

## LÓGICA Y CONTROL DE PROGRAMAS

(INSTRUCCIONES DE CONTROL Y ETIQUETAS)

### Introducción

En función de que los programas o problemas a resolver usualmente no son lineales, dentro del programa se requiere realizar transferencias del control a rutinas o procedimientos que no están inmediatamente después de la que se está ejecutando en un momento dado.

Esto implica que el control se deba transferir a instrucciones que no se han ejecutado o a repetir la ejecución de otras instrucciones ejecutadas anteriormente. Un elemento importante en este proceso o representa el uso de etiquetas para identificar rutinas o secciones de operaciones en el código a las cuales se deba hacer referencia.

A continuación se listan las categorías en las que se agrupan las instrucciones que permiten realizar transferencias del control del programa:

INSTRUCCIONES DE COMPARACIÓN	INSTRUCCIONES DE TRANSFERENCIA
<b>CMP</b> , comparación	<b>JMP</b> , salto no condicional
<b>TEST</b> , testeo de bits	<b>Jnn</b> , salto condicional
<b>Jnn</b> , salto condicional	<b>CALL</b> , llamada a procedimientos
<b>LOOP</b> , ciclos	

### 1. INSTRUCCIÓN CMP

Es usada para comparar dos campos de datos que pueden estar contenidos en un registro.

**SINTAXIS:** CMP operando1, operando1 ;el operando puede ser una variable o una etiqueta

Ejemplo:

```
CMP  BX, 00          ;compara BX con cero
JZ   otro            ;si es cero salta a "otro"
```

### 2. INSTRUCCIÓN TEST

Es usada para comparar dos campos de datos bit por bit (de igual manera a como se compara con AND), pero no cambia los bits en los operandos, sino solamente establece valores en el registro de banderas.

**SINTAXIS:** TEST operando1, operando1 ;el operando2 usualmente es un valor definido (máscara)

Ejemplo:

```
TEST  BL, 1111 0000B ;¿alguno de los bits de más a la izquierda en BL
                        no es cero?
JZ    otro            ;si es cero salta a "otro"
```

### 3. INSTRUCCIÓN DE SALTO JMP (NO CONDICIONAL)

Este tipo de salto se realiza sin importar las condiciones que se han dado o los resultados de las operaciones anteriores. Se utiliza para establecer bifurcaciones en la ejecución del programa.

**SINTAXIS:** JMP destino el destino refiere a una etiqueta

Ejemplo:

```
JMP   otro           ; salta si verificar alguna condición
```

#### 4. INSTRUCCIONES DE SALTO CONDICIONAL

Los lenguajes ensambladores permiten una variedad de instrucciones de salto condicional que transfieren el control dependiendo de las configuraciones en el registro de banderas.

SINTAXIS GENERAL:

<etiqueta> Jnnn destino ; "nnn" refiere a los tipos de salto condicional, el destino  
; refiere a una etiqueta

Ejemplo:

CMP BX, 00 ; compara BX con cero  
JZ otro ; si es cero salta a "otro"

También es importante tomar en cuenta si se está trabajando con datos sin signos (sin negativos) o con datos con signos (positivos y negativos) en cuyo caso el bit más a la izquierda indica el signo del número, en donde 0 es positivo y 1 es negativo.

Por ejemplo, si se tiene AX = 11000110 y BX = 00010110, la instrucción:

##### **CMP AX,BX**

compara el contenido de ambos registros y con los siguientes posibles resultados:

- Para datos sin signo, el valor de AX es mayor
- Para datos con signo, el valor de BX es mayor a causa del signo negativo

##### **SALTOS CON BASE EN DATOS SIN SIGNO**

INSTRUCCION	DESCRIPCIÓN	BANDERA EXAMINADA
JE o JZ	Salta si es igual o salta si resultado es cero	ZF
JNE o JNZ	Salta si no es igual o salta si resultado no es cero	ZF
JA o JNBE	Salto si es mayor o salta si no es menor o igual	CF, ZF
JAE o JNB	Salto si es mayor o igual o salta si no es menor	CF
JB o JNAE	Salto si es menor o salta si no es mayor o igual	CF
JBE o JNA	Salto si es menor o igual o salta si no es mayor	CF, AF

##### **SALTOS CON BASE EN DATOS CON SIGNO**

INSTRUCCION	DESCRIPCIÓN	BANDERA EXAMINADA
JE o JZ	Salta si es igual o salta si resultado es cero	ZF
JNE o JNZ	Salta si no es igual o salta si resultado no es cero	ZF
JG o JNLE	Salto si es mayor o salta si no es menor o igual	ZF, SF, OF
JGE o JNL	Salto si es mayor o igual o salta si no es menor	SF, OF
JL o JNGE	Salto si es menor o salta si no es mayor o igual	SF, OF
JLE o JNG	Salto si es menor o igual o salta si no es mayor	ZF, SF, OF

**SALTOS CON PRUEBAS ARÍTMÉTICAS ESPECIALES:** para usos especiales.

INSTRUCCION	DESCRIPCIÓN	BANDERA EXAMINADA
JS	Salta si signo es negativo (s=1)	SF
JNS	Salta si signo es positivo (s=0)	SF
JC	Salta si hay acarreo (c=1) (igual que JB)	CF
JNC	Salta si no hay acarreo (c=0)	CF
JO	Salta si hay desbordamiento (O=1) (overflow)	OF
JNO	Salta si no hay desbordamiento (O=0)	OF
JP / JPE	Salta si hay paridad de bits o salta si la paridad es par (P=1)	PF
JNP / JPO	Salta si no hay paridad de bits o salta si la paridad es impar (P=0)	PF

**INSTRUCCIÓN LOOP**

Esta instrucción se utiliza para realizar ciclos en cantidades de veces definidos o hasta que se cumplan condiciones. Para ello, requiere un valor inicial en CX. En cada iteración, el valor en CX se disminuye automáticamente en 1. Si el valor en CX es cero, el control pasa a la siguiente instrucción; si no, entonces pasa a la dirección del operando destino.

**SINTAXIS:** <etiqueta> LOOP destino ; el destino refiere a una etiqueta

Ejemplo:

```

                MOV    CX, 5          ; el ciclo se hará 5 veces
ciclo:         ADD     AX, 1          ; suma 1 al registro AX
                LOOP   ciclo          ; salta a la etiqueta "ciclo" y disminuye CX en 1
                                         ; hasta que CX = 0

```

**INSTRUCCIONES CALL Y RET**

La instrucción CALL transfiere el control a un procedimiento llamado y la instrucción RET regresa del procedimiento llamado al que hizo la llamada. RET debe ser la última instrucción en un procedimiento llamado.

Para efectos de orden en el código, es conveniente incluir el programa principal en un procedimiento, aunque no es obligatorio.

**SINTAXIS:** <etiqueta> CALL procedimiento ; el destino refiere a una etiqueta

Ejemplo:

```
CALL calculo ; llamada al procedimiento "cálculo"
```

**SINTAXIS DE PROCEDIMIENTOS:**

```

procedimiento PROC <NEAR/FAR>      ; se indica el salto para llamar al
                                     ; procedimiento, en el mismo u otro
                                     ; segmento
    Código
    ....
    RET                             ; debe haber un retorno del procedimiento
procedimiento ENDP                  ; fin del procedimiento

```