



## Ensayo

Nombre del alumno: Yahir Aguilar Sicalhua.

Nombre del tema: Estructura de bits y bytes. Modos de direccionamiento.

Parcial: 1.

Nombre de la materia: Microprocesadores.

Nombre del profesor: Juan José Ojeda Trujillo.

Nombre de la licenciatura: Ingeniería en Sistemas Computacionales.

Cuatrimestre: 7.

Mientras el bit es la unidad mínima de información byte es un conjunto fijo de bits. Aunque hay excepciones, en la mayoría de casos (sobre todo en relación al sector de la informática), un byte cuenta con 8 bits, de ahí su equiparación con el concepto de <<octeto>>.

## **UNIDAD I. ESTRUCTURA DE BITS Y BYTES.**

### **1.1.- Sistemas numéricos decimal, binario y hexadecimal.**

Bases numéricas decimal La forma de interpretar los números decimales es a través de su posición (lugar que ocupa el dígito en el número total. unidad, decena, centena, etc.), que se pueden relacionar con las denominaciones de unidades, decenas, centenas, etc.

### **1.2.- Formatos de datos.**

Buses en los microprocesadores, partir del desarrollo de la arquitectura Von Newman, se introdujo en los microprocesadores el concepto de programa almacenado y la ejecución secuencial del programa almacenado, para que el microprocesador pueda comunicarse con los diversos periféricos que tiene existen unos medios llamados buses.

### **1.3.- Funcionamiento interno de una PC.**

Los sistemas de cómputo basados en microprocesadores se componen de tres bloques fundamentales, la Unidad Central de Procesamiento – CPU, Dispositivos de memoria y Puertos de Entrada / Salida, de igual forma el microprocesador está conformado por varios bloques, entre ellos están la Unidad Aritmético – lógica (ALU), una Unidad de Control (UC) y un bloque o matriz de registros.

### **1.4.- La evolución de microprocesadores.**

En los comienzos de la implementación de circuitos lógicos el diseñador debía poseer amplios conocimientos tanto en lógica digital como en los componentes y dispositivos que debía acoplar, esto presentaba inconvenientes, la llamada lógica cableada no permitía

hacer modificaciones sin afectar la construcción física del circuito, en 1970 esto cambia con la aparición del microprocesador, con lo que aparece lo que se denomina Lógica programable, la cual permite modificar el comportamiento lógico digital del circuito sin tener que cambiar su configuración física.

## **1.5.- Arquitectura del microprocesador 80x86.**

Los procesadores de 16 bits fueron una nueva generación de microprocesadores desarrollados para reemplazar o completar a las microcomputadoras de 8 bits de los años setenta, que fueron las que comenzaron la revolución de las microcomputadoras.

El 8086 fue diseñado para trabajar con lenguajes de alto nivel, disponiendo de un soporte hardware con el que los programas escritos en dichos lenguajes ocupan un pequeño espacio de código y pueden ejecutarse a gran velocidad. Esta concepción, orientada al uso de compiladores, se materializa en un conjunto de facilidades y recursos, y en unas instrucciones entre las que cabe destacar las que permiten efectuar operaciones aritméticas de multiplicar y dividir, con y sin signo; las que manejan cadenas de caracteres, etc.

## **1.6.- Estructura de un programa ejecutable cargado en memoria.**

El 8086 usa un esquema llamado segmentación, para acceder correctamente a un megabyte completo de memoria, con referencias de direcciones de sólo 16 bits, y todo esto gracias a la utilización de registros de segmento que dividen esencialmente el espacio de memoria en segmentos de 64K de longitud, que pueden estar separados entre sí, adyacentes o superpuestos, y que comienzan en una dirección divisible por 16.

## **1.7.- Arreglo de registros internos.**

El 8086/88 dispone de 4 registros de datos, 4 registros de segmento, 5 registros de índice y 1 registro de estado. Registros de datos. Los registros de datos son de 16 bits, aunque están divididos. lo que permite su acceso en 8 bits. Estos registros son de propósito general, aunque todos tiene alguna función por defecto.

## **1.8.- Operación en modo real.**

El modo real (también llamado modo de dirección real en los manuales de Intel) es un modo de operación del 8086 y posteriores CPUs compatibles de la arquitectura x86. El modo real está caracterizado por 20 bits de espacio de direcciones segmentado (significando que solamente se puede direccionar 1 MB de memoria), acceso directo del

software a las rutinas del BIOS y el hardware periférico, y no tiene conceptos de protección de memoria o multitarea a nivel de hardware. Todos los CPUs x86 de las series del 80286 y posteriores empiezan en modo real al encenderse el computador; los CPUs 80186 y anteriores tenían solo un modo operacional, que era equivalente al modo real en chips posteriores.

## **1.9.- Operación en modo protegido.**

El modo protegido es un modo operacional de los CPUs compatibles x86 de la serie 80286 y posteriores. El modo protegido tiene un número de nuevas características diseñadas para mejorar la multitarea y la estabilidad del sistema, tales como la protección de memoria, y soporte de hardware para memoria virtual como también la conmutación de tarea. A veces es abreviado como p-mode y también llamado Protected Virtual Address Mode (Modo de Dirección Virtual Protegido) en el manual de referencia de programador del iAPX 286 de Intel, (Nota, iAPX 286 es solo otro nombre para el Intel 80286).

## **UNIDAD II. MODOS DE DIRECCIONAMIENTO.**

### **2.1.- Direccionamiento por registros.**

L Cuando ambos operando son un registro.

Ejemplo: MOV AX, BX; transfiere el contenido de BX en AX

### **2.2.- Direccionamiento inmediato.**

Cuando el operando origen es una constante.

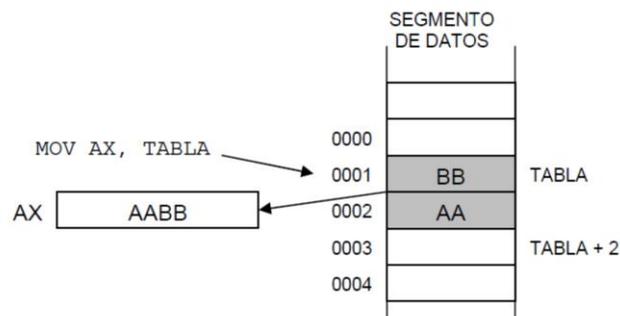
Ejemplo: MOV AX,500; carga en AX el valor 500.

### **2.3.- Direccionamiento directo.**

Cuando el operando es una dirección de memoria. Ésta puede ser especificada con su valor entre [ ], o bien mediante una variable definida previamente (cómo definir etiquetas se verá más adelante).

Ejemplo: MOV BX, [1000]; almacena en BX el contenido de la dirección de memoria DS:1000.

MOV AX, TABLA; almacena en AX el contenido de la dirección de memoria DS: TABLA.



**MOV [BP],CX** ; almacena en la dirección apuntada por BP en contenido de CX.

## 2.4.- Direccionamiento base más índice.

E Cuando el operando está en memoria en una posición apuntada por el registro BX o BP al que se le añade un determinado desplazamiento Ejemplo:

MOV AX, [BP] + 2; almacena en AX el contenido de la posición de memoria que resulte de sumar 2 al contenido de BP (dentro de segmento de pila). Equivalente a MOV AX, [BP + 2].

## 2.5.- Direccionamiento relativo.

Cuando la dirección del operando se obtiene de la suma de un registro base (BP o BX), de un Índice (DI, SI) y opcionalmente un desplazamiento.

Ejemplo: MOV AX, TABLA[BX][DI]; almacena en AX el contenido de la posición de memoria apuntada por la suma de TABLA, el contenido de BX y el contenido de DI.

### 2.5.1.- Instrucciones para transferencia de datos.

Las instrucciones de transferencia de datos copian datos de un sitio a otro y son: MOV, CHG, XLAT, LEA, LDS, LES, LAHF, SAHF, PUSH, PUSHF, POP, POPF.

MOV realiza la transferencia de datos del operando de origen al destino. Como ya hemos visto en la parte de los modos de direccionamiento, MOV admite todos los tipos de direccionamiento. Ambos operandos deben ser del mismo tamaño y no pueden estar ambos en memoria.

## 2.5.2.- Instrucciones aritméticas y lógicas.

Este tipo de instrucciones realizan operaciones aritméticas con los operandos. Y son: ADD, ADC, DAA, AAA, SUB, SBB, DAS, AAS, NEG, MUL, IMUL, AAM, DIV, IDIV, AAD, CBW, CWB, INC, DEC.

ADD y ADC realizan la suma y la suma con acarreo (bit CF del registro de estado) de dos operandos, respectivamente, y guardan el resultado en el primero de ellos. Admiten todos los tipos de direccionamiento (excepto que ambos operando estén en memoria).

## 2.5.3.- Instrucciones para control de programa.

Se utilizan para el control del programa, son instrucciones de salto, bucles y llamadas a procedimientos.

### Instrucciones de salto.

Estas instrucciones permiten saltar a otras partes del código. Todas cambian el registro IP (contador de programa) y el registro CS (segmento de código) si es un salto lejano. Un salto es lejano cuando la dirección a la que se salta no está en el mismo segmento de código.

Existen dos tipos de saltos: los absolutos; en lo que se especifica la dirección absoluta a la que se salta; y los relativos; que son saltos hacia delante o hacia atrás desde el valor de IP.

### JMP.

Realiza un salto incondicional a la dirección especificada. La siguiente tabla relaciona los tipos de saltos y los argumentos que puede tomar esta instrucción.

El microprocesador (CPU o simplemente procesador) es el chip más importante del ordenador. Sin él, no podría funcionar el sistema.

Actúa como el conductor y supervisor de los componentes de hardware del sistema.

Es fundamental tener estas herramientas y su actualización para poder desempeñarnos de una manera más eficiente en nuestra profesión teniendo en cuenta que vivimos en un mundo globalizado, donde la tecnología es una necesidad básica y dentro de esta la UCP.

Fuentes de información:

<https://plataformaeducativauds.com.mx/libro.php?idLibro=169566880616>