

EL MICROPROCESADOR.

La capacidad de direccionamiento de un procesador se define como la cantidad máxima de memoria principal a la cual puede tener acceso. Dicha capacidad se define antes de fabricar el procesador ya que de acuerdo a eso, será la cantidad de patillas que tenga para el bus de direcciones. Específicamente el procesador 8088 tiene una capacidad de direccionamiento de 1 MB (2^{20} Bytes, dicho de otra manera puede manipular 2^{20} localidades de memoria de 8 bits c/u), lo cual se logra con 20 bits en el bus de direcciones (20 patillas). Al fabricar este procesador, la compañía Intel, decidió conservar los registros internos de 16 bits, pero que tuviera 1 MB de capacidad de direccionamiento. El problema presentado es que para alcanzar 1 MB en principal, deberían utilizarse 20 bits. Como los registros internos deberían ser de 16 bits, decidieron dividir la memoria en partes a la que llamaron segmento. Y para hacer referencia a una localidad específica se utiliza el desplazamiento.

Un segmento se define como una porción de memoria de 64 KB (como máximo), la cual puede estar situada en cualquier posición dentro del MB de memoria RAM. Dado que las localidades de memoria que contiene un segmento son contiguas, entonces solamente es necesario conocer la dirección de la primera posición de memoria del segmento. Todos los segmentos inician en una localidad cuyos últimos 4 bits (menos significativos) son 0, de esta manera solo es necesario almacenar 16 bits de la dirección de inicio (coincidiendo con el tamaño de los registros).

Desplazamiento es una dirección dentro del segmento. Debido a que el segmento tiene una capacidad de 64 KB como máximo, cualquier localidad de memoria dentro del segmento se manipula con 16 bits, por tal razón los desplazamientos posibles están comprendidos en el rango de 00000h a FFFFFh. De esta manera también un segmento se manipula en un registro de 16 bits.

La dirección efectiva de memoria a la cual se tendrá acceso la calcula el procesador multiplicando el valor almacenado en el registro de segmento por 10H, y sumando el desplazamiento. El resultado son 20 bits que señalan la localidad de memoria a la cual se tendrá acceso. Un programa en ensamblador consta de 4 segmentos: de código, de pila, de datos y extra.

El procesador 8088 dispone de registros de 16 bits divididos en cuatro grupos: propósito general, apuntador e índice, de segmento y señalizadores. Los registros de segmento guardan la dirección de inicio (solamente los 16 bits mas significativos) del respectivo segmento los cuales son: CS, SS, DS y ES. Los registros que se utilizan para guardar una dirección de desplazamiento son: BP, IP, SP, DI y SI. A continuación se describen las características y uso de los registros internos del procesador 8088.

REGISTROS DE PROPOSITO GENERAL. Se usan generalmente como registros de 16 bits. Sirven para manipular los datos del programa en ejecución. Cuando se requieren usar datos de 8 bits, existe la posibilidad de convertirlos en dos registros de 8 bits. Se les llama de propósito general porque sirven para almacenar datos o resultados (eso es lo general) de algunas operaciones. Además cada uno de ellos tiene funciones específicas que no tienen los demás, mismas que se explican a continuación:

AX ACUMULADOR. Guarda el producto después de una multiplicación o el dividendo para efectuar una división. Su parte baja se utiliza para almacenar un caracter leído del teclado. La parte alta del registro es AH, y parte baja AL. Además se utiliza para la programación en general porque sirve para almacenar los resultados temporalmente después de operaciones aritméticas o lógicas.

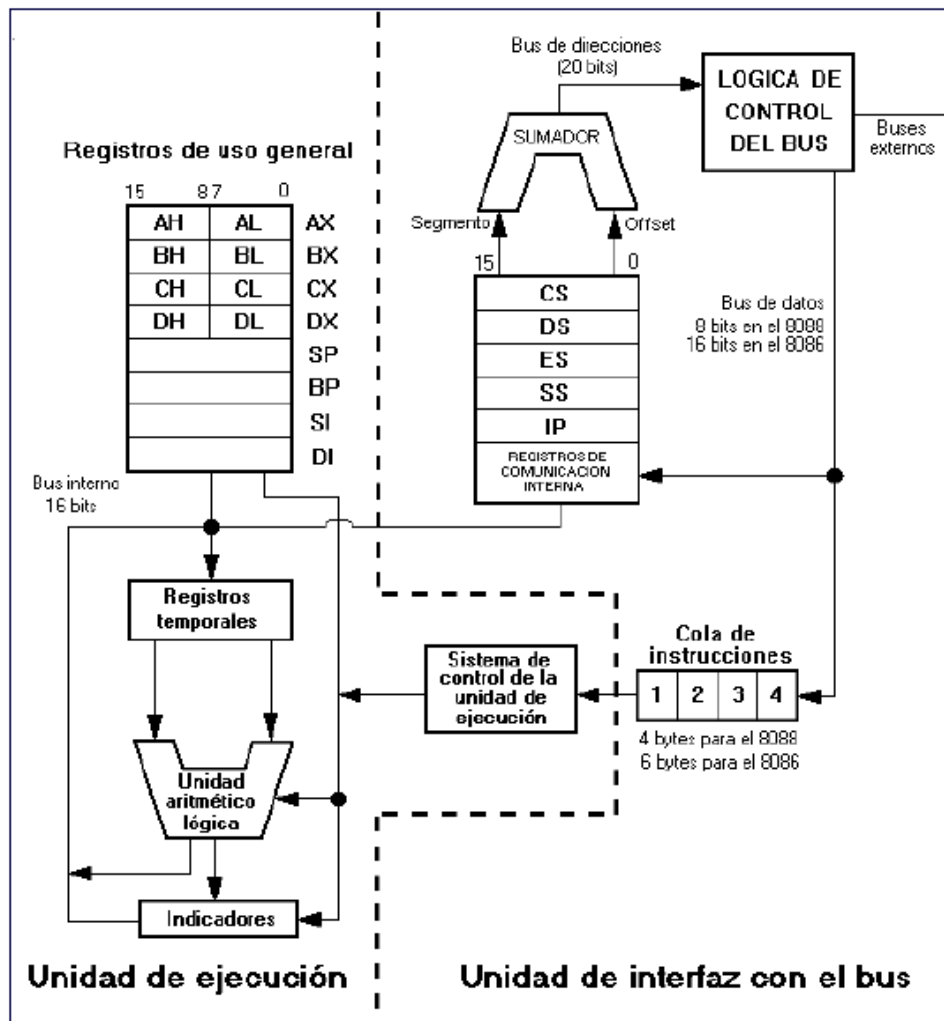


Figura 1. Estructura interna del procesador 8088

- BX** BASE. Además de ser un registro de propósito general, sirve para guardar la dirección base de los datos localizados en memoria (segmento de datos) como una dirección indirecta ya sea para lectura o escritura. Dicho de otra manera, se utiliza para almacenar la dirección de un dato del segmento de datos. Sirve también para guardar la dirección base de la tabla de datos referenciados por la instrucción XLAT. La parte alta del registro es BH, y parte baja BL.
- CX** CONTADOR. Este registro se utiliza para guardar el valor que indica la cantidad de ciclos a ejecutar, además, para contar rotaciones y desplazamientos de bits. También sirve para guardar el valor de repetición de instrucciones de manejo de cadenas. La parte alta del registro es CH, y parte baja CL.
- DX** DATOS. Sirve para guardar la parte más significativa del resultado de una multiplicación de 16 bits, la parte más significativa del dividendo para una división de 32 bits. También sirve para almacenar el número de puerto de entrada/salida. Su parte baja se utiliza para almacenar un caracter que se va a imprimir. La parte alta del registro es DH, y parte baja DL. Además, este registro sirve como de propósito general.

REGISTROS DE SEGMENTO. Estos registros se utilizan para almacenar los 16 bits más significativos de la dirección de inicio de segmento. Con ellos se tiene acceso a la pila, al código y a los datos. Debido a que un programa en ensamblador tiene 4 segmentos, existen cuatro registros (uno para cada tipo de segmento), mismos que se explican a continuación:

- CS SEGMENTO DE CODIGO. Este segmento se utiliza para almacenar las instrucciones del programa. Este registro se utiliza para almacenar la dirección de inicio del segmento de código del programa en ejecución. El valor del registro cambia por alguna instrucción de salto, llamada o retorno de un procedimiento. El procesador calcula la dirección efectiva de la siguiente instrucción a ejecutar haciendo la siguiente operación: $CS*10h+IP$.
- DS SEGMENTO DE DATOS. El segmento de datos sirve para almacenar los datos del programa en ejecución. El registro DS guarda la dirección de inicio del segmento de datos. La dirección efectiva de los datos manipulados por un programa se obtiene cuando el procesador hace cualquiera de las siguientes operaciones: $DS*10h+BX$, $DS*10h+DI$, $DS*10h+SI$.
- ES SEGMENTO EXTRA. Cuando la cantidad de datos que manipula un programa excede 64 KB (un segmento) puede utilizarse el segmento extra para almacenar más datos. Además, cuando un programa maneja cadenas de caracteres debe utilizarse obligatoriamente este segmento.
- SS SEGMENTO DE PILA. El segmento de pila es necesario para cualquier programa, aún cuando no utilice instrucciones de pila explícitamente, ya que la pila se usa para controlar la ejecución de un programa para manipular los retornos de procedimientos. Este registro guarda la dirección del inicio del segmento de pila. La dirección efectiva del elemento que está en el tope de la pila se genera sumando al valor de $SS*10h$ el contenido del registro apuntador de pila, de acuerdo a lo siguiente: $SS*10h+SP$, también puede hacerse referencia a datos en este segmento con el registro apuntador base BP. En este caso la dirección efectiva se calcula con la siguiente fórmula: $SS*10h+BP$

REGISTROS APUNTADOR E INDICE. Los registros apuntador e índice son registros de desplazamiento, puesto que al tener almacenada una dirección, hacen referencia a una localidad de memoria. Para tal efecto, cada registro hace una referencia a una dirección de memoria dentro de un segmento específico, esto significa que ya está definido el segmento con el cual trabaja cada uno de estos registros. De su función principal viene el nombre apuntador e índice. Además de su función especial de puntero, en caso de ser necesario, pueden utilizarse como de propósito general. Estos cuatro registros son de 16 bits, mismos que se explican a continuación:

- SP APUNTADOR DE PILA. Se utiliza para indicar la dirección del elemento que está en el tope de la pila. El procesador calcula la dirección efectiva del elemento en el tope de la pila con la siguiente operación: $SS*10h + SP$.
- BP APUNTADOR BASE. En algunas ocasiones es necesario modificar o almacenar datos en la pila sin mover el tope, para tal efecto se utiliza este registro guardando en él la dirección del dato a leer o modificar. La dirección efectiva dentro de la pila se calcula con: $SS*10h+BP$

- SI** INDICE FUENTE. Se usa con instrucciones de manejo de cadenas caracteres para indicar la posición de uno de ellos. Por tal motivo, almacena la dirección de la posición de un carácter específico dentro del segmento de datos. La dirección efectiva del dato dentro del segmento de datos la calcula el procesador con el siguiente cálculo: $DS*10h+SI$.
- DI** INDICE DESTINO. Este registro también se utiliza cuando el programa manipula cadenas de caracteres. En él se almacena la dirección del carácter en el segmento extra. La dirección efectiva del dato la calcula el procesador de la siguiente forma: $ES*10h+DI$.
- IP** APUNTADOR DE INSTRUCCIÓN. Este registro lo utiliza el procesador para señalar la dirección (dentro del segmento de código) de la siguiente instrucción a ejecutarse. La dirección efectiva está dada por: $CS*10h+IP$.

REGISTRO DE BANDERAS. Una bandera es en sí una celda con capacidad de almacenar un bit. Se dice que una bandera está activada cuando guarda un 1 lógico, y desactivada cuando guarda un cero. Las banderas que utiliza el procesador sirven para interpretar correctamente los resultados de las operaciones aritméticas, lógicas, o de desplazamiento o rotación. Además, algunas de ellas sirven para controlar de alguna manera el funcionamiento del procesador. El registro de banderas es de 16 bits y tiene el formato que se indica en la figura 2

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	OF	DF	IF	TF	SF	ZF	-	AF	-	PF	-	CF

Fig. 2 Registro de banderas

- AF** BANDERA DE ACARREO AUXILIAR. Se activa cuando existe acarreo o préstamo del bit 3 al bit 4 de una operación aritmética o lógica. Tiene importancia solamente en suma y resta. El bit manipulado por esta bandera sirve para corregir el resultado de operaciones en aritmética BCD.
- ZF** BANDERA DE CERO. Se activa cuando el resultado de una operación aritmética o lógica es cero, en caso contrario se desactiva.
- SF** BANDERA SE SIGNO. Tiene significado solo para operaciones de números con signo. Se activa cuando el resultado de una operación aritmética genera un valor negativo. En otras palabras, se dice que esta bandera almacena el signo (1 para negativo y 0 para positivo) de una operación.
- TF** BANDERA DE TRAMPA. Se activa cuando entra al modo de operación simple paso, ejecutando solo una instrucción a la vez. Tiene utilidad solo para depurar programas.

NOTAS DE LENGUAJES ENSAMBLADOR

- IF BANDERA DE INTERRUPCION. Habilita o deshabilita la terminal INTR (interrupt request) del procesador para atender o no las solicitudes interrupciones de hardware. Las atiende cuando la bandera está activada.
- DF BANDERA DE DIRECCION. Es utilizada para seleccionar el modo de operación de los registros índice fuente e índice destino en operaciones con cadenas. Cuando la bandera está desactivada se decrementan SI y/o DI, en caso contrario se incrementa(n). Esta bandera permite que se revise o copie una cadena de caracteres, haciéndolo byte por byte consecutivamente.
- CF BANDERA DE ACARREO. Indica acarreo o préstamo dentro de la posición del bit mas significativo después de una operación aritmética. El valor de este bit tiene importancia solo en sumas y restas. También esta bandera almacena el bit que sale cuando se ejecutan instrucciones de desplazamiento o rotación de bits.
- PF BANDERA DE PARIDAD. Se activa cuando el resultado de una operación aritmética o lógica tiene un número par de unos, se desactiva cuando el número de unos es impar. En la actualidad no tiene importancia el valor de esta bandera
- OF BANDERA SE SOBREFLUJO. Se activa cuando el resultado de una operación aritmética es mayor que el máximo valor que es posible representar con el número de bits del operando destino. Dicho de otra manera, se activa cuando si al sumar dos número con signo semejante o al restar dos números con signo opuesto produce un operando que excede la capacidad del registro de destino. También se activa cuando cambia el bit más significativo del operando durante una operación de desplazamiento o rotación.