

TEMA 3. EL MICROPROCESADOR

0. Introducción

La informática es la técnica que se ocupa del tratamiento de la información. El ordenador es una máquina digital que tiene la posibilidad de recibir datos. Después de tomarlos y procesarlos, puede emitir unos resultados. Serán los periféricos de E/S a través de los canales los que suministren la Información necesaria al ordenador, y por los que se obtendrán los resultados después de haber sido procesados.

Es dentro del ordenador, en la CPU, la que se encarga de procesar la información, haciendo funcionar todas sus partes.

En general podemos decir que el ordenador tiene dos partes claramente diferenciadas:

- La Unidad Central de Procesos (CPU) que es la que se encarga de ejecutar los programas y que está compuesta de memoria principal, de unidad aritmético-lógica y de la unidad de control. En el caso de microordenadores, a la agrupación de unidad de control y unidad aritmético-lógica se le puede denominar de forma más técnica microprocesador.
- Las unidades periféricas o periféricos de E/S, que permiten la comunicación de la CPU con el exterior. Los periféricos registran y leen datos, en general sirven para poder registrar Información procesada por la CPU.

Además de todo lo considerado también deberemos tener en cuenta los buses como líneas de conexión.

1. El microprocesador

1.1. Concepto

El microprocesador es el elemento principal del ordenador. Sus características definen en gran parte la configuración y prestaciones del ordenador: organiza el flujo de datos en el interior de la placa base, entre los periféricos y la memoria, y entre el resto de componentes de la placa. Éste realiza una serie de operandos según el ritmo marcado por el reloj. Los más extendidos y comercializados son los fabricados por Intel (aliado de Microsoft que hace que su software sea compatible para estos procesadores). Otros fabricantes son: AMD, CYRIX, etc.

En un principio las funciones del microprocesador estaban repartidas en varios módulos. Al conjunto de todos ellos se les denominaba CPU. (Unidad Central de Proceso). Posteriormente, la evolución tecnológica permitió integrar en una sola pastilla estos módulos denominándose Microprocesador.

Las Funciones del Microprocesador son las siguientes:

- 1) Ejecutar Instrucciones. Para ello dispones de una Sección de Ejecución.
- 2) Generar señales de Control. Para ello dispone de la Sección de Control.

Su velocidad depende de:

- *Frecuencia de reloj*: velocidad a la que trabaja el procesador y se mide en MHz que indica la cantidad de millones de operaciones (impulsos) que realiza por segundo.

La velocidad tienen que estar dentro de unos límites correctos y exactos sin oscilaciones; gracias a que en cada placa base hay un cristal de cuarzo que genera impulsos de forma sincronizada.

- *Buses*: son las vías internas por las que circulan los datos e influyen seriamente en la velocidad real de proceso de un ordenador. Su eficacia depende de su ancho (número de bits que pueden circular por ellos) y la rapidez con la que estos bits circulan. La velocidad se mide en MHz.
- *Coprocesador matemático*: gracias a esto la velocidad de procesamiento del ordenador aumenta considerablemente, ya que este componente libera al procesador central de la realización de operaciones puramente matemáticas.

1.2. Estructura General de un Microprocesador

La estructura más general de la Unidad Central de Proceso viene dada en la siguiente figura:

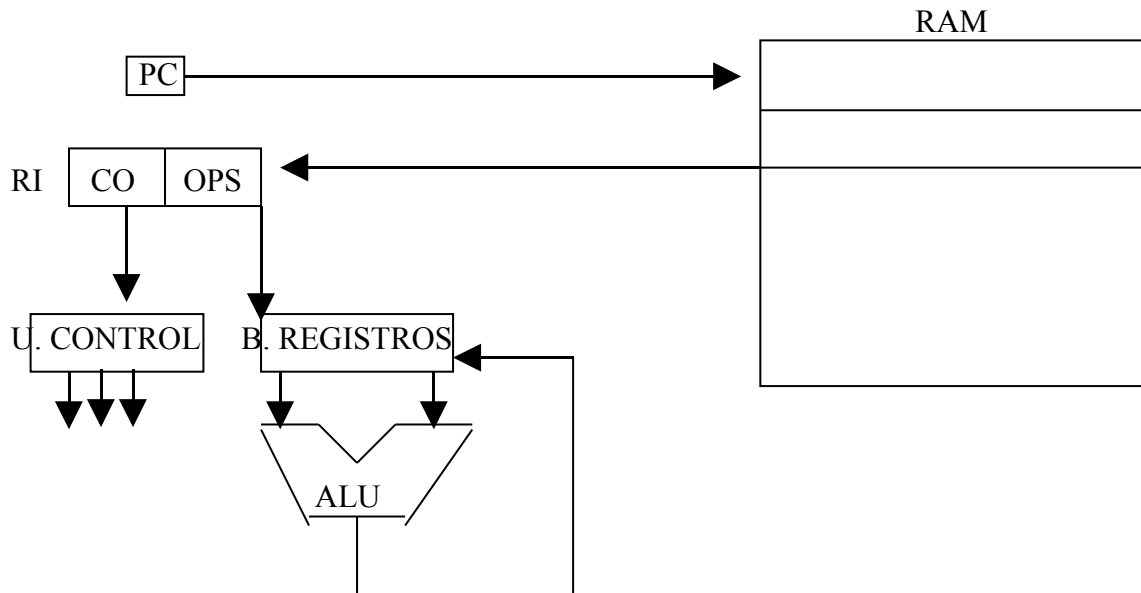
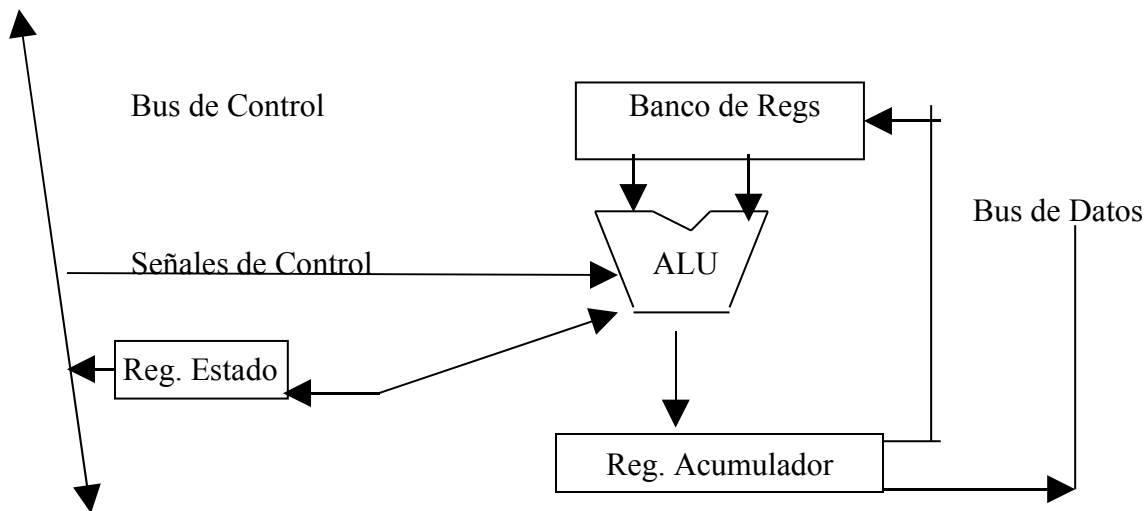


Diagrama de la CPU

El contador de programa (PC) es un registro que apunta a la dirección de memoria donde se encuentra la próxima instrucción a ejecutar. Una operación de lectura carga la instrucción en el registro de instrucciones (RI). Cada instrucción tiene dos campos. El primero denominado, código de operación (CO), expresa el tipo de operación que se va a realizar. El segundo, denominado Operandos (OPS) indica en qué lugar se encuentran los operandos.

La Unidad de Control obtiene, a partir del código de operación, las señales de control necesarias para ejecutar la instrucción. Estas señales de control son: tipo de operación que ejecuta la ALU, señales de activación de RAM, de lectura/escritura, etc. Con el campo operandos se activan los registros que intervienen en la operación.

La sección de procesamiento tiene la siguiente estructura:



El Registro de Estado son un conjunto de Biestables que indican condiciones del resultado, por ejemplo overflow o el signo.

El Registro Acumulador almacena el resultado de las operaciones. Puede realimentar el resultado al Banco de Registros para operaciones encadenadas o bien enviar el resultado al Bus de Datos.

La sección de procesamiento consta además de la Unidad Aritmético-Lógica (ALU) y de un Banco de Registros.

La **ALU** está formada a su vez por una unidad aritmética y una unidad lógica. La primera realiza las operaciones aritméticas básicas y suele estar formada por un sumador en C2 y, en algunos casos, por un multiplicador. Ambas unidades trabajan en punto fijo. Para realizar las operaciones en punto flotante se utiliza un coprocesador aritmético que es opcional aunque, a partir del 486, suele venir integrado en el microprocesador. La unidad lógica realiza las operaciones booleanas básicas AND, OR, XOR, NOT y operaciones de desplazamiento y rotación.

El **Banco de Registros** se utiliza para almacenar los operandos. Está formado por varios registros con funciones distintas según el tipo de microprocesador. En general suele tener entre 14(Intel) y 32(Motorola) registros.

1.3. Funcionamiento de la Unidad Central de Proceso

Un programa a ejecutar está compuesto por un conjunto de instrucciones almacenadas en memoria. La CPU es la encargada de ejecutar las instrucciones especificadas en el programa.

El Procesamiento que requiere una instrucción se denomina *Ciclo de Instrucción* está constituido básicamente por dos etapas:

- ✓ *Etapas de captación de una instrucción:* es una operación común a todas las instrucciones y consiste en la lectura de la instrucción de una posición de memoria. También se denomina Ciclo de Captación.
- ✓ *Etapas de ejecución de una instrucción:* puede suponer varias operaciones y depende de la naturaleza de la instrucción. También se denomina Ciclo de Ejecución.

Al comienzo de cada ciclo de instrucción, la CPU capta una instrucción de memoria. A no ser que se indique otra cosa, la CPU siempre incrementa el PC después de captar cada instrucción. La instrucción captada se almacena en el registro de instrucción. Posteriormente la Unidad de Control interpreta el Código de Operación y generará las señales de control necesarias para llevar a cabo la ejecución de la instrucción.

La ejecución del programa consiste en la repetición del proceso de captación y de ejecución de las instrucciones del programa.

En la ejecución de un programa, los componentes estudiados anteriormente funcionan de forma sincronizada realizando las siguientes etapas:

- ✓ La CPU lee de memoria la primera instrucción.
- ✓ La CPU decodifica la instrucción para conocer el tipo de operación que hay que realizar y donde se encuentran los operandos.
- ✓ Se leen los operandos bien de memoria o de E/S.
- ✓ Se ejecuta la operación.
- ✓ Se guardan los resultados en memoria o E/S.
- ✓ Se lee la siguiente instrucción.

Este proceso se repite hasta terminar el programa.

1.4. Características

Todos los Microprocesadores realizan las mismas funciones, pero existen unas características que configuran su personalidad y definen sus prestaciones:

ARQUITECTURA

Se refiere a los elementos hardware que forma el microprocesador y su distribución.

TECNOLOGÍA

Se refiere a los elementos materiales sobre los que se construye los circuitos electrónicos que forman el microprocesador.

El micro está compuesto por millones de transistores hechos de un material semiconductor conectados por un material conductor. La integración se realiza en capas. La tecnología que se usa hoy en día para integrar los transistores se llama CMOS. De la tecnología de integración depende:

- N° de transistores que pueden integrar
- El voltaje que necesitará el chip
- El espacio físico que ocupa

ASPECTO

Viene empaquetado en una cápsula aislante (generalmente de cerámica), que actúa como absorbente del calor disipado por los circuitos electrónicos. Las conexiones con el exterior se realizan a través de las patillas (pines). Cada patilla tiene asignada una señal de comunicación. La clasificación es la siguiente:

1. Bus de Datos.
2. Bus de Direcciones
3. Señales de Control.
4. Interrupciones.
5. Alimentación.

1.5. Memoria interna (al microprocesador)

Los más internos son los más rápidos y más cercanos a la UC. También son los de menor tamaño.

Se usan para almacenar información que usa el micro, que procede o de cálculos intermedios internos o de la RAM.

De más interno a más externo:

1º Banco de registros

- Guardan los datos que sirven de operandos para la instrucción-es en curso.
- Guardan resultados intermedios
- Guardan la dirección de la siguiente instrucción
- Guardan el código de la instrucción que está siendo ejecutada
- Etc.

2º Búffer de prebúsqueda de instrucciones /datos

Se utiliza si se van a ejecutar varias instrucciones a la vez, entonces éstas tendrán que estar preparadas en algún lugar y es aquí donde se almacenan.

3º Caché de primer nivel (L1)

Está separada en dos partes: una para datos, y otra para instrucciones. De esta forma en una misma etapa se consulta simultáneamente datos e instrucciones.

4º Caché de segundo nivel (L2)

Sirve para aumentar la velocidad de acceso a la memoria.

1.6. Medidas de los microprocesadores

La velocidad en los micros se mide con los siguientes parámetros:

- *Frecuencia*: Medida indicativa de la potencia de un micro. Medir la velocidad con este término es un error, ya que no sólo hay que tener en cuenta la arquitectura interna del micro.
- *Tiempo de CPU*: Tiempo que invierte un procesador en la ejecución de un programa. La fórmula sería la siguiente:

$$T_{cpu} = NI * CPI * T = NI * CPI * 1/F$$

NI: número de instrucciones en que se transforma el programa

CPI: número medio de ciclos de reloj que se necesitan para ejecutar cada instrucción

T: periodo o tiempo de ciclo de reloj (inversa de la frecuencia, F)

Se mide en microsegundos.

- *Los MIPS y los MFLOPS.* Nos indica los millones de instrucciones que el micro realiza por segundo. El inconveniente es que depende de la cantidad de instrucciones. Para solucionar este problema se introdujo los MFLOPS, que nos indica los millones de operaciones en coma flotante por segundo. El problema de esta unidad es que no mide el rendimiento del sistema en general, sino sólo la unidad de coma flotante (FPU).

$$\text{MIPS} = \text{NI}/\text{Tcpu} * 10^6$$

1.7. Disipadores y ventiladores

Todos los componentes electrónicos de un ordenador producen calor. Por eso los ordenadores deben incorporar mecanismos para disminuir temperaturas.

En los primeros era suficiente colocando un ventilador en la fuente de alimentación.

Conforme empezó a disminuir el tamaño de integración, fue necesario añadir el disipador: se coloca en contacto directo con el micro.

Cuando el disipador dejó de ser suficiente, hubo que añadir un ventilador para el micro, que se coloca sobre el disipador.

Ventilador + disipador = combo. Se reduce la temperatura en unos 40° y aumenta el rendimiento de éste en un 30%

El disipador será mejor cuanto más grande

1.8. Evolución

Desde su aparición en la década de los 30, los computadores han ido evolucionando, tanto en su arquitectura como en su tecnología. Esto nos permite establecer una serie de etapas en el desarrollo de computadores a las que denominamos generaciones. Podemos distinguir las siguientes generaciones:

1ª generación

Comprende los primeros grandes ordenadores basados en la arquitectura Von Newman. Cronológicamente se corresponde con finales de los años 30. Sus características principales son:

- Utilizar tecnología basada en válvulas
- Fines científicos
- Tenían nombre propio: ENIAC y EDVAC

2ª generación

Coincide con la aparición del transistor (1956). Las funciones del transistor son similares a las de las válvulas de vacío pero con un ahorro significativo de volumen y consumo. Sus características son:

- Tecnología basada en transistores
- Aparecen los primeros lenguajes de programación
- Fines comerciales y científicos
- Aparece la serie IBM 370

- Primeros periféricos
- Aparece el concepto de Supercomputador

3ª generación

La tercera generación de ordenadores se caracteriza por la aparición de los circuitos integrados. Se trata de integrar en un solo chip todos los transistores y circuitos analógicos que realizan las operaciones básicas de un computador. A estos chips se les denominan puertas lógicas. Esta generación se caracteriza por:

- Utilizar la tecnología basada en una escala de integración pequeña (MSI)
- Aparecen los lenguajes de programación de propósito general (C, pascal, basic, etc.)
- Nuevos soportes (disco flexible, monitor, etc.)
- Nuevas arquitecturas
- Aparece el concepto de miniordenador (estación de trabajo)

4ª generación

La cuarta generación se caracteriza por la popularización del microordenador. La tecnología permite integrar más circuitos en una sola pastilla. Esto reduce el espacio y el consumo haciendo asequible el ordenador a cualquier persona. Sus características son:

- Tecnología LSI (Alta escala de integración)
- Aparecen los microprocesadores
- Proliferan los lenguajes de programación
- Uso del ordenador para fines particulares

5ª generación

Aparece la tecnología VLSI. Cualquier operación algorítmica utilizada mediante software puede implementarse directamente en un circuito integrado. Sus características son:

- Tecnología VLSI (Muy alta escala de integración)
- Aparecen los microprocesadores de uso específico
- Avance muy rápido de la tecnología
- Tecnología multimedia

2. Instrucciones

2.1. Concepto de instrucción

Cuando se diseña un microprocesador, el primer paso que se da es definir su repertorio de instrucciones. Este conjunto de operaciones va a definir la estructura que va a tener el microprocesador y las conexiones entre sus elementos. El formato general de una instrucción es:

CO	Operandos
----	-----------

El campo CO (Código de operación) indica el tipo de operación que se va a realizar y el campo Operandos el lugar en el que se encuentra los operandos que intervienen.

El campo CO es utilizado por la unidad de control para generar las señales de sincronismo y preparar así los elementos hardware que intervienen en la operación. El campo Operandos se utiliza para activar los registros en los cuales se encuentran los operandos y el que va a recibir el resultado de la operación. Cuando la operación es de Lectura/Escritura en memoria o en E/S, el campo OP indica la dirección de la operación. Este proceso se denomina decodificación y lo realiza un elemento hardware denominado *decodificador*.

El tamaño de una instrucción depende del número de instrucciones que forman el repertorio del microprocesador y del número de registros utilizados y de direcciones de memoria. Así, un microprocesador de 16 instrucciones necesitará un CO de 4 bits para codificarlas. Si el micro posee cuatro registros para guardar operandos necesitará 2 bits para codificarlos. Por último, si se trata de una máquina de tres operandos necesitaremos tres campos para indicar los operandos.

Así, el formato de una instrucción típica de la ALU sería:

1001	00	01	10
------	----	----	----

Este código correspondería a la instrucción:

SUMA R0,R1,R2 \Leftrightarrow R0=R1+R2

El tamaño de cada instrucción será por tanto de 10 bits.

Cuando la instrucción se refiere a una operación con memoria el formato de la instrucción debe incluir un campo que haga referencia a la dirección de memoria y otro al registro que utiliza el dato que se transfiere. Así, el formato típico de una instrucción de este tipo puede ser:

0101	00	1100111001
------	----	------------

En este caso, estamos suponiendo que la memoria disponible es de 1K(=2¹⁰) y que necesitamos por tanto, 10 bits para indicar cada una de las posiciones de memoria. El esquema anterior se refiere a una instrucción del tipo:

CARGA R0,[DIR] \Leftrightarrow R0 \leftarrow [DIR]

siendo la dirección 1100111001.

Cuando la instrucción es de salto, sólo debemos indicar el código de operación y la dirección a la cual hay que saltar. El formato sería:

1101	0011001100
------	------------

Que corresponde a una instrucción del tipo:

SALTA [DIR]

En este caso sólo hay que especificar como comando la dirección de salto.

Si comparamos los tres formatos vemos que poseen distinto número de campos y que además, cada campo puede tener un significado distinto. Este tipo de codificación de instrucciones se denomina formato variable. Cada uno de los tipos de instrucciones (ALU, Memoria y Salto) tiene un formato distinto y la interpretación de los campos la realiza la Unidad de Control en función del contenido del código de operación. Existe otra alternativa y es asignar siempre el mismo significado a cada campo. En este caso tendríamos que ampliar el tamaño de las instrucciones aunque en determinados campos haya campos que no intervengan. Este tipo de instrucciones se denominan formato fijo.

Así, en el caso anterior, las instrucciones en formato fijo quedarían como:

CO	01	02	03	DIRECCION
----	----	----	----	-----------

En formato fijo, todas las instrucciones tienen el mismo tamaño incluyendo un campo específico para cada tipo de operación. En formato variable cada instrucción tiene un tamaño distinto. La ventaja de un formato fijo es la facilidad en la decodificación. Su inconveniente es el incremento de memoria para almacenamiento.

2.2. Clasificación

Las instrucciones de un microprocesador se pueden clasificar en los siguientes grupos:

- **Instrucciones Aritméticas**
Las realiza la Unidad Aritmética y son operaciones aritméticas básicas tales como suma, resta, desplazamiento, comparación, etc.
- **Instrucciones Lógicas**
Se realizan en la Unidad Lógica de la ALU. Pertenecen a este grupo las instrucciones AND, OR, XOR, etc.
- **Instrucciones de Entrada/Salida**
Son operaciones de transferencia de datos entre la CPU y E/S. Existen dos formatos estándar que son IN, OUT.
- **Instrucciones de Acceso a memoria**
Representan operaciones de lectura o escritura en memoria. Los formatos estándar son LOAD y SAVE.
- **Instrucciones de salto**
Implican una ruptura en la secuencialidad del programa. Puede ser condicional o incondicional. Los formatos estándar son JMP y J <condición>

2.3. RISC y CISC

Cuanto más amplio sea el repertorio de instrucciones de un microprocesador más versatilidad presenta y más posibilidades ofrece al programador. Sin embargo, a medida que aumenta el número de instrucciones aumenta también el campo CO necesario para codificarlas. Al aumentar el tamaño de la instrucción hace que la memoria que ocupa el

programa sea mayor y el número de accesos a memoria también aumente. Se nos presenta, por tanto, dos planteamientos distintos en el diseño de microprocesadores:

- Un planteamiento basado en el diseño de un conjunto de instrucciones reducido con el objetivo de disminuir el tamaño de los programas a costa de aumentar el número de instrucciones que forman el programa.
- Un planteamiento basado en un amplio repertorio de instrucciones aumentando la potencia del microprocesador a costa de aumentar el tamaño de las instrucciones.

Al primer planteamiento se le denomina RISC y al segundo CISC. Pertenecientes al primer grupo los microprocesadores de Motorola y AMD. Son microprocesadores CISC la familia Intel.

2.4. Extensiones de instrucciones

Extensiones MMX

Se introdujeron al conjunto de instrucciones de los micros para el manejo de multimedia. De esta forma se consigue grandes ganancias para estas aplicaciones y le da capacidad a la CPU para realizar operaciones gráficas y de sonido a la vez.

Garantiza la compatibilidad con las máquinas que no contengan MMX.

Extensiones SIMD

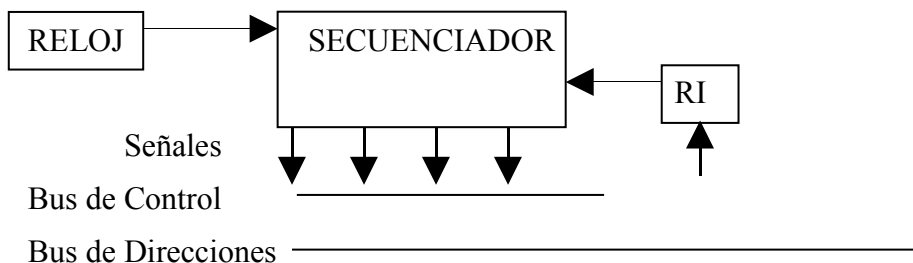
Se da para operaciones en coma flotante.

Es efectiva cuando se realiza en paralelo la misma operación en coma flotante para muchos números, ya que se aumenta mucho el rendimiento para realizar las operaciones.

3. Sección de control

3.1. Sección de control

Tiene como función generar las señales de control necesarias para sincronizar el funcionamiento combinado de todos los elementos que intervienen en la ejecución de una instrucción. Consiste en una estructura hardware que tiene como entrada el campo código de operación del registro de instrucciones y la señal de reloj, y como salida un conjunto de señales de control:



Estas señales se pueden clasificar en:

- Control de E/S

Son señales que controlan a los elementos externos al microprocesador. Las señales de control de E/S genéricas son:

- Activación de Memoria (Enable Memory)
- Lectura/Escritura (R/W)
- Memoria / Entrada_Salida

- Control de ALU

Expresan la operación que debe realizar la ALU y dan información sobre el resultado de la operación.

- Operación de la ALU
- Signo del resultado de la ALU
- Resultado cero
- Overflow

- Control de Registros

Controlan el estado de carga de un registro.

- Carga de un registro (Load)
- Inicialización del contenido del registros(Reset).

- Control multiplexores

Controlan el contenido de la salida del multiplexor de la Memoria Principal

Existen dos tipos de estrategias cuando se diseña una Unidad de Control: Control cableado y Control Microprogramado. En el primero, cada señal se obtiene mediante un conjunto de puertas lógicas. En el segundo, las señales de control están almacenadas en memoria.

3.2. Control Cableado

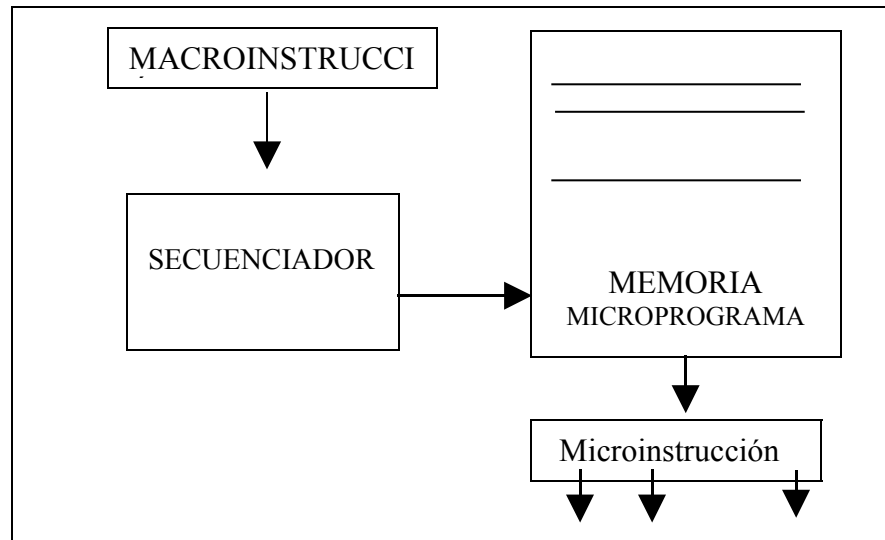
Consiste en un circuito que genera cada una de las señales de control como salida de una secuencia de puertas lógicas cuyas entradas son el CO.

El control cableado tiene como principal ventaja su simplicidad y rapidez pero tiene como inconveniente que si queremos añadir alguna instrucción nueva tenemos que modificar todas las señales de control y comenzar un diseño nuevo.

3.3. Control Microprogramado

En el control microprogramado cada instrucción tiene sus señales de control almacenadas en una ROM denominada *memoria microprogramada*. Cada línea de esta ROM se denomina *microinstrucción* y contiene todas las señales de control con los valores correspondientes al CO de la instrucción. De esta forma, cuando la Unidad de Control recibe un CO, lo utiliza como dirección para leer de la ROM la *microinstrucción* correspondiente. A la instrucción del programa se le denomina *macroinstrucción* y a la que contiene las señales de control se le denominan *microinstrucción*. En algunos casos, las señales de control ocupan varias *microinstrucciones* agrupadas en un *microprograma*.

La estructura de una sección de control *microprogramada* viene dada en la siguiente figura:



La macroinstrucción contiene información sobre el código de operación y los operandos. El código de operación es utilizado por el secuenciador como dirección de memoria para leer en la memoria microprogramada la microinstrucción correspondiente. Esta microinstrucción está formada por todas las señales de control del sistema que le corresponden a la macroinstrucción inicial.

4. Segmentación

4.1. Concepto de segmentación

Durante la ejecución de un programa se sigue una secuencia ordenada formada por varias etapas (generalmente 4):

- *Búsqueda de la instrucción (IF)*

El contador de programa (PC) guarda la dirección de la instrucción que se va a ejecutar. Esta dirección es utilizada para leer en memoria la instrucción y depositarla en el registro de instrucciones (RI). Mientras, el PC se va incrementando para apuntar a la siguiente instrucción.

- *Decodificación (ID)*

El código de operación de la instrucción es utilizado por la Unidad de Control para generar las señales de sincronismo oportunas y el campo operandos se utiliza para seleccionar los registros que intervienen.

- *Ejecución (EX)*

La ALU realiza la operación que haya sido seleccionada mediante el Código de Operación y los registros seleccionados en la etapa anterior.

- *Postescritura (WB)*

El resultado obtenido por la ALU hay que escribirlo en el registro adecuado.

Teniendo en cuenta esto y suponiendo que cada etapa se realice en un ciclo de reloj, la ejecución de una instrucción requeriría cuatro ciclos de reloj. Para acelerar el proceso de ejecución de los programas se introduce el concepto de segmentación.

La segmentación consiste en dividir la sección de procedimiento en varias unidades que trabajen en paralelo y cada una de ellas realice una de las etapas que constituyen la ejecución de una instrucción.

De esta forma, mientras una instrucción se está ejecutando, otra puede estar decodificándose y una tercera escribiendo el resultado en el banco de registros. Al conjunto de estas unidades se les denomina cauce.

En una sección de procesamiento segmentada, cada etapa del cauce realiza una función distinta y puede emplear un tiempo diferente. Por tanto, hay que sincronizar el funcionamiento del cauce de forma que la transferencia de datos entre etapas se realice de forma unísona. Para ello, entre dos etapas consecutivas se inserta un registro (registro de acoplo) que guarde el dato que se transfiere entre etapas. Todos los registros están controlados por la misma señal de reloj.

Para analizar la evolución del cauce se utiliza una representación Etapa-Ciclo en la cual, en el eje OX se representan los ciclos de reloj y en el eje OY las etapas de cauce. En cada celda XY se escribe la instrucción que en el ciclo X se encuentra en la etapa Y.

Cuando el cauce está lleno se ejecuta una instrucción en cada ciclo. Suponiendo que cada etapa consume un ciclo de reloj, si no existiese segmentación necesitaríamos 4 ciclos para ejecutar cada instrucción. Con la segmentación se reduce a 1 ciclo. De esta forma se consigue mejorar el rendimiento del microprocesador.

4.2. Riesgos de la segmentación

La segmentación es una técnica que aumenta el rendimiento de un microprocesador reduciendo el número de ciclos por instrucción ejecutada. Sin embargo, existen situaciones en las cuales se produce una ruptura del cauce. Estas situaciones se denominan riesgos de la segmentación y, en general, obligan a detener el cauce y en algún caso, a limpiarlo y empezar de nuevo.

Existen dos tipos de riesgos de segmentación:

- ***Riesgo por dependencia de datos.***

Se produce cuando una instrucción utiliza como entrada el resultado de la operación de la instrucción anterior.

Supongamos la siguiente secuencia de instrucciones:

I1: SUM R1,R2,R3 R3=R2+R1

I2: SUM R2,R3,R4 R4=R2+R3

La instrucción I2 utiliza el registro R3 como operando para calcular el valor de R4, sin embargo, el registro R3 se está calculando en la instrucción I1. Por tanto, para poder utilizarlo hay que esperar a que I1 termine de ejecutarse.

Para resolver este problema existen varias soluciones:

- *Detención del cauce.* Es la solución más trivial. Consiste en detener la ejecución de I2 hasta que termine de ejecutarse I1. Esta solución es la más

fácil de implementar físicamente pero es la que más disminuye el rendimiento del cauce.

- *Ejecución fuera de orden.* Consisten en aplazar la ejecución de I2 hasta que termine I1 y mientras ejecutar otras instrucciones posteriores que no utilicen el registro que origina la dependencia de datos.

- ***Riesgos por salto.***

Ocurren cuando aparece una instrucción de salto condicional. Si la condición es cierta, se ejecuta una serie de instrucciones, si es falsa, se ejecuta otro grupo distinto. Puesto que el resultado de la condición no se sabe hasta finalizar la instrucción de salto, no sabemos que instrucción es la que sigue.

Para solucionar este problema tenemos dos alternativas:

- *Detención del cauce.* Hay que esperar hasta que sepamos el resultado de la condición y seguir luego por la instrucción que corresponda. Este método implica detener el cauce y, por tanto, disminuye el rendimiento de la segmentación.
- *Ejecución especulativa o Predicción de salto.* Consiste en suponer que el salto no va a producirse y seguir ejecutando la siguiente instrucción sin detener el cauce. Cuando sepamos el resultado de la condición, si el salto no se produce, la elección que hicimos era correcta y el cauce sigue. Si el salto si se produce, hay que limpiar el cauce y empezar por la instrucción que le corresponda. También existen circuitos que calculan estadísticamente si debe o no realizar el salto, en este caso el número de aciertos es muy alto, dicho chip se llama BTB.

4.3. La segmentación en los microprocesadores de Intel

El primer microprocesador de Intel (el 8086) tenía un cauce formado por dos etapas: decodificación y ejecución. La unidad de decodificación se limitaba a traerse instrucciones de memoria y decodificarlas. Las instrucciones, una vez decodificadas, se guardaban en una pila FIFO. La unidad de ejecución recogía las instrucciones de la pila y las ejecutaba. Este proceso era asíncrono, es decir, cada etapa trabajaba a un ritmo distinto. Cuando la pila estaba llena la unidad de decodificación se paraba.

A partir del 386, la sección de procesamiento está segmentada en cuatro etapas (IF, ID, EX, WB) pero la conexión entre ellas no es síncrona, y por tanto, no existen registros de acoplo sino que se sitúan pilas para pasarse los datos. El pentium II introduce como novedad la ejecución especulativa y fuera de orden para resolver los problemas de riesgos en la segmentación.

4.4. Otras técnicas de paralelismo a nivel de instrucción

- ***Superescalaridad***

Se duplican componentes dentro de un mismo chip para que se puedan ejecutar varias instrucciones a la vez.

- ***Supersegmentación***

Consiste en tener varios niveles de segmentación. Esta técnica requiere elementos duplicados, ya que hay que cambiar a veces el orden de las instrucciones para hacerlo más eficiente. Cuando tenemos esta técnica, cada ciclo de reloj se encuentra subdividido en ciclos más pequeños. Esto da lugar a frecuencias de reloj muy altas.

5. Tipos de micros según número de bits que componen el bus de datos

Independientemente de su arquitectura encontramos los siguientes tipos:

1. *Microprocesadores de 4 bits*: 4004 de Intel. Memoria direccionable: 640 bits.
2. *Microprocesadores de 8 bits*: 8008, 8085, 8088 de Intel; Z80 de Zilog. Memoria Direccionable: 16 KB / 64 KB.
3. *Microprocesadores de 16 bits*: Hasta ahora sólo existía la posibilidad de direccionar 64KB de memoria. Para poder utilizar una mayor cantidad de memoria se diseñaron los microprocesadores de 16 bits. 8086 MD: 1 MB, 80286 MD: 16 MB.
4. *Microprocesadores de 32 bits*: Las necesidades aumentan tanto en la cantidad de memoria a direccionar como en la rapidez de ejecución de las instrucciones. 386DX MD: 4 Gb, Multitarea; 386SX, 486DX MD: 4 Gb.
5. *Microprocesadores de 64 bits*: Pentium; Pentium Pro; Pentium MMX; Pentium II; Pentium II/100Mhz; Pentium II para Slot 2; Pentium Deschutes; Celeron; Celeron Mendocino; Pentium II Xeon; Pentium III; Celeron (2ª Generación; Pentium III Xeon; Pentium III Xeon Mejorado; Pentium 4; Merced (Estudio para estaciones de trabajo y servidores)
AMD (Advanced Micro Devices): 386, 486, k5, k6, k6-2 3DNow!, k6-III, k7.

6. Nuevas tecnologías

Tecnología Hyper-Threading

Permite que el procesador ejecute subprocesos de instrucciones múltiples en paralelo, es decir, hace parecer a un solo procesador físico como múltiples procesadores lógicos. Para lograr que el sistema operativo vea dos procesadores, hay una copia del estado de arquitectura por cada procesador lógico, y los procesadores lógicos comparten un único conjunto de recursos físicos de ejecución.

Dual Core

Es la solución más económica para poder disfrutar de un procesador donde en realidad encontramos dos procesadores, ya que la tecnología Dual Core es la inclusión de dos procesadores dentro de la carcasa siendo la comunicación entre los dos núcleos directa.

En dicho micro, la caché se reparte a partes iguales entre los dos núcleos del procesador.

Los índices de overclocking serán muy reducidos a no ser que dispongamos de un excelente sistema de refrigeración por agua, intercambio de fase o electrorefrigeración.