

Acceso Directo a Memoria (DMA)

En el procesador x86 de Intel existen las instrucciones de ensamblador **IN** y **OUT** para la transferencia de información hacia y desde los dispositivos periféricos respectivamente. La comunicación se realiza entre los registros de la CPU y las interfaces de entrada/salida, las cuales se activan con una dirección (conocida como de entrada/salida) cuyo valor está también en el rango de las direcciones de memoria. Para poder distinguir cuando se trata de una transferencia desde/hacia dispositivos periféricos y transferencia entre RAM se requiere de una señal de control de salida de la CPU denominada IO/M', la cual si tiene el valor lógico uno activará el procesador de E/S para preparar la interface involucrada en la transferencia. Por el contrario, si tiene el valor lógico cero, activa los módulos de memoria

Para la transferencia hacia y desde la memoria se emplea la instrucción **MOV**, con la cual si se desea leer entonces la dirección o el nombre de la variable de memoria se escribe como segundo parámetro u origen de la transferencia, esto es:

```
mov ax, dato
mov ax, [bx]
```

En el primer caso, el origen de la transferencia es la variable de memoria **dato**, mientras que en el segundo se hace referencia a una localidad de memoria cuyo valor de desplazamiento está contenido en el registro **BX**.

Si la transferencia es de escritura, entonces se anota primero la dirección de memoria o el nombre de una variable como se muestra en estos ejemplos:

```
mov dato, ax
mov [bx], ax
```

Un caso especial lo representa la carga en memoria RAM de un archivo que se encuentra en un dispositivo de almacenamiento masivo de datos (unidad de disco), pues se tendrían que ejecutar las instrucciones IN y MOV en secuencia para cada byte del archivo y lograr transferir hacia la RAM dicho archivo completo. Las gráficas siguientes mostrarían el sentido de las transferencias con cada instrucción.

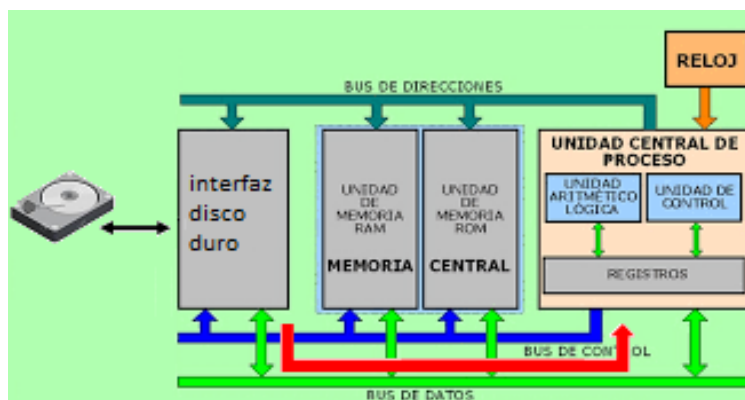


Fig. a) Instrucción IN

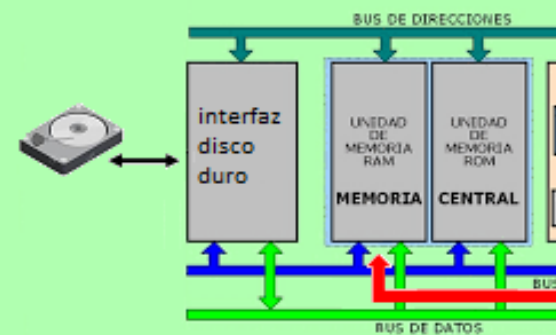


Fig b) Instrucción MOV

Tanto la tarea de cargar como la de guardar un archivo en el disco duro son muy cotidianas, y con el propósito de reducir el tiempo de dichas tareas se implementa el Acceso Directo a Memoria (DMA)

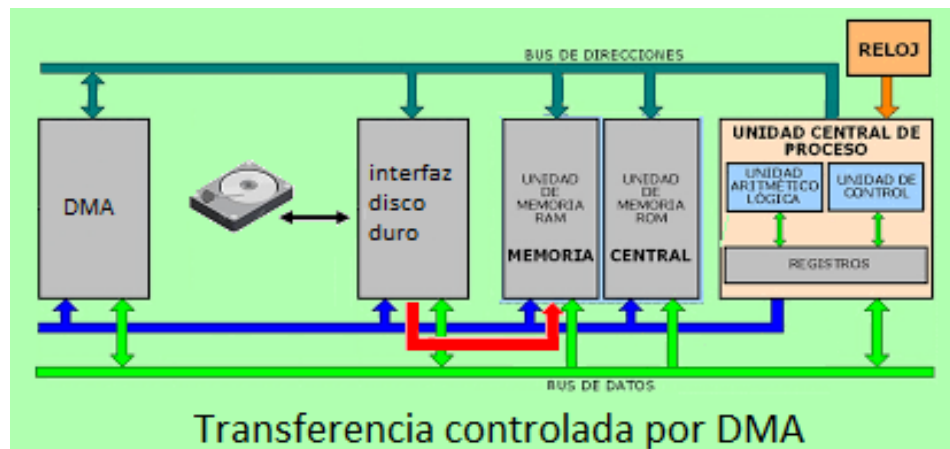
– *Direct Memory Access*) mediante un circuito controlador que es activado cuando el CPU requiere realizar una de estas dos tareas.

Cuando esto sucede, el CPU debe enviar al controlador DMA la dirección de inicio de la memoria donde se cargará el archivo y el tamaño del mismo para saber cuándo terminar la transferencia.

Para iniciarla transferencia DMA, el CPU o su procesador de entrada/salida debe hacer una solicitud hacia el controlador DMA (chip 8237) activando la línea de control DMA-REQ a lo que el controlador responde con la señal DMA-ACK que hará que el CPU libere los buses y lo comunique activando su señal HLDA (*Hold Acknowledge*). Hay que recordar que los buses conectan con el CPU mediante registros separadores implementados con compuertas de tres estados lo que permite una “desconexión física”.

Para el caso de la transferencia de un archivo de disco a memoria, el controlador DMA envía al dispositivo una señal de lectura y a la RAM una de escritura manteniendo la dirección de memoria en un registro interno que se irá incrementando hasta llegar al tamaño del archivo, momento en el que el circuito 8237 notifica mediante una interrupción que ha finalizado el proceso.

A continuación se muestra el diagrama de la transferencia directa entre un dispositivo Entrada/salida y memoria RAM.



La tarea descrita anteriormente permite al CPU encargarse de cálculos o tareas que no requieran accesos a la RAM eficientando la operación del mismo.

La siguiente liga muestra algunas tareas esenciales del controlador DMA el cual deberás leer.

<https://conceptosarquitecturadecomputadoras.wordpress.com/acceso-directo-a-memoria-dma/>