

## LOS MICROPROCESADORES

Ing. CELESTE O. D'INCA Profesor titular de Computadoras Electrónicas en la Facultad de Ingeniería en Electrónica y Electricidad.

### I. Introducción

Un micro procesador es un componente de integración en gran escala, controlado por programa, capaz de ejecutar operaciones aritméticas y lógicas sobre ciertos datos. La forma de empleo de un microprocesador es la de llevar a cabo un sistema donde las funciones lógicas no son realizadas mediante interconexión de una serie de circuitos, sino mediante una serie de instrucciones, que conforman el programa, almacenadas en una memoria de semiconductor.

La flexibilidad del sistema así constituido, frente a los circuitos de lógica cableada, es enorme, por cuanto cambiando el programa se modifica la función. Este cambio puede hacerse en forma parcial, para adaptarse a nuevas situaciones, o en forma total para realizar nuevas operaciones.

Hasta el año 1971, los proyectistas podían elegir entre dos posibilidades para resolver un problema de control, o una minicomputadora o una lógica cableada. La aparición de los microprocesadores, ofreció una tercera alternativa, dado que se inserta directamente entre los dos sectores considerados, disminuyendo los costos y facilitando la realización de estos sistemas de control.

Dentro de las causas que contribuyen en la actualidad al gran desarrollo de los sistemas a microprocesador, existen cuatro factores primordiales, a saber:

- a) Costos
- b) Tiempo de desarrollo de un proyecto.
- c) Confiabilidad del sistema
- d) Posibilidad de adecuación de los equipos desarrollados a nuevas especificaciones.

En cuanto hace al primer factor, o sea a los costos, la tendencia

del mercado es la de entregar al consumidor productos de sofisticación creciente a precios decrecientes. Inclusive estos productos más económicos son más sofisticados, pues aumenta año a año la densidad de componentes en un factor del ciento por ciento.

El tiempo de desarrollo de un proyecto, una vez que los encargados han alcanzado una experiencia mínima, presenta varias ventajas sobre los sistemas de lógica cableada, por cuanto puede recurrirse al desarrollo de sistemas standard, cuyo cambio de función es realizable mediante el simple cambio de una memoria de lectura solamente, correctamente programada.

Debe siempre tenerse en cuenta que el tiempo de desarrollo de los programas no es despreciable, ni tampoco es despreciable el costo de dicho programa, pues es necesario cumplir una serie de procedimientos de control del mismo que necesitan ya sea de una computadora, o ya sea de la implementación del sistema. Solo la experiencia del proyectista puede obviar algunos de los pasos.

Confiabilidad, es un término que Implica seguridad, ya sea seguridad en el funcionamiento, o dicho de otro modo, cuan poca es la cantidad de fallas o errores que comete el sistema en funcionamiento, ya sea eficiencia durante un prolongado lapso de tiempo, o sea que tan grande es el intervalo de tiempo que transcurre entre dos fallas sucesivas del sistema. Es evidente que cuanto menor es el número de componentes, el número de conexiones, el número de soldaduras, etc., menor será la probabilidad de fallas, asimismo en un sistema programado que debe responder siempre de cierta manera ante una misma solicitud, es simple hacerle un programa de pruebas que ayude a detectar eventuales fallas.

Los sistemas a microprocesador permiten una rápida modificación de sus funciones a fin de adecuar los dispositivos a nuevos requerimientos, esto hace que puedan actuar durante mayores lapsos de tiempo sin quedar obsoletos.

#### **I. 1: Significado del uso de un microprocesador:**

El microprocesador como elemento lógico, presenta al proyectista un cierto número de nuevos instrumentos y métodos de aproximación para la resolución de problemas de control. Pero si bien hay notables ventajas, también existen dificultades, sobre todo en los comienzos, que no son fácilmente evitables. El primer obstáculo que debe superar un proyectista, familiarizado con las técnicas de la lógica cableada, radica en el "SOFTWARE" o sea en la programación del problema.

En consecuencia, previo al inicio de cualquier proyecto, debe tenerse un buen conocimiento de lo que se pretende obtener con la programación a desarrollar, además de lo que puede alcanzarse programando ese particular microprocesador, contando con todo el apoyo

dado por los fabricantes, y como alcanzar los resultados deseados mediante las herramientas disponibles para la resolución del problema en cuestión.

La diferencia más notable entre la utilización de un microprocesador y la lógica convencional, pueden sintetizarse en lo siguiente:

- a) En la lógica tradicional, un conjunto de componentes es cableado en forma tal que satisfaga una función lógica, llevándose a cabo todas las operaciones elementales en paralelo.
- b) En un microprocesador existe una sola lógica central, la que puede interconectarse de distintas formas mediante la interpretación de un programa, en forma tal de satisfacer la misma función. Es evidente que cada operación lógica elemental será llevada a cabo en una cierta unidad de tiempo que dependa de cada procesador, y en consecuencia la función final será implementada en forma serie.

El proyectista de lógica discreta define un conjunto de funciones y las interconecta para obtener el resultado específico.

El proyectista de lógica programada debe decir al dispositivo como interconectarse en forma automática instante por instante para llevar a cabo su función.

El objetivo del "SOFTWARE" es el de ayudar al proyectista en la puesta a punto instante por instante de la forma de actuar del procesador. Tal descripción debe ser interpretada por el circuito y llevada a cabo mediante un programa. Este programa, es un conjunto ordenado de instrucciones pertenecientes al conjunto propio del microprocesador. En realidad el programa en definitiva no es más que una secuencia de unos y ceros ubicados en una zona determinada de la memoria que dicen a la unidad de procesamiento que debe hacer en cada instante de tiempo.

## **I. 2: Interacción Hardware - Software**

Según dije antes el software presenta algunos inconvenientes que dependen exclusivamente de la formación de las personas que hacen uso de los microprocesadores. Si se trata de individuos familiarizados con las técnicas de la lógica discreta, debe operarse en ellos un cambio de mentalidad, incluyendo una actualización de sus conocimientos, para poder pasar a trabajar con una lógica programada.

Esto no quiere decir que aquellas personas familiarizadas con el uso de computadoras, generalmente programadores o analistas de sistemas, tengan menos dificultades, dado que el usuario de micro-procesadores debe no sólo conocer el software sino también tener bien en claro el funcionamiento del hardware del dispositivo.

Quien está acostumbrado a programar computadores digitales, en las máquinas grandes o medianas, tiene un gran apoyo por parte de las mismas, con sus sistemas operativos, sus detectores de errores en la compilación y su operación en lenguajes elevados, tales como Fortran, Cobol, Algol, etc., y además, posee un gran número de periféricos a su alcance, así como grandes capacidades de memoria, y un sistema supervisor. En cambio en un microprocesador ocurre generalmente todo lo contrario, tendrá a su disposición una limitada capacidad de memoria, un limitado número de periféricos y todos ellos muy simples y que además deberán ser interconectados mediante los interfaces apropiados. También en cuanto hace a los lenguajes, éstos son muy primitivos, con un limitado número de instrucciones, que si bien pueden procesarse mediante una computadora en un lenguaje elevado y después hacer la traducción, si se desea aprovechar íntegramente las capacidades del dispositivo, deberá conocerse a fondo su lenguaje, y aún más en caso de necesitar hacer correcciones en su memoria, deberá trabajarse en binario.

Todo esto hace que el trabajo del proyectista vuelva a hacerse de lógica discreta, o por lo menos muy similar a él, dado que deberá conocerse perfectamente la arquitectura y la filosofía de diseño no sólo del microprocesador sino de la unidad de memoria que lo acompañará, y por qué no de sus interfaces y circuitos asociados.

Resumiendo, podemos decir que si bien es necesaria cierta especialización de los individuos que deban hacer uso del microprocesador, también es cierto que dicha especialización es preferible que haya sido orientada hacia el trabajo con lógica discreta, dado que los mismos siempre tendrán en cuenta no sólo el origen específico de los microprocesadores, sino también estarán en condiciones de analizar con mayor profundidad la arquitectura y la filosofía de diseño de los mismos.

## **II: Organización de un microprocesador.**

Un microprocesador consiste en un conjunto de bloques funcionales, cada uno de los cuales es responsable de llevar a cabo una determinada tarea en el conjunto.

Un microprocesador puede estar contenido en uno o en varios circuitos integrados monolíticos, pero de cualquier forma deberá estar formado por los siguientes bloques funcionales:

- 1) Memoria
- 2) Unidad lógica aritmética
- 3) Registros
- 4) Barras ómnibus
- 5) Unidad de control

A continuación analizaremos en particular cada uno de estos componentes.

## II. 1: Memoria

El elemento que caracteriza a todo microprocesador como un dispositivo de programa interno, lo mismo que a toda computadora, es la memoria. Esta es un bloque lógico digital que contiene las informaciones sobre las cuales se deberá operar.

Estas informaciones son de dos tipos, datos e instrucciones.

Los datos representan las cantidades que deberán ser elaboradas de acuerdo a lo que indiquen las instrucciones, que son las que indican al microprocesador la secuencia de operaciones a llevar a cabo sobre los datos.

Cada una de las informaciones almacenadas en memoria lleva el nombre de "palabra", debiendo entenderse por palabra a un conjunto de dígitos binarios que tomados por cualquier unidad del dispositivo, son elaborados en paralelo. La longitud, o sea la cantidad de dígitos binarios, de una palabra determina el llamado "paralelismo" de un microprocesador. Si decimos paralelismo ocho, significa que ese microprocesador es capaz de elaborar ocho dígitos binarios en paralelo, o sea que su palabra es de ocho dígitos binarios, también llamados "BITS".

La longitud de una palabra no limita la longitud de una instrucción o de un dato, por cuanto los mismos pueden descomponerse entre varias palabras.

Debemos hacer la distinción aquí de lo que es una palabra dato y qué es una palabra instrucción. Como se lo indica, una palabra dato, es una palabra de "X" bits, tantos como sea el paralelismo del microprocesador, que contiene un dato, mientras que una palabra instrucción, es una palabra que contiene una instrucción.

Una palabra instrucción siempre se divide en dos partes, una de ellas contiene un código de operación o código operativo, mientras que la otra contiene una dirección, o una referencia a ella.

El código operativo indica al microprocesador los pasos necesarios para cumplir con esa instrucción, o sea que le hará habilitar o no ciertos circuitos lógicos a fin de producir los eventos deseados.

La dirección indica la locación de memoria donde se encuentra el dato sobre el cual debe operar. No siempre en esta parte correspondiente a la dirección se encuentra directamente el número que lleva la locación de memoria donde se encuentra el dato, sino que puede llevar un número sobre el cual se deberá realizar alguna operación o comparación a fin de encontrar dicha locación. Esto se hace para facilitar la programación.

La longitud de palabra, o el paralelismo de un microprocesador depende de lo que se desee de él, pues cuanto mayor es dicha longitud, menor será el tiempo de operación, y mayor será la cantidad de

instrucciones distintas que se puedan ejecutar, así como mayor será la capacidad de memoria que es posible direccionar.

Por ejemplo, si se dispone un microprocesador con palabra de doce bits, y se deben reconocer treinta y dos instrucciones, son necesarios cinco bits para el código operativo, con lo cual quedan siete bits para direccionar la memoria, o sea que se podrán direccionar ciento veintiocho locaciones en forma directa.

Sin embargo, la cosa no es tan simple, pues debemos distinguir en primer lugar entre dos tipos de instrucciones, las que tienen referencia a memoria, y las que no tienen esta referencia. Las primeras son las que provocan directamente las operaciones o transferencias entre registros y memorias y las segundas solamente las transferencias y operaciones entre dos registros especializados, o alguna otra que realice operaciones sin intervención de la memoria.

Una vez que se fija la longitud de la parte correspondiente a la dirección de una instrucción, y en consecuencia la capacidad de memoria directamente direccionable, el número de operaciones distintas ejecutables, o sea de los códigos operativos distintos, puede resultar mucho mayor de aquel dado por el número de dígitos binarios destinados a tal fin. Además cuando la instrucción no hace referencia a memoria, todos los bits de la palabra pueden ser empleados como código operativo.

Para esto, es necesario indicar al microprocesador cuando una instrucción hace referencia a memoria y cuando no, para lo cual puede emplearse uno de los bits, por ejemplo el primero, y cuando este sea un cero diremos que la instrucción no hace referencia a memoria, y si lo hace cuando es un uno. No es esta la única forma posible, sino solamente la más sencilla.

Si continuamos con nuestro ejemplo de paralelismo de doce bits, el fabricante puede disponer las cosas para tener:

- a) el primer bit para indicar si la instrucción es o no con referencia a memoria.
- b) tres bits para indicar el código operativo de las instrucciones con referencia a memoria, con lo cual se puede tener ocho operaciones distintas con referencia a memoria.
- c) ocho bits para la dirección, lo cual permite direccionar directamente doscientas cincuenta y seis posiciones de memoria.
- d) En el caso de instrucciones sin referencia a memoria, se podrán tener once bits para codificarlas, lo que da un total de dos mil cuarenta y ocho instrucciones distintas sin referencia a memoria.

El último número indicado es verdaderamente muy grande, y prácticamente imposible de completar.

Otra forma, cuando se construyen microprocesadores de bajo número de paralelismo, por ejemplo de cuatro u ocho bits, se emplea más de una palabra para completar una instrucción.

En cuanto a la longitud de palabra, podemos decir que cuando la misma es corta, ocho o menos bits, se tendrá un microprocesador económico pero con instrucciones de poco poder, o sea que para realizar aun operaciones sencillas es necesario un gran número de instrucciones. En cambio cuando la palabra es larga, el dispositivo será más caro, pero se tendrá generalmente mayor velocidad de operación, más facilidad para ejecutar operaciones, y sus instrucciones en general serán más poderosas, o sea que permitirán ejecutar operaciones con un menor número de instrucciones, con lo cual se tendrá una mejor ocupación de la memoria con una mayor capacidad de direccionamiento.

La selección de uno u otro microprocesador dependerá del tipo de actuación que se pretenda. Para un sistema de control simple, convendrá un microprocesador de palabra corta, mientras que para la construcción minicalculadoras, convendrá un microprocesador de palabra larga.

## II.2: Unidad lógica aritmética.

La unidad lógica aritmética, es la encargada de ejecutar las operaciones aritméticas fundamentales, conjuntamente con algunas operaciones lógicas. Su complejidad y la complejidad de las operaciones que puede realizar dependen del paralelismo y la filosofía de construcción y de aplicación del microprocesador.

Las operaciones que puede llevar a cabo una unidad lógica aritmética de un microprocesador son siempre muy elementales, pues es inútil lograr gran complejidad de la misma siendo que en las aplicaciones standard de estos dispositivos no se presentan grandes requerimientos.

Por ejemplo, la unidad lógica aritmética puede realizar operaciones tales como:

- a) Suma algebraica de números binarios.
- b) Complementación de números binarios.
- c) Operación lógica AND entre dos números binarios.
- d) Operación lógica OR entre dos números binarios.
- e) Operación lógica OR EXCLUSIVO entre dos números binarios.
- f) Suma decimal entre dos números decimales codificados en binario.

En algunos microprocesadores inclusive puede llegarse hasta la

realización del producto, aunque éstos son casos especiales de dispositivos especialmente contruidos para la fabricación de microcomputadores.

### **II.3: Registros.**

Los registros son, como su nombre lo indica, lugares donde puede almacenarse una palabra para poder operar sobre ella. En todo microprocesador existe una cierta cantidad de registros, cada uno de ellos orientado a una función en particular.

#### **II.3.1: Registro de direcciones de memoria.**

Para poder acceder a las informaciones guardadas en la memoria, es necesario alcanzar una dada posición de la misma y extraerlas. Con este fin es necesario disponer un registro de direcciones de memoria, que almacene durante los breves instantes durante los cuales se efectúa la lectura de una locación de memoria, o sea la transferencia del contenido de la misma hacia otro lugar, la identificación o dirección de esa locación.

#### **II.2.2: Registro de datos.**

Como normalmente la memoria está subdividida en dos regiones, una de ellas destinada a las instrucciones, y otra a los datos, este registro de datos es el encargado de tomar a los mismos en caso de lectura, o entregarlos a la memoria en caso de escritura, o sea cuando se transfiere un dato a una cierta locación de memoria.

No siempre existe este registro, pudiendo ser reemplazado por otros registros, que a la vez cumplen otros propósitos.

#### **II.3.3: Registro de instrucciones.**

Cumple la misma función que el registro de datos, pero para las instrucciones, con un agregado, en él lee la unidad de control para poder hacer ejecutar la instrucción correspondiente.

Este registro solamente es capaz de almacenar la parte correspondiente al código operativo.

#### **II.3.4: Registro acumulador.**

Es un registro cuya función específica es la de contener un operando en la actuación de la unidad lógica aritmética, y luego de realizada la operación, contener el resultado, de aquí el nombre de acumulador, pues es el encargado de acumular los resultados.

#### **II.3.5: Registro de propósitos generales.**

En todo microprocesador existe una cierta cantidad de registros



accesibles desde las demás unidades. Generalmente forman parte de la memoria, o mejor dicho, se reservan ciertas locaciones de memoria a fin de que funcionen como registros, a los cuales se les puede asignar tareas específicas por programa. No necesariamente deben ser locaciones de la memoria central, sino que pueden formar parte de una memoria especial denominada "memoria borrador", que es la encargada de soportar resultados parciales en la ejecución de algunas operaciones.

Estos registros facilitan notablemente la tarea del microprocesador, haciéndola más rápida y segura, mientras que al mismo tiempo son una gran ayuda al programador, pues puede echar mano de ellos en caso de necesitar un almacenamiento temporario de alguna información, que se precisará para seguir la misma elaboración en ejecución.

#### **II.4: Barras ómnibus.**

El flujo de instrucciones entre las distintas partes de un microprocesador se realiza mediante líneas de interconexión entre bloques funcionales diversos. Estas líneas de interconexión son denominadas "Barras Ómnibus" o abreviadamente "BUS".

Normalmente existe más de un juego de barras ómnibus, especializadas para ciertas funciones. Algunos microprocesadores tienen más de un juego para los datos, otro juego para los resultados, otro para las instrucciones, otro para la conexión con la memoria exterior, etc.

#### **II.5: Unidad de control.**

Es la unidad encargada de realizar varias funciones, algunas en forma automática, otras por programa.

Entre las primeras, se encuentra la de dar a todas las otras unidades componentes las señales de tiempo necesarias para el funcionamiento conjunto. Entre las segundas, podemos citar:

- 1) Decodificador de instrucciones.
- 2) Circuitos de control.
- 3) Memoria de microprogramas.
- 4) Contador de programa.

##### **II.5.1: Decodificador de instrucciones.**

Este decodificador, traduce las instrucciones almacenadas en el registro de instrucciones, que a veces suele considerarse incorporado a la unidad de control, en forma tal de entregar una serie de señales a los circuitos de control.

### II.5.2. Circuitos de control.

Las señales de salida del decodificador son alimentadas juntamente con las señales del reloj central a los circuitos de control, les que derivan dichas señales a todos los componentes del microprocesador en la forma y tiempo adecuados para cumplir con i a instrucción contenida en ese momento en el registro de instrucciones.

Estos circuitos son generalmente una serie de compuertas y multivibradores para retardar señales.

### II.5.3: Memoria de microprogramas.

Esta es una pequeña unidad de memoria, generalmente del tipo de lectura exclusiva, que almacena una pequeña cantidad de micro-instrucciones, o sea instrucciones de reducido alcance necesarias para realizar alguna función especial. Por ejemplo puede contener una serie de instrucciones para realizar la suma algebraica de dos números decimales codificados en binario.

Asimismo, puede contener pequeños programas supervisores del funcionamiento del dispositivo.

La presencia de esta memoria de microprogramas, hace que sea más simple programar el microprocesador, pues permite emplear instrucciones más poderosas.

En caso de cambiar la memoria de microprogramación, se cambia el funcionamiento del microprocesador, obteniendo así posiblemente una actualización de sus funciones.

### II.5.4: Contador de programa.

Este es un registro especializado que contiene en cada instante la dirección de memoria de la próxima instrucción que deberá ejecutarse.

Cada vez que se ejecuta una instrucción, el contador de programa aumenta su contenido en una o más unidades de acuerdo a la cantidad de palabras de una instrucción, y en caso de salto condicionado o no, su contenido es modificado mediante sustracción o adición haciéndolo pasar por la unidad lógica aritmética.

Normalmente el contador de programa es auxiliado por otros contadores o registros auxiliares, denominados registros apuntadores, que se encargan de guardar la última dirección antes de ejecutar un salto, o la dirección a la cual se debe ir en un salto incondicionado o condicionado.

### **III: Otros componentes auxiliares.**

Estos son elementos que no hacen al funcionamiento básico de un microprocesador, pero que ayudan al mismo a ejecutar sus funciones en forma más segura y práctica, tanto desde el punto de vista del "HARDWARE" como del "SOFTWARE".

Entre estos elementos podemos citar a los más convencionales, que son:

- 1) Multiplexores de entrada y salida.
- 2) Registro de pila.
- 3) Registro indicador de estados.

#### **III. 1: Multiplexores de entrada y salida.**

Son dispositivos que permiten la conexión del microprocesador a distintos periféricos, lo cual puede realizarse por programación, aunque en algunos casos es posible mediante una llave externa realizar también el cambio o modificación del lugar al cual se debe acceder desde o hacia el microprocesador..

Son circuitos de pasaje de muchas entradas o salidas a una sola o entrada. Inclusive a veces pueden ser circuitos de adaptación de niveles de tensión o de velocidades.

#### **III .2: Registro de pila.**

Es un registro especial, capaz de almacenar varias palabras, y que se accede a él con sólo citarlo mediante su código.

Las palabras pueden ser almacenadas de varias formas, de acuerdo al tipo de pila que se use. En general hay dos variantes principales que son conocidas como:

- 1) Primero que entra primero que sale.
- 2) Ultimo que entra primero que sale.

##### **III.2.1: Pila tipo primera entrada primera salida.**

En este caso el acceso de la información se realiza por el primer registro de la pila y la salida por el último, en forma tal que la primera información que se almacena, será la primera en aparecer por el otro extremo. Para poder ser leído, es necesario primero llenarlo en su totalidad, o dar tantos desplazamientos de la información como sea necesario para obtener la salida de la misma.

##### **III.2.2: Pila tipo última entrada primera salida.**

Ahora, la última información entrada al registro es la primera a la cual puede accederse citando la lectura del mismo.

Este es más conveniente que el anterior, pues puede guardarse cualquier número de informaciones, de acuerdo a la longitud o altura de la pila, y siempre la última estará inmediatamente accesible.

La pila es una forma muy útil de guardar información de utilización inmediata en el cálculo, pues es de rápido y sencillo acceso por parte del microprocesador, y es de fácil programación.

### **III.3: Registro de estados.**

Este es un registro muy especial y muy útil, pues permite dar avisos de la marcha de un programa, o hacer que desde el exterior se tomen decisiones sobre la marcha de una elaboración en base al punto alcanzado por la misma. Es posible mediante programa modificar uno o más bits del registro de estados, de acuerdo a los caminos que ha tomado la elaboración, también por programa es posible hacer una prueba de esos bits y de acuerdo a su valor modificar la ejecución del programa, o hacer detener la ejecución en curso para indicar en forma externa el curso de la acción a seguir.

## **IV: Direccionamiento y métodos de direccionamiento.**

Se denomina direccionamiento a la forma en la cual puede indicarse una cierta locación de memoria.

Según decíamos antes, una parte de una instrucción es la dirección de memoria sobre la cual se puede acceder, o dar una clave para alcanzar una cierta locación.

Existen dos métodos básicos para lograr el direccionamiento, que son:

- 1) Direccionamiento directo.
- 2) Direccionamiento indirecto.

### **IV. 1: Direccionamiento directo.**

En este caso, la parte correspondiente de la instrucción da directamente la dirección donde se debe buscar el operando. Según vimos este método permite acceder a muy pocas locaciones de memoria, por cuanto el número de bits empleados en la parte dirección es limitado.

### **IV.2: Direccionamiento indirecto.**

Ahora, la parte correspondiente a la dirección de una instrucción no contiene el rótulo o número de una locación de memoria,

sino que se da la clave para encontrarla. Podemos encontrar tres tipos distintos de direccionamiento indirecto, tales como:

- 1) Básico.
- 2) Relativo.
- 3) Indexado.

#### **IV. 2 .1: Direccionamiento indirecto básico.**

La parte dirección de la instrucción contiene el número de una locación de memoria donde se encuentra alojada la dirección del dato buscado. Así si tenemos un paralelismo de ocho bits, usando este sistema podremos direccionar doscientos cincuenta y seis locaciones distintas.

#### **IV.2.2: Direccionamiento indirecto relativo.**

En este caso, el contenido de la parte dirección de una instrucción es sumado al contenido del contador de programa, o a otro registro especializado para obtener la dirección de la locación buscada. Esto tiene como ventaja de que cuando se almacenan los datos en forma secuencial al final del programa, el contenido de la instrucción en la parte dirección, para realizar la búsqueda en memoria será siempre el mismo, pues lo que varía es el contador de programa, y si se debe realizar la lectura de una gran cantidad de datos, no es necesario repetir la instrucción tantas veces como lecturas deban hacerse, sino que alcanza con una sola.

#### **IV.2.3: Direccionamiento indirecto indexado.**

Se tiene un direccionamiento indexado cuando el contenido de la parte dirección de la instrucción es sumada algebraicamente al contenido de un registro apuntador, o registro índice.

#### **V.: Operación de un microprocesador.**

Cualquier computadora, lo mismo que un microprocesador, ejecuta una instrucción por lo menos en dos pasos o fases bien distintas:

- 1) Fase de búsqueda.
- 2) Fase de ejecución.

Durante la primera fase, la de búsqueda, se realiza la localización y transferencia de la instrucción y sus partes componentes a sus lugares de destino, o sea el mando o código operativo al registro de instrucciones en contacto con la unidad de control, y la parte dirección al registro de direcciones de la memoria, así como también se producen todas las operaciones accesorias a estas transferencias.

Durante el transcurso de la segunda fase se ejecuta la instrucción transferida, también con todas las operaciones accesorias a dicha ejecución.

La ejecución de una instrucción puede llevar un tiempo de reloj o varios tiempos, dependiendo ello de la complejidad de operaciones necesarias para satisfacerla.

Para ilustrar lo dicho supongamos un sencillo ejemplo, en el contador de programa tenemos un cierto número que corresponde a la dirección de una instrucción. En el registro de direcciones de memoria, se tiene este mismo número, que es el correspondiente a la locación a la cual debe accederse para extraer la información.

Ante la actuación de un pulso de reloj, el contenido de la locación citada pasa al registro de datos, de donde ante otro pulso se derivará la parte dirección nuevamente al registro de direcciones de memoria y la parte mando o código operativo es llevada al registro de instrucciones donde se procederá a su decodificación.

Al concluir esta serie de operaciones, el contador de programa es aumentado en una unidad.

Este conjunto de operaciones constituyen la fase de búsqueda de una instrucción, y es iniciado con un mensaje interno del microprocesador especial para ello. Esta señal interna se denomina "Señal de Búsqueda".

La fase de búsqueda es siempre igual sea cual sea la instrucción, variando solamente entre distintos tipos de máquina o de microprocesadores.

La fase de ejecución en cambio dependerá del tipo de mando a ejecutar. Siguiendo con el ejemplo, supongamos que el mando indique realizar una suma. Uno de los sumandos ya se encuentra, o por lo menos lo suponemos así, en el acumulador, mientras que el otro está alojado en la dirección que indica ahora el registro de direcciones de memoria.

Ante la actuación de un nuevo pulso de reloj, el dato es transferido de la locación de memoria conectada a la barra ómnibus por el selector de direcciones a la unidad lógica aritmética. Lo mismo ocurrirá a través de otro juego de barras ómnibus con el contenido del acumulador.

El resultado es vuelto al mismo tiempo al acumulador, ante la acción de un nuevo pulso de reloj. Como ya se ha ejecutado la instrucción, el control pasará nuevamente a la condición de Búsqueda para reiniciar el proceso,