de instrucciones (fetch): se accede a la memoria
 - Decodificación de la instrucción (CPU)
 - Búsqueda de los operandos: se accede a la memoria en modo lectura
 - Ejecución de la instrucción: realizada por el camino de datos de la CPU
 - Escritura del resultado: se accede a la memoria para almacenar.
- ✓ Las dos etapas que afectan al procesador se realizan en un ciclo cada una. Las otras tres etapas (1-3-5) ocupan un tiempo equivalente al acceso a la memoria principal.
- ✓ En lugar de sustituir la memoria DRAM por caché, se opta por aplicar la jerarquía de memoria, que consiste en interponer una pequeña memoria ultrarrápida (caché) entre la CPU y la DRAM.
- ✓ La CPU realiza una petición de información al controlador de la caché, que se encarga de trasladar la petición a la memoria caché. Si ésta contiene la información solicitada (presencia) se produce un Acierto, y entrega la información a la CPU. En caso de no encontrarse la información (ausencia), se produce un fallo. Entonces se debe obtener la info de la memoria principal.
- ✓ La petición a la memoria principal puede realizarse simultáneamente a la petición a la caché o cuando se produce el fallo.
- ✓ La memoria caché tiene como función principal acelerar la transmisión de datos e instrucciones entre el procesador y cualquier componente de almacenamiento
- ✓ El movimiento de datos se realiza en forma que, al producirse un fallo, la memoria caché recibe de la memoria principal el dato pedido y otros continuos que previsiblemente va a pedir la CPU. Así se consigue optimizar la transferencia de bloques siempre que se cumpla con la vecindad espacial y temporal (instante que se va a necesitar)
- ✓ Factor de velocidad = T_p (tiempo de acceso a DRAM) / T_c (acceso a la caché)
- ✓ Factor de eficiencia = T_c / T (tiempo medio de acceso)
- ✓ $T = \alpha \times T_c + (1 - \alpha) \times (T_c + T_p)$ (siendo α la tasa de acierto)

PRINCIPIO DE FUNCIONAMIENTO DE LA CACHE

- ✓ Es de tipo SRAM (RAM estática).
- ✓ Los componentes más importantes que constituyen la caché son:
 - El controlado de caché: se encarga de gobernar cada uno de los elementos de que consta la memoria caché
 - El directorio caché o Bloque de etiquetas, RAM-CAM: es una memoria RAM de acceso por contenido que contiene una lista de etiquetas que hacen

referencia a las direcciones de la memoria principal cuyos datos están en la caché.

- La memoria de datos caché o Bloque de datos asociados, SRAM: usada para almacenar réplicas de instrucciones y datos que serían accedidas muy lentamente desde la DRAM. La información se agrupa en líneas. De acuerdo a este componente se dice el tamaño de la caché
- Lógica de control: comparadores de tantos bits como tenga la etiqueta.
- ✓ De la dirección que el bus de direcciones aplica a la caché, se toman los 21 bits de mas peso (etiqueta) y se comparan con cada etiqueta.
- ✓ Si el comparador indica presencia, la línea de datos buscada es la asociada a la etiqueta que produjo la coincidencia. Los bits de posición seleccionan el dato concreto dentro de la línea recuperada.

ARQUITECTURA Y CARACTERIZAS DE LA CACHE

EL TAMAÑO DE LA CACHE

- ✓ Suele oscilar entre 8 KB y 512 KB.
- ✓ Grandes caches pueden que no aumenten el rendimiento, influyen los algoritmos de transferencia de la memoria caché.

TIPOS DE ORGANIZACIÓN

- ✓ Se divide en unidades de igual tamaño denominadas bloques.
- ✓ Existen tres tipos de organización:
 - Totalmente asociada
 - Cada bloque de la memoria principal debe ubicarse en cualquiera de los bloques o línea de la memoria caché.
 - El acceso se basa en que:
 - La caché es asociativa. Cada línea tiene asociada una etiqueta.
 - La memoria CAM del bloque de etiquetas se accede por contenido.
 - Existe un comparador para cada etiqueta.
 - La etiqueta debe tener tantos bits como sea necesario para identificar completamente al bloque.
 - Permite aprovechar todas las líneas de la caché independientemente del bloque de memoria, obteniendo flexibilidad total.
 - Sólo es necesario sobrescribir una línea cuando la caché se llenó
 - El inconveniente es que las etiquitas son muy largas y se necesita una memoria asociativa (CAM) grande.
 - Asociativa de una vía
 - Se divide la memoria principal en un conjunto de grupos de información que contienen los mismos datos o líneas que la memoria caché.
 - La memoria principal se divide en grupos y cada grupo tiene tantos bloques o líneas como líneas tiene la caché, a cada bloque de uno de los grupos le corresponde una línea de la caché.
 - Una consecuencia es que a bloques consecutivos de la memoria principal le corresponde líneas consecutivas de la caché.

- Para conexionar la memoria principal se necesita un bus de direcciones que tenga 32 líneas ($4G = 2^{23}$)
- La etiqueta sólo necesita 20 bits ($1M = 2^{20}$) para indicar el número de grupo
- De esta forma se ahorran 12 bits porque una vez que se selecciona el grupo la posición relativa que ocupa el bloque en la memoria principal es la misma que la que ocupa en la caché
- No se necesita una memoria asociativa (CAM)
- No se necesita implementar ningún algoritmo de reemplazo, ya que cuando una línea de la caché no contiene el bloque de memoria requerido se reemplaza.
- El inconveniente es que es posible que se expulsen bloques requeridos cuando hay posiciones libres todavía.
- Sólo se necesita comparar los primeros bits (etiqueta) de la dirección.
- El dato concreto a manipular dentro de la línea se indexa utilizando los últimos bits (posición) de la dirección, pero este sólo es accedido en caso de acierto.
- Asociativa de N vías
 - La DRAM y la caché se descomponen en bloques de igual tamaño.
 - Las líneas de memoria caché se agrupan en conjuntos y el número de líneas de cada conjunto se denomina número de vías.
 - La memoria principal reúne los bloques en grupos, tantos bloques como líneas tiene una vía.
 - Cuando la CPU solicita una dirección, se calcula el conjunto en el que debe encontrarse el bloque. Se compara la etiqueta de la dirección con las etiquetas del directorio caché correspondiente al conjunto. Habrá N comparadores. Con el bit de posición se accede a la posición requerida dentro de la línea.
 - Tiene la ventaja de que en esta correspondencia sólo es necesario aplicar la búsqueda de bloques y el algoritmo de reemplazo sobre los bloques de un conjunto completo.
 - También se evita desalojar siempre el mismo bloque de la caché, ya que se puede seleccionar uno de los N bloques que forman el conjunto.

LA ESTRUCTURA FÍSICA DE LA CACHÉ

- ✓ El Pentium tiene dos memorias cachés independientes: una para datos y otra para instrucciones, permitiendo el paralelismo entre instrucciones y datos. Con un tamaño, cada una, de 8 KB. Son cachés asociativas de 2 vías
- ✓ Las etiquetas contienen los 20 bits de más peso de la dirección además de dos bits de control: WP protege contra escritura y V indica la validez de la línea.
- ✓ La sustitución de información se realiza mediante el algoritmo LRU

ACTUALIZACIÓN DE LA MEMORIA CACHÉ

- ✓ Tipos de algoritmos de reemplazo son: RANDOM, LRU, FRQ (leaste frequently used), NLU (not last used), FIFO (first input – first output)
 - RANDOM: aleatoriamente se elige y se sobrescribe una cualquiera de las posiciones ocupadas de una de las vías.

- LRU – Last Recently Used: debe conocer cual de todas las líneas de la caché ha sido la que hace más tiempo que no se usa, y será esta línea la eliminada para alojar la nueva línea requerida por la CPU
- ✓ Hay dos opciones para actualizar las líneas:
 - El dato pedido va en primer lugar: se conoce con el nombre de lectura o carga directa. Consiste en leer primeramente el dato requerido por el procesador y a continuación de este leer el resto de los datos hasta completar la línea. Esto libera al procesador mientras se cargan las líneas
 - El dato pedido va en último lugar: el procesado debe esperar que se termine de cargar la línea para transferir la info a la CPU.

ACTUALIZACIÓN DE LA MEMORIA PRINCIPAL

- ✓ Actualización por escritura inmediata: cada vez que la CPU modifica la caché, esta última manda una orden al bus de sistema y se transfiere la información a la CPU, consiguiendo que no haya errores en la coherencia y actualizando así la memoria principal.
- ✓ Actualización por escritura diferida: la caché dispone de registros internos donde carga temporalmente las modificaciones que ha habido. Actualiza la memoria principal cuando el bus del sistema está libre.
- ✓ Actualización por escritura obligada: la actualización de memoria principal se produce cuando no queda otro remedio, por lo que nunca hay fallo. La memoria se actualiza obligatoriamente cuando:
 - Se accede a una posición de la memoria principal modificada en la caché
 - Hay que eliminar línea en la caché porque está llena, y en ella hay un dato modificado.

TIPO DE CONEXIONADO DE LA MEMORIA CACHE

CONEXIÓN EN SERIE

- ✓ La CPU se conecta directamente a la caché y sólo con ella, por lo que todas las peticiones que la CPU se hace al bus del sistema tienen lugar a través de la memoria caché.
- ✓ Ventajas: si la memoria caché contiene el dato solicitado se evitará manejar el bus del sistema permitiendo el paralelismo
- ✓ Inconvenientes: la memoria cachés es de uso obligatorio por lo que hay a veces una penalización de tiempo si no se encuentra el dato en la caché.

CONEXIÓN EN PARALELO

- ✓ La CPU se conecta indirectamente a la caché a través del bus del sistema
- ✓ Ventajas: la caché es de uso opcional. Cada vez que la CPU realiza una petición, la envía simultáneamente a la caché y a la memoria principal, por lo que no hay penalización de tiempo. En caso de acierto el tiempo es el de acceso a la caché y en caso de fallo es el de acceso a la memoria principal.
- ✓ Inconvenientes: el bus de sistemas queda sobrecargado.

NIVELES DE JERARQUÍA DE LA CACHE

- ✓ Para conseguir un aumento en los aciertos se puede mejorar los algoritmos de carga en la caché y aumentar el tamaño de la caché.

- ✓ Caché de nivel 1, caché L1, caché primaria: es una memoria caché integrada en la CPU. Es la memoria más rápida del ordenador.
- ✓ Caché de nivel 2, caché L2, caché secundaria: se encarga de almacenar aquellos datos e instrucciones muy usados recientemente, pero que no han sido guardados por la caché L1.

CONEXIONADO DE CACHÉS DE VARIOS NIVELES (monoprocesador)

Conexión en serie

- ✓ Ventajas: el tráfico de peticiones a la memoria principal disminuye considerablemente, y por lo tanto el bus de sistema está desocupado la mayor parte del tiempo
- ✓ Inconvenientes: las cachés L1 y L2 son obligatorias. Hay penalización de tiempo

Conexión en paralelo

- ✓ Ventajas: si la caché L1 da fallo, se envía la petición a través del bus de sistema, a la caché L2 y a la memoria principal. La caché L2 es optativa
- ✓ Inconvenientes: siempre se está empleando el bus de sistemas. Hay menos penalización de tiempo.

CONEXIONADO DE CACHÉS EN SISTEMAS MULTIPROCESADOR

- ✓ Simple: cada procesador tiene su propia jerarquía de cachés, pero hay una alta utilización del bus.
- ✓ Multipuesto: la caché L2 es multipuesto, permitiendo acceso simultáneo a las cachés L1 conectadas a ella. Hay una baja ocupación del bus de sistema.
- ✓ Bus: se usa un sistema subbuses para la interconexión. Tiene una baja ocupación del bus principal.

PROTOCOLOS DE COHERENCIA DE CACHE

EL PROTOCOLO MESI

- ✓ Está basado en cuatro estados:
 - M, Modificado: Indica que la línea está modificada por una escritura del procesador y a la espera de actualizar la memoria principal.
 - E, Exclusiva: Indica el controlador de la caché que la línea de la memoria principal es coherente con el contenido de la caché y que ninguna otra caché contiene una copia de la línea.
 - S, Simultáneo o Compartido: permite que otra copia de la misma línea de memoria esté almacenada en alguna otra caché.
 - I, Inválido: indica que la línea marcada no es válida y puede sustituirse.
- ✓ Las operaciones que se pueden efectuar sobre las líneas son:
 - L, Lectura
 - E, Escritura
 - R, Rastreo: operación que se produce al cargar un dato en la caché, se busca el dato en otras cachés de las otras CPU

Estado Inicial	Operación	Estado Final	Acción a realizar
M	L	M	
	E	M	Escritura Obligatoria
	R	S	

E	L	E	
	E	M	
	R	S	
S	L	S	
	E	E (I)	Escritura Inmediata a la memoria principal (invalidación en las demás cachés)
	R	S	
I	L	E	Actualización de la caché
	E	I	Escritura inmediata a memoria principal
	R	I	

EL PROTOCOLO FUTUREBUS +

- ✓ Toma la ventaja del uso de un bus síncronico, basado en el protocolo MESI pero implementando post-escritura (copy-back)
- ✓ Cuando un procesador intenta leer datos de la caché pero el bus está ocupado, la caché toma la copia del bloque.
- ✓ También usa cuatro estados:
 - Exclusivo modificado (modified en MESI)
 - Exclusivo no modificado (exclusive en MESI)
 - Compartido no modificado (shared en MESI)
 - Inválido (invalid en MESI)

EL PROTOCOLO MOESI

- ✓ Soporta intervención directa de datos.
- ✓ Usa cinco estados:
 - Exclusivo (E) es un estado de bloque válido escrito sólo por la CPU
 - Compartido (S) indica un bloque válido
 - Inválido (I) indica un bloque inválido
 - Modificado (M) indica un bloque válido, escrito por la CPU y por alguna otra CPU, sin haber sido observado por un controlador de bus o snoop
 - El estado propietario (O) indica un bloque válido, escrito por varias CPU y controlado por el snoop.

EL PROTOCOLO N + 1

- ✓ Usa una memoria principal inteligente con un bit de modo de uso mantenido en la memoria por cada línea potencial de caché, para indicar cuándo una línea pertenece a la memoria principal o a la memoria caché (pública o privada)
- ✓ Usa tres estados:
 - Inválido
 - Válido
 - Sucio

ARQUITECTURA DEL PENTIUM

CAPÍTULO 6 (Angulo)

INTRODUCCIÓN

- ✓ El microprocesador Pentium tiene una frecuencia de funcionamiento inicial de 60 y 66 Mhz (112 millones de instrucciones por segundo)
- ✓ Tiene una caché interna dividida en un bloque de 8 KB para datos y otro independiente de 8KB para instrucciones.
- ✓ Verificación interna de la paridad para asegurar la correcta transferencia de las informaciones+
- ✓ Unidad de coma flotante mejorada
- ✓ Bus externo de 64 bits
- ✓ Diseño superescalar

ARQUITECTURA INTERNA DEL PENTIUM

- ✓ Es un micro de 32 bits.
- ✓ Su rango de direccionamiento de memoria física o principal es de 4 GB
- ✓ Su rango de direcciones de memoria virtual es de 64 TB
- ✓ Tiene cinco bloques principales

SUBSISTEMA DE MEMORIA CACHÉ

- ✓ Está constituido por dos memorias internas, una de instrucciones y otra de datos de 8 KB cada una, independientes entre sí.
- ✓ La organización es de tipo asociativo de dos vías. Cada vía está configurada por 128 etiquetas correspondiendo a cada una los datos de una línea (32 bytes).
- ✓ Las etiquetas tienen un tamaño de 22 bits: los 20 de mayor peso es la dirección a la que desea acceder la CPU y dos bits de atributos (WP y V)
- ✓ Que sean independientes permite realizar simultáneamente una prebúsqueda en la caché de instrucciones y un acceso a una información en la caché de datos.
- ✓ Las cachés son de escritura obligada (Write Back), lo que implica que los resultados de las operaciones quedan almacenados dentro de la caché hasta que sea preciso actualizarla.
- ✓ Nunca se produce fallo y se consigue un ahorro de tiempo
- ✓ Se usa el algoritmo de sustitución de líneas LRU
- ✓ En los sistemas multiprocesador se utiliza el protocolo MESI

UNIDAD DE ENTEROS SUPERESCALAR

- ✓ La arquitectura superescalar significa que dispone de dos cauces para la ejecución de instrucciones que maneja número enteros. Además tiene otro cauce para la ejecución de instrucciones en coma flotante
- ✓ Los dos cauces de números enteros están segmentados en cinco etapas:
 - Prebúsqueda de instrucciones
 - Decodificación
 - Cálculo de la dirección afectiva (búsqueda de operandos)
 - Ejecución
 - Escritura de los resultados
- ✓ Los cauces se denominan U (ejecución de instrucciones simples de tipo RISC) y V (ejecución de instrucciones complejas de tipo CISC) y su funcionamiento es paralelo

- ✓ Cada cauce posee su UAL (ALU) con un circuito de generación de direcciones y una interfaz con la memoria caché de datos exclusivos.

Funcionamiento de la segmentación

- ✓ En la primera etapa del cauce, la unidad de prebúsqueda manda una dirección al caché de instrucciones. Si está la envía a uno de los buffers de prebúsqueda que la pasará a la unidad de decodificación.
- ✓ Si entre dos instrucciones no existe dependencia irán una a cada cauce simultáneamente.
- ✓ El Pentium posee una Unidad de Predicción de Saltos, para que la ejecución de la instrucción de salto no introduzca burbujas en el cauce.
- ✓ El hecho de introducciones en cada etapa del cauce se llama “paring”

FRU: COPROCESADOR MATEMÁTICO

- ✓ Opera en paralelo con la unidad de números enteros compartiendo el bus de sistema
- ✓ Dispone de un cauce segmentado de 8 etapas que permite obtener resultados en cada ciclo de reloj:
 - Prebúsqueda de instrucciones
 - Decodificación
 - Cálculos de la dirección efectiva
 - Ejecución
 - Ejecución de las instrucciones de coma flotante
 - Ejecución de las instrucciones de coma flotante
 - Escritura de los resultados
 - Informe de los posibles errores
- ✓ Para la ejecución de las instrucciones específicas, el coprocesador posee registros de datos y registros especiales.

SISTEMA DE PREDICCIÓN DE SALTOS CONDICIONALES

- ✓ Se utilizan dos recursos:
 - Software: potente algoritmo estadístico
 - Hardware: buffer de destino de las bifurcaciones BTB (“Branch Target Buffer”)
- ✓ La BTB es una caché ultrarrápida que registra el resultado de las últimas 256 instrucciones de salto condicional y que permite realizar una predicción dinámica de la dirección destino de las instrucciones de salto condicional.
- ✓ El sistema de predicciones de saltos condicionales se pone en marcha cuando el buffer de prebúsqueda encuentra una instrucción de salto condicional.
- ✓ El concepto de ejecución dinámica viene determinado por tres conceptos:
 - Predicción de Salto Múltiple: se predice el flujo del programa a través de varias ramificaciones mediante un algoritmo de predicción de salto múltiple.
 - Análisis del Flujo de Datos: se analiza y ordenan las instrucciones a ejecutar en una sucesión óptima
 - Ejecución especulativa: se aumenta la velocidad de ejecución analizando las instrucciones posteriores a la del contador de programa y ejecutando las instrucciones que “posiblemente” van a necesitarse

BIU: UNIDAD DE INTERFAZ CON EL BUS

- ✓ Es un bloque encargado de soportar todas las transferencias con elementos externos al micro.
- ✓ Controla los ciclos del bus que acceden a la memoria y a las E/S.

- ✓ El bus de datos del Pentium es de 64 bits por lo que la organización del subsistema de memoria se debe llevar en bloques de 8 bits. Cada posición de memoria constará de 8 bloques de 8 bits cada uno. Cada bloque dispone de una señal de control que indica su validez en cada acceso – BEn#
- ✓ En el bus del procesador se utiliza la técnica de “Pipeline” que permite comenzar el siguiente ciclo de bus antes que haya finalizado el anterior.
- ✓ La BIU consta de tres bloques:

Monitor de Prestaciones

El procesador posee un conjunto de contadores y unidades de rastreo y traza que exploran y archivan todos los acontecimientos significativos del flujo de control. Permiten conocer el estado del procesador, el tiempo que se emplea en la realización de ciertas operaciones y las instrucciones que se ejecutan. También obtener el número de ciclos que el procesador emplea en operaciones internas.

Es posible conocer cuánto tiempo tiene que esperar el procesador para conseguir el control del bus externo, todo permite optimizar la ejecución del programa

La unidad de rastreo realiza una traza cuando se produce cierta condición de bifurcación en los saltos o una selección del código, por lo tanto se pueden detectar los cuellos de botella

Bus a Ráfagas

- ✓ Permite transferir 256 bits en un solo ciclo.
- ✓ Como es el tamaño de una línea de caché permite acelerar el tiempo de actualización de la caché

Unidad de Redundancia Funcional

- ✓ Para asegurar la integridad de los datos y se basa en mecanismos de paridad.
- ✓ Cada octeto del bus de datos lleva asociado un bit de paridad que son comprobados por el procesador en cada lectura.
- ✓ Pentium genera un bit de paridad por cada octeto de los 64 bits que componen cada escritura hacia el exterior.
- ✓ El bus de direcciones añade un bit de paridad por cada dirección generada que se comprueba en cada acceso de lectura o escritura.
- ✓ El Pentium implementa un sistema de redundancia funcional con dos procesadores Pentium en el mismo bus. Los dos ejecutan las mismas instrucciones y uno se encarga de comparar los resultados con el otro. Si existe diferencia se producirá una interrupción.

MODOS DE FUNCIONAMIENTO DEL PENTIUM

- ✓ Modo real: Empleando sólo 20 líneas para el bus de direcciones, de modo que la CPU sólo puede direccionar 1 MB.
- ✓ Modo Protegido: Capacidad del proceso multitarea y el manejo de memoria virtual.
- ✓ Modo Virtual 8086: multitarea y con un sistema de protección entre tareas, y se permiten ejecutar tareas del modo Real
- ✓ Modo de Manejo del Sistema (SMM): proporciona un sistema operativo transparente para el programador que implementa dos funciones – la primera relacionada con la mejora de la seguridad de todo el sistema y la segunda en un sistema de regulación de la alimentación del consumo del procesador.

MODELO DEL PENTIUM PARA EL PROGRAMADOR DE APLICACIONES

CAPÍTULO 7 (Angulo)

PROGRAMADORES DE SISTEMAS Y PROGRAMADORES DE APLICACIONES

PROGRAMADOR DE APLICACIONES

- ✓ Es el encargado de codificar programas para los usuarios finales
- ✓ Suelen desarrollar aplicaciones compuestas por uno o más programas en un lenguaje de alto nivel comercial.
- ✓ Contempla la CPU como un conjunto de registros de trabajo que le permiten confeccionar dichas aplicaciones del usuario.

PROGRAMADOR DE SISTEMAS

- ✓ Es el encargado de desarrollar programas y utilidades del sistema operativo
- ✓ La misión es construir un sistema de exploración óptimo que sea capaz de soportar todas las aplicaciones previstas
- ✓ Funciones:
 - Organizar el sistema para el correcto tratamiento de las tareas de los diferentes usuarios
 - Confección de objetos para sistemas operativos, depuradores, compiladores, etc.
 - Asignar a cada tarea su nivel de privilegio y un sistema de protección adecuado
 - Organizar toda la memoria y el procesador para que las tareas consigan un mejor rendimiento
- ✓ Los micros Pentium disponen de una serie de registros y recursos especiales – registros del sistema – que se encargan de gestionar el funcionamiento general, aprovechando todas las prestaciones.

REGISTROS INTERNOS PARA EL PROGRAMADOR DE APLICACIONES

- ✓ El Pentium dispone de 32 registros en su arquitectura interna de los cuales la mitad son para el programador de aplicaciones que se clasifican en cuatro grupos:

REGISTROS DE PROPÓSITO GENERAL

- ✓ Pueden usarse para almacenar datos como direcciones.
- ✓ Estos son (E de “extendido”):
 - EAX: acumulador (por una parte contiene un operando y por otra se carga el resultado de las operaciones de la ALU)
 - EBX: base (contiene una dirección que apunta a la base de un conjunto de datos)
 - ECX: contador (se carga con el número de veces que se ejecuta una instrucción)
 - EDX: datos (se emplea para contener las direcciones de los puertos de entrada y salida en las instrucciones que manejan el mapa de E/S)
 - ESP: puntero de pila (como el de base sirven para controlar el direccionamiento de la pila)
 - EBP: puntero de base
 - ESI: índice fuente (son dos punteros – con el EDI – para trabajar concadena de caracteres.
 - EDI: índice destino

- ✓ Cuando se accede a los 16 bits de menos peso se designan AX, BX, CX, DX, SP, BP, SI, DI
- ✓ Las operaciones en la pila son soportadas por tres registros:
 - Registro de segmento de pila (SS): especifica las características del segmento de pila que reside en la memoria
 - Registro puntero de pila (ESP): contiene el desplazamiento de la cima de la pila en el segmento de la pila actual
 - Registro puntero base de la pila (EBP): se usa para acceder a estructuras de datos pasadas a la pila.
- ✓ Se pueden acceder a los dos bytes de menos peso de los registros AX, BX, CX y DX y se los llama por AL, BL, CL y DL al acceder al de menos peso y si accedo al byte de más peso es AH, BH, CH y DH.

EIP: REGISTRO PUNTERO DE INSTRUCCIONES

- ✓ Puede trabajar en dos modos:
 - En modo Nativo o Protegido tiene una longitud de 32 bits y recibe el nombre de EIP. Almacena el desplazamiento que hay que añadir a la base del segmento de código para obtener la dirección donde está la siguiente instrucción a ejecutar. La base se obtiene a partir del registro de segmento CS.
 - En modo Real se emplea un tipo de direccionamiento reducido que sólo precisa de 16 bits para indicar el desplazamiento. Son los dos bytes de menos peso de EIP que se denomina IP.

REGISTRO DE ESTADO O DE SEÑALIZADORES (EFLAGS)

- ✓ Consta de 32 bits de los que la mayoría son señalizadores controlados por la UAL

<u>Tipo de bit</u>	<u>Cuando toma el valor 1</u>	<u>Cuando toma el valor 0</u>
CF: acarreo	indica que ha existido acarreo en el bit de más peso de una operación suma aritmética	no ha habido acarreo
PF: bit de paridad impar	toma este valor para generar la paridad impar con los bits que conforman el resultado de una operación	generar paridad impar
AF: señalizador de acarreo auxiliar	cuando hubo acarreo en el bit 3 del resultado	no hubo
ZF: señalizador de cero	cuando todos los bits del resultado son ceros	caso contrario
SF: señalizador de signo	si el bit de más peso del resultado de la operación es 1	si es 0
TF: excepción al terminar la ejecución de la instrucción	provoca una excepción cuando se termina de ejecutar la instrucción en curso	No provoca excepción
IF: flag de habilitación de interrupciones mascarables	permite el reconocimiento de las peticiones de interrupción mascarables	prohíbe el reconocimiento de la interrupción externa e ignora las peticiones de interrupciones mascarables
DF: flag de dirección de exploración de las cadenas de caracteres	Postdecremento automático de los registros ESI y EDI, que direccional la cadena	Post incremento automático de ESI y EDI
OF: flag de desbordamiento (overflow)	En la operación con números enteros con signo se activa si el resultado es muy grande	No existe desbordamiento

IOPL: nivel de privilegio de las entradas y salidas. (2 bits). Se emplea en modo Protegido y determina el nivel de privilegio que debe iguala o superar el segmento de código en el que se desea ejecutar instrucciones de E/S que manejan datos en el espacio de E/S del procesador que está reservado a guardar info de los periféricos.		11: nivel 2 (puede acceder a todos) 10: nivel 2 01: nivel 1 00: nivel 0 (únicamente puede acceder el SO)
NT: tarea anidada	La tarea en curso está anidada con la anterior. Hay que retornar a la tarea previa	La conmutación de tareas el libre
RF: flag de reanudación	Se ignoran los puntos de depuración o parada	No se ignoran los puntos de parada
VM: modo Virtual. Permite el paso desde el modo Protegido al modo virtual	Estado en modo Protegido se pasa a modo virtual	No hay paso al modo virtual
AC: bit de chequeo de alineamiento	Se produce una excepción cuando se encuentra una palabra desalineado	No hay excepción por desalineamiento
VIF: IF pero en modo virtual. Permite o prohíbe la atención a las peticiones mascarables	permite el reconocimiento de las peticiones de interrupción mascarables	prohíbe el reconocimiento de la interrupción externa e ignora las peticiones de interrupciones mascarables
VIP: interrupción (mascarable) pendiente en modo virtual	Interrupción pendiente	No hay interrupción pendiente
ID: bit de identificación. Informa si Pentium soporta la instrucción CPUID que sirva para su identificación	El Pentium soporta la instrucción CPUID	Caso contrario

REGISTROS DE SEGMENTO

- ✓ Hay tres tipos de segmentos en los programas de aplicaciones: de pila, de código, de datos
- ✓ Cuando el Pentium trabaja en modo Protegido la dirección lógica o virtual de todos los elementos accesibles en la memoria está formada por una información que consta de los siguientes campos:
 - Selector: son los 14 bits de más peso que referencia el segmento al que se desea acceder y con ellos se encuentran la base del segmento, el límite o tamaño y sus atributos. Los dos bits de menos peso son el campo RPL que indican el nivel de privilegio del segmento peticionado
 - Desplazamiento: es un valor que se añade a la base del segmento para localizar la dirección que hay que acceder de él.
- ✓ Cada registro de segmento hace referencia a un tipo de segmento:
 - CS: (Segmento de Código), contiene el segmento en curso. El desplazamiento reside en el Registro Puntero de Instrucciones EIP
 - SS: (Segmento de Pila), guarda el valor del selector de pila en curso. El registro ESP contiene el desplazamiento
 - DS: (Segmento de Datos) soporta el valor del selector del segmento de datos y el desplazamiento viene especificado en el modo de direccionamiento usado en la instrucción para expresar operandos y el resultado.

LA SEGMENTACIÓN EN MODO REAL

- ✓ Un segmento queda definido por:
 - Base: 20 bits

- Desplazamiento: 16 bits (por ser el tamaño máximo de un segmento 64KB) para indicar el desplazamiento que hay que agregarle a la base
- ✓ La base y el desplazamiento se obtienen en dos campos de 16 bits:
 - Selector: referencia la base del segmento. Para obtener la base del segmento se añaden cuatro ceros a los 16 bits del registro
 - Desplazamiento: en el caso del segmento de código, el desplazamiento lo almacena IP. El desplazamiento en segmentos de pila está en SP y en segmento de datos lo expresa el modo de direccionamiento de los operando o del resultado de la instrucción en curso.

$$\text{Dirección efectiva} = \text{RS} \times 16 + \text{Desplazamiento}$$

	Base	Desplazamiento
Código	$\text{CS} \times 16$	IP
Pila	$\text{SS} \times 16$	SP
Datos	$\text{DS} - \text{ES} - \text{FS} - \text{GS} \times 16$	Desplazamiento

LA SEGMENTACIÓN EN MODO PROTEGIDO

- ✓ El segmento queda caracterizado por tres parámetros – descriptor de segmento:
 - Base de 32 bits
 - Límite, es un valor de 20 bits que determina el tamaño del segmento (máximo 1 MB)
 - Atributos o derechos de acceso, campo de 12 bits que proporciona las características relevantes al segmento:
 - Tipo de segmento
 - Nivel de privilegio
 - Indicadores sobre aspectos relacionados con la gestión de la memoria.
- ✓ Los descriptores de los segmentos se guardan en una “tabla de descriptores” que residen en la memoria principal y se crean y manejan desde el SO.
- ✓ En modo protegido, los registros de segmento actúan como selectores visibles con los que se accede a tablas con los parámetros que son cargados automáticamente por la CPU en los registros caché asociados que son invisibles

JUEGO DE REGISTROS DE LA UNIDAD EN COMA FLOTANTE

- ✓ Los registros del coprocesador son: ocho generales de 80 bits cada uno y tres específicos de 16 bits, siendo estos últimos empleados en labores de control y presentación de estado.
- ✓ Todas las operaciones del coprocesador se realizan utilizando los ocho registros generales que están organizados en forma de pila.
- ✓ La Palabra de Estado del coprocesador matemático tiene 16 bits:
 - TOP: campo de 3 bits apunta el primer registro
 - B y ES: siempre tienen el mismo valor e informan del estado de error
 - SF: señala y distingue las operaciones inválidas causadas por el desbordamiento de la pila y otras causas
 - Los cuatro flags de condiciones de código (C0 a C3) indican el resultado de operaciones aritméticas y de comparación en coma flotante
 - Cuando uno de los siguientes bits toma el valor 0 se produce una excepción y si toma 1 que se ha generado la condición de excepción:

- PE: Precisión
 - UE: Underflow
 - OE: Overflow
 - ZE: División por cero
 - DE: Operando no normalizado
 - IE: Operación inválida.
 - ID: no tiene significado cuando no se manejan valores infinitos
 - RC: los dos bits controlan el redondeo
 - PC: el campo de control de precisión
 - PM, UM, OM, ZM, DM e IM: controla una excepción determinada
 - ✓ Hay disponible un registro de estado más denominado “Palabra Tag”. La FPU usa los valores de los tag para detectar desbordamiento en condiciones de overflow o underflow.
-

MEMORIA SEGMENTADA

CAPÍTULO 8 (Angulo)

ORGANIZACIÓN DE LA MEMORIA

- ✓ Esta organizada en bytes, palabras (2 bytes), dobles palabras (4 bytes) y cuádruples palabras (8 bytes).
- ✓ La memoria se organiza en posiciones de 32 divididos en 4 grupos de 8
- ✓ También existen instrucciones que reconocen los siguientes tipos de datos: enteros, ordinales, enteros BDC empaquetados y sin empaquetar, punteros de direcciones, campos, cadenas y número en coma flotante.
- ✓ Pentium maneja:
 - Segmentos: bloques de memoria variable que contiene información de la misma clase y constituyen el objeto principal sobre el que se basa el mecanismo de protección.
 - Páginas: trozos de longitud fija que tienen un tamaño de 4 KB o 4 MB
- ✓ La segmentación paginada utiliza la paginación para descomponer los segmentos en páginas de 4 KB o de 4 MB de tamaño, que se distribuyen aleatoriamente en la memoria física. En la memoria principal sólo se cargan aquellas páginas del segmento que va a ser usadas por el procesador.

LA MEMORIA EN MODO REAL

- ✓ La memoria está restringida al espacio diseccionado por las 20 líneas de menos peso del bus de dirección (rango máximo de 1 MB)
- ✓ Hay zonas reservadas por el hardware (A000 a C000)
- ✓ La zona de memoria física comprendida entre la dirección decimal 0 y la 1.025 está reservada para contener los 256 vectores de interrupciones
- ✓ Para determinar la posición en un segmento se precisa de dos indicadores: la base y el desplazamiento.
- ✓ En cada momento la CPU tiene activados a un segmento de código (referenciado por CS), otro de pila (SS) y hasta cuatro de datos, referenciados por DS, ES, FS y GS

LA MEMORIA EN MODO PROTEGIDO

- ✓ La memoria en modo Protegido distingue tres espacios:
 - Espacio virtual o lógico: abarca toda la dimensión de la memoria virtual. Como la dirección virtual es de 46 bits el tamaño del espacio virtual es de $2^{46} = 64$ TB. Los programas utilizan este rango pero no la CPU que con su bus de direcciones de 32 líneas sólo accede a 4 GB.
 - Espacio lineal
 - Espacio físico
- ✓ La Unidad de Segmentación traduce las direcciones virtuales a “lineales”
- ✓ Cuando la Unidad de Paginación optativa no funciona, la dirección lineal coincide con la dirección física. Si la está activada, cada segmento se descompone en un número variable de páginas del mismo tamaño y deposita algunas sobre la memoria física en los huecos que encuentra libres. La dirección lineal se debe traducir a física de acuerdo con esta distribución aleatoria de las páginas.
- ✓ Los programas sólo hacen referencias a direcciones virtuales y es la MMU (Unidad de Manejo de Memoria) lo que se encargan de traducirlas a direcciones físicas y destaca ausencias de los elementos al SO

EL ESPACIO VIRTUAL O LÓGICO

- ✓ La dirección lógica consta de dos partes:
 - Selector: 14 bits que seleccionan determinado segmento
 - Desplazamiento: 32 bits que determina una posición de segmento que puede alcanzar un máximo de 4 GB
- ✓ Los dos bits restantes del registro de segmento hacen referencia al nivel de privilegio RPL
- ✓ El bit de menos peso del selector es TI (Indicador de Tabla) e indica si la tabla de descriptores a acceder es local o global.
- ✓ La MMU convierte direcciones virtuales de 46 bits a físicas de 32. En ella residen la Unidad de Segmentación, formada por las tablas de los descriptores de los segmentos y la Unidad de paginación.

EL ESPACIO LINEAL

- ✓ La segmentación referencia a bloques que tienen todas sus posiciones ordenadas consecutiva o linealmente.
- ✓ Tiene el inconveniente de trabajar con segmentos de tamaño variable, lo que complica el mecanismo de transferencia entre las memorias virtual y física, así como la optimización en el comportamiento del espacio de la física.
- ✓ El fallo del segmento provoca una excepción que pone en marcha una rutina del SO que se encarga de trasladar dicho segmento desde la memoria virtual a la física.

DESCRIPTORES DE SEGMENTO

- ✓ Descriptor: en modo protegido es un segmento que queda especificado con tres parámetros: Base de 32 bits, Límite o tamaño de 20 bits y Atributos de 12 bits.
 - Base: donde comienza el segmento
 - Límite: tamaño del segmento. En el campo de atributos la granularidad (bit G) indica si el límite está expresado en bytes o en páginas.
 - Atributo o derecho de acceso: con características relevantes al segmento:
 - Bit de presencia (P): indica si el segmento al que referencia el descriptor se halla en la Memoria Principal o no.

- Nivel de privilegio (DPL): campo de dos bits
 - Clase de segmento (S): 1 es normal, 0 segmento de sistema
 - Tipo: tres bits que distingue a los segmentos normales y determina el acceso permitido
 - Accedido (A):
 - Granularidad (G)
 - Defecto/Grande (D/B): permite distinguir los segmentos nativos de 32 bits para el Pentium. Si D=0 la dirección es de 16 bits.
 - Disponible (AVL): a disposición del usuario para diferenciar ciertos segmentos.
- ✓ Todos los componentes del descriptor son empleados por la Unidad de Segmentación para comprobar si se satisface el conjunto de reglas que protegen a los segmentos.

TIPOS DE SEGMENTOS NORMALES

- ✓ De los tres bits que componen el campo TIPO, el bit de más peso E (Ejecutable)
- ✓ Si E=1 los otros dos bits son, segmento de código:
- Ajustable o Conforming (C): si C=0 al ser accedido el segmento no cambia su nivel de privilegio. C=1 se llama “segmento ajustable” porque toma el privilegio de quien lo invocó.
 - Legible (R): el segmento de código puede leer si R=1
- ✓ Si E=0 y se hace referencia a un segmento de datos:
- Expansión Decreciente (ED): Si ED=0 se trata de un segmento de datos normal, lo que supone que el crecimiento se realiza incrementando el valor de la dirección. Cuando ED=1 se trata de un segmento de pila y su crecimiento es decrementando el valor de la dirección que apunta a la cima.
 - Escribible (W)

MANEJO DE LOS DESCRIPTORES

- ✓ Cada segmento ubicado en la memoria principal dispone de un descriptor en la Tabla de Descriptores que crea el SO.
- ✓ Cada registro de segmento tiene:
- Una parte visible que puede manejarse con una instrucción directa (MOV, POS, LDS,...) o una implícita, que cargue el contenido del registro segmento.
 - Y otra parte invisible que es cargada por el propio procesador. Es la información que existe en el registro caché oculto asociado a cada registro segmento.

TABLA DE DESCRIPTORES

- ✓ Un sistema multitarea distribuye el espacio de memoria en un área global, en la que residen los segmentos comunes a todas las tareas y en un área local exclusiva para cada tarea, con los segmentos propios de cada una.
- ✓ El SO primero distribuye la memoria en las tareas necesarias, posteriormente deposita los segmentos correspondientes, y después construye las tablas de descriptores.
- ✓ Cada segmento del área global estará definido por un descriptor, existiendo una Tabla de Descriptores Globales (GDT)

- ✓ Existirán tantas Tablas de Descriptores Locales (LDT) como tareas soporte el sistema.
- ✓ La tabla de descriptores contiene un máximo de 8 K descriptores de 8 bytes cada uno (64 KB ocupa la tabla). Se pueden controlar un máximo de 16 K de descriptores (dos tablas al mismo tiempo)
- ✓ TI es el bit indicador de tabla. Cuando es uno se selecciona la LDTn, si es cero la CGDT. Los 13 bits más significativos del selector (Índice) apuntan a una entrada de la tabla seleccionada. El valor del índice se multiplica por 8 para apuntar a la dirección concreta de inicio del descriptor.
- ✓ Se puede borrar el contenido de un registro de segmento cargándolo con ceros. (Índice=0, TI=0)
- ✓ A partir del selector y a través de la tabla de descriptores, la Unida de Segmentación localiza la base del segmento a la que suma el desplazamiento para obtener la dirección lineal que se convierte en la física.

EL MODO PLANO

- ✓ Como la segmentación no puede desactivarse se simula su con “plano”.
 - ✓ En este modo se puede saltar la restricción de 64 KB y acceder desde el modo Real, a toda la memoria por encima del primer MB
 - ✓ Se cargan todos los registros de segmento con selectores que apuntan a las tablas a descriptores caracterizado porque le valor de su base es 00000000H y el límite FFFFFFFFH. Así la CPU maneja un único segmento que abarca todo el espacio lineal.
 - ✓ El procesador maneja 6 segmentos.
-

EL MECANISMO DE PAGINACIÓN

CAPÍTULO 9 (Angulo)

INTRODUCCIÓN

- ✓ Divide y manipula los programas y los datos en trozos de tamaño fijo, llamados páginas, que no guardan relación con la estructura lógica
- ✓ La mayor ventaja de la paginación se obtiene en la transferencia e intercambio de elementos entre la memoria virtual y la física. El hecho de que las páginas tengan siempre el mismo tamaño facilita la ocupación de la memoria, así como el rendimiento en su explotación.
- ✓ El mayor inconveniente es el mal aprovechamiento de la memoria.
- ✓ Por otro lado sólo es necesario que un reducido número de páginas de la tarea en curso esté ubicado en la memoria principal en cada momento.

MECANISMO DE PAGINACIÓN

- ✓ Para su habilitación basta con poner a 1 un bit (PG) de uno de los registros de control (CR0). Para ello se utiliza la instrucción MOV CR0 FFFF, cuando estoy en modo protegido solamente.
- ✓ La Unidad de Paginación traduce la dirección lineal a física. Teniendo en cuenta que el espacio físico máximo de 4 GB, la paginación lo descompone en un millón de páginas de 4 KB.
- ✓ Los 32 bits de la dirección lineal se llevan a la Unidad de Paginación. Esta desarrolla la misma función que la segmentación pero con una tabla de páginas

- ✓ Para referenciar la base de una página bastan 20 bits. A la base se le suma el valor del desplazamiento, que son los 12 bits de menos peso de la dirección lineal, para obtener la dirección física de 32 bits que apunta a la posición referenciada en una página de la memoria principal.
- ✓ Con esta organización de la DRAM, la Unidad de Paginación maneja una tabla de un millón de entradas en las que se guardaría la dirección base de la página y sus atributos.
- ✓ La tabla tendrá un tamaño de $1\text{ M} \times 4\text{ bytes} = 4\text{ MB}$

TABLAS DE PÁGINAS

- ✓ Quedan definidas por tres parámetros: base, límite y atributos
- ✓ La tabla de descriptores de páginas reside en la memoria principal para poder ser manejada por la CPU
- ✓ Para reducir el espacio que ocupa la tabla se recurrió a una traducción en dos niveles:
 - Un primer nivel de traducción soporta la tabla denominada Directorio de Tablas de Páginas, que consta de 1 K entradas de 32 bits cada una, ocupa 4 KB y controla hasta 1.024 entradas. Estas entradas se seleccionan mediante los 10 bits de más peso de la dirección lineal. Cada entrada apunta a otra página que se llama Tabla de Páginas y que tiene la misma estructura que el Directorio.
El acceso a una entrada del Directorio se calcula sumando a la base (CR3) el valor de los diez bits de más peso de la dirección.
Para acceder a las entradas de las Tablas de Páginas se suma a la base el valor de los 10 bits centrales de la dirección línea.
 - De esta forma sólo se necesitan 8 KB (Directorio y Tabla de Páginas 4 KB cada uno) de la memoria principal para mantener operativa la paginación.
 - Cuando las páginas son de 4 MB la traducción es sólo de primer nivel porque con el directorio me basta para un 1 MB de direcciones.
 - Tiene el inconveniente de la lentitud. Para traducir una dirección línea a física la CPU debe realizar dos accesos a la memoria para consultar el Directorio y la Tabla de Páginas y un tercer acceso al dato solicitado.
- ✓ Otra solución para deducir el tamaño de la tabla de páginas es la Tabla de Páginas Inversa. Esta sólo da información de las páginas guardada en los marcos. Luego si se produce un fallo, debe existir un mecanismo de traducción de la información a memoria secundaria. En esta hay tantas entradas como marcos de página hay asignados al proceso por el SO. Su principal ventaja es que el tamaño de la tabla es constante, independientemente del tamaño del proceso. Como inconveniente tiene que la paginación es un poco más compleja.

FALLO DE PÁGINA

- ✓ Cuando la Unidad de Paginación detecta que la página no está en la memoria principal genera un fallo de página que origina una excepción que llama a una rutina del SO que se encarga de trasladar dicha página desde la memoria virtual o disco hasta la memoria física o RAM y actualiza los contenidos de las entradas de las tablas de traducción.

PROTECCIÓN A NIVEL PÁGINA

- ✓ En la paginación sólo existen dos niveles de protección: usuario, nivel de privilegio 3 y supervisor 0 1 y 2
- ✓ Los bits U/S (Usuario/Supervisor) y R/W de los atributos se utilizan para dar info sobre el grado de protección
- ✓ Siempre se toman como válidos los U/S y R/W más restrictivos entre los que hay en el Directorio de páginas y los de la Tabla de páginas de la página.

U/S	R/W	Protección de Usuario (nivel 3)	Protección de Supervisor (niveles 0, 1, 2)
0	0	Ninguno	Lectura/Escritura
0	1	Ninguno	Lectura/Escritura
1	0	Sólo Lectura	Lectura/Escritura
1	1	Lectura/Escritura	Lectura/Escritura

FORMATO DE LAS ENTRADAS DEL DIRECTORIO Y DE LAS TABLAS DE PÁGINAS

	Cuando vale 1	Cuando vale 0
P (bit de presencia)	Indica que la página está cargada en la memoria física Una vez cargada la página el S pone P=1, escribe los 20 bits de más peso de la entrada con el valor correspondiente a la dirección de la base de la página y actualiza los restantes bits de los atributos.	Indica que la página está cargada en la memoria física
A (bit accedido)	Cada vez que se accede a dicha página. Para usar el algoritmo LRU	No hubo acceso
SIZ: sólo es válido en CR4 y se llama PSE	Páginas de 4 MB	Páginas de 4 KB
D (bit sucio)	Se ha escrito en la página	Se puede sobrescribir en esta página
R/W (bit de Lectura/Escritura)	Lectura y Escritura	Sólo lectura
U/S (bit usuario/supervisor)	Nivel supervisor	Nivel usuario
PCD (bit de aceptación de la caché)	La página es cacheable	
PWT (bit de escritura obligada)	Además de ser cacheable funciona en modo de Escritura Obligada	
Definibles: tres bits a disposición del programador de sistema.		

TABLA DE TRADUCCIÓN DE DIRECCIONES LINEALES (TLB)

- ✓ Para acelerar la traducción se introdujo en el micro una caché especial ultrarrápida para la traducción de direcciones lineales se denomina TLB: “Translation lookaside buffer”
- ✓ La TLB es una caché de acceso por contenido, que contiene etiquetas con sus respectivos datos asociados.
- ✓ Como etiqueta actúa la dirección lineal y como dato asociado la dirección física correspondiente.
- ✓ En Pentium hay una TLB de 32n entradas para la caché de instrucciones y otra similar para la de datos.

ESTRUCTURA Y FUNCIONAMIENTO DE LA TLB

- ✓ Con las 32 entradas se controlan las 32 últimas páginas accedidas, que supone un espacio de 128 KB de memoria. El Pentium maneja 64 páginas y 256 KB,
 - ✓ Cada entrada es una memoria de acceso por contenido (CAM) como la TLB se compone de una etiqueta y un dato asociado. Cuando se quiere obtener una información se suministra un valor, que se compara con los campos de etiqueta de todas las posiciones. La comparación en paralelo y a gran velocidad.
 - ✓ Un fallo en la TLB puede ser:
 - La página está presente en memoria y solamente se necesita crear la entrada de la TLB que falla.
 - La página no está presente en memoria y se necesita transferir el control al SO para que trate un fallo de página.
 - ✓ En el 386 las 32 entradas de la TLB están organizadas en cuatro grupos de ocho entradas cada uno, que operan en paralelo, ya que utilizan una caché de 4 vías
 - ✓ Dichas entradas constan de 20 bits: 16 de la dirección lineal, uno de validez y tres más de atributo (D: sucio, U: usuario, W: Escritura). Mediante cuatro comparadores que trabajan en paralelo se comparan las cuatro entradas o etiquetas seleccionadas con los bit 15 a 31 de la dirección línea.
 - ✓ Cada vez que se modifican las Tablas de Páginas al cambiar de Directorio como consecuencia de modificarse del registro CR3 en la conmutación de tareas hay que borrar la TLB.
 - ✓ Al aumentar el número de páginas que contiene la TLB incrementa el número de aciertos.
-

MODELO DE PENTIUM PARA EL PROGRAMADO DE SISTEMAS

CAPÍTULO 11 (Angulo)

REGISTROS DEL SISTEMA

- ✓ En un entorno protegido y multitareas el programador del sistema debo poseer un conocimiento de todos los mecanismos de protección de la memoria
- ✓ Los registros del sistema usados por el programador de sistemas son:
 - Los registros de segmento (CS, DS, SS, ES, EF y GS)
 - Los registros de tablas del sistema
 - Registros de señalizadores EFLAGS con los bits propios del control de sistemas
 - Registros de control
 - Registros de depuración
 - Registros de prueba de la TLB
 - Registros específicos

REGISTROS DE SEGMENTO

- ✓ En cada momento la CPU maneja los segmentos propios de la tarea en curso, que se hallan situados en el área local de dicha tareas. También puede manejar segmentos compartidos por las tareas que se ubican en el área global.

REGISTROS DE TABLAS DEL SISTEMA

- ✓ Existen cuatro registros dedicados a la manipulación de estas tablas:

- GDTR: registro de la base de la GDT y el tamaño de la misma. Para manipular este registro se utilizan las instrucciones LGDT y SGDT (carga y almacenamiento). Tiene un tamaño de 48 bits, 32 para la base y 16 para el tamaño.
- IDTR: registro de base de la IDT (tabla de descriptores de interrupción) y el límite. Tiene un tamaño de 48 bits, 32 para la base y 16 para el tamaño.
- LDTR: registro del selectore del descriptor de la LDT. 16 bits. Para cargar y almacenar este registro se usan las instrucciones LLDT y SLDT. Por ser direccionamiento indirecto a través de la GDT, la base y el límite de la LDT se cargan en un registro caché oculto de 48 bits asociado al LDTR.
- TR: registro del selector del descriptor del segmento de estado de la tarea, TSS. Tiene 16 bits e igual que el LDTR los valores obtenidos en la GDT se cargan en un registro oculto.

REGISTROS DE SEÑALIZADORES (EFLAGS)

- ✓ Consta de 32 bits que son banderas o realizan algún control
- ✓ Los EFLAGS que controla el programador de sistemas son:
 - IOLP: nivel de privilegio de las E/S
 - NT: tarea anidada. El procesador pone este flag a 1 cuando se hace una llamada a una tarea mediante una instrucción CALL, una interrupción o una excepción, y examina y modifica este flag cuando se retorna de una tarea mediante la instrucción IRET
 - RF: flag de reanudación del punto de parada.
 - AC: Chequeo de alineamiento.
 - VIF: Flag de interrupción en modo Virtual
 - VIP: Flag de interrupción pendiente en modo Virtual
 - VM: modo virtual-86
 - ID: Identificación del procesador. ID=1 indica que la CPU admite la ejecución de la instrucción CPUID, que devuelve las características de la misma en algún registro.

REGISTROS DE CONTROL

- ✓ CR0, CR2, CR3 y CR4 son accesibles al programador de sistemas y pueden ser leídos y escritos mediante instrucciones del tipo MOV desde los registros de propósito general.
- ✓ Determinan el modo de operación del procesador y las características de la tarea en ejecución

CR0: DOBLE PALABRA DE ESTADO DE LA MÁQUINA (POR BIT)

- ✓ PG: Paginación. En modo protegido y en uno está habilitada la Unidad de Paginación
- ✓ CD: Caché desactivada (caché interna)
- ✓ NW: escritura obligada. NW=0 se pone en marcha la escritura obligada y los ciclos de invalidación de la caché
- ✓ AM: Máscara de alineamiento. AM=1 y AC (EFLAG)=1 se permite comprobar si la palabra está alineada.
- ✓ WP: Error del coprocesador
- ✓ TAS: tarea conmutada. En 1 produce una conmutación de la tarea

- ✓ PE: habilitación del modo Protegido.
- ✓ Pentium no utiliza:
 - ET: tarea de coprocesador
 - EM: emulación.
 - MP: monitor de coprocesador.

CR2: DIRECCIÓN LINEAL DE FALLO DE PÁGINA

- ✓ Cuando se produce un fallo de página, es necesario disponer de nuevo de la dirección lineal, por lo que se almacena en este registro de control

CR3: BASE DEL DIRECTORIO DE LAS TABLAS DE PÁGINAS

- ✓ Es un registro de 32 bits de los cuales 20 corresponden a la base del Directorio de Páginas de la tarea en curso.
- ✓ Un bit es el PCD: página cacheable y otro PWT: Página de escritura obligada

CR4: EXTENSIÓN DE LA ARQUITECTURA

- ✓ Sus bits soportan diversas extensiones de la arquitectura

	En 1
VME: extensión del modo Virtual 8086	Se amplía el soporte para el modo Virtual 86 activando el flag VIF para interrupciones mascarables
PVI: Interrupciones virtuales en modo Protegido	Se activa el soporte que activa el flag VIF para las interrupciones en modo Protegido
TSD: desactivación del “time-stamp”	Se permite la ejecución de esta instrucción sólo desde el nivel de privilegio PL=0
DE: depuración de extensiones:	Se activan los puntos de ruptura en el espacio de E/S
PSE: extensión del tamaño de página	Páginas de 4 MB
MCE: activación del chequeo de máquina	Se activan las excepciones correspondientes al cheque de la máquina
PGE: permiso de páginas globales	En 0 se deshabilitan todas las páginas del área global
PCE	Permite la ejecución de instrucciones RDPCM que lee el PCM

REGISTROS DE DEPURACIÓN

- ✓ Son 8 registros de 32 bits (DR0-DR7) que usa el programador de sistema para depurar los programas
- ✓ Los DR4 y DR5 son de uso reservado para el fabricante
- ✓ Las direcciones lineales asociadas a cada una de las condiciones de los puntos de ruptura se almacenan en los registros DR0-DR3.
- ✓ Las entradas en el registro de control de depuración DR7 y el registro de estado de depuración DR6 controlan el comportamiento del Pentium cuando en la ejecución de la instrucción se encuentra un punto de ruptura
- ✓ Cada registro de direcciones de depuración tiene en DR7 sus propios campos de dos bits para la habilitación de puntos de ruptura de Lectura/Escritura (RWx), Longitud (LENx) y Local/Global (Lx y Gx)
- ✓ Los campos R/W toman los valores:
 - 00: ruptura en ejecución de una instrucción
 - 01: ruptura en escritura de datos
 - 10: operaciones L/E (DE debe ser 1)
 - 11: ruptura en escritura o lectura de datos, pero no en la búsqueda de instrucciones

- ✓ LENx que especifica la longitud del campo que está siendo vigilado:
 - 00: un byte
 - 01: dos bytes
 - 10: no utilizado
 - 11: cuatro bytes
- ✓ Lx representan las habilitaciones locales de las condiciones x de los puntos de ruptura.
- ✓ Gx representan las habilitaciones globales de las condiciones x de los puntos de ruptura.
- ✓ DR7 tiene además tres bits más llamados LE, GE, GD:
 - GD en 0 deshabilita la protección de los registros de depuración. Uno es caso contrario. Este bit es puesto a cero cuando entra al manipulador de excepciones
- ✓ DR6 permite determinar las condiciones que se han producido con el error a depurar.
 - Cuando se detecta una excepción de depuración habilitada, el bit asociado Bn se pone en 1. Está formado por
 - El campo BT que funciona como un bit de excepción en depuración que se guarda en la primera posición del TSS. Se pone en 1 antes de introducir al manipulador de depuración, si se ha producido una excepción de depuración por una conmutación de tareas y el bit de excepción del TSS está a 1.
 - BS funciona con el bit del señalizador de excepción del TF. Se pone en 1 cuando el manipulador de depuraciones actúa como resultado de una ejecución paso a paso.
 - BD indica si la siguiente instrucción leerá o escribirá
- ✓ Otras herramientas complementarias a los registros de depuración son: los Puntos de ruptura por software; y paso a paso: cuando TF=1 se produce una excepción cada vez que se ejecuta una instrucción

REGISTROS DE PRUEBA DE LA TLB

- ✓ Son dos registros de 32 bits, TR6 y TR7, con los que se puede leer y escribir los contenidos de la TLB
- ✓ TR6 – comando o control: contiene el campo de etiqueta de la TLB, el bit de comando OP, que si vale 1 lee y 0 escribe, el bit CD que indica acceso y los bits de estado:
 - V: indica si la entrada a la TLB es válida (1) o inválida (0)
 - D: indica si el acceso de escritura es permitido (1) o no (0)
 - U: si esta en 0, permite el acceso a todos los niveles de preferencia y en 1 sólo accede la preferencia 0
 - W: indica si está permitida la escritura (1) o sólo la lectura (0)
 - PD: cuando vale 0 el tamaño de la página es de 4 KB y si vale 1 4 MB
- ✓ TR7: contiene los 20 bits de más peso de la dirección física, un campo de dos bits llamado REP, que es un campo que identifica cuál de los bloques es el seleccionado y el bit HT, que en la lectura expresa si hubo presencia (1) o ausencia (0), mientras que en escritura siempre vale 1.

REGISTROS ESPECÍFICOS DEL PENTIUM

- ✓ Son registros de 64 bits:

- Registro de chequeo de dirección de máquina
 - Registro de chequeo de tipo de máquina
 - Registro de chequeo programables del TR1 al TR12
 - Contador Time-Stamp
 - Registro de selección de control
 - Contadores 1 y 0
 - ✓ Se incorporan las instrucciones que permiten leer (RDMSR – Read Model-Specific Register) y escribir (WDMSR - Write Model-Specific Register) los registros específicos.
-

INTERRUPCIONES Y EXCEPCIONES

CAPÍTULO 14 (Angulo)

CONCEPTOS GENERALES

- ✓ Las interrupciones y excepciones son acontecimientos que provocan la desviación el flujo de control de la CPU
- ✓ Las interrupciones son acontecimientos externos
- ✓ Las excepciones son provocadas automáticamente como consecuencia de alguna anomalía en la CPU
- ✓ Para manejar las interrupciones y las excepciones hay una Tabla de Interrupciones (IDT) de 256 entradas. Esta ocupa un segmento cuya base y límite están contenidos en el Registro de Tablas de Descriptores de Interrupciones IDTR

TIPOS DE INTERRUPCIONES

INTERRUPCIONES EXTERNAS

- ✓ Se trata de señales eléctricas por componentes hardware externos, que provocan la activación de una de las patitas del Pentium. Esta activación es destacada por el Controlador de Interrupciones Programables Avanzado local (APIC)
- ✓ Las patitas de Pentium que soportan interrupciones externas son la INTR y la NMI:
 - NMI: es una interrupción no mascarable (no puede ser ignorada)
 - INTR: es una interrupción mascarable (puede ser atendida o ignorada por la CPU). Su aceptación y ejecución depende del valor del flag IF

INTERRUPCIONES INTERNAS

- ✓ Son interrupciones que se provocan a través de software
- ✓ Las instrucciones que pueden provocar este tipo de interrupciones son INT n e INTO
 - INT n permite que se genere una interrupción no mascarable suministrando el número del vector de interrupciones de la IDT como un operador
 - INTO comprueba el valor del señalizador OF, si vale 1 se producirá una interrupción que es atendida por la entrada cuatro de las IDT

TIPO DE EXCEPCIONES

- ✓ Se clasifican en:
 - Faltas o errores: que se encargan de corregir la anomalía producida al intentar ejecutar una instrucción. El programa retorna a la instrucción que provocó la excepción.

- Trampa: se generan tras la finalización de la instrucción. Permite continuar con la ejecución del programa y retorna a la siguiente instrucción a la que provocó la excepción.
- Aborto: son aquellas excepciones generadas por la CPU en las que no se sabe la localización exacta de la instrucción que la originó.

TABLA DE DESCRIPTORES DE INTERRUPCIONES Y EXCEPCIONES (IDT)

- ✓ En modo real cada entrada o puerta tiene 4 bytes y en modo Protegido 8 bytes
- ✓ Las 20 primeras entradas de la IDT están reservadas para situaciones concretas, las próximas 10 para operaciones del procesador y el resto están a disposición del usuario
- ✓ Para localizar la IDT se utiliza el registro IDTR (32 bits - base y límite)
- ✓ El contenido de IDTR es manipulado por el procesador mediante la instrucción LIDT y la SIDT (hace una copia en la memoria de los valores de la base y del límite almacenados en el IDTR)

VECTOR	DESCRIPCIÓN	FLAG	CAUSA	CÓDIGO DE ERROR	CLASE
0	Error de división	#DE	DIV IDIV (división por 0 o no entra el resultado)	No	Falta
1	Excepción de depuración	#DB	Cualquier código/dato de referencia/INT1	No	Falta/trampa
2	Interrupción NMI	--	Interrupción externa no mascarable	No	Interrupción
3	Punto de ruptura	#BP	INT3 (instrucción de punto de ruptura ejecutada)	No	Trampa
4	Desbordamiento	#OF	INTO (OF=1)	No	Trampa
5	Comprobación de límites	#BR	BOUND (se intenta leer o escribir fuera del límite de un segmento)	No	Falta
6	Código OP no válido	#UD	UD2 o código OP reservado (intenta ejecutar una instrucción de Protegido no estando en ese modo)	No	Falta
7	Coprocesador matemático no disponible	#NM	WAIT/FWAIT o coma flotante	No	Falta
8	Doble falta	#DF	Instrucción que origine una excepción, MNI o INTR (se encuentra una segunda excepción cuando se está tratando a otra)	Sí (cero)	Aborto
9	Desbordamiento del segmento del coprocesador	--	Instrucción de coma flotante (detecta la violación de una página o segmento mientras se transfería una parte del operando del coprocesador)	No	Falta
10	TSS no válido	#TS	Acceso TSS o conmutación de tareas (se ha intentado realizar una conmutación de tareas y que la info del TSS no es válida)	Si	Falta
11	Segmento no presente	#NP	Carga de registros de segmento o acceso a segmentos que no están presentes en la DRAM	Si	Falta
12	Excepción en la pila	#SS	Operaciones de pila y carga registros SS (se detecta una violación del límite durante una operación al registro SS).	Si	Falta

			Segmento de pila no presente al cargar el registro SS)		
13	Protección general	#GP	Referencia a memoria y comprobación (el procesador detecta una clase de violación de protección)	Si	Falta
14	Fallo de página	#PF	Referencia a memoria (la página no está en la memoria física, el proceso no tiene suficiente privilegio para acceder a la página, usuario intenta escribir una página de solo lectura, bits reservados en una página son 1)	Si	Falta
15	Reservado por Intel	--	--	No	--
16	Error de coma flotante FRU	#MF	WAIT/FWAIT o coma flotante	No	Falta
17	Comprobación de alimentación	#AC	Cualquier dato referenciado en memoria	Si (cero)	Falta
18	Comprobación de la máquina	#MC	Código de error y fuentes son modelos dependientes (cuando se detecta un error de bus)	No	Aborto
19	Excepción de coma flotante SIMD	#XF	SSE SSE2	No	Falta
30-31	Reservados por Intel	--	--	--	---
32-255	Interrupciones definidas por el usuario.	--	Interrupciones externas o INT n	--	Interrupción

Clase	Número de vector
Excepciones benignas	1-2-3-4-5-6-7-8-9-16-17-18-19
Excepciones contribuyentes	0-10-11-12-13
Falta de páginas	14

Cuando se producirá o no una doble falla:

Primera excepción	Segunda excepción		
	Benigna	Contribuyente	Falta de página
Benigna	Excepción	Excepción	Excepción
Contribuyente	Excepción	Doble falla	Excepción
Falta de página	Excepción	Doble falla	Doble falla

CÓDIGO DE ERROR

- ✓ Cuando se produce una interrupción o excepción el procesador coloca en la cima de la pila del gestor de interrupciones un código de error, mediante el cual la CPU sabe cuál es el origen de la interrupción o excepción y donde tiene que acceder para tratarla.
- ✓ Antes de ejecutarse la instrucción de retorno de la interrupción a la tarea principal (IRET) se deberá sacar de la pila los códigos de error que se han producido y atenderlos
- ✓ El código de error consta de:
 - Bit 1: EXT indica el origen del error. Interrupción (1) o excepción (0)
 - Bit 2: IDT: indica que el error se ha producido en la tabla IDT (1) o en la GDT o LDT (0)

- Bit 3: indica que el error proviene de la GDT (0) y de la LDT (1)
- 3-15: Índice: indica la entrada de la tabla correspondiente que produjo el error
- 16-31 no usado

INTERRUPCIONES Y EXCEPCIONES EN MODO REAL

- ✓ En cada entrada de la IDT se apunta el segmento de código donde reside la rutina que atiende a la interrupción. Cada vector es de 32 bits y guarda los registros CS e IP. Añadiendo cuatro ceros a los 16 bits del CS se obtiene la dirección de la base del segmento y con los 16 bits del IP se dispone del desplazamiento.

FASES DE ATENCIÓN DE UNA INTERRUPCIÓN O EXCEPCIÓN

- ✓ 1ª fase: se comprueba si hay interrupciones pendientes de ser atendidas entre cada dos instrucciones. Si hay varias peticiones de interrupciones pendientes, se atenderán por orden de prioridad.
- ✓ 2ª fase: al comenzar la interrupción hay que salvar el estado actual para que al finalizar la interrupción pueda retornar. Se guarda en la pila el SC, el IP y el registro FLAGS.
- ✓ 3ª fase: la CPU localiza la entrada correspondiente de la IDT y carga los registros CS e IP con el valor contenido en dicha entrada. Se ejecuta
- ✓ 4ª fase: la rutina de interrupción finaliza con la instrucción IRET, recuperando de la pila el CS, IO y FLAGS.

Prioridad	Descripciones
1 (mayor)	Reset del hardware y comprobación de la máquina
2	Depuración en le cambio de tareas
3	Intervenciones de hardware externos
4	Depuración de la instrucción previa
5	Interrupciones externas
6	Faltas procedentes de la captura de la instrucciones
7	Falta precedente de la codificación de las instrucciones
8 (menor)	Faltas de la ejecución de una instrucción

INTERRUPCIONES Y EXCEPCIONES EN MODO PROTEGIDO

- ✓ La IDR está compuesta por entradas que contienen descriptores de puertas, que permiten acceder al segmento donde se encuentra la rutina
- ✓ Cada entrada es de 64 bits.
- ✓ Los descriptores de la IDT responde a tres tipos de puertas:
 - Puertas de tarea (PT), que sirve para realizar una conmutación de tareas. Esto significa que la rutina de interrupción está en otra tarea distinta a la tarea en curso. La CPU carga el descriptor TSS en el registro TR produciéndose la conmutación de tareas. Dicha tarea finaliza con la instrucción IRET que devolverá el control a la tarea precia
 - Puertas de Interrupciones y Excepciones: conmuta dentro de la misma tarea a un segmento de código donde reside la rutina de interrupción

EL BUS Y LOS CICLOS DE BUS

CAPÍTULO 16 (Angulo)

INTRODUCCIÓN

- ✓ El procesador del Pentium combina un bus caché L2 dedicado de alta velocidad más un bus del sistema con anticipación que hace posible múltiples transacciones simultáneas. Al ser independientes Pentium está habilitado para acceder a datos desde cualquier de sus buses simultáneamente y en paralelo.
- ✓ El bus puede funcionar en modo pipeline lo que le permite comenzar un segundo ciclo de bus antes de haber terminado el primero
- ✓ Las acciones del procesador se realizan en forma sincronizada y controladas por una unidad interna del procesador llamada Unidad de Interfaz con el Bus

CARACTERÍSTICAS DEL CICLO DE BUS

- ✓ Se conoce como ciclo de bus al tiempo en el que la CPU realiza una transferencia de datos completa con memoria o con los periféricos de E/S
- ✓ La frecuencia de funcionamiento del Pentium es el ciclo de reloj interno (CLK)
- ✓ Se denomina período de la CPU, y se representa por T_{CPU} , al tiempo que dura un ciclo de trabajo.
- ✓ Un estado de bus (T) es el tiempo mínimo en el que la CPU puede realizar una operación elemental y dura un período de CLK
- ✓ Tipos de ciclos de bus:
 - Ciclo de bus en modo Sencillo o Simple (No Burst)
 - Ciclo de bus en modo ráfaga (Burst Bus Cycle): para transferir grandes cant
 - Ciclos especiales: reconocimiento de interrupciones, “shutdown”, etc.
- ✓ El ciclo de bus queda definido con las señales de control del ciclo de bus: M/IO#, D/C# y W/R#
- ✓ Cada vez que se activa ADS# comienza un nuevo ciclo de bus y al mismo tiempo aparece una dirección válida por las líneas A3-A31/BE0#-BE7#
- ✓ Cuando el procesador no está realizando ningún ciclo de bus, significa que está en estado de reposo (T_i) o en estado de reconocimiento de HOLD (T_n)

CARACTERÍSTICAS DE LA MEMORIA Y SUBSISTEMAS DE E/S

- ✓ Pentium dispone de un bus externo de datos de 64 bits
- ✓ La caché L2 y la memoria principal están organizadas en posiciones de 64 bits, pero Pentium puede direccionar bits, palabras y dobles palabras por medio de señales
- ✓ Incluye un modo por ráfaga para el llenado y salvado rápido de la línea de la caché
- ✓ Los accesos al área de direcciones de E/S son de 32 bits

SEÑALES ENTRE LA CPU Y LA MEMORIA

- ✓ Para leer o escribir datos de la memoria hay que direccionarla físicamente y transferir los datos. Cuando se transfieren datos se sigue una secuencia definida y estricta de señales de direcciones, datos y control.
- ✓ Se dispone de un contador de bus adicional, que hace que todas las señales estén disponibles para el dispositivo de memoria
- ✓ Si desea leer o escribir datos de memoria transmite una dirección al buffer de dirección por el bus de direcciones.

- ✓ BE#-BE7# definen qué líneas del bus de datos deben transferir los datos en grupos de 8 bits
- ✓ Los datos leídos son transferidos de la memoria principal al buffer de memoria quien los transfiere al bus de datos, de donde Pentium los lee.

CICLOS DE MEMORIA

POSIBLES ESTADOS DE UN CICLO

- ✓ El bus del procesador Pentium tiene seis estados de bus:
 - T_i Estado de Reposo o Inactividad: no se está ejecutando ningún ciclo
 - T₁ Tiempo de defeccionado: estado correspondiente al primer ciclo de reloj de un ciclo de bus
 - T₂ Tiempo de transferencia de datos: segundo estado del ciclo de reloj de un ciclo de transferencia de bus
 - T₁₂ Tiempo de diseccionado (segundo ciclo en modo pipeline) y tiempo de datos (primer ciclo ya en progreso): hay dos ciclos de bus pendientes. El procesador está todavía en el T₂ y ha comenzado el T₁ para el próximo ciclo en modo pipeline
 - T_{2P} Tiempo de datos (primer ciclo) y Tiempo de datos (segundo ciclo en modo pipeline): dos ciclos de bus pendientes ambos en T₂
 - T_D Estado Muerto: indica un ciclo pendiente, en el que el Pentium debe permitir transcurrir un ciclo de reloj para ejecutar una operación de escritura después de una operación de lectura y viceversa

CICLOS SIMPLES DE TRANSFERENCIA

- ✓ Durante su ejecución, los datos de 8, 16, 32 o 64 bits son transferidos de la memoria al Pentium, o viceversa
- ✓ En el modo de transferencia simple (escritura o lectura) una transferencia de datos sin estados de espera requiere dos ciclos de reloj.

Ciclos de Lectura

- ✓ Durante el estado T₁, el Pentium hace válidas las siguientes señales:
 - Las direcciones A₃₁:A₃
 - Las señales BE₀#:BE₇#
 - Las señales de control W/R # y CACHE #
 - Activa ADS # para indicar la validez de la dirección y de las señales de control del bus
- ✓ Después de un período de tiempo, éste transmite los datos diseccionados y activa la señal de listo BRDY #

Ciclos de escritura (sin estados de espera)

- ✓ Se validan las siguientes señales:
 - Las direcciones A₃₁:A₃
 - Las señales BE₀#:BE₇#
 - Las señales de control W/R # y CACHE #
 - Activa ADS # para indicar la validez de la dirección y de las señales de control del bus
- ✓ Durante T₂ el Pentium transmite los datos a escribir y los bits de paridad necesario. El subsistema toma los datos y entonces devuelve activa la señal de BRDY, dando por terminado el ciclo de bus.

ESTADOS O CICLOS DE ESPERA

- ✓ Si la memoria o el dispositivo periférico no puede terminar una petición de escritura o lectura en los dos estados, T1 y T2, entonces el controlador de memoria mantiene la señal de BRDY # a nivel alto, lo que indica al Pentium que debería insertar otro estado T2, para dar a la memoria el tiempo para ajustarse a la respuesta

LÍMITES PARA LAS CUÁDRUPLES PALABRAS

- ✓ La escritura o lectura de una cuádruple palabra que no comienza en una dirección múltiplo de 8, siempre requiere de 2 ciclos de bus, mientras que el acceso a una con una dirección múltiplo de 8 requiere un ciclo de bus.

CICLOS A RÁFAGAS (BURST)

- ✓ Pentium usa el modo Burst o a ráfaga para ciclos de recuperación y lectura de la caché. Se emplean cuatro ciclos de bus para transferir 32 bytes.
- ✓ Según las señales BDRY #, CACHE0020# y KEN #, el Pentium decide independientemente cuándo debe ejecutarse un ciclo de bus Burst o si sólo un ciclo de transferencia simple es suficiente.

Ciclo de lectura

- ✓ Lo único que cambia es que si CACHE # está en nivel bajo el objeto direccionado es cacheable y si la señal KEN # está activa llena la caché
- ✓ Como los ciclos a ráfagas están limitados a un área de memoria, después de recibir la dirección, el subsistema de memoria calcula las otras tres direcciones del ciclo Burst, sin la necesidad de decodificar ninguna señal de Pentium

Ciclo de escritura

- ✓ Un ciclo de escritura en modo Burst es siempre un ciclo de recuperación de una línea modificada en la caché de datos. El Pentium indica esto con una señal activa CACHE # a nivel bajo.

CICLOS “INQUIRY (PETICIÓN) Y “SNOOPING” (RASTREO INTERNO)

CICLOS “INQUIRY”

- ✓ Utilizados por el protocolo MESI
- ✓ Con ellos es posible para una unidad externa determinar cuándo los datos de una dirección específica están almacenados en la caché del Pentium.
- ✓ Para uno de estos ciclos la unidad externa debe transferir la dirección física de la memoria del dato al Pentium (señal AHOLD). Se esperan dos ciclos CLK y se transfiere al Pentium la dirección solicitada a través de A31-A5. La activación final de la señal EADS # indica al Pentium que obtenga la dirección suministrada y la use como una dirección “inquiry”. La señal INV debe también enviarse con A31-A5
- ✓ El comparador de direcciones analiza el contenido de la caché y compara las direcciones transferidas para determinar un fallo o un acierto. Si se produce un acierto envía una señal activa de nivel bajo por la patita HIT #. En caso contrario toma un alto nivel.

CICLOS DE “SNOOPING” O RASTREO INTERNO

- ✓ Usado principalmente para chequear la consistencia de los dos cachés
- ✓ El ciclo de rastreo interno se puede producir en tres casos:

- El kernel de la CPU del Pentium accede a la caché de instrucciones y produce un error. En este caso, el ciclo snooping provoca que la línea de la caché de datos se recupere antes de cargar la caché de código.
- El kernel de la CPU accede a la caché de datos y este acceso origina un error.
- Cuando el bit A (accedido) o D(sucio) de los atributos de entrada de la Tabla de Páginas o del Directorio son sobrescritos

ESPACIO DE DIRECCIONES DE E/S Y PERIFÉRICOS

- ✓ El Pentium maneja el espacio de E/S que puede ser accedido mediante las instrucciones IN/OUT, que incluye 64 K puertos de 8 bits, o 32 K de 16 bits o 16 K de 32 bits.
- ✓ Los accesos a direcciones de E/S se permiten con un tamaño máximo de 32 bits, los restantes 32 bits del bus de datos de Pentium quedan sin uso.
- ✓ El Pentium identifica un ciclo de bus de E/S por la señal M/IO# a nivel bajo

DIRECCIONAMIENTO DE E/S

- ✓ Modos de acceso:
 - Usando las instrucciones E/S
 - Usando las instrucciones de propósito general, lo cual requiere de mapear la memoria de E/S

ACCESO DIRECTO A E/S

- ✓ Si el Pentium desea acceder a un puerto, pasa a nivel bajo la patita M/IO#
- ✓ Por medio de RD/WR# se determina la dirección de la transferencia como lectura o escritura

ACCESO MEDIANTE MEMORIA E/S MAPEADA

- ✓ Se pueden usar todas las instrucciones que permitan acceder a memoria.
- ✓ Mapeando el área de E/S en memoria se pueden utilizar todas las instrucciones generales, para acceder a la misma.

LOS BUFFERS DEL BUS INTERNO

- ✓ La interfaz de bus del Pentium tiene dos buffers de escritura de 64 bits (cada buffer a un cause U y V) y su misión es evitar las retenciones y retrasos que pueden producirse cuando el bus externo se halla ocupado
 - ✓ Sólo los accesos de escritura son almacenados temporalmente en los buffers
 - ✓ Además Pentium contiene otros tres buffers de escritura obligada de 32 bytes. Cada uno contiene una línea completa de la caché y actúa durante la recuperación de línea de la caché en la caché de datos.
 - ✓ Uno almacena la línea de caché a recurrar en el proceso de reemplazo de una línea de caché modificada que debe recurrarse en la memoria externa. Otro se actualiza durante el ciclo de rastreo externo y el tercero se si genera rastreo interno.
 - ✓ Hay otros dos buffers (32 bytes) que soportan llenado de línea de caché. El primero para la caché de datos y el segundo para la de instrucciones.
-

REPERTORIO DE INSTRUCCIONES

CAPÍTULO 17 (Angulo)

TIPOS DE DATOS

- ✓ Los tipos de datos son: byte, palabra, doble palabra y cuádruple palabra.
- ✓ Los bytes de más peso ocupan las direcciones más altas de la memoria.
- ✓ Los datos que manejan las instrucciones de Pentium se pueden clasificar en:
 - Entero y ordinales: son números con y sin signo, para instrucciones aritméticas.
 - Números BDC (Decimal Codificado en Binario): BCD empaquetados, cada byte representa dos dígitos BCD. BCD desempquetado, cada byte sólo representa un dígito BCD utilizando los cuatro bits de menor peso.
 - Cadenas: puede alcanzar un tamaño de 4 GB
 - Campo de bits: presenta la posibilidad de modificar únicamente ciertos bits en un campo, cuyo tamaño máximo puede alcanzar los 32 bits, manteniendo los demás con su valor anterior.
 - Punteros de direcciones: el puntero corto o cercano dispone de un desplazamiento de 32 bits que permite un salto dentro del segmento que se trabaja. El puntero largo o lejano, además del desplazamiento, contiene el valor de un selector para uno de los seguir registros de segmento. Se puede realizar un salto a otro segmento (48 bits)

TIPOS DE DATOS DEL COPROCESADOR MATEMÁTICO

- ✓ Almacena la información en coma flotante y precisión extendida con un tamaño de 80 bits
- ✓ Trabaja con 6 tipos distintos de datos que se clasifican en tres grupos:
 - Enteros: largo (64 bits), corto (32 bits) y palabra (16 bits)
 - Decimal empaquetado: guarda 18 dígitos BCD empaquetados en 80 bits.
 - Coma flotante: de simple precisión (32 bits), doble precisión (64 bits) y precisión extendida (80 bits)

MODO DE DIRECCIONAMIENTO

- ✓ Formas que dispone Pentium para referenciar los operando de sus instrucciones:
 - Modo de direccionamiento inmediato: Instrucción registro, operando inmediato (el operando reside en la propia instrucción)
 - Por registro: Instrucción registro (el operando reside en un registro interno del procesador)
 - Modo de diseccionado en memoria: el operando reside en la memoria. Cuando se trabaja con 32 bits, el Pentium calcula la dirección efectiva del operando mediante:

Dirección = variable + registro base + registro índice * factor escalar + desplazamiento

Variable: permite precisar un identificador:

Registro Base: apunta al comienzo de una estructura de datos de dirección variable

Registro Índice: mueve el apuntador dentro de las estructuras de datos para acceder a los diferentes campos.

Factor escala: tiene en cuenta el tamaño del elemento

- Desplazamiento: se usa para seleccionar un dato de dirección conocida.
- Cuando se trabaja con operando de 16 bits se emplea:
Dirección = variable + registro base + registro índice + desplazamiento
- Las instrucciones que introducen o sacan datos por la puerta de E/S disponen de un operando que especifica la posición de éstas en el espacio de E/S

CLASIFICACIÓN Y CARACTERÍSTICAS GENERALES DEL REPERTORIO DE INSTRUCCIONES

INSTRUCCIONES ESPECIALES

- ✓ Se clasifican en instrucciones protegidas e instrucciones privilegiadas

INSTRUCCIONES PRIVILEGIADAS

- ✓ Pueden ejecutarse únicamente con el máximo nivel de privilegio
- ✓ Se distinguen cuatro grupos:
 - Cualquier instrucción que pueda modificar el campo IOPL, como IRET, POPF y las relacionadas con la conmutación de tareas, como CLTS
 - Instrucciones que afectan a los registros que hacen referencia a tablas que controlan el sistema en modo Protegido:
 - LGDT-LIDT: para cargar los registros GDTR y IDTR
 - SGDT-SIDT: almacenan los contenidos de GDTR y IDTR
 - LLDT-LTR: cargan a LDT y a TR
 - SLDT-STR: almacenan el valor de LDTR y TR
 - Instrucciones que afectan a las MSW, la palabra baja del CR0
 - LMSW: carga en la MSW el valor del operando
 - SMSW: almacena el valor de MSW
 - Instrucción HLT: lleva al procesador a un estado de parada del que se sale mediante un RESET o una interrupción

INSTRUCCIONES PROTEGIDAS:

- ✓ Mediante el control del campo IOPL el sistema puede establecer dinámicamente en cada tarea, un control riguroso a las E/S
- ✓ Las instrucciones protegidas son:
 - IN, OUT, INS y OUTS: corresponden a instrucciones de E/S de operando y cadenas
 - SEI y CLI: permiso y prohibición de las interrupciones enmascarables

INSTRUCCIONES ARITMÉTICAS

- ✓ AAA: después de realizar una suma, ajusta el último byte del resulta a BCD
- ✓ AAD: ajusta el dividendo para que proporcione el resultado en formato cociente-resto
- ✓ AAM: ajusta el resultado de dos BCD a BCD
- ✓ AAS: después de realizar una resta, ajusta el último byte del resulta a BCD
- ✓ DAA: tras sumar dos BCD, transforma dicho resultado en dos dígitos BCD
- ✓ DAS: igual que DAA pero con resta
- ✓ ADD: suma dos operandos
- ✓ ADC: suma dos operando y el acarreo
- ✓ SUB: resta dos operandos
- ✓ SBB: resta el minuendo del sustraendo más el acarreo
- ✓ DEC: decrementa

- ✓ INC: incrementa
- ✓ MUL: multiplica el operando, sin signo, por AL, AX o EAX dejando el resultado en AX, EAX o EDX:EAX
- ✓ IMUL: multiplica los operandos con signo
- ✓ DIV: divide a AL, AX o EAX por el operando especificando en la instrucción, dejando el resultado en dos registros, en forma de Resto:Cociente
- ✓ IDIV: división con signo
- ✓ CBW/CWD: convierte AL y AX a doble tamaño
- ✓ NEG: realiza el complemento a dos del operando
- ✓ CMP: compara dos operandos, reflejando el resultado únicamente en los señalizadores

NUEVAS INSTRUCCIONES ARITMÉTICAS DEL PENTIUM

- ✓ CWDE: convierte una palabra en doble palabra
- ✓ CDQ: convierte una palabra en cuádruple.

INSTRUCCIONES LÓGICAS

- ✓ AND-OR-XOR-NOT
- ✓ ROL/ROR: rotación a la izquierda o derecha de los bits del primer operando, tantas veces como lo indique CL u otro operando.
- ✓ RCL/RCR: rotación a izquierda o derecha de los bits del primer operando junto con el acarreo, el número de veces especificado por CL
- ✓ TEST: realiza la operación lógica AND de los operandos, sin resultado. Sólo afecta a los señalizadores
- ✓ SAL/SAR: desplazamiento aritmético a izquierda o derecha
- ✓ SHL/SHR: desplazamiento lógico (sin preservar el bit de signo) a izquierda o derecha

NUEVAS INSTRUCCIONES LÓGICAS DE PENTIUM

- ✓ SHLD: desplaza el contenido del primer y segundo operando, conjuntamente, a izquierda el número de veces especificado en el tercer operando.
- ✓ SHRD: realiza el desplazamiento a derecha

INSTRUCCIONES DE MANIPULACIÓN DE CADENAS

- ✓ CMPS-CMPVS-CMPSW: sirven para comparar cadenas, restando el contenido del elemento apuntado por ES:EDI, con el de DS:ESI, afectando sólo los señalizadores
- ✓ MOVS: mueve el contenido de la dirección apuntada por DS:ESI a la referenciada por el puntero ES:EDI, incrementando el valor de los registros índice, según el tamaño en bytes de los operandos
- ✓ LODS-LODSB-LODSW: transfiere al acumulador el elemento apuntado por ES:EDI
- ✓ STOS-STOSB-STOSW: transfiere el contenido del acumulador a la dirección apuntada por ES:EDI
- ✓ SCAS-SCASB-SCASW: resta el contenido diseccionado por ES:EDI el valor de la puerta de E/S apuntada por DX
- ✓ OUTS-OUTSB-OUTSW: saca por la puerta apuntada por DX, uno o dos bytes a partir de la dirección señalizada por ES:EDI
- ✓ REP: repite la instrucción a la que antecede tantas veces como indica el valor del contenido ECX
- ✓ REPE-REPZ: se repite la instrucción a la que antecede hasta que ECX valga 0, o bien el señalizador Z=0

- ✓ REPNE-REPZ: termina la repetición con Z=1 o ECX=0
- ✓ XLAT-XLATB: deposita en AL el contenido de la dirección obtenida al sumar ABX+AL. (para ficheros)

INSTRUCCIONES DE TRANSFERENCIA DE CONTROL

- ✓ JMP: realiza un salto a la dirección que se indica en el descriptor referenciado
- ✓ LOOP: provoca un salto a una etiqueta corta
- ✓ LOOPZ-LOOPE-LOOPNZ-LOOPNE: salto corto mientras ECX no sea 0 y Z=0 o 1
- ✓ CALL: llamada a un salto a una rutina
- ✓ RET: retorno de una rutina

INSTRUCCIONES DE TRANSFERENCIA DE CONTROL ESPECIALES

- ✓ JA: salta si el primer operando es mayor que el segundo. Sin signo
- ✓ JAE: salta si el primer operando es menor que el segundo. Sin signo
- ✓ JB: salta si el primer operando es mejor igual que el segundo. Sin signo
- ✓ JBE: salta si el primer operando es mayor igual que el segundo. Sin signo
- ✓ JE: salta si Z=1
- ✓ JO: salta si el overflow=1
- ✓ JNO: salta si el overflow=0
- ✓ JCXZ: salta si CX < > 0
- ✓ JS: salta si el flag S=1
- ✓ JNS: salta si el bit S=0
- ✓ JNA: salta si el primer operando es mayor que el segundo. Con signo
- ✓ JNAE: salta si el primer operando es menor que el segundo. Con signo
- ✓ JNB: salta si el primer operando es mejor igual que el segundo. Con signo
- ✓ JNBE: salta si el primer operando es mayor igual que el segundo. Con signo
- ✓ JC: salta si el bit de carry=1
- ✓ JNC: salta si carry=0
- ✓ JP: salta si la paridad es impar
- ✓ JNP: salta si la paridad es par

INSTRUCCIONES DE TRANSFERENCIA DE DATOS

- ✓ IN-OUT: entrada y salida de información desde las puertas de E/S
- ✓ POP: se transfiere desde la cima de la pila una información a un registro
- ✓ POPA: transfiere desde la pila ocho palabras que se cargan con los ocho registros generales
- ✓ POPF: transfiere desde la cima de la pila una palabra al registro de señalizadores
- ✓ PUSH: transfiere un registro a la pila
- ✓ PUSHA: transfiere a la cima de la pila ocho registros generales de 16 bits
- ✓ PUSHF: transfiere a la pila el registro de señalizadores
- ✓ MOV: transfiere al primer operando el valor del segundo
- ✓ XCHG: intercambia el contenido de los operandos
- ✓ LEA: transfiere a un registro el valor de la dirección que corresponde al segundo operando

NUEVAS INSTRUCCIONES DE TRANSFERENCIA DE DATOS

- ✓ El Pentium admite nuevas instrucciones para operando de 32 bits, POPAD, POPFS, PUSHFD, PUSHAD. MOVSX extiende el signo de un byte o una palabra a 16 o 32

INSTRUCCIONES DE CONTROL DE LOS SEÑALIZADORES

- ✓ CLC-STC: ponen a 0 o a 1 el señalizador de acarreo
- ✓ CLD-STD: ponen a 0 o a 1 al señalizador D
- ✓ CMC: complementa el valor del señalizador C
- ✓ CLI: pone a 0 el señalizador 1. inhabilita interrupciones enmascarables
- ✓ LAHF: transfiere al registro AH el byte de menos peso del registro de señalizadores
- ✓ SAHF: transfiere el contenido de AH al byte de menos peso del registro de señalizadores

INSTRUCCIONES DE ASIGNACIÓN CONDICIONAL (cambio en el operando, a uno si se cumple)

- ✓ SETA: pone a uno si el primer operando es mayor que el segundo. Sin signo
- ✓ SETAE: pone a uno si el primer operando es mejor que el segundo. Sin signo
- ✓ SETB: si el primer operando es menor igual que el segundo. Sin signo
- ✓ SETBE: si el primer operando es mayor igual que el segundo. Sin signo
- ✓ SETC: si el carry=1
- ✓ SETNC: si carry=0
- ✓ SETE: si Z=1
- ✓ SETO: si OF=1
- ✓ SETNO: si OF=0
- ✓ SETCXZ: si $CX < > 0$
- ✓ SETS: si S=1
- ✓ SETNS: si S=0
- ✓ SETNA: pone a uno si el primer operando es mayor que el segundo. Con signo
- ✓ SETNAE: pone a uno si el primer operando es mejor que el segundo. Con signo
- ✓ SETNB: si el primer operando es menor igual que el segundo. Con signo
- ✓ SETNBE: si el primer operando es mayor igual que el segundo. Con signo
- ✓ SETP: si la paridad es impar
- ✓ SETNP: si la paridad es par

INSTRUCCIONES DE BIT

- ✓ BT: asigna a CF el valor del bit del primer operando, quedando especificada su posición por el segundo operando.
- ✓ BRC: realiza la misma operación que BT, pero también complementa el bit especificado en la instrucción
- ✓ BTR: igual que BT, pero pone a 0 el bit especificado en la instrucción
- ✓ BTS: opuesto BTR
- ✓ BSF: si todos los bits recorridos de menor a mayor peso, del primer operando, especificado por el segundo, son cero Z=1. si se encuentra algún bit a 1, Z=0, guardando en el primer operando la posición del primer bit a 1 que se haya encontrado
- ✓ BSR: igual que la anterior instrucción, pero la exploración se realiza desde el bit de más peso al de menos peso.

INSTRUCCIONES DE ALTO NIVEL

- ✓ BOUND: comprueba que un operando está comprendido entre dos límites y, en caso contrario, genera la excepción 5

- ✓ ENTER: sirve para crear en la nueva pila que se usa, tras una llamada a un procedimiento, un espacio reservado para el uso de parámetros. El primer operando expresa el número de bytes que se conservan en la nueva pila. El segundo expresa el grado de anidamiento.
- ✓ LEAVE: elimina el espacio reservado en la pila del procedimiento saliente, asignando como nueva pila la del procedimiento anterior en anidamiento.

INSTRUCCIONES MULTISEGMENTO

- ✓ CALL: llamada a rutina o procedimiento
- ✓ RET: retorno de un procedimiento
- ✓ INT: llamada a un programa de manejo de una interrupción
- ✓ INTO: llamada a la entrada 4 de la tabla de interrupciones, si OF=1
- ✓ IRET: retorno de un programa de interrupción
- ✓ LDS: carga simultáneamente el registro DS y otro general de 7 bytes de memoria
- ✓ LES: igual que el anterior pero cargando el registro ES
- ✓ JMP: salto a un segmento de código

NUEVAS INSTRUCCIONES MULTISEGMENTO

- ✓ LFS-LGD-LSS: Igual que LDS y LES, pero afectando a los registros FS, GS y SS
- ✓ MOV-POP-PUSH: permiten realizar transferencia de registros de segmento a otros registros de pila.

INSTRUCCIONES DEL SISTEMA OPERATIVO

- ✓ ARPL: comprueba el RPK del primer operando con el del segundo. Se es menor Z=1 e igual los RPL el valor del segmento
- ✓ CLTS: pone a 0 el bits TS de la MSW
- ✓ HLT: detiene la ejecución de un programa
- ✓ LAR: carga en el primer operando los derechos de acceso del descriptor el que hace referencia el segundo operando.
- ✓ LGDT: carga el GDTR desde una posición de memoria
- ✓ LIDT: carga el IDTR desde una posición de memoria
- ✓ LLDT: carga el registro LDTR
- ✓ LMSW: carga a MSW
- ✓ LSL: carga el primer operando con el límite del segundo especificado por el segundo operando
- ✓ LTR: carga el registro TR
- ✓ SGDT-SIDT-SLDT-SMSW-STR: almacena en la memoria a los registros GDTR, IDTR, LDTR, MSW, TR
- ✓ VERR: verifica si puede ser leído el segmento definido por el selector que se proporciona en el operando de la instrucción. En caso afirmativo Z=1
- ✓ VERW: verifica si puede ser escrito un segmento

INSTRUCCIONES PARA EL COPROCESADOR

- ✓ ESC: precede a las instrucciones que debe procesar el procesador matemático
- ✓ WAIT: se detiene el procesador hasta que la patilla BUSY se desactive. Así el procesador espera al coprocesador

OTRAS INSTRUCCIONES

- ✓ NOP: no efectúa operación alguna

- ✓ LOCK: impide la cesión del bus hasta la finalización de la instrucción con LOCK

NUEVAS INSTRUCCIONES DE PENTIUM

- ✓ CMPXCHG8B: compara el valor de 64 bits ubicado en EDX:EAX con un valor de 64 bits situado en memoria. Si son iguales, el valor de memoria se reemplaza por el contenido de ECX:EBX, y el indicador ZF se pone en 1. en caso contrario el valor en la memoria se carga con EDX:EAX y ZF a cero
 - ✓ CPUID: le informa al software acerca del modelo del microprocesador en que está ejecutando.
 - ✓ RDMSR: el valor en ECX especifica uno de los registros de 64 bits específicos del modelo del procesador.
 - ✓ RDTSC: copia el contenido del contador de tiempo en EDX:EAX
 - ✓ RSM: el estado del procesador se restaura utilizando la copia que creó al entrar al modo de manejo del sistema
 - ✓ WSMR: el valor en ECX especifica uno de los registros de 64 bits específicos del modelo del procesador. El contenido de EDX:EAX se carga en ese registro.
-

PENTIUM PRO

CAPÍTULO 20 (Angulo)

CARACTERÍSTICAS GENERALES

INTRODUCCIÓN

- ✓ Arquitectura de 64 bits
- ✓ Contiene dos chips: uno para el procesador y el otro para la caché de segundo nivel
- ✓ La frecuencia de reloj tuvo como límite superior 200 Mhz, partiendo de un mínimo de 150 Mhz.
- ✓ Bus de direcciones de 36 líneas capaz de direccionar hasta 64 giga bytes
- ✓ Desventajas:
 - Alto precio de fabricación
 - La micro arquitectura utilizada estaba optimizada para software de 32 bits.
 - Un consumo de energía y disipación de calor elevados.
- ✓ Microprocesador con los siguientes aspectos arquitectónicos:
 - Integración total de una caché de segundo nivel con el procesador
 - Potenciación del paralelismo mediante la ejecución dinámica de las instrucciones fuera de orden
 - Potenciación del paralelismo mediante la incorporación de más unidades de ejecución que trabaja simultáneamente
 - Orientación a los procesadores RISC
 - Incremento de la velocidad mediante la disminución del ciclo de reloj, aumentando el número de etapas del cause que se eleva a 14

ARQUITECTURA INTERNA

- ✓ Unidad de Bus externo:
- ✓ Unidad de bus trasero: interfaz directa entre el núcleo de la CPU y la caché L2
- ✓ Caché L2 unificada: actúa cuando la caché L1 falla
- ✓ Caché de datos L1: da servicio a las peticiones de carga y almacenamiento de datos

- ✓ Caché de código L1: atiende las peticiones de búsqueda de instrucciones formuladas por la unidad de prebúsqueda de instrucciones
- ✓ Microprocesador: responsable de:
 - Búsqueda de instrucciones
 - Predicción de saltos
 - Análisis del flujo de instrucciones CISC y RISC
 - Despacho, ejecución y retirada de micro operaciones
- ✓ Unidad local APIC: responsable de recibir las peticiones de interrupciones de otros procesadores, de las patitas propias de interrupciones y del temporizador APIC

APORTACIONES Y NUEVOS RECURSOS ARQUITECTÓNICOS

REGISTROS DE PROPÓSITO GENERAL

- ✓ El Pentium-Pro soluciona el problema de los escasos registros con 40 registros nuevos. Las instrucciones son traducidas a micro operaciones tipo RISC antes de ser ejecutadas.
- ✓ En vez de una carga inmediata de los resultados de la instrucción en los registros reales, el microprocesador almacena el resultado. Si una vez ejecutada, otra operación requiere el resultado producido por las micro operaciones que le preceden, los resultados son enviados a la micro operación peticionaria.

ELIMINACIÓN DE DEPENDENCIAS

CACHÉ L1 Y L2

- ✓ Caché interna L1 de 8 KB destinada a contener instrucciones y otra del mismo tamaño para contener datos. De de instrucciones es asociativa de dos vías y la de datos de cuatro
- ✓ La caché de segundo nivel L2 tiene un tamaño de 256 KB y una estructura asociativa de cuatro vías. Tiene una conexión mediante el bus trasero con el procesador lo que permite funcionar a la misma frecuencia que el procesador pudiendo transferir 64 bits por el bus de datos en un ciclo de reloj.
- ✓ El procesador se comunica con la memoria principal con los módulos de E/S mediante el bus frontal.
- ✓ Pentium-Pro opera en forma independiente y simultánea con el bus frontal y con el trasero. Mientras espere que se complete un acceso a la memoria, se puede iniciar otro, para lo cual existe un Buffer de Ordenamiento de Acceso a Memoria (MOB), que guarda hasta 8 accesos o “transacciones”, pudiendo iniciarse un acceso aunque existen 8 pendientes de realizar.
- ✓ El MOB actúa en coordinación con el ROB (Buffer de Reordenamiento) que almacena las peticiones de lectura y escritura pendientes de ejecutarse hasta que desaparecen las causas que les impiden.
- ✓ Emplea el proceso MESI.

APLICACIÓN DE LA TÉCNICA RISC

- ✓ El problema de la creciente complejidad de las instrucciones y el intento por mantener compatibilidad con viejos sistemas, y los problemas inherentes a las instrucciones RISC se eliminan en los procesadores con arquitectura CISC. Estas últimas instrucciones son simétricas, tiene el mismo tamaño casi todas, son poco numerosas y se ejecutan en un ciclo.

- ✓ Para aprovechar las ventajas de los procesadores RISC, el Pentium-Pro transforma las instrucciones en CISC x86 en microinstrucciones RISC en el bloque denominado Unidad de Decodificación, que consta de cuatro bloques
- ✓ Las instrucciones son pre-buscadas en la memoria y almacenadas en la caché L2 y en la caché de código L1. el microprocesador decodifica el bloque de instrucciones que recibe en microoperaciones.
- ✓ Se puede obtener hasta un máximo de seis microoperaciones por ciclo de reloj.

SUPERSEGMENTACIÓN

- ✓ Consta de 1 etapas
 - 1^{ra}: se calcula el valor que hay que cargar en el contador de programa (EIP) para apuntar a la dirección de la siguiente instrucción
 - 2^{da}, 3^{ra}, 4^{ta}: se destinan a la búsqueda de las instrucciones. Estas proceden de la caché L1 de instrucciones. Para delimitar las instrucciones se toman los 16 bytes que apunta el EIP y se introducen a uno de los decodificadores. una vez delimitada se introducen al decodificador de instrucciones que transforma el código binario en secuencia de microprocesadores RISC
 - 7^{ma}: se encarga de realizar un renombrado de los registros de propósito general que emplean las instrucciones para los números enteros y de coma flotante. Pentium-Pro dispone en esta etapa de una Tabla de Alias de Registros (RAT)
 - 8^{va}: se encuentra el ROB (buffer de reordenamiento), que es una CAM con 40 entradas de 256 bits que puede contener cada una, una micro operación con lo que le corresponde. En esta etapa está también la BTB
 - Tras el ROB se encuentra RS que es una Estación de Reserva que tiene como objetivo planificar el orden con que se van a entregar las microoperaciones a las Unidades de Ejecución que comprenden las etapas 10^a y 11^a.

ARQUITECTURA SUPERESCALAR

- ✓ Existen cinco unidades de ejecución que trabajan de forma simultánea e independiente, alcanzando un máximo de 5 microoperaciones por ciclo de reloj
- ✓ Dos unidades de ejecución (IU1 e IU2) trabajan sobre microoperaciones que manejan números enteros, otras dos (AGU1 y AGU2), calculan las direcciones de los operandos y las instrucciones de carga y almacenamiento, y la FPU que trata instrucciones con números de coma flotante.
- ✓ Los resultados producidos sobre las microoperaciones por la unidad de ejecución vuelven a la RS y al ROB. El ROB determina cuando se ha completado y puede ser retirada.
- ✓ La Unidad de Extracción puede tratar en cada ciclo hasta la terminación de tres microoperaciones, por lo que se dice que Pentium-Pro es un “procesador superescalar en tres vías”

NUEVAS INSTRUCCIONES

- ✓ Mover datos condicionalmente, esto permite que las bifurcaciones de datos dependientes sea eliminada.

ANÁLISIS DEL RENDIMIENTO

EJECUCIÓN DINÁMICA

- ✓ La ejecución fuera de orden se consigue explorando 20 a 30 instrucciones por delante de la que se halla en curso.
- ✓ Una vez las instrucciones en el ROB, este se comunica con las tres unidades operativas del procesador. La Unidad de Selección y Ejecución explora las microoperaciones del ROB determina cuales se pueden ejecutar y devuelve los resultados al ROB para que la unidad de Extracción las retire y almacene.
- ✓ La ejecución dinámica se implementa mediante el uso de: Predicción de ramificaciones múltiples, análisis del flujo de datos y ejecución especulativa.

ORIENTACIÓN DEL SOFTWARE DEL PENTIUM-PRO

- ✓ Orientado a sistemas operativos de 32 bits.
-

PENTIUM II

CAPÍTULO 22 (Angulo)

CARACTERÍSTICAS GENERALES

INTRODUCCIÓN

- ✓ Emplea como soporte la ranura “Slot One” y el encapsulado SECC (“Single Edge Contact Cartridge”)
- ✓ La memoria caché L2 y algunos componentes auxiliares están soportados sobre una tarjeta de circuito impreso
- ✓ Al no contener la caché L2 su rendimiento es mejor que el de Pentium-Pro
- ✓ 233 y 266 Mhz
- ✓ Otras cualidades:
 - Arquitectura de bus dual e independiente
 - Tecnología MMX
 - Tecnología de ejecución dinámica
 - Cartucho de contactos de contacto simple
- ✓ Características que potencia el procesador en el cálculo de coma flotante, las operaciones multimedia, y el cálculo con números enteros.

VERSIONES DEL PROCESADOR

- ✓ Celeron: se trata de una versión económica. Posteriormente incluyó una caché L2 de 512 KB. Este procesador está optimizado para aplicaciones de 32 bits y se comercializa en frecuencias de 266 hasta 466 Mhz
- ✓ Xeon: es un Pentium II de alto rendimiento. Dispone una caché L2 de tamaño comprendido entre 513 KB y 2 MB y se presenta en un cartucho con conexión “Slot Two” que incorpora un sensor de temperatura para evitar el sobrecalentamiento.

APORTACIONES Y NUEVOS RECURSOS ARQUITECTÓNICOS

APORTACIONES GENERALES

- ✓ Traducen a microoperaciones RISC
- ✓ Es un procesador supersegmentado en 12 etapas y tiene 5 líneas de ejecución
- ✓ Existen dos buses diferenciados que se denominan “doble bus” de datos, uno para comunicarse con la memoria principal y otro con la caché LW”

- ✓ La caché L2 es una memoria BSRAM (Burst static RAM) con un tamaño de 256 o 512 KB.

MEMORIA CACHÉ

- ✓ Para abaratar costos la caché L2 está separada y solamente alcanza la mita de la frecuencia que la CPU
- ✓ Tiene 32 KB de caché de primer nivel, separados en dos bloques de 16 KB para el código y datos.

ALIMENTACIÓN

- ✓ Emplea diversos estados de bajo consumo de energía que persiguen el ahorro de energía durante tiempos muertos.
- ✓ Maneja una unidad de suministro de energía capaz de aceptar una señal de 5 bits que selecciona el voltaje necesario.
- ✓ Posee un ventilador que permanece siempre disponible y está construido sobre el encapsulado SECC

ARQUITECTURA DE DOBLE BUS INDEPENDIENTE

- ✓ Esto es para resolver las limitaciones en el ancho de banda
- ✓ Estos buses posibilitan acceder a datos desde cualquiera de ellos simultáneamente y en paralelo.
- ✓ La CPU se comunica de forma independiente con la caché y con el bus de sistema

EJECUCIÓN DINÁMICA ESPECULATIVA Y FUERA DE ORDEN

- ✓ Posibilita al procesador manejar más eficientemente los datos a procesar, incrementando el paralelismo en la ejecución de las instrucciones en un mismo período de tiempo.

UNIDAD DE PREDICCIÓN DE SALTOS

- ✓ Cuando detecta y ejecuta por primera vez una instrucción de bifurcación apuntan en una tabla (BTB), la dirección de la instrucción, la dirección del salto y un histórico que indica la última vez que se ha realizado un salto o no.
- ✓ La primera vez que se trata una de estas instrucciones condicionales se especula el resultado del destino con una política prefijada. Las siguientes veces se hace la especulación según resultados anteriores
- ✓ La BTB tiene 512 entradas, organizadas como una caché asociativa de cuatro vías

NUEVAS INSTRUCCIONES

INSTRUCCIONES MMX

La tecnología MMX posibilita al procesador mejorar la ejecución de aplicaciones en las que se maneja audio, video y gráficos a través del procesamiento de datos en paralelo.

PENTIUM IV

CAPÍTULO 24 (Angulo)

CARACTERÍSTICAS GENERALES

- ✓ Proporciona altas prestaciones para procesar video y audio
- ✓ La micro-arquitectura NetBurst le permite funcionar a velocidades altas.
- ✓ Se abandona la conexión por Slot y se conecta en el Socket 423
- ✓ La memoria RAM es de tipo RIMM que trabaja a altas velocidades
- ✓ La caché L1 es de 20 KB, 12 para instrucciones y 8 para datos.
- ✓ Caché L2 de 256 KB

APORTACIONES Y NUEVOS RECURSOS ARQUITECTÓNICOS

- ✓ Tecnología hiper segmentada: 20 etapas permitiendo alcanzar mayores frecuencias
- ✓ Bus del sistema a 400 Mhz: mejoran el rendimiento de aplicaciones “profesionales” y multimedia y juegos 3D.
- ✓ Duplicación de unidades aritmético-lógicas: dos unidades de números enteros que consiguen tiempos de espera iguales a un semiciclo de reloj
- ✓ Caché y otros recursos: la caché L2 está integrada en la cápsula y tiene un bus de datos de 256 bits. Dispone de técnicas que mejoran la ejecución especulativa, pudiendo romper el orden de las instrucciones para acelerar su procesamiento y reforzando la predicción de saltos y ramificaciones.
- ✓ Descripción de la arquitectura: la primera etapa es la Bebo, después la instrucción llega al decodificador, después al renombramiento y se almacenan en una cola.

NUEVAS INSTRUCCIONES

- ✓ Las nuevas instrucciones SSE2 constan de 144 nuevas instrucciones de tratamiento de datos enteros y reales en simple y doble precisión.
- ✓ El tamaño de los operandos es de 128 bits, duplicando la longitud de palabra
- ✓ La ventaja es el aumento de rendimiento en la reproducción de audio y video. Además facilita la codificación y cifrado de información con claves de gran longitud.
- ✓ La desventaja es que los softwares no están preparados todavía.

ITANIUM

CAPÍTULO 25 (Angulo)

CARACTERÍSTICAS GENERALES

INTRODUCCIÓN

- ✓ 64 bit pudiendo direccionar más memoria (16 terabytes)
- ✓ Procesa el doble de instrucciones por ciclo.

ARQUITECTURA BÁSICA

- ✓ Frecuencia de 733 y 800 Mhz.
- ✓ Realiza hasta seis instrucciones a la vez
- ✓ La etapa fetch (que se encarga de las predicciones) está basada en:
 - Predicción: predicción en ramas
 - Especulación: puede cargar instrucciones y datos a la CPU antes que sean necesarios limitando los efectos de las burbujas.
- ✓ En la segunda etapas, se dirigen hacia la zona donde se manejan los registros que consta de un juego de 128 registros para números enteros (de 64 bits cada uno) , 128 registros para números en coma flotante (de 82 bits cada uno), 8 registros para

saltos y varios registros con diversas funciones. El bloque Snack Engine se encarga de reducir el tiempo para las llamadas y retornos de subrutinas.

- ✓ Después pasa a la etapa de Recursos paralelos (etapa de ejecución) donde existen cuatro unidades de números enteros, cuatro para instrucciones MMX, la FRU con dos FMACS y la tabla ALAT con 32 entradas para direcciones adelantadas.
- ✓ Las etapas de búsqueda de instrucciones, predicción de saltos y buffer de desacoplamiento constan de los siguiente bloques funcionales:
 - IPG: generación de instrucciones
 - FET: búsqueda de instrucciones
 - ROT: rotación de registros para generar las burbujas necesarias para evitar la dependencia de datos.
- ✓ En las etapas de emisión de instrucciones y renombrado de registros residen:
 - EXP: tiene como misión repartir las seis instrucciones por los diferentes puertos
 - REN: renombra los registros en los lugares adecuados de la pila, para optimizar su tamaño
- ✓ Las etapas de acceso a operandos, y marcaje y predicción consta de:
 - WLD: decodificación
 - REG: lectura de operandos
- ✓ Para la ejecución y terminación de instrucciones:
 - EXE: ejecución
 - DET: detección de excepciones
 - WRB: Post-escritura

APORTACIONES Y NUEVOS RECURSOS ARQUITECTÓNICOS

- ✓ Procesador de 64 bits
- ✓ Memoria caché: L1 de 32 KB y L2 de 96 KB integrados. Una externa de 2 a 4 MB
- ✓ Mejora en operaciones de coma flotante: 128 registros de 82 bits
- ✓ Compatibilidad con las instrucciones de 32 bits
- ✓ Registros de monitorización de prestaciones
- ✓ Un elevado número de registros
- ✓ EPIC (Explicitly Parallel Instruction Computing): quince unidades de ejecución. Capaces de ejecutar hasta 20 instrucciones por ciclo de reloj. EPIC puede indicar al procesador qué elementos del programa pueden ser ejecutados con total paralelismo

NUEVAS INSTRUCCIONES

- ✓ Compatibilidad con las instrucciones de tipo IA-32. que son paquetes de instrucciones de 128 bits que pueden albergar hasta tres instrucciones.
- ✓ Se usan técnicas para la mejora del procesamiento. La principal es la del procesamiento de más de una instrucción en la misma ranura de tiempo.