



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

Estructura y Programación de Computadoras

Grupo: 02 - Semestre: 2021-1

Tarea 2:
Acceso a memoria en modo real y modo protegido

FECHA DE ENTREGA: 14/10/2020

Alumno:
Téllez González Jorge Luis



Índice

1. Direccionamiento de memoria en modo real	2
1.1. Funcionamiento y cálculo de direcciones	2
1.2. Registros de segmento y desplazamiento predeterminados	4
2. Direccionamiento de memoria en modo protegido	5
2.1. Funcionamiento	6
2.2. Cálculo de una dirección de memoria	7

1. Direcccionamiento de memoria en modo real

El modo real es un modo de operación presente en los procesadores basados en la arquitectura Intel x86 como el 80286 y sus versiones superiores. La operación en **modo real** permite a un microprocesador direccionar únicamente el primer megabyte de espacio de memoria, al cual se le conoce como **sistema de memoria real, convencional o DOS**.

Este modo de operación es requerido por el sistema operativo DOS para funcionar, mientras que Windows no hace uso de este modo. Además, el modo real carece de conceptos de protección de memoria o multitarea a nivel de hardware. Es importante destacar que todas las CPU basadas en x86 a partir del 80286 empiezan su funcionamiento en modo real por defecto al momento de energizarse o cuando se reinicia el procesador.

La operación en modo real permite que el software de aplicación escrito para los microprocesadores de la serie 8086/8088 (los cuales contienen únicamente un 1[MB] de memoria) funcionen en los microprocesadores 80286 y superiores.

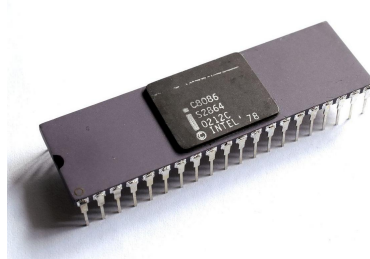


Figura 1: En modo real el procesador imita el funcionamiento de un Intel 8086.

1.1. Funcionamiento y cálculo de direcciones

Para acceder a una posición de memoria en este modo se utiliza la combinación de una dirección de segmento y otra dirección de desplazamiento: todas las direcciones de memoria en modo real deben de cumplir con tal estructura.

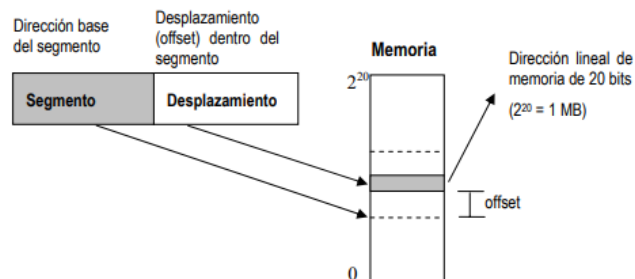


Figura 2: Selección de una posición de memoria segmento más desplazamiento.

La **dirección de segmento**, ubicada dentro de uno de los registros de segmento, define la dirección inicial de cualquier segmento de memoria de 64[Kbytes]. Por otra parte, la **dirección de desplazamiento** selecciona cualquier posición dentro del segmento de memoria de 64[Kbytes]. A raíz de esto, los segmentos en el modo real siempre tienen una longitud total de 64[Kbytes].

En la ilustración que se muestra a continuación se tiene un segmento de memoria que empieza en la posición **10000H** y termina en la posición **1FFFFH**. También se muestra cómo una dirección de desplazamiento de **F000H** selecciona a la posición **1F000H** en el sistema de memoria. De allí que el desplazamiento *es la distancia por encima del inicio del segmento*.

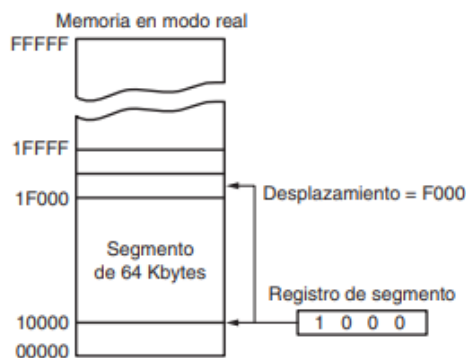


Figura 3: Selección de una posición de memoria segmento más desplazamiento.

El registro en la figura contiene 1000H, y aún así, direcciona hacia un segmento inicial en la posición 10000H, esto es debido a que en el modo real a cada registro de segmento se le adjunta internamente un **0H** en su extremo más a la derecha. De esta forma, se generan direcciones de memoria de 20[bits], lo cual le permite acceder al inicio de un segmento.

El microprocesador debe de generar una dirección de memoria de 20[bits] para poder acceder a una posición dentro del primer megabyte de memoria. Por ejemplo, si un registro de memoria contiene el valor **1201H**, direccionará a un segmento de memoria que inicia en la posición **12010H**. Debido a que el valor 0H se adjunta de forma interna, los segmentos del modo real pueden comenzar sólo dentro de un límite de 16 [bits] en el sistema de memoria. A este límite se le denomina como **párrafo**.

Entre otros aspectos, se encuentran los siguientes:

- Una vez conocida la dirección inicial de un segmento de memoria es posible encontrar la dirección final sumándole FFFFH. Por ejemplo, si un registro de segmento contiene 3000H, la primera dirección del segmento es 30000H y la última dirección será $30000H + FFFFH = 3FFFFH$.
- La dirección de desplazamiento, que forma parte de la dirección, se suma al inicio del segmento para direccionar una posición de memoria dentro del segmento de memoria. Por ejemplo, si la dirección de segmento es 1000H y la dirección de desplazamiento es 2000H, el microprocesador direcciona la

posición de memoria 12000H.

La dirección de segmento y desplazamiento algunas veces se escribe como 1000:2000 para representar una dirección de segmento de 1000H, con un desplazamiento de 2000H.

- Algunos modos de direccionamiento combinan más de un registro y un valor de desplazamiento para formar una dirección de desplazamiento. Cuando esto ocurre, la suma de estos valores podría exceder de FFFFH.

Por ejemplo, la dirección a la que se accede en un segmento cuya dirección de segmento es 4000H y cuya dirección de desplazamiento se especifica como la suma de F000H más 3000H, accederá a la posición de memoria 42000H en vez de la posición 52000H. Al sumar F000H y 3000H forman una suma de 16 bits (módulo 16) de 2000H, que se utiliza como la dirección de desplazamiento, y no 12000H, la verdadera suma. El acarreo de 1 ($F000H + 3000H = 12000H$) se descarta en esta suma para formar la dirección de desplazamiento de 2000H. De este modo, la dirección generada es de 4000:2000, o 42000H.

1.2. Registros de segmento y desplazamiento predeterminados

El microprocesador tiene un conjunto de reglas que se aplican a los segmentos siempre que se direcciona la memoria. Por ejemplo, el registro de segmento de código siempre se utiliza con el apuntador de instrucciones para direccionar la siguiente instrucción en un programa. Esta combinación es **CS:IP** o **CS:EIP**, dependiendo del modo de operación del microprocesador. El registro de segmento de código define el inicio del segmento de código y el apuntador de instrucciones localiza la siguiente instrucción dentro del segmento de código.

Esta combinación localiza la siguiente instrucción que va a ejecutar el microprocesador. Por ejemplo, si CS = 1400H e IP/EIP = 1200H, el microprocesador obtiene su siguiente instrucción de la posición de memoria 14000H + 1200H, o 15200H.

<i>Registro de segmento</i>	<i>Dirección inicial</i>	<i>Dirección final</i>
2000H	20000H	2FFFFH
2001H	20010H	3000FH
2100H	21000H	30FFFFH
AB00H	AB000H	BAFFFFH
1234H	12340H	2233FH

Figura 4: *Combinaciones predeterminadas para direcciona memoria en un procesador Intel de 16 bits.*

Otra de las combinaciones predeterminadas es la **pila**. Se hace referencia a los datos de la pila a través del segmento de pila en la posición de memoria direccionada por el apuntador de la pila (SP/ESP)

o por el apuntador (BP/EBP). Se hace referencia a estas combinaciones como SS:SP (SS:ESP) o SS:BP (SS:EBP). Por ejemplo, si SS = 2000H y BP = 3000H, el microprocesador direcciona la posición de memoria 23000H para la posición de memoria del segmento de la pila. En modo real *sólo los 16 bits más a la derecha de la dirección del registro extendido direccionan una posición dentro del segmento de memoria.*

2. Direccionamiento de memoria en modo protegido

El direccionamiento de memoria en modo protegido se encuentra en los procesadores Intel 80286 y superiores. Su principal característica es que permite el acceso a los datos y programas que se encuentren localizados por encima del primer megabyte de memoria, además de aquellos ubicados dentro de este. Sistemas operativos como **Windows** utilizan este modo al iniciar su funcionamiento. A diferencia del modo real, en el modo protegido el procesador utiliza un direccionamiento segmentado (no lineal) en lugar de un direccionamiento lineal.

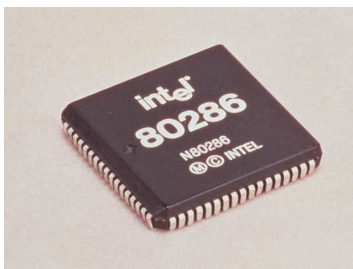


Figura 5: Intel 80286.

Para direccionar esta sección extendida del sistema de memoria se requiere un cambio en el esquema de direccionamiento mediante un segmento más un desplazamiento que se utiliza con el direccionamiento de memoria en modo real. Cuando los datos y los programas se direccionan en memoria extendida, la dirección de desplazamiento se sigue utilizando para acceder a la información localizada dentro del segmento de memoria. Sin embargo, la dirección de segmento **no se encuentra presente en el modo protegido.**

En su lugar, el registro de segmento contiene un **selector** que escoge un descriptor de una tabla de descriptores. El **descriptor** presenta la posición, longitud y derechos de acceso del segmento de memoria. Como el registro de segmento y la dirección de desplazamiento siguen accediendo a la memoria, las instrucciones en modo protegido son idénticas a las instrucciones en modo real. La diferencia entre los modos es la manera en que el microprocesador interpreta el registro de segmento para acceder al segmento de memoria.

Otra diferencia en los microprocesadores 80386 y superiores es que la dirección de desplazamiento puede ser de 32 [bits], en vez de 16 [bits] en modo protegido. Una dirección de desplazamiento de 32 [bits] permite que el microprocesador acceda a los datos dentro de un segmento que puede tener hasta 4[GB] de longitud.

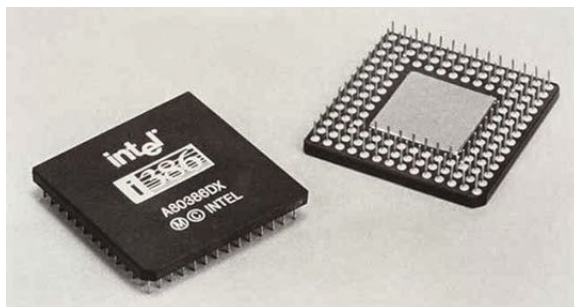


Figura 6: Intel 80386.

2.1. Funcionamiento

El **selector**, que se encuentra en el registro de segmento, selecciona uno de 8192 descriptores de una o dos tablas de descriptores. El descriptor presenta la posición, longitud y derechos de acceso del segmento de memoria. El registro de segmento sigue seleccionando un segmento de memoria, pero no de forma directa como en el modo real; de allí que el direccionamiento es segmentado.

Hay dos tablas de descriptores que se utilizan con los registros de segmento: una contiene descriptores globales y la otra contiene descriptores locales. Los **descriptores globales** contienen definiciones de segmentos que se aplican a todo los programas, mientras que los **descriptores locales** son por lo general únicos para una aplicación. Al descriptor global también se le conoce como **descriptor de sistema** mientras que al descriptor local se le puede denominar como **descriptor de aplicación**.

Descriptor del 80286				
7	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	6	
5	Derechos de acceso	Base (B23–B16)	4	
3	Base (B15–B0)		2	
1	Límite (L15–L0)		0	

Descriptor del 80386 al Pentium 4				
7	Base (B31–B24)	G D 0 A V	Límite (L19–L16)	6
5	Derechos de acceso	Base (B23–B16)		4
3	Base (B15–B0)		2	
1	Límite (L15–L0)		0	

Figura 7: Formatos de descriptores para los microprocesadores 80286 y 80386.

Cada tabla contiene 8192 descriptores, por lo que hay un total de 16,384 descriptores disponibles para una aplicación en un momento dado. Como el descriptor describe un segmento de memoria, esto permite que se describan hasta 16,384 segmentos de memoria para cada aplicación. Como un segmento de memoria puede tener hasta 4[GB] de longitud, esto significa que una aplicación podría tener acceso a $(4[\text{GB}] \times (16,384 [\text{bytes}]))$ de memoria, es decir, 64 [TB].

La porción de dirección base del descriptor indica la posición inicial del segmento de memoria. Para el microprocesador 80286 la dirección base es de 24 [bits], por lo que los segmentos empiezan en cualquier posición de sus 16 [MB] de memoria. La limitación en cuanto al límite de los párrafos se elimina en los microprocesadores que trabajan en este modo, por tanto, los segmentos pueden iniciar en cualquier dirección. Los microprocesadores 80386 y superiores pueden utilizar una dirección base de 32 bits que permite que los segmentos inicien en cualquier posición dentro de sus 4 Gbytes de memoria.

El límite de segmento contiene la última dirección de desplazamiento de un segmento. Por ejemplo, si un segmento comienza en la posición de memoria F00000H y termina en la posición F00FFH, la dirección base es F00000H y el límite es FFH. Para el microprocesador 80286, la dirección base es F00000H y el límite es 00FH. Para los microprocesadores 80386 y superiores, la dirección base es 00F00000H y el límite es 000FFH.

Existe otra característica incluida en el descriptor de los microprocesadores del 80386 al Pentium 4 que no se encuentra especificada en el descriptor del 80286: el bit G, o **bit de granularidad**. Si G = 0, el límite especifica un límite de segmento de 00000H a FFFFFH. Si G = 1, el valor del límite se multiplica por 4 Kbytes (se adjunta a FFFH). El límite sería entonces de 00000FFFFH a FFFFFFFFH si G = 1.

2.2. Cálculo de una dirección de memoria

Con lo anterior, si se tiene un segmento con dirección base de 10000000H y un límite de 001FFH con un bit de granularidad G=0, el resultado es el siguiente:

$$Base = Inicio = 10000000H$$

$$G = 0$$

$$Final = Base + Límite = 10000000H + 001FFFFH = 10001FFFH$$

Ahora, con G=1:

$$Base = Inicio = 10000000H$$

$$G = 1$$

$$Final = Base + Límite = 10000000H + 001FFFFFH = 101FFFFFH$$

El límite queda adjuntado con FFFH para determinar la dirección de segmento final.



Referencias

- [1] Arquitectura Intel IA-32 Modo Real. Recuperado de: http://artemisa.unicauca.edu.co/~emezav/operativos/materiales/modo_real.pdf. Fecha de consulta: 12/10/2020.
- [2] Modo protegido. Recuperado de: http://usuarios.sion.com/iop-chagui/modo_protegido.html. Fecha de consulta: 12/10/2020.
- [3] Modos de direccionamiento. Recuperado de: <https://lenguajeensambladorherramientasj.wordpress.com/2013/02/24/modos-de-direccionamiento/>. Fecha de consulta: 12/10/2020.
- [4] Brey, B. B. (1991). *Microprocesadores Intel*. Prentice Hall, 7th edition.

Los créditos de las fotografías pertenecen a sus respectivos autores. ©