

---

# ARQUITECTURA DEL PENTIUM

---

# 6

<b>6.1.- Introducción .....</b>	<b>2</b>
<b>6.2.- Arquitectura interna del pentium .....</b>	<b>3</b>
6.2.1.- Subsistema de Memoria Caché .....	5
6.2.2.- Unidad de enteros superescalar .....	7
6.2.3.- FPU: coprocesador matemático .....	10
6.2.4.- Sistema de predicción de saltos condicionales .....	12
6.2.5.- BIU: unidad interfaz con el bus .....	13
6.2.5.1.- Monitor de prestaciones .....	14
6.2.5.2.- Bus a ráfagas .....	14
6.2.5.3.- Unidad de redundancia funcional .....	15
<b>6.3.- Modos de funcionamiento del pentium .....</b>	<b>15</b>

## 6.1. INTRODUCCIÓN

El 17 de mayo 1993 Intel lanzó al mercado el primer microprocesador Pentium (5ª generación), y bastó solo un lustro para que el mundo entero se inundara de millones de computadoras que soportaban diferentes versiones de este microprocesador. En el segundo semestre de 1993 se consiguieron vender 100.000 Pentium a un precio de 958 dólares cada uno, cifra poco relevante si comparamos con los 30 millones de 80486 que vendieron ese mismo año y muchísimo menos si se tiene en cuenta los miles de millones de diversos Pentium que se han fabricado. Hoy en día el computador se ha convertido en una herramienta imprescindible en la vida de los seres humanos.

El microprocesador Pentium sigue teniendo una arquitectura interna IA-32 de 32 bits, similar a la de sus predecesores 80386 y 80486. Además mantiene la compatibilidad binaria de software de manera que se ejecutan en los nuevos todo lo que se desarrolló a partir del 8088. Esta política comercial ha proporcionado a Intel su indiscutible éxito de ventas dejando atrás así a sus competidores más directos como Motorola. El Pentium es básicamente evolutivo por lo que se ha denominado como un diseño “80486+”.

El primer modelo de Pentium, el Pentium I fue construido con 3,1 millones de transistores (muchos más que el 8086 que poseía 29.000 transistores aproximadamente), fabricado con una tecnología BICMOS de 0,8 micras y funcionaba con una frecuencia de 60 MHz. Utilizaba una tensión de alimentación de 5 voltios y su rendimiento era de 100 MIPS-VAX. En el pasado año, en el 2002, el Pentium IV ya contenía 42 millones de transistores (se puede apreciar un gran aumento en el número de éstos). Su tecnología ya era de 0,3 micras, usando una frecuencia de 2,4 GHz y utilizando un voltaje inferior a 3 voltios, consiguiendo un rendimiento de 500 MIPS-VAX.

En la siguiente tabla podemos apreciar la evolución de las primeras versiones del Pentium:

PROCESADOR	FRECUENCIA	TECNOLOGÍA	VOLTAJE	BUS	MULTIPLICADOR
P60	60 MHz	0,8 µ	5 V	60 MHz	-
P66	66 MHz	0,8 µ	5 V	66 MHz	-
P75	75 MHz	0,6 µ	3,52 V	50 MHz	1,5
P90	90 MHz	0,6 µ	3,52 V	60 MHz	1,5
P100	100 MHz	0,6 µ	3,52 V	66 MHz	1,5
P120	120 MHz	0,35 µ	3,52 V	60 MHz	2
P133	133 MHz	0,35 µ	3,52 V	66 MHz	2
P150	150 MHz	0,35 µ	3,52 V	60 MHz	2,5
P166	166 MHz	0,35 µ	3,52 V	66 MHz	2,5
P200	200 MHz	0,35 µ	3,52 V	66 MHz	3

Tabla 6.1 – Tabla con las características relevantes de las primeras versiones de Pentium

El incremento del rendimiento y la frecuencia de trabajo supusieron un problema para el primer Pentium. El inconveniente consistía en la alta temperatura que podía llegar a alcanzar debido al aumento de transistores y a la miniaturización de los mismos. Había que disipar 16 vatios cuando se trabajaba a la máxima potencia ( 5 Voltios x 3,2 Amperios = 16 Watios ). Por este motivo, se disminuyó la tensión de alimentación hasta 3,3 VDC. La solución era clara, conseguir un mejor sistema de refrigeración.

Existen dos segmentos de mercado para los cuales un ordenador basado en el procesador de la familia Pentium puede ser una solución muy aconsejable. Uno de ellos es el de los ordenadores personales con altas prestaciones, y el otro para el uso como servidores de redes de área local y sistemas multiprocesador. Como ordenador de sobremesa, los procesadores de la familia Pentium tienen la enorme ventaja de que sobre él funcionan todos los principales sistemas operativos, como pueden ser UNIX, Windows 95, Windows NT, OS/2, Solares... Todas las aplicaciones actuales funcionan mucho más rápido bajo un Pentium gracias a sus grandes prestaciones y especialmente se desarrollan estos procesadores para aumentar al máximo las posibilidades de las aplicaciones 3D, de tratamiento de imágenes, de vídeo, sonido y de reconocimiento de la voz.

Por último, mencionar que una novedad que incorporó el Pentium fue la división de la memoria caché de primer nivel L1 en dos secciones independientes de la misma capacidad (8 KB), una dedicada a las instrucciones y otra a los datos. También destaca la ampliación del bus de datos externos a 64 líneas bidireccionales y la potenciación de la Unidad de Coma Flotante, que hasta este modelo, Intel no había prestado mucho interés.

## 6.2. ARQUITECTURA INTERNA DEL PENTIUM

El procesador Pentium es un miembro de la familia Intel de microprocesadores de propósito general de 32 bits. Al igual que los demás miembros de su familia, el 80386 y el 80486, su rango de direccionamiento es de 4 GBytes de memoria física o principal, ya que sus direcciones físicas son de 32 bits ( $2^{32} = 4\text{GB}$ ) y 64 TBytes de memoria virtual, ya que sus direcciones virtuales son de 46 bits ( $2^{46} = 64\text{TB}$ ). Proporciona unas prestaciones más elevadas gracias a una arquitectura mucho más optimizada. Su bus de datos es de 64 bits, el de direcciones es de 32 bits y el de control de 1 bit.

A continuación vemos en la figura 6.1, un diagrama de bloques simplificado. En él, podemos apreciar los cinco bloques (uno de los cuales se subdivide en tres partes) en los que se divide la arquitectura interna del pentium. Estos bloques son:

- Subsistema de memoria caché
- Unidad de enteros superescalar
- FPU: coprocesador matemático
- Sistema de predicción de saltos condicionales
- BIU: unidad de interfaz con el bus
  - Monitor de presentaciones
  - Bus a ráfagas
  - Redundancia funcional

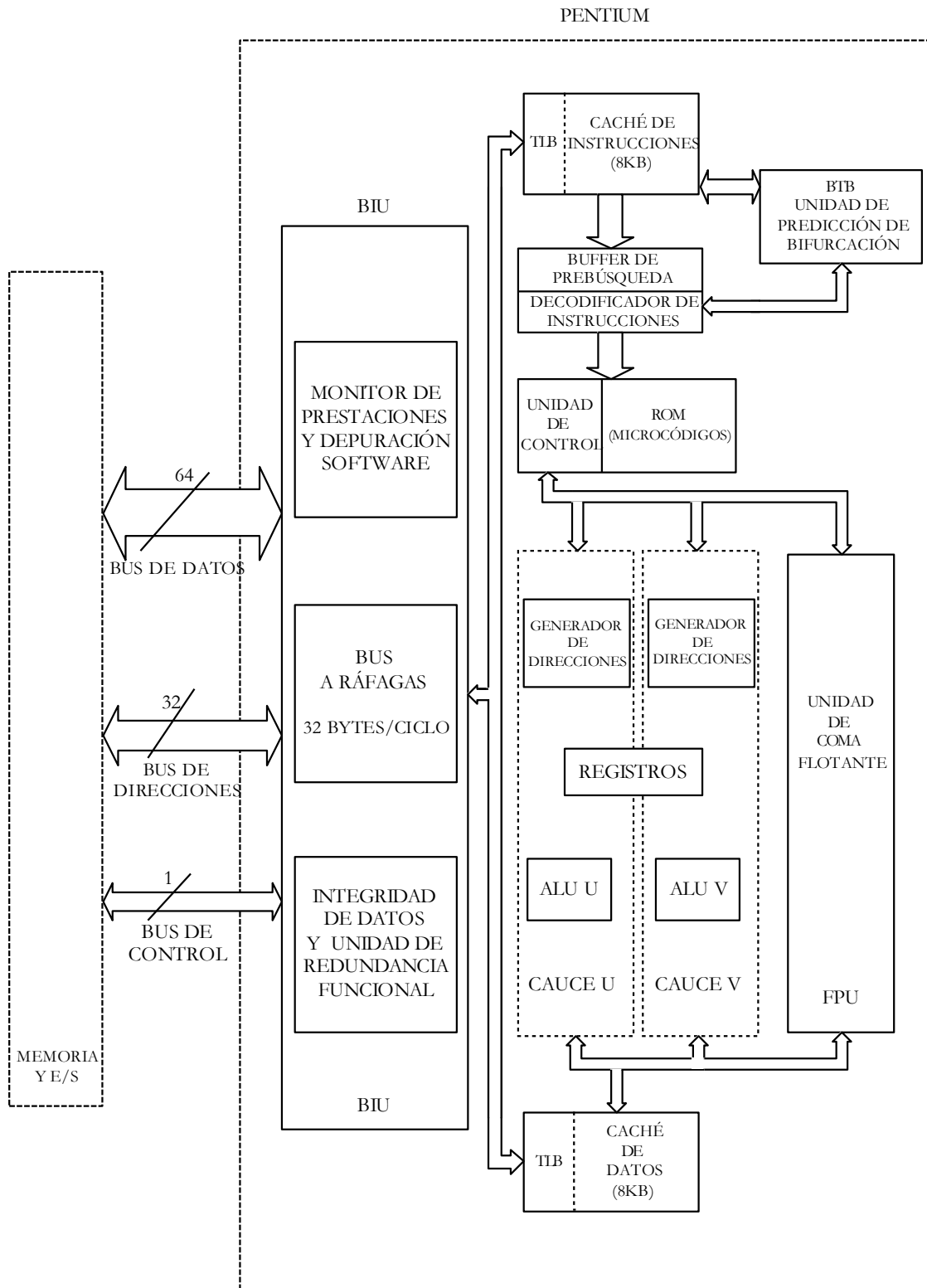


Figura 6.1 – Arquitectura del Pentium

### 6.2.1. Subsistema de memoria caché

El subsistema de memoria caché se compone de dos memorias caché independientes de 8KB cada una: una para almacenar instrucciones (código) y otra para almacenar datos. Este esquema acelera las prestaciones y la capacidad de transferencia del procesador. Por ejemplo, durante la prebúsqueda las instrucciones se obtienen de la memoria caché de instrucciones. Si hubiese una única memoria caché, no podría realizarse un acceso a un dato al mismo tiempo. Con memorias caché independientes para instrucciones y datos ambas operaciones, de prebúsqueda y acceso a datos, pueden realizarse simultáneamente.

Las dos memorias cachés son memorias asociativas de dos vías (ver figura 6.2) que utilizan como unidad de información una línea que es de 32 Bytes; 8 dobles palabras, es decir  $8 \times 2 \times 16 \text{ bits} = 256 \text{ bits} = 32 \text{ Bytes}$  (el doble que en su antecesor 80486), ya que el bus externo del Pentium es de 64 bits. De este forma, en un acceso de tipo ráfaga se puede llenar una línea completa de caché, igual que ocurría con el 80486. Los buses independientes que abastecen a las cachés internas desde la unidad de bus externo son de 64 bits cada uno.

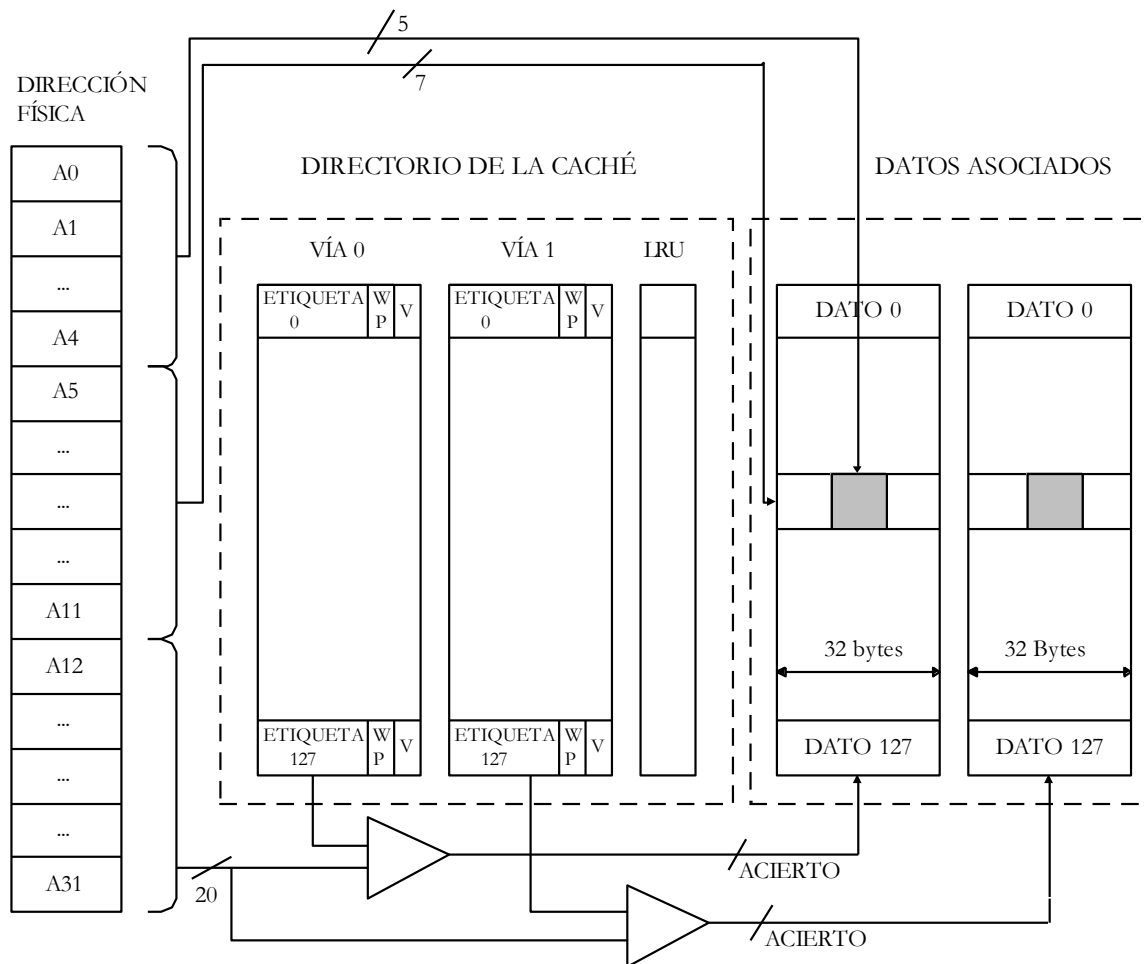


Figura 6.2 – Arquitectura del Pentium

Como podemos observar en la figura 6.3, el bus que parte de la caché de datos es de 64 bits, mientras que el que conecta la caché de instrucciones con los registros de prebúsqueda de instrucciones es de 256 bits.



Figura 6.3 – Caché de datos y caché de instrucciones del Pentium

La caché de datos utiliza el protocolo MESI (modified exclusive shared invalid) para asegurar la consistencia de datos entre la memoria principal y los cachés de todos los procesadores que integren el sistema multiprocesador, es decir, puede haber varias cachés y puede darse que dos CPU's estén empleando la misma posición de la memoria principal. Así dicha posición se encontrará en ambas cachés. Por este motivo, dicho protocolo asegura que se lea el dato que este más actualizado. Por ello, cada línea de la caché puede tener cuatro estados diferentes: M (modificado, por lo que la línea no está actualizada en la memoria principal), E (exclusivo, si únicamente se encuentra en esa caché), S (simultaneo, si la línea está repetida en varias cachés) e I (inválida, si su contenido no sirve).

Cuando se precisa almacenar instrucciones o datos en la caché correspondiente y está totalmente ocupada con valores válidos, se usa el algoritmo de sustitución de líneas LRU (Last Recently Used). Se reemplaza la línea menos recientemente usada, es decir, la que hace más tiempo que no se usa. Por ejemplo, cuando se accede a una página, el bit accedido A, que se encuentra en el registro de control CR3, se pone a 1. Este bit lo maneja el sistema operativo para llevar la cuenta del número de accesos que tiene cada página. De este modo, el algoritmo LRU consulta este número de accesos para eliminar de la memoria, cuando ésta esté llena, la página que menos se hay usado recientemente.

Las caches son de escritura obligada (Write Back), mientras que su antecesor 80486 era de escritura inmediata, es decir, se almacenaban en la caché y en la memoria principal al mismo tiempo. Que una caché sea de escritura obligada implica que los resultados de las operaciones no se transfieren a memoria principal sino que se quedan dentro de la caché hasta que sea preciso actualizarla, porque no queda otro remedio. Su principal ventaja por tanto es que nunca da fallo.

Existen dos situaciones que obligan a actualizar la memoria principal:

- Cuando se va a machacar una línea de la caché porque está llena y ésta no ha sido transferida a memoria principal.
- Cuando algún otro procesador, por ejemplo DMA intenta acceder a una posición de memoria principal cuyo dato está en la caché y ha sido modificado por la CPU. Por tanto, habrá que detener este acceso hasta que se actualice la memoria principal. Una vez actualizada, se prosigue con el acceso.

En el tipo de escritura obligada, al no existir transferencias de resultados a la memoria principal, las operaciones se terminan antes.

Por último, cabe destacar que dos cachés que integran el subsistema de memoria caché son de primer nivel (L1) y admiten una interconexión fácil con caches de segundo nivel (L2) con lo que se aumentan significativamente las prestaciones.

### 6.2.2. Unidad de enteros superescalar

Primeramente, aclararemos el significado de la palabra superescalar. Este término hace referencia a que en su interior existe más de una unidad de ejecución dedicadas a realizar las mismas funciones. Así queda reflejado en la siguiente figura 6.4.

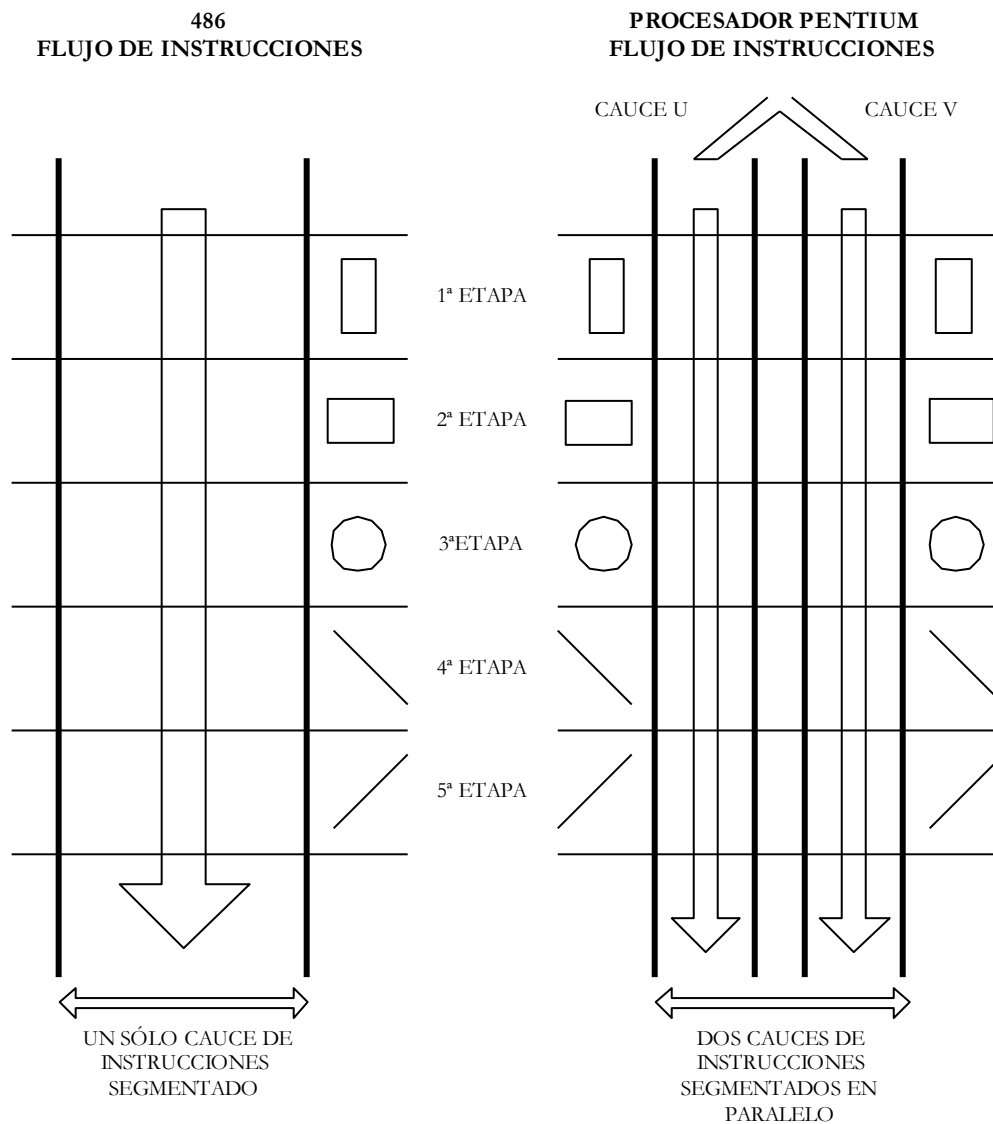


Figura 6.4 – Arquitectura superescalar del Pentium

En este caso, existen dos cauces (U y V) que operan en paralelo para ejecutar las instrucciones de números enteros. Son independientes entre sí, ya que son capaces de funcionar uno independientemente del otro. Es como si existieran dos procesadores del tipo 80486 trabajando al mismo tiempo, por lo que el Pentium podría proporcionar dos resultados enteros por

ciclo de reloj, es decir, ejecutamos dos instrucciones en cada ciclo de reloj. Cada unidad de enteros tiene un cauce segmentado de instrucciones de cinco etapas:

- Prebúsqueda de instrucciones
- Decodificación
- Cálculo de la dirección efectiva (búsqueda de operandos)
- Ejecución
- Escritura de los resultados

Estas etapas del cauce segmentado siguen el orden de la siguiente figura 6.5.



Figura 6.5 – Etapas del cauce segmentado

Cada unidad de proceso interno tiene su propia unidad aritmético lógica (ALU) con un circuito de generación de direcciones exclusivo y un interfaz específico con la memoria caché de datos. Los resultados de las operaciones se almacenan en la caché interna y no se transfieren a la memoria principal a no ser que sea necesario. Estas dos ALUs tienen características diferentes.

La ALU A que pertenece al cauce U es más simple que la del cauce V. El cauce U sólo ejecuta instrucciones simples y del núcleo RISC. Por otra parte la ALU B correspondiente al cauce V es más potente y utiliza instrucciones complejas de tipo CISC.

El bloqueo en la ejecución paralela de instrucciones se realiza de forma transparente para el software y el usuario, y también cuando existe dependencia entre los operandos de las instrucciones. Por ejemplo, si una instrucción realiza una operación que deja el resultado en el registro EDX, la siguiente si utiliza el registro EDX como uno de los operandos origen para cualquier otra operación.

El Pentium intenta paralizar al máximo la ejecución de instrucción, sólo si se garantiza la integridad de los datos, y es capaz de ejecutar alrededor de 1,3 instrucciones por cada ciclo de reloj, rompiendo por tanto la mítica barrera de conseguir la ejecución de la instrucción en cada ciclo de reloj.

Ahora pasaremos a explicar el funcionamiento de la segmentación. La unidad de prebúsqueda manda una dirección a la caché de instrucciones. Si la caché tiene dicha dirección manda una línea de información (32 bytes) a uno de los buffer de prebúsqueda que a su vez pasará la dirección en cuestión a la unidad decodificadora donde decodificará la información. Inicialmente las instrucciones son decodificadas para ver si pueden ser ejecutadas a la vez. En caso afirmativo,



una instrucción irá al cauce U y otra al V para realizar simultáneamente, ya que no existen dependencias entre ellas. En caso contrario, es decir, que existen dependencias entre ellas, la primera deberá completar su ejecución antes de que comience la segunda.

Cuando se predice un salto, la dirección de esta instrucción es demandada por la caché de instrucciones. Si se encuentra allí, una línea de código se manda al otro buffer de prebúsqueda de tal manera que se impide cualquier retraso cuando la instrucción branch se ejecute. Si no hay instrucciones de este tipo ambos cauces son tratados conjuntamente, realizando las prebúsquedas linealmente.

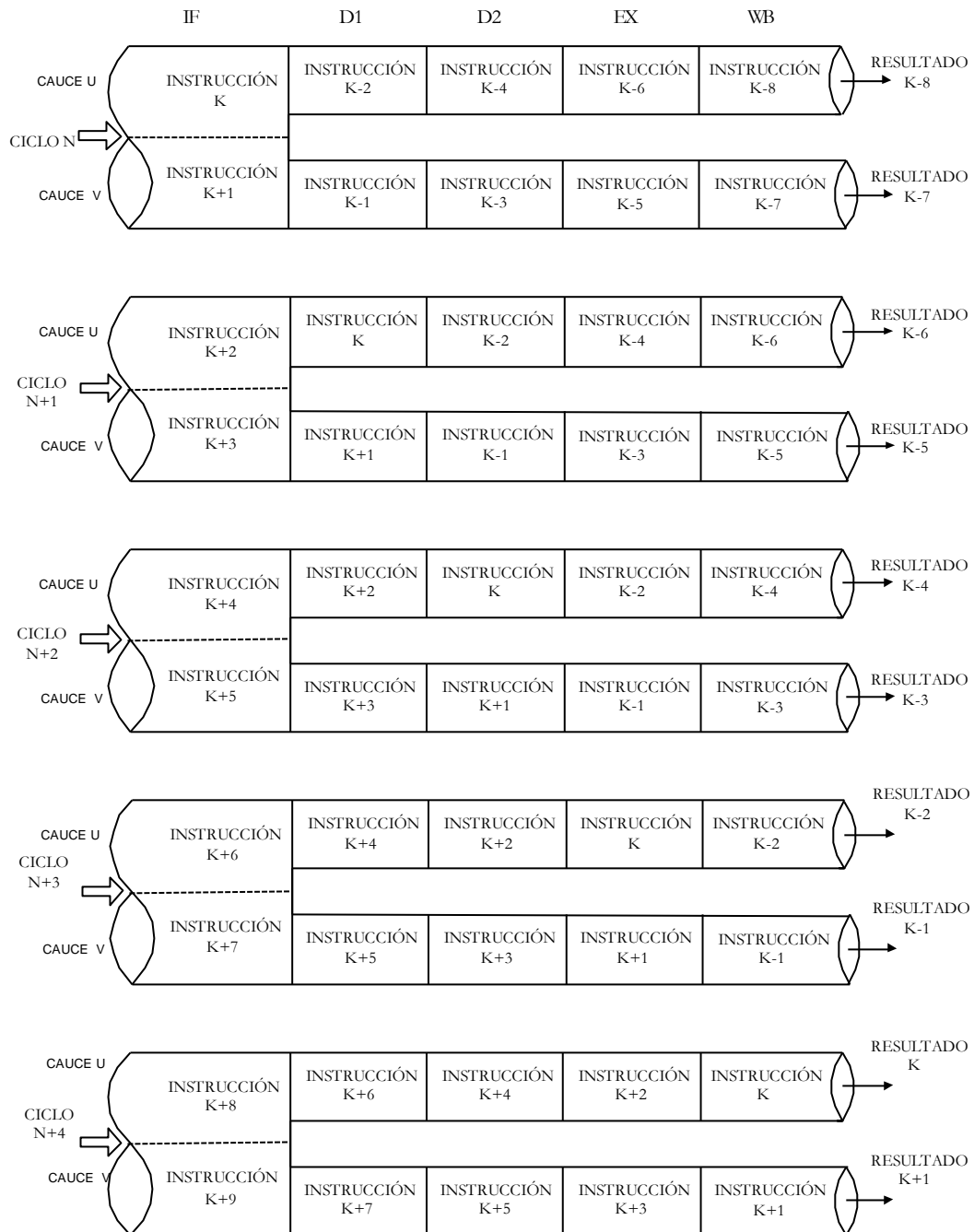


Figura 6.6 – Paso de las instrucciones a través de los dos cauces

Con esta arquitectura se pueden introducir y obtener dos instrucciones en cada etapa del cauce (ver figura 6.6). A esto en inglés se le denomina “pairing” o emparejamiento. Durante el primer ciclo de reloj un par de instrucciones realizan la prebúsqueda; en el segundo ciclo de reloj las dos instrucciones se tratan en paralelo, una en el cauce U y otra en el cauce V , claro está, sino existen dependencias entre ambas instrucciones. En un tercer ciclo de reloj se decodificarán para q en el último se ejecuten. Por todo ello se deduce que el máximo número de instrucciones que puede ejecutar el Pentium son dos.

### 6.2.3. FPU: Coprocesador matemático

La FPU es un coprocesador que opera con unidades enteras de otros procesadores, de los cuales coge sus instrucciones desde el mismo decodificador y secuenciador que la unidad de enteros, compartiendo con esta última el bus del sistema. Cabe destacar que la unidad de enteros y la FPU operan independientemente y en paralelo, como podemos observar en la figura 6.7.

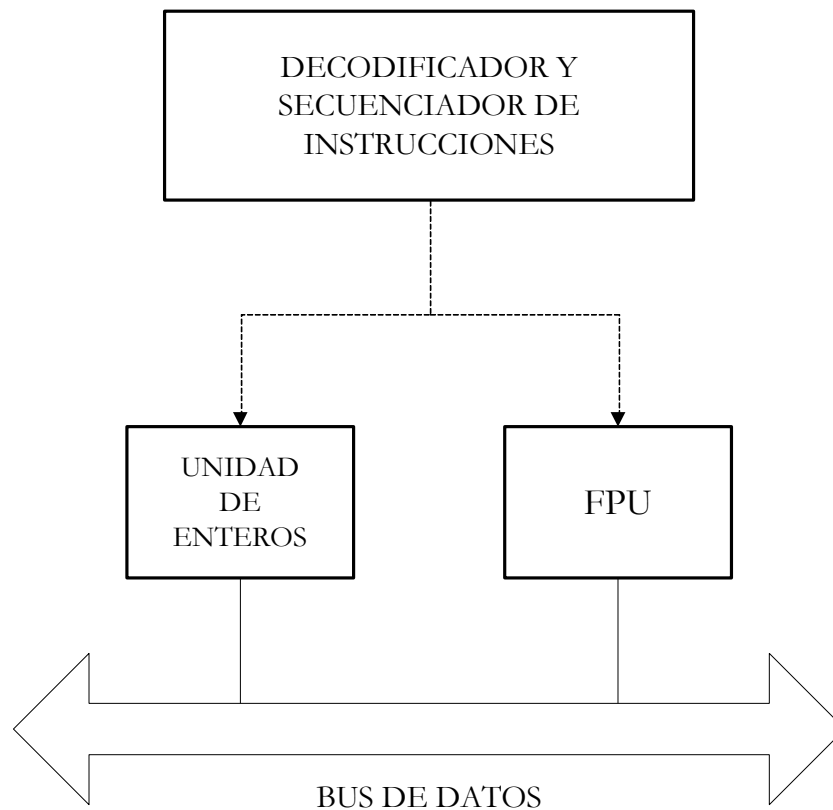


Figura 6.7 – Relación de la FPU con la unidad de enteros

Este bloque se ha rediseñado totalmente respecto al que se utilizaba en el 80486. Sin embargo sigue siendo compatible 100%. Utiliza nuevos algoritmos que aceleran la ejecución de las operaciones (hasta tres veces más rápido que con el 80486) e incluye elementos de hardware dedicados (un multiplicador, un sumador y un divisor).

La FPU consta de un cauce segmentado de instrucciones de 8 etapas que permite obtener resultados partiendo de instrucciones de coma flotante en cada ciclo de reloj. Las etapas son las siguientes:

- 1° : Prebúsqueda de instrucciones
- 2° : Decodificación
- 3° : Cálculos de la dirección efectiva
- 4° : Ejecución
- 5° : Ejecución de las instrucciones de coma flotante
- 6° : Ejecución de las instrucciones de coma flotante
- 7° : Escritura de los resultados
- 8° : Informe de posibles errores

Para llevar a cabo estas instrucciones el coprocesador matemático internamente posee registros. Son 8 registros de datos y los siguientes registros especiales:

- Registros de datos, que es donde se guardan los operandos y los resultados. Consta de 8 registros (de R0-R7) de 80 bits de longitud cada uno. Un bit es para el signo, 15 bits para el exponente y los 64 restantes para la mantisa.
- Registros de estado, son registros de 16 bits que indican la situación actual de la FPU, los flags incluidos en estos registros son activados por la FPU para mostrar el resultado de sus operaciones.
- Registros de control, estos registros controlan la precisión de la FPU y los métodos de redondeo usados.
- Registros de palabra, tienen 16 bits y están divididos en campos de 2 bits cada uno (se le denomina tag). Así contendrá 8 tag cada uno del 0 al 7. Cada tag hace referencia a un registro de datos, numerados del R0 a R7.
- Registro puntero de instrucciones, se trata de un registro de 48 bits que guarda la dirección RS (16 bits) y el desplazamiento (32 bits) de las direcciones virtuales de las últimas instrucciones que hemos usado.
- Registro puntero al último operando, también denominado puntero dato, se trata de un registro de 48 bits que guarda la dirección RS (16 bits) y el desplazamiento (32 bits) de las direcciones virtuales de los últimos datos que hemos usado.
- Registro de código, se trata de un registro que consta de 11 bits. La FPU almacena el código de las últimas instrucciones ejecutadas que no sean de control.

El coprocesador puede obtener y escribir datos en memoria de los siguientes tipos:

- Entero: Words de 16 bits, Dword de 32 bits y Qwords de 64 bits.
- Real: Words de 16 bits, Dword de 32 bits, Qwords de 64 bits y Twords de 80 bits.
- Simple precisión en coma flotante.
- Doble precisión en coma flotante.
- Doble precisión expandida en coma flotante.
- Entero con signo.
- BCD.

#### 6.2.4. Sistema de predicción de saltos condicionales

Una de las razones que más influyen en el bajo rendimiento del procesador son los saltos condicionales, que obligatoriamente introducen tres burbujas en el cauce (tres ciclos del cauce sin nada) ya que no se sabe la siguiente instrucción a ejecutar por lo que hay que esperar a ver si se cumple la condición para saber cuál es dicha instrucción.

Los fabricantes intentan eliminar estas burbujas del cauce. Intel lo hace prediciéndolo en un solo ciclo.

Para ello utiliza dos elementos:

- Software: Consiste en un potente algoritmo estadístico.
- Hardware: (BTB) Buffer de destino de las bifurcaciones. (“Branch Target Buffer”). Es una caché ultrarrápida que tiene 256 posiciones donde se guardan los resultados de las 256 últimas instrucción de salto BRANCH (salto condicional).

Cuando una instrucción supone un salto la BTB recuerda dicha instrucción y la dirección de salto efectuada y predice, aplicando ciertos algoritmos en qué dirección se va a producir el salto la próxima vez que se ejecute. Hay una tasa de acierto del 90%. En este caso, si la predicción es correcta, la bifurcación se realizará en 0 ciclos de reloj, puesto ésta ya se realizó, y se siguieron buscando instrucciones en dicha dirección. Por otro lado, si la predicción falla (en un 10% de los casos) habrá una penalización de tiempo debido a que se han metido en el cauce tres instrucciones erróneas por lo que hay que deshacer todas las operaciones realizadas con dichas instrucciones.

En la figura 6.8 se muestra como está conectada la BTB a la caché de instrucciones y al decodificador de las mismas, de tal forma que si la BTB acierta existe una recompensa de tiempo ya que no hay que recoger la instrucción siguiente de la caché.

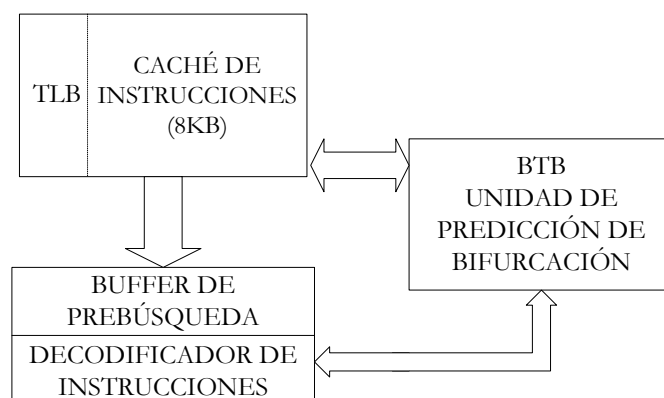


Figura 6.8 – Conexionado de la BTB

### 6.2.5. BIU: unidad de interfaz con el bus

La BIU (“Bus Interface Unit”) es el bloque encargado de soportar todas las transferencias con el mundo exterior. Controla los ciclos del bus que acceden a la memoria y a las E/S. En la siguiente figura 6.9, podemos ver las distintas partes en las que se divide la unidad de interfaz con el bus.

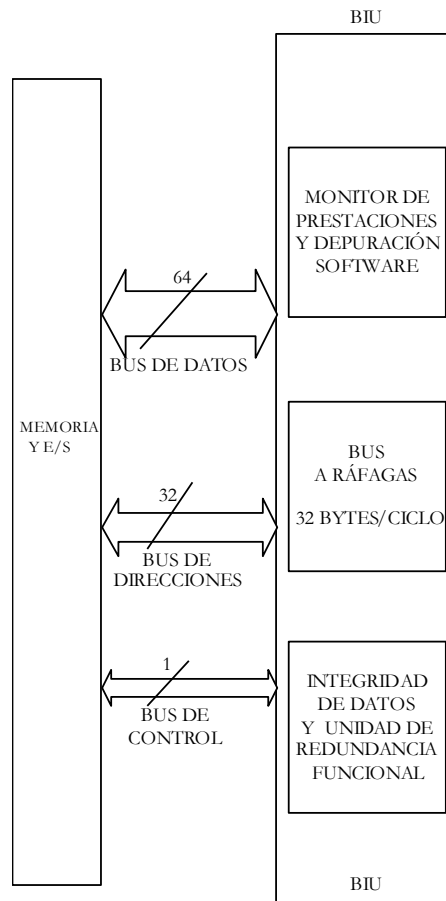


Figura 6.9 – Conexión de la BIU

El Pentium tiene el mismo rango de direccionamiento que los 80386 y los 80486. Esto es debido a que también usa un bus de direcciones de 32 bits. Sin embargo, el bus de datos externo es de 64 bits, el doble que sus predecesores. El subsistema de memoria debe por tanto, estar organizado en ocho grupos de 8 bits cada uno, es decir, un total de 64 bits para adecuarse al bus del procesador.

El bus externo de 64 bits, es capaz de transferir datos entre la memoria y el procesador a velocidades que pueden llegar hasta 584 MBytes/sg.

La BIU consta de tres partes:

- Monitor de prestaciones
- Bus a ráfagas
- Unidad de redundancia funcional

A continuación se explican cada una de ellas.

### 6.2.5.1. Monitor de prestaciones

Desarrollar aplicaciones es cada vez más complejo y precisa de una cuidadosa realización para evitar que la mayor parte del tiempo se pierdan ciertas rutinas o selecciones del código que no son excesivamente importantes. Para facilitar el trabajo de los desarrolladores de software, el procesador Pentium incorpora un monitor de prestaciones y una unidad de depuración software.

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. Es una herramienta hardware que la puede usar el programador para saber cómo funciona el programa. Se creó para facilitar el trabajo de los desarrolladores de software.

Estos contadores junto a las unidades de rastreo y traza, permiten conocer el estado del procesador, el tiempo que se emplea en la realización de operaciones y las instrucciones que se ejecutan. Desde el exterior del procesador, por medio del puerto serie, se puede interactuar con esta unidad.

A parte de todo esto, se puede obtener el número de ciclos que el procesador emplea en operaciones internas que de algún modo afecten a:

- La lectura y escritura de datos.
- La ausencia o presencia de datos o código en las memorias caché internas del Pentium.
- Las interrupciones.
- La utilización del bus.

También se puede saber cuanto tiempo tiene que esperar el procesador para conseguir el control del bus externo. Gracias a todo lo anterior, se consigue sistema más rápido debido a la fácil optimización del mismo.

La unidad de traza cuando se produce cierta condición de bifurcación o los saltos a subrutinas o si la ejecución se produce en determinada sección de código o que instrucción ha provocado una interrupción, etc. Por tanto, se pueden detectar los cuellos de botella, es decir, dónde el sistema se ralentiza o la aplicación pierde mucho tiempo inútilmente. Por ello, se optimiza consiguiendo mejores prestaciones y mejor tiempo de respuesta.

### 6.2.5.2. Bus a ráfagas

Es una circuitería de silicio que permite cargar 256 bits (32 Bytes que es igual a la línea de caché) en la caché de datos de una sola vez, es decir en un ciclo.

En cuanto al tipo de ciclos del bus, los valores medios corresponden a:

- 36%: Prebúsqueda de instrucciones
- 21%: Lecturas de datos
- 36%: Escritura de datos
- 7%: Escrituras obligadas de datos (L1)

Como se acaba de comentar, el tipo de ciclo de bus de ráfaga puede cargar 256 bits de una sola vez. El bus externo de 64 bits es capaz de transferir datos entre la memoria y el procesador a una velocidad que puede llegar a ser de 528 MBytes por segundo. Esto significa que, por ejemplo, el contenido completo de un disco fijo de 100 MBytes pasaría por este bus en menos de un quinto de segundo. Esta velocidad de transferencia es superior en más de tres veces al ancho de banda del bus de un 80486 a 50 MHz.

### **6.2.5.3. Unidad de redundancia funcional**

Es un recurso que emplea diferentes técnicas para la detección de errores tanto externa como internamente, para asegurar la integridad de los datos. Cada octeto del bus de datos, lleva asociado un bit de paridad lo que hace un total de 8 bits de paridad para todo el bus de datos. Los bits de paridad son comprobados por el procesador en cada lectura. A su vez el Pentium genera un bit de paridad por cada octeto de los 64 bits que componen cada escritura hacia el exterior, lo que hace un total de 8 bits de paridad. También el bus de direcciones añade un bit de paridad por octeto; de este modo hay 4 bits de paridad para las direcciones que se generan y comprueban en cada acceso de escritura o lectura respectivamente. Con todo ello, el procesador Pentium es capaz de detectar errores en el bus de direcciones y en el de datos. Por tanto, el Pentium, no sólo detecta que el dato leído o escrito es correcto sino que también es capaz de saber si la dirección de memoria es correcta.

Internamente, también se hacen controles de paridad en la caché interna, en los registros internos y en la memoria ROM que almacena el microcódigo. Hay otro tipo de recursos que aseguran la fiabilidad del procesador. Siempre después de una inicialización se realiza un autodiagnóstico interno que comprueba que al menos un 70% de los dispositivos internos funcionan adecuadamente.

El Pentium implementa un sistema de redundancia funcional de una forma muy sencilla. Basta con poner dos procesadores Pentium en el mismo bus, uno trabajando en modo maestro y otro como comprobador. Los dos procesadores ejecutan las mismas instrucciones en el mismo tiempo. El que hace de comprobador chequea cada resultado obtenido por el maestro con el suyo propio. Si existe diferencia se produce una interrupción de máxima prioridad que detiene el sistema y avisa que los dos procesadores no están de acuerdo en los resultados de la ejecución del programa.

## **6.3. MODOS DE FUNCIONAMIENTO DEL PENTIUM**

Las tres formas de trabajo que tiene el Pentium son el modo real, el modo protegido y el modo de manejo del sistema. En cada una el Pentium funciona de manera diferente. Para seguir correctamente el funcionamiento se aconseja consultar la figura 6.10

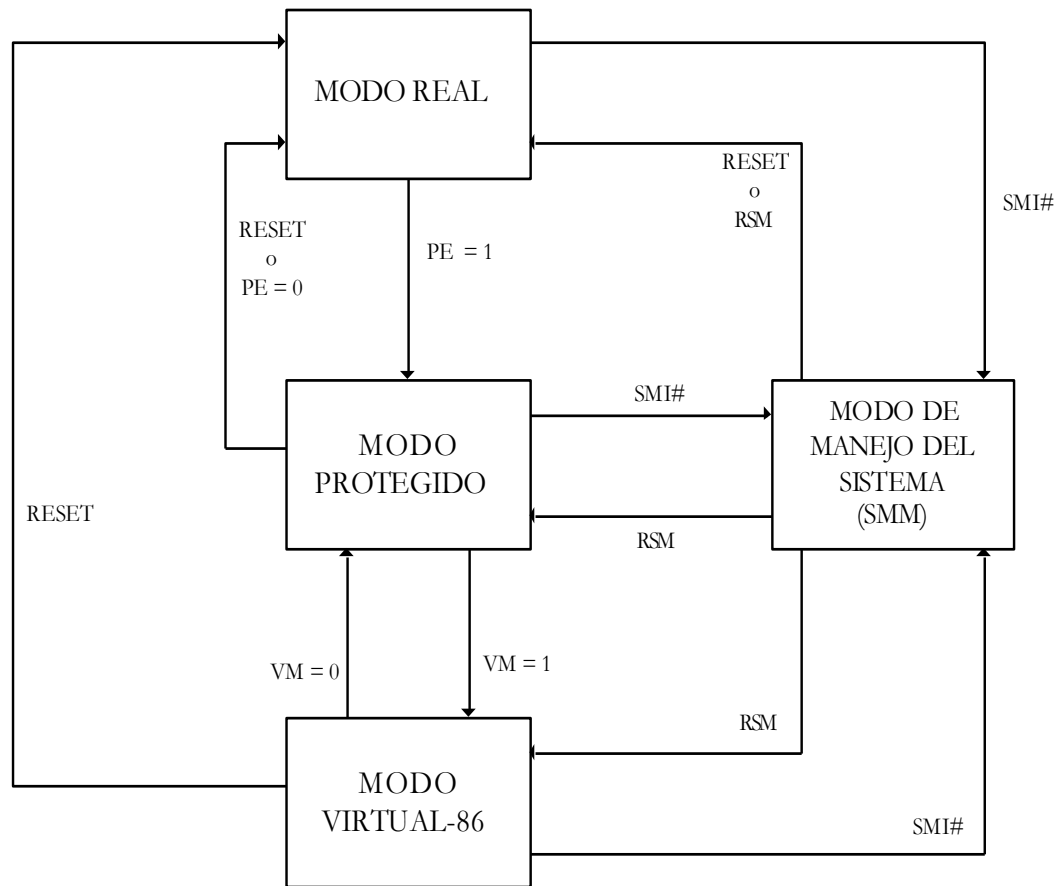


Figura 6.10 – Modo de funcionamiento del Pentium

- Modo real:

Funciona como si fuese un 8086 aunque posee algunas extensiones, por ejemplo la frecuencia es mayor (es la propia de trabajo del Pentium). En este modo entra cuando se produce un Reset (se usa la segmentación y funciona con 16 bits). En este modo el Pentium solo trabaja con el primer MByte de la memoria, siendo 64 KBytes la capacidad máxima que puede ocupar un segmento. Por último citar que el bus de direcciones utiliza los 20 bits de menos peso. Este modo es monotarea y los registros de propósito general (AX, BX, CX, DX, SI, DI, SP, BP), IP (Puntero de Instrucciones) son de 16 bits; lo mismo que el registro de flags. La tabla IDT es 256 entradas y en cada una de ellas hay un vector de interrupciones que apunta al inicio de la rutina que atiende esa entrada.

En el modo real se empieza a trabajar con un reset o cuando se conecta la alimentación por primera vez, es decir cuando se produce un reset automático.

- Modo protegido:

Este es el modo nativo con toda su potencia. Los registros ya tienen una extensión de 32 bits. Es un modo opcional. Puede trabajar con memoria virtual y con una memoria física de hasta 32 GBytes, pudiéndose dar la segmentación y la paginación. Se trabaja ya



con multitarea y por ello funciona el sistema de protección y aislamiento entre tareas, pudiendo alcanzar los segmentos un tamaño de 32 GBytes.

Estando en modo real para pasar a modo protegido hay que poner el bit PE a 1 (“Protection Enabled”) que se encuentra en el CR0 (doble palabra de estado de la máquina) del registro de control.

Al pasar al modo protegido hay que tener en cuenta que el flag IF puede estar a 1, también hay que crear una nueva IDT ya que ésta cambia en contenido y en capacidad respecto a la del modo real. El sistema operativo debe crear la IDT justo al empezar el modo protegido.

Además para funcionar, se necesitan las tablas de descriptores, GDT y LDT, para localizar un segmento. También hay que actualizar los registros de segmento.

Como resumen de lo comentado hasta ahora, los pasos que se deben seguir para pasar a este modo son los siguientes:

1. Poner el bit IF a 0.
2. Instalar en la memoria la tabla de descriptores global (GDT), mediante el registro GDTR que apunta a la base y al límite de la GDT.
3. Poner el bit PE a 1. Esto se realiza cargando el registro EAX con 00....01 (32 bits) y moviendo el contenido de este a CR0.
4. Hacer una instrucción JUMP para eliminar así las instrucciones que se habían cargado en la cola de prebúsqueda. En el modo real hay un elemento que es la cola de prebúsqueda que va guardando las instrucciones que se van a ejecutar a continuación. Así al pasar a modo protegido hay que limpiar esta cola ya que contiene instrucciones de modo real.
5. Cargar la LDT y el registro LDTR pasa a contener las características de la tabla en curso.
6. Cargar el registro de tarea (TR).
7. Actualizar los registros de segmentos, el CS, SS.....GS para que apunten a los segmentos de modo protegido.
8. Poner en marcha la IDT que una vez cargada se cargará en el registro LDTR, la base y el límite de la misma.
9. Permitir las interrupciones mascarables.

- **Modo Virtual 8086**

Es una mezcla de modo real y modo protegido. Trabajamos en un ambiente igual que en modo protegido, en un ambiente multitarea y con sistema de protección entre las tareas, pero se permiten ejecutar tareas del 8086 (del modo real). Así algunas tareas son por tanto del modo real y como tienen únicamente 20 bits, las convierte a 32 bits para que así puedan apuntar a más del primer MByte de memoria.

Para pasar del modo protegido al modo virtual, basta con poner el bit VM a 1 del registro de estado, en caso contrario habrá que poner el bit VM a 0.

- Modo de manejo del sistema (también conocido como SMM)

En este modo el Pentium proporciona un sistema operativo que es transparente para el programador y que implementa dos funciones muy importantes: la primera función está relacionada con la seguridad de todo el sistema y mejora dicha seguridad; la otra función es un sistema de control de la alimentación que controla el consumo del procesador del sistema y lo mejora.

El procesador aísla completamente, cuando se trabaja en este modo, un espacio de memoria reservado para él donde se salva todo el contenido de la tarea que se va a ejecutar. Para pasar de cualquiera de los otros tres modos a SSM hay que activar por hardware una patita del Pentium. Se trata de la patita SMI# que se activa por nivel bajo. Hay otra forma que no es por hardware y que consiste en provocar una interrupción desde el controlador programable de interrupciones SMI. Dicha interrupción, la atiende una entrada específica de la IDT.

Para pasar a modo real desde este modo, basta con que se produzca un reset. Además existe una instrucción del repertorio de instrucciones del Pentium, la RSM mediante la cual se puede pasar al modo real, al modo protegido o al modo virtual 86. Para saber a cual de los tres modos se ha pasado, se consultarán los bits PE y VM. Si el bit PE es 0, haremos pasado al modo real y no es necesario consultar el VM. Si el bit PE y VM están a 1, habremos pasado al modo virtual 86. Por último, habremos pasado a modo protegido si el bit PE está a 1 y el VM a 0.