

## TÉCNICAS DIGITALES II

### UNIDAD N°2

#### LENGUAJE DE MÁQUINA

Es el lenguaje que interpreta el microprocesador, el cual se encuentra conformados por ceros (0) y unos (1), ya que, como lo venimos viendo desde la asignatura Técnicas Digitales I, los circuitos digitales síncronos como asíncronos, lo único que interpretan son niveles de tensión alto o bajo. Los microprocesadores están preparados para poder ejecutar un programa el cual se construye a base de **Instrucciones**. Cada una de las instrucciones van a estar conformada por una cierta cantidad de bytes, el primero de ellos lleva el código de operación de la instrucción a ejecutar, en los bytes restante si son necesarios se encuentran los datos (operando) o la dirección adonde ir a buscar o a guardar los mismos. El 8085 posee 239 instrucciones.

Elegimos como ejemplo una instrucción, a saber: Instrucción **ANI dato**. Esta instrucción hace la operación lógica AND entre el dato almacenado en el acumulador y el contenido del segundo byte de la instrucción. Recordar cómo se mencionó en la Unidad N°1 que el resultado se guarda en el acumulador. En la figura a continuación se representan los dos bytes de la instrucción, el primero que especifica el **Código de Operación** y en el segundo byte se encuentra el otro dato que se va a utilizar la unidad aritmética lógica para ejecutar la operación AND.

Código de Operación	1	1	1	0	0	1	1	0
BYTE 2	DATO							

Muchas veces, para escribir un programa hacemos uso de palabras abreviadas o especiales, que representan las instrucciones que utiliza el microprocesador. A estas palabras se las conoce con el nombre de **Nemotécnico**. En el ejemplo anterior el nemotécnico de la instrucción AND entre el registro acumulador y el dato que se guarda en el segundo byte es: **ANI dato**. Recordando que en la primera unidad se desarrollaron los cuatro modos de direccionamiento en este caso podemos decir que la instrucción que venimos utilizando posee un modo de direccionamiento inmediato, es decir que la instrucción tiene el dato.

#### Set de instrucciones

El set de instrucciones del microprocesador 8085, como lo dijimos anteriormente posee 239, a las cuales las podemos clasificar como:



**Instrucciones Aritméticas:** Estas son las que ejecuta el microprocesador haciendo uso de la Unidad Aritmética Lógica. Estas operaciones pueden ser: sumar, restar, incrementar o decrementar datos; los operando pueden estar almacenados en registros los registros internos del 8085 o en alguna posición de la memoria. Estas instrucciones pueden generar una modificación de las banderas que se encuentran almacenadas en el Registro de estados. A continuación utilizaremos como ejemplo la instrucción **SUB r, (Resta Registros)**. En este caso se realiza la operación de la resta, entre el registro acumulador y el contenido que se encuentra almacenado en el registro r. En este caso el modo de direccionamiento por registro. El microprocesador realiza la resta a partir del formato de números, donde representan los mismos por medio del complemento A2 y que fue objeto de estudio en la Asignatura Técnicas Digitales I.

El nombre nemotécnico es SUB r, su simbología de funcionamiento es:  $(A) \leftarrow (A) - (r)$ , lo que dice que hace la resta del registro acumulador menos el registro r y el resultado se guarda en el acumulador. Las banderas que puede modificar son: Z, S, P, CY y AC.

**Instrucciones Lógicas:** al igual que las anteriores estas instrucciones se ejecutan en la Unidad Aritmética Lógica del microprocesador 8085, algunas de las operaciones que realiza son: AND, OR, XOR, etc. Como ejemplo veremos la Instrucción **ORA M**, esta instrucción hace la operación lógica OR entre el valor de que se encuentra almacenado en el registro acumulador y el contenido ubicado en la dirección de la memoria especificada por los registros H y L. En este caso el modo de direccionamiento indirecto por registros.

El nombre nemotécnico de la Instrucción es, ORA M, su simbología de funcionamiento es:  $(A) \leftarrow (A) \vee ((H)(L))$  que dice que la instrucción va a hacer la operación OR entre el Acumulador y el contenido que se encuentra ubicado en la dirección de memoria que especifican los registros H y L y guarda el resultado en el acumulador. Las banderas que puede modificar son: Z, S, P, CY y AC.

**Instrucciones de Transferencia de Datos:** son el conjunto de instrucciones que se utilizan para transferir de datos, entre registros o entre posiciones de memoria y registros y viceversa. En este caso vemos como ejemplo la



instrucción, **MOV r, M**. Dicha instrucción hace que el contenido de la posición de memoria cuya dirección está indicada a través de los registros H y L es transferido al registro r. En este caso el modo de direccionamiento indirecto por registros.

El nombre nemotécnico de la Instrucción es, MOV r, M; su simbología de funcionamiento es:  $(r) \leftarrow ((H)(L))$  que dice que transfiere al registro r el contenido de la memoria que se encuentra en la dirección especificada por los registros H y L.

**Instrucciones de bifurcación:** Estas instrucciones se dan a partir de dos tipos de saltos, a saber: incondicionales o condicionales. En el primero de los casos los mismos realizan su operación sin necesidad de ninguna comparación; para el caso de los saltos condicionales, dependiendo de la instrucción que ejecutan tienen que verificar una de las banderas de estado, para determinar si salta o no. A continuación se muestra a modo de ejemplo la instrucción, **JMP addr**. Esta instrucción es de salto incondicional y su funcionamiento es el siguiente: el contador de programa del microprocesador se carga con los byte 2 y 3 que se encuentran a continuación del código de operación de la instrucción (addr), en este caso deja de seguir ejecutando el programa en la secuencia que lo venía haciendo, para pasar a la a ejecutar la instrucción que se encuentra ubicada en la nueva dirección que se ha cargado en el registro del contador de programa. Modo de direccionamiento Inmediato.

El nombre nemotécnico de la Instrucción es, JMP addr; su simbología de funcionamiento es:  $(PC) \leftarrow ((\text{Byte } 3)(\text{byte } 2))$  que el contador de programa se carga con los Bytes 2 y 3 de la instrucción.

**Instrucciones de llamado y vuelta de subrutinas:** Estas instrucciones se dan a partir de dos tipos de llamados a subrutinas: incondicionales o condicionales. Su ejecución es semejante al utilizado por las instrucciones de saltos. A continuación se muestra como ejemplo la instrucción **CALL addr**. Esta instrucción es su funcionamiento parecido al salto condicional, la gran diferencia, es que, previo a que se genere al salto a la subrutina se guarda la dirección del contador de programa en la pila, para que cuando se termine de ejecutar la subrutina, se pueda retornar a la ejecución del programa. Posteriormente se carga en el contador de programa con los bytes que



Facultad Regional

**UTN** VILLA MARIA

acompañan a la instrucción (addr), dichos bytes contienen la dirección de donde se encuentra la subrutina a la cual el programa va a saltar.

Simbología de funcionamiento:

$((SP) - 1) \leftarrow (PCH)$

$((SP) - 2) \leftarrow (PCL)$

$(SP) \leftarrow (SP) - 2$

$(PC) \leftarrow (\text{byte } 3)(\text{byte } 2)$

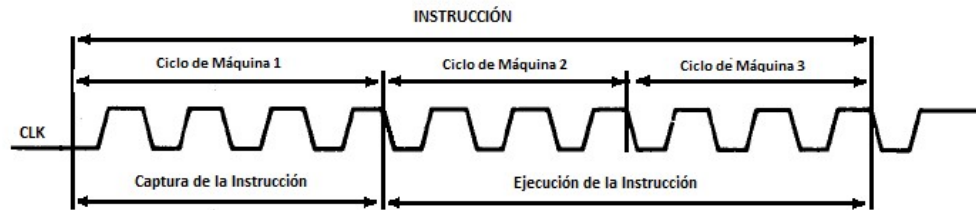
**Instrucciones de comunicación:** Este tipo de instrucciones permiten la comunicación con otros procesadores.

### Ciclos de máquina

En este punto en particular es importante mencionar que cada instrucción está compuesta por uno o más ciclos de máquinas, dichos ciclos tienen una duración entre 3 y 6 periodos de reloj. En la tabla que se muestra a continuación, se mencionan los diferentes ciclos de máquinas del 8085, con las señales de estados y control que se son necesarias para la ejecución de cada uno de ellos.

Ciclos de Máquinas	Señales de Estados			Señales de Control		
	IO/#M	S1	S0	#RD	#WR	#INTA
Buscar Código (opcode)	0	1	1	0	1	1
Lectura de Memoria	0	1	0	0	1	1
Escritura de Memoria	0	0	1	1	0	1
Lectura Puerto E/S	1	1	0	0	1	1
Escritura Puerto E/S	1	0	1	1	0	1
Reconocimiento de Interrupción	1	1	1	1	1	0
ACK de RST	1	1	1	1	1	0

En la figura que se muestra a continuación se puede observar, como se van procesando los distintos ciclos de máquinas para dar cumplimiento a una instrucción. Por otra parte en la figura queda marcado como una instrucción se divide en dos partes, la primera de ellas la búsqueda del código de la instrucción (opcode) y la segunda de ellas la ejecución de la instrucción.



La etapa de búsqueda (captura) de una instrucción que por otro lado también corresponde al primer ciclo de máquina, debe realizarse en todas las instrucciones, este ciclo de máquina tiene una duración de 4 periodos de reloj. Por otro lado para la etapa de la ejecución de una instrucción dependiendo de la misma se ejecutan los otros ciclos de máquina.

A continuación se muestra un diagrama de tiempo provisto por el fabricante en su hoja de datos, sobre las señales que se activan en una instrucción, en este caso la misma, tiene tres ciclos de máquina uno de Búsqueda de la Instrucción, otro de Lectura de un dato en memoria y por último la escritura del dato en un Puerto de E/S. Se recomienda buscar la Hoja de Dato del Fabricante para obtener más Información.

