

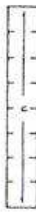
## ADD A, n

Suma el acumulador con el dato inmediato n.

**Función:**

$$A \leftarrow A + n$$

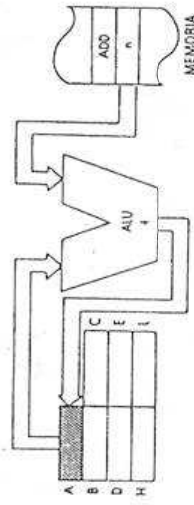
**Formato:**



**Descripción:**

El contenido del acumulador se suma al de la posición de memoria inmediata al código de operación. El resultado se almacena en el acumulador.

**Flujo de datos:**



**Tiempo:**

2 ciclos M; 7 estados T: 3.5  $\mu$ seg @ 2 MHz.

**Direccionamiento:**

Inmediato.

**Banderas**



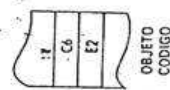
**Ejemplo:**

ADD A, E2

Antes:



Después:



## ADD A, r

Suma el acumulador con el registro r.

**Función:**

$$A \leftarrow A + r$$

**Formato:**

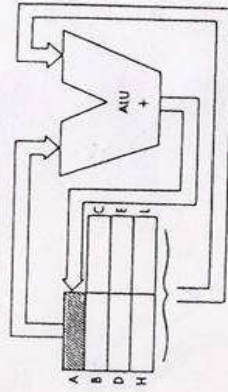


**Descripción:**

El contenido del acumulador se suma al del registro especificado. El resultado se coloca en el acumulador; r puede ser:

A - 111  
B - 000  
C - 001  
D - 010  
E - 011  
H - 100  
L - 101

**Flujo de datos:**



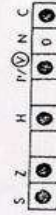
**Tiempo:**

1 ciclo M; 4 estados T; 2  $\mu$ seg @ 2 MHz.

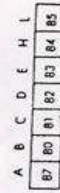
**Direccionamiento:**

Implicito.

**Códigos byte:**



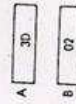
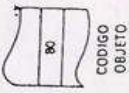
**Banderas:**



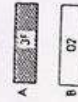
**Ejemplo:**

ADD A, B

Antes:



Después:



## AND s

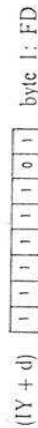
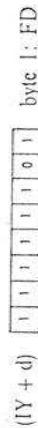
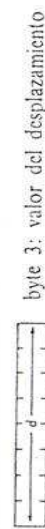
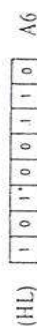
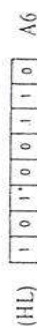
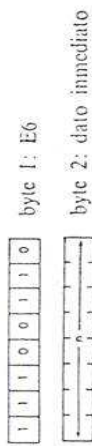
"Y" lógica del acumulador y el operando s.

**Función:**

$A \leftarrow A \wedge s$

**Formato:**

s puede ser: r, n, (HL), (IX + d), (IY + d)



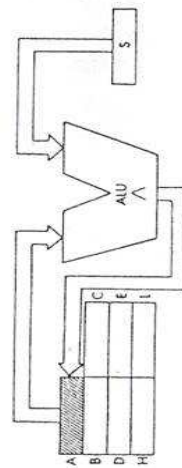
r puede ser:

A - 111      E - 011  
B - 000      H - 100  
C - 001      L - 101  
D - 010

**Descripción:**

El acumulador y el operando especificado se someten a la operación lógica "Y" (AND), y el resultado se almacena en el acumulador; s se define en la descripción de instrucciones ADD similares.

**Flujo de datos:**



Tiempo:

| S        | Ciclos M | Estados T | $\mu\text{seg}$ @<br>2 MHz |
|----------|----------|-----------|----------------------------|
| r        | 1        | 4         | 2                          |
| n        | 2        | 7         | 3.5                        |
| (HL)     | 2        | 7         | 3.5                        |
| (IX + d) | 5        | 19        | 9.5                        |
| (IY + d) | 5        | 19        | 9.5                        |

Direccionamiento:

r: implícito; n: inmediato; (HL): indirecto; (IX + d), (IY + d): indexado.

Códigos byte:

AND r

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 |
|----|----|----|----|----|----|----|----|

Banderas:

S Z H N C

|   |   |   |   |   |   |
|---|---|---|---|---|---|
| ● | ● | 1 | ● | 0 | 0 |
|---|---|---|---|---|---|

Ejemplo:

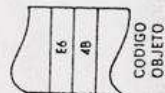
AND 4B

Antes:

A 36

Después:

A 03



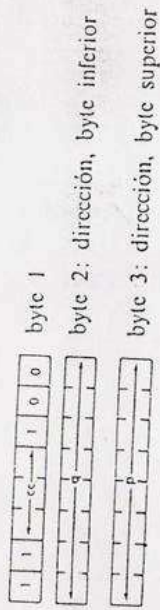
## CALL cc, pq

Llamada condicional a subrutina.

### Función:

Si cc cierto:  $(SP - 1) \leftarrow PC_{sup}; (SP - 2) \leftarrow PC_{inf}; SP \leftarrow SP - 2;$   
 $PC \leftarrow pq$   
 Si cc falso:  $PC \leftarrow PC + 3$

### Formato:



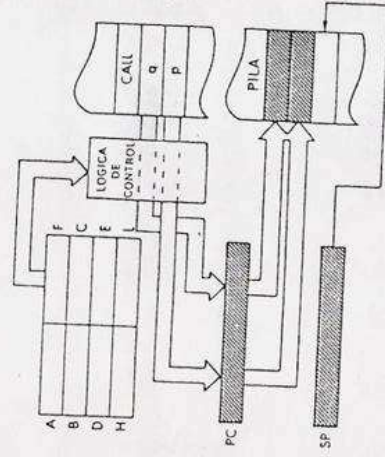
### Descripción:

Si se satisface la condición, el contenido del contador del programa se empuja en la pila tal como se describe en las instrucciones PUSH. A continuación, el contenido de la posición de memoria inmediatamente siguiente al código de operación se carga en la parte inferior del PC y el contenido de la posición de memoria siguiente se carga en la mitad superior del PC. La siguiente instrucción se tomará de esta nueva dirección. Si la condición no se satisface, se ignora la dirección pq y se ejecuta la instrucción siguiente; cc puede ser:

NZ - 000 PO - 100  
 Z - 001 PE - 101  
 NC - 010 P - 110  
 C - 011 M - 111

Al final de la subrutina llamada puede usarse una instrucción RET para restablecer el PC.

### Flujo de datos:



Tiempo:

|                     | Ciclos <i>M</i> | Estados <i>T</i> | $\mu\text{seg}$<br>(@ 2 MHz) |
|---------------------|-----------------|------------------|------------------------------|
| condición<br>cierta | 5               | 17               | 8.5                          |
| condición<br>falsa  | 3               | 10               | 5                            |

Direccionamiento:

Inmediato

Códigos byte:

CC: NZ, Z, NC, C, PO, PE, P, M

|    |    |    |    |    |    |    |    |     |
|----|----|----|----|----|----|----|----|-----|
| C4 | CC | D4 | DC | E4 | EC | F4 | FC | -qp |
|----|----|----|----|----|----|----|----|-----|

Banderas:

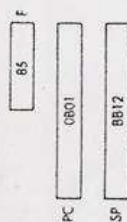
|   |   |   |     |   |   |  |
|---|---|---|-----|---|---|--|
| S | Z | H | P/V | N | C |  |
|---|---|---|-----|---|---|--|

(efecto nulo)

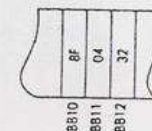
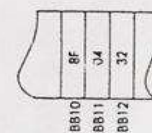
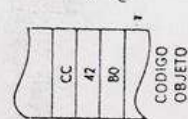
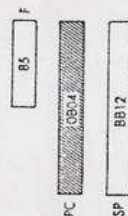
Ejemplo:

CALL Z, B042

Antes:



Después:





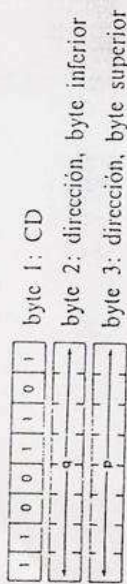
## CALL pq

Llamada a subrutina a la posición pq.

**Función:**

$(SP - 1) \leftarrow PC_{sup}; (SP - 2) \leftarrow PC_{inf}; SP \leftarrow SP - 2; PC \leftarrow pq$

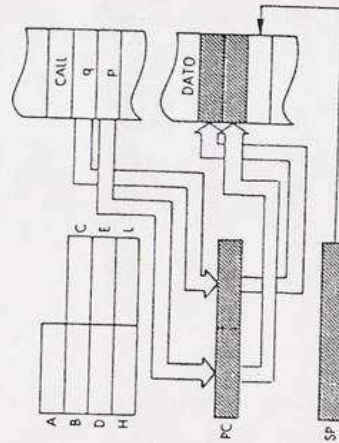
**Formato:**



**Descripción:**

El contenido del contador del programa se empuja en la pila tal como se describe en las instrucciones PUSH. A continuación se carga el contenido de la posición de memoria siguiente al código de operación en la mitad inferior del PC y el de la posición siguiente en la mitad superior del PC. La instrucción siguiente se trará de esta nueva dirección.

**Flujo de datos:**



**Tiempo:**

5 ciclos M; 17 estados T; 8.5  $\mu$ seg @ 2 MHz.

**Direccionamiento:**

Instantáneo.

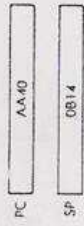
**Banderas:**



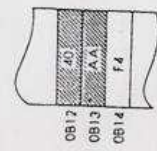
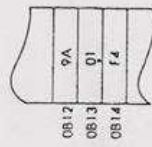
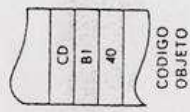
Ejemplo:

CALL 40B1

Antes:



Después:





## CPL

Complementar el acumulador.

**Función:**

$$A \leftarrow \bar{A}$$

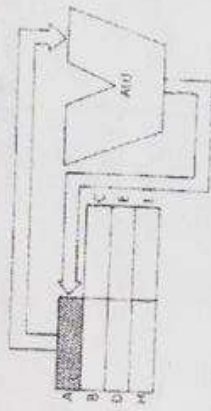
**Formato:**



**Descripción:**

El contenido del acumulador se complementa (se invierte) y el resultado vuelve a almacenarse en el acumulador (complemento a 1).

**Flujo de datos:**



**Tiempo:**

1 ciclo M; 4 estados T; 2  $\mu$ seg @ 2 MHz.

**Direccionamiento:**

Implícito.

**Banderas:**



**Ejemplo:**



Antes:



Después:



## DEC m

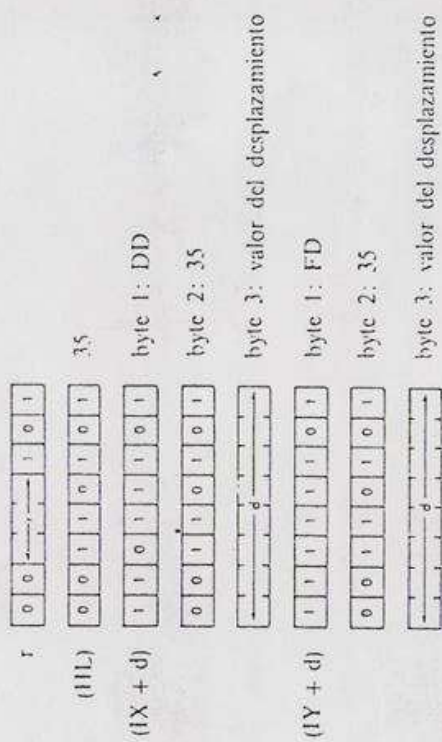
Decrementa el operando m.

**Función:**

$m \leftarrow m - 1$

**Formato:**

m puede ser r, (HL), (IX + d), (IY + d).



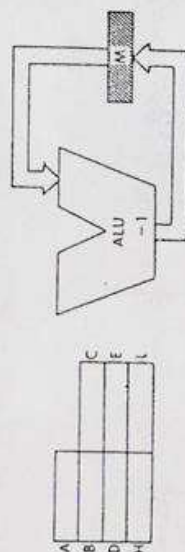
r puede ser:

- A - 111      E - 011
- B - 000      H - 100
- C - 001      L - 101
- D - 010

**Descripción:**

El contenido de la posición direccionada por el operando especificado se decrementa y vuelve a almacenarse en esa posición; m se define en la descripción de instrucciones INC similares.

**Flujo de datos:**



Tiempo:

| $m$      | Ciclos $M$ | Estados $T$ | $\mu\text{seg}$<br>@ 2 MHz |
|----------|------------|-------------|----------------------------|
| $r$      | 1          | 4           | 2                          |
| (HL)     | 3          | 11          | 5.5                        |
| (IX + d) | 6          | 23          | 11.5                       |
| (IY + d) | 6          | 23          | 11.5                       |

Direccionamiento:

$r$ : implícito; (HL): indirecto; (IX + d), (IY + d): indexado.

Códigos byte:

DEC  $r$

$r$ :

|    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|
| A  | B  | C  | D  | E  | H  | L  |
| 30 | 05 | 00 | 15 | 10 | 25 | 20 |

Banderas:

|   |   |   |   |   |   |
|---|---|---|---|---|---|
| S | Z | H | P | N | C |
| ● | ● | ● | ● | ● | 1 |

Ejemplo:

DEC C

Antes:

|    |   |
|----|---|
| 0F | C |
|----|---|

Después:

|    |   |
|----|---|
| 0F | C |
|----|---|



## HALT

Detiene la CPU.

**Función:**

La CPU deja de actuar.

**Formato:**

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 76

**Descripción:**

La CPU deja de funcionar y ejecuta instrucciones NOP continuamente, para proseguir con los ciclos de refresco de la memoria, hasta que recibe una interrupción o una orden de reinicio.

**Tiempo:**

1 ciclo.M; 4 estados T; 2  $\mu$ seg @ 2 MHz + el tiempo necesario para ejecutar un número indefinido de instrucciones NOP.

**Direccionamiento:**

Implicito.

**Banderas:**

|   |   |   |     |   |   |
|---|---|---|-----|---|---|
| S | Z | H | P/V | N | C |
|   |   |   |     |   |   |

 (efecto nulo)

## IN A, (N)

Carga el acumulador a partir del puerto de entrada N.

**Función:**

$A \leftarrow (N)$

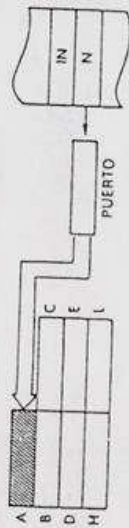
**Formato:**



**Descripción:**

Se lee el dispositivo periférico N, y el resultado se carga en el acumulador.  
El literal N se sitúa en las líneas A0 a A7 del *bus* de direcciones.  
A proporciona los bits A8 a A15.

**Flujo de datos:**



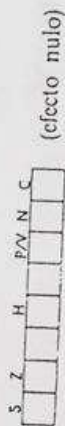
**Tiempo:**

3 ciclos M; 11 estados T; 5.5  $\mu$ seg @ 2 MHz.

**Direccionamiento:**

Externo.

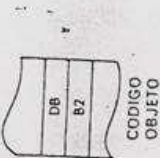
**Banderas:**



**Ejemplo:**

IN A, (B2)

Antes:



Después:



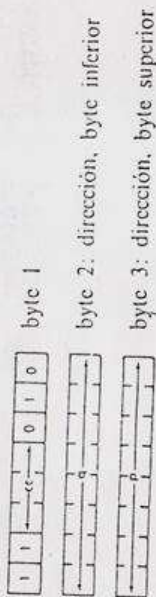
## JP cc, pq

Salto condicional a la posición pq.

**Función:**

Si cc es cierto: PC ← pq

**Formato:**

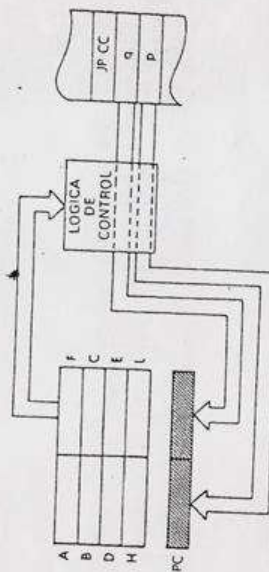


**Descripción:**

Si se cumple la condición especificada, la dirección de dos bytes inmediatamente siguiente al código de operación se cargará en el contador del programa; el código de operación se cargará en la parte inferior del PC. Si la condición no se satisface, se ignora la dirección; cc puede ser:

|    |   |     |               |
|----|---|-----|---------------|
| NZ | - | 000 | no cero       |
| Z  | - | 001 | cero          |
| NC | - | 010 | sin acarreo   |
| C  | - | 011 | acarreo       |
| PO | - | 100 | paridad impar |
| PE | - | 101 | paridad par   |
| P  | - | 110 | más           |
| M  | - | 111 | menos         |

**Flujo de datos:**



**Tiempo:**

3 ciclos M; 10 estados T; 5 μseg @ 2MHz.

**Direccionamiento:**

Inmediato.

**Códigos byte:**

cc:

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| NZ | Z  | NC | C  | PO | PE | P  | M  |
| C7 | CA | D7 | DA | E7 | EA | F7 | FA |



Banderas:

| S | Z | H | P/V | N | C |
|---|---|---|-----|---|---|
|   |   |   |     |   |   |

(efecto nulo)

Ejemplo:

JP C, 3B24

|    |
|----|
| DA |
| 24 |
| 3B |

CODIGO  
OBJETO

Antes:

|      |
|------|
| SI   |
| 0032 |

PC

Después:

|      |
|------|
| SI   |
| 3B24 |

PC

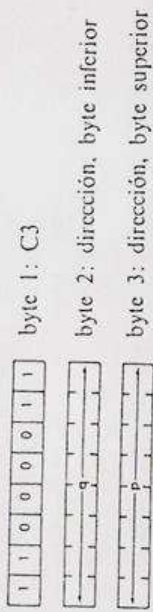
## JP pq

Salto a la posición pq.

**Función:**

PC ← pq

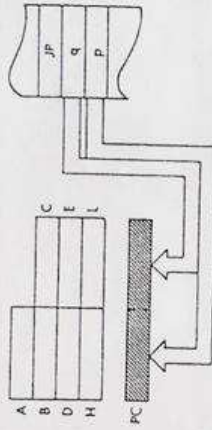
**Formato:**



**Descripción:**

El contenido de la posición de memoria inmediatamente siguiente al código de operación se carga en la mitad inferior del contador del programa y el de la posición siguiente a la anterior en la mitad superior del mismo contador. La siguiente instrucción se toma de esta nueva dirección.

**Flujo de datos:**



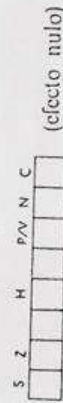
**Tiempo:**

3 ciclos M; 10 estados T; 5  $\mu$ seg @ 2 MHz.

**Direccionamiento:**

Inmediato.

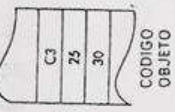
**Banderas:**



**Ejemplo:**

JP 3025

**Antes:**



**Después:**



Carga el registro r con el dato inmediato n.

 $\Gamma \leftarrow \Pi$ 

|   |   |   |   |   |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 |
|---|---|---|---|---|

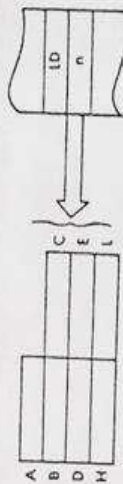
byte 1

byte 2: dato immediato

El contenido de la posición de memoria inmediatamente siguiente al código de operación se carga en el registro especificado; r puede ser:

|         |         |
|---------|---------|
| A - 111 | E - 011 |
| B - 000 | H - 100 |
| C - 001 | L - 101 |
| D - 010 |         |

**Flujo de datos:**



2 ciclos M; 7 estados T; 3.5  $\mu\text{seg}$  @ 2 MHz.

**Immediato.**

| r: | A  | B  | C  | D  | E  | H  | I  |
|----|----|----|----|----|----|----|----|
|    | 3E | 06 | 0E | 16 | 1E | 26 | 2E |

| S | Z | H | P/V | N | C |
|---|---|---|-----|---|---|
|   |   |   |     |   |   |

(efecto nulo)

LD C, 3B



10

2

# LD r, r'

Carga el registro r con el contenido del r'.

**Función:**

$r \leftarrow r'$

**Formato:**

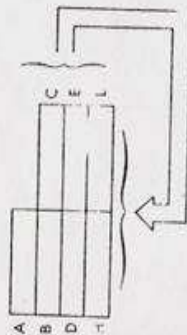


**Descripción:**

El contenido del registro fuente especificado se carga en el registro destino especificado; r y r' pueden ser:

- A - 111
- B - 000
- C - 001
- D - 010
- E - 011
- H - 100
- L - 101

**Flujo de datos:**



**Tiempo:**

1 ciclo M; 4 estados T; 2  $\mu s$ ; a 2 MHz.

**Direccionamiento:**

Implicito.

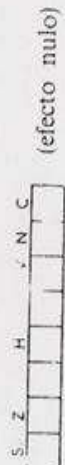
**Códigos byte:**

A B C D E H L (fuente)

|   |    |    |    |    |    |    |    |
|---|----|----|----|----|----|----|----|
| A | 7F | 78 | 79 | 7A | 7B | 7C | 7D |
| B | 47 | 40 | 41 | 42 | 43 | 44 | 45 |
| C | 4E | 48 | 49 | 4A | 4B | 4C | 4D |
| D | 57 | 50 | 51 | 52 | 53 | 54 | 55 |
| E | 5E | 58 | 59 | 5A | 5B | 5C | 5D |
| H | 67 | 60 | 61 | 62 | 63 | 64 | 65 |
| L | 6F | 68 | 69 | 6A | 6B | 6C | 6D |

(dest.)

**Banderas:**



**Ejemplo:**

LD H, A

antes:



Después:



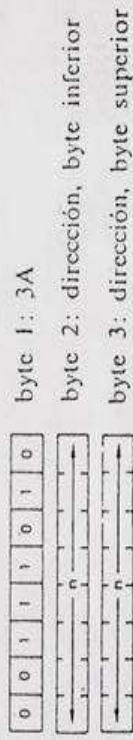
## LD A, (nn)

Carga el acumulador a partir de la posición de memoria (nn).

**Función:**

$A \leftarrow (nn)$

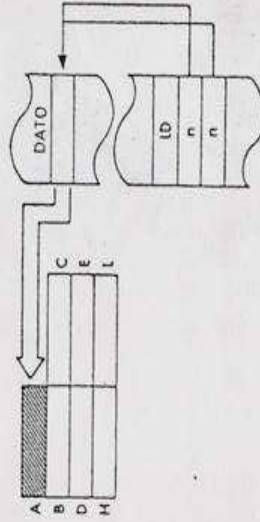
**Formato:**



**Descripción:**

El contenido de la posición de memoria direccionada por el contenido de las dos posiciones de memoria que siguen al código de operación se cargan en el acumulador. El byte de orden inferior de la dirección aparece justo a continuación del código de operación.

**Flujo de datos:**



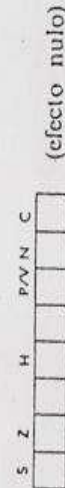
**Tiempo:**

4 ciclos M; 13 estados T; 6.5  $\mu\text{seg}$  @ 2 MHz.

**Direccionamiento:**

Directo.

**Banderas:**



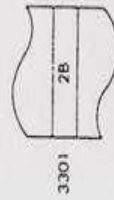
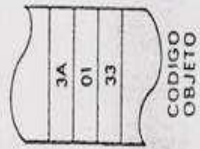
**Ejemplo:**

**LD A.(3301)**

**Antes:**

A 

|    |
|----|
| 0A |
|----|



**Después:**

A 

|    |
|----|
| 2B |
|----|





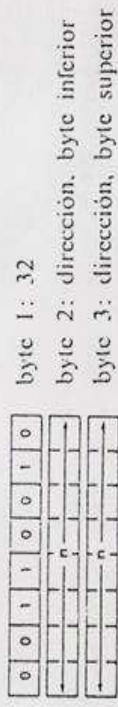
## LD (nn), A

Carga la posición de memoria directamente direccionada (nn) a partir del acumulador.

**Función:**

(nn) ← A

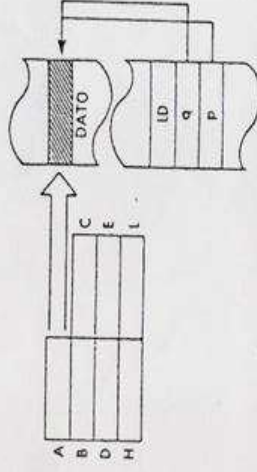
**Formato:**



**Descripción:**

El contenido del acumulador se carga en la posición de memoria direccionada por el contenido de las posiciones de memoria que siguen al código de operación. El byte inferior de la dirección aparece justo a continuación del código de operación.

**Flujo de datos:**



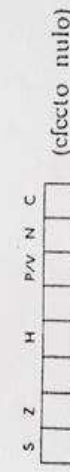
**Tiempo:**

4 ciclos M; 13 estados T; 6.5  $\mu$ seg @ 2 MHz.

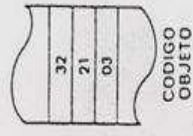
**Direccionamiento:**

Directo.

**Banderas:**

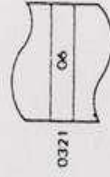


Ejemplo:



LD (0321), A

Antes:



Después:



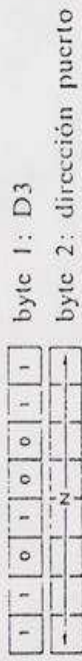
## OUT (N), A

Salida del acumulador al puerto N.

**Función:**

$(N) \leftarrow A$

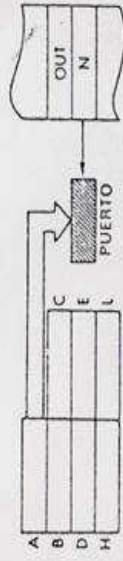
**Formato:**



**Descripción:**

El contenido del acumulador se lleva al dispositivo periférico direccionado por el contenido de la posición de memoria que sigue al código de operación.

**Flujo de datos:**



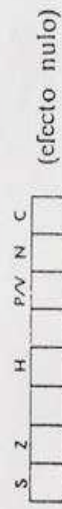
**Tiempo:**

3 ciclos M; 11 estados T; 5.5  $\mu$ s (a 2 MHz).

**Direccionamiento:**

Externo.

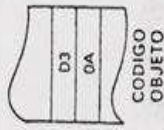
**Banderas:**



**Ejemplo:**

OUT (0A), A

Antes:



Después:



# RET

Retorno de subrutina.

**Función:**

$PC_{inf} \leftarrow (SP); PC_{sup} \leftarrow (SP + 1); SP \leftarrow SP + 2$

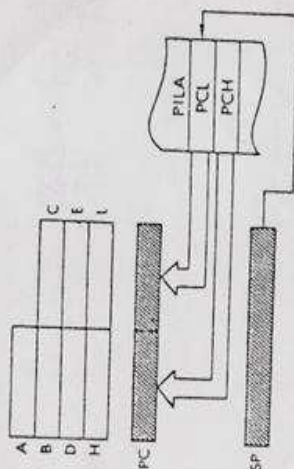
**Formato:**



**Descripción:**

El contador del programa se extrae de la pila tal como se describe en las instrucciones POP. La siguiente instrucción se toma de la posición señalada por el PC.

**Flujo de datos:**



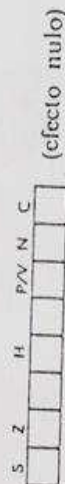
**Tiempo:**

3 ciclos M; 10 estados T; 5  $\mu$ seg @ 2 MHz.

**Direccionamiento:**

Indirecto.

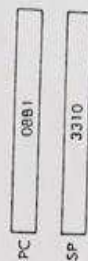
**Banderas:**



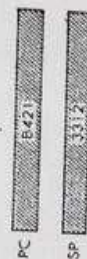
**Ejemplo:**

RET

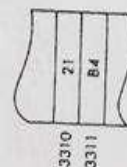
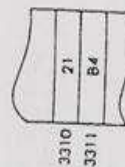
Antes:

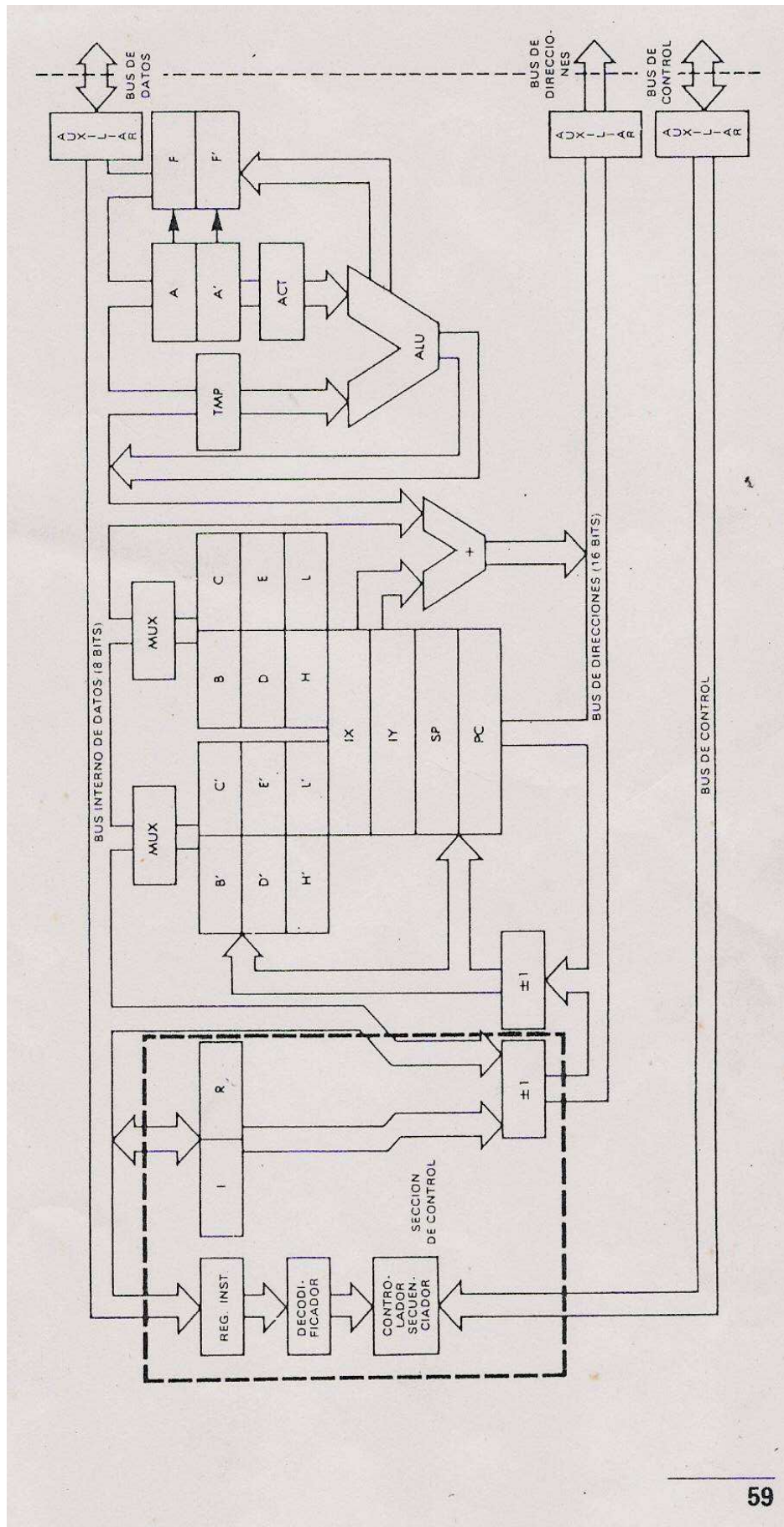


Después:



CODIGO OBJETO







# ORGANIZACIÓN BÁSICA DEL COMPUTADOR

