

Repertorios de Instrucciones: Caraterísticas y Funciones

Daniel Araya Román

Elementos de una instrucción máquina

Los elementos esenciales de las instrucciones de los computadores son, el código de operación (**codop**). Referencias a operandos de origen y destino y referencia a la siguiente instrucción. Los operandos de origen, son valores de entrada para la operación que se va a realizar. Los operandos de destino, son valores de salida de la operación que se va a realizar.

Es importante destacar que los operandos de origen y destino pueden ser almacenados en memoria principal o virtual, registros de la CPU o en dispositivos de entrada/salida.

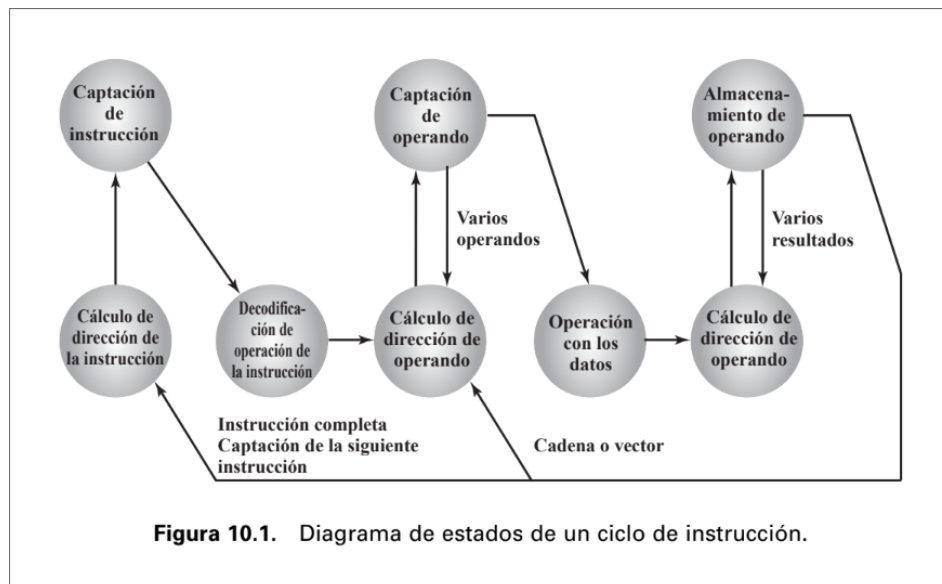


Figure 1: Ciclo de una instrucción. [1]

Tipos de las Instrucciones

Un lenguaje de alto nivel, expresa las operaciones de forma concisa. Mientras que un lenguaje máquina expresa las operaciones de una manera elemental. Es fundamental que el repertorio de instrucciones máquina sea lo suficientemente rico para que se puedan expresar las operaciones de un lenguaje de alto nivel. Los tipos de instrucciones se pueden clasificar en:

- **Transferencia de Datos:** instrucciones de entrada y salida de datos.
- **Almacenamiento de Datos:** instrucciones de memoria.
- **Procesamiento de Datos:** instrucciones aritméticas y lógicas.
- **Control de Flujo:** instrucciones de comprobación y de bifurcación.

Representación de las Instrucciones

Dentro del computador cada instrucción se representa mediante una secuencia de bits. Dividida en campos correspondientes a los elementos que la constituyen. En la mayoría de repertorios de instrucciones se utiliza más de un formato de instrucción. Durante la ejecución del programa, la instrucción se almacena en el registro de instrucción (IR) de la CPU. La CPU extrae los campos de la instrucción y los utiliza para realizar la operación correspondiente. Los códigos de operación (codop) se representan mediante abreviaturas denominadas **mnemónicos**.

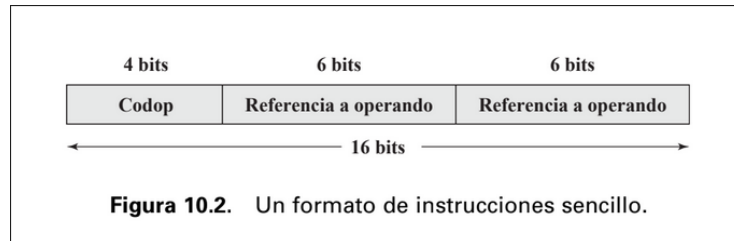


Figure 2: Formato de una instrucción. [1]

Número de direcciones

La mayoría de las instrucciones de los computadores actuales son una, dos o tres direcciones. Las instrucciones de tres direcciones no son comunes ya que requieren formatos relativamente largos.

Instrucción	Comentario
SUB Y, A, B	$Y \leftarrow A - B$
MPY T, D, E	$T \leftarrow D \times E$
ADD T, T, C	$T \leftarrow T + C$
DIV Y, Y, T	$Y \leftarrow Y \div T$

(a) Instrucciones de tres direcciones

Instrucción	Comentario
MOVE Y, A	$Y \leftarrow A$
SUB Y, B	$Y \leftarrow Y - B$
MOVE T, D	$T \leftarrow D$
MPY T, E	$T \leftarrow T \times E$
ADD T, C	$T \leftarrow T + C$
DIV Y, T	$Y \leftarrow Y \div T$

(b) Instrucciones de dos direcciones

Instrucción	Comentario
LOAD D	$AC \leftarrow D$
MPY E	$AC \leftarrow AC \times E$
ADD C	$AC \leftarrow AC + C$
STOR Y	$Y \leftarrow AC$
LOAD A	$AC \leftarrow A$
SUB B	$AC \leftarrow AC - B$
DIV Y	$AC \leftarrow AC \div Y$
STOR Y	$Y \leftarrow AC$

(c) Instrucciones de una dirección

Figura 10.3. Programas para calcular $Y = \frac{(A - B)}{(C + D \times E)}$.

Figure 3: Número de direcciones. [1]

La instrucción de una sola dirección es muy simple. Para que funcione, una segunda dirección debe estar implícita. Esto era usual en las primeras máquinas. La dirección implícita era un registro conocido como el **acumulador (AC)**.

Incluso es posible que una instrucción no tenga direcciones. Esto se logra utilizando la pila. El número de direcciones es una decisión básica de diseño. Menos direcciones significa instrucciones más primarias, lo que requiere un procesador menos complejo. Por otro

lado, un mayor número de direcciones significa instrucciones más complejas, pero menos instrucciones para realizar una tarea.

Diseño del Repertorio de Instrucciones

Es importante entender que **el repertorio de instrucciones es el medio que tiene el programador para poder controlar el procesador**. En consecuencia debe de adaptarse a las necesidades del programador. Los aspectos fundamentales del diseño del repertorio de instrucciones son.

- **Repertorio de Operaciones:** El repertorio de operaciones es el conjunto de operaciones que se pueden realizar en el procesador. Cuántas y cuán complejas deben ser.
- **Tipos de Datos:** Los distintos tipos de datos que se pueden utilizar en las operaciones.
- **Formatos de Instrucciones:** Longitud de la instrucción (en bits), número de direcciones, tamaño de los campos, etc.
- **Registros:** número de registros del procesador que se pueden referenciar por las instrucciones y el uso de ellos.
- **Modos de direccionamiento:** Los distintos modos de direccionamiento que puede especificarse la dirección de un operando.

Tipos de operandos

Los operandos de una instrucción pueden ser de distintos tipos tales como: **direcciones**, **números**, **caracteres** o **datos lógicos**.

Tipos de operaciones

Las operaciones que se pueden realizar en un procesador se pueden clasificar en la siguiente forma:

- **Transferencia de datos:** Mueven datos entre registros y memoria. Deben de especificarse la dirección de origen y la dirección de destino. Que podrían ser de memoria, un registro, cabecera de la pila o un dispositivo de entrada/salida. Indicarse la longitud de datos a transferir. Especificar el modo de direccionamiento.
- **Operaciones aritméticas:** Realizan operaciones aritméticas sobre datos numéricos. Suma, resta, multiplicación, división, etc.
- **Operaciones lógicas:** Realizan operaciones lógicas sobre datos lógicos. Manipulación de bits, operaciones booleanas, desplazamiento, rotación, etc.
- **Operaciones de conversión:** Arquellas que cambian el formato de los datos. Conversión binario a decimal, decimal a binario, etc.
- **Operaciones de control del sistema:** Realizan llamadas al sistema operativo, interrupciones. Instrucciones privilegiadas, que solo pueden ser ejecutadas por el sistema operativo.
- **Operaciones de control de programa:** Cambiar la secuencia de ejecución de las instrucciones. Saltos, bifurcaciones, llamadas a subrutinas, etc.

Instrucciones de bifurcación

Las instrucciones de bifurcación son aquellas que cambian la secuencia de ejecución de las instrucciones. Las instrucciones de bifurcación pueden ser condicionales o incondicionales. Por ejemplo:

```
is_zero proc
again:
    mov ax, 0
    je end          ; conditional jump
    jmp again       ; unconditional jump

end:
    ret
is_zero endp
```

Instrucciones de llamada a procesamiento

Las razones principales para utilizar procedimientos son la economía y la modularidad. Ya que un procedimiento permite que la misma porción de código se puede reutilizar muchas veces. También estos permiten que programas largos, se puedan subdividir en unidades más pequeñas.

```
; GetMousePosition
;
; int 33H
; ax = 03H
; bx: button status
; cx: x position
; dx: y position
;
GetMousePosition proc far
mov ax, 03H
int 33H
mov [mouseX], cx
mov [mouseY], dx
mov [mouseStatus], bx
ret
GetMousePosition endp

; SetMousePosition
;
; int 10H
; ax = 02H
; bx = 00H
; dh = x position
; dl = y position
;
SetMousePosition proc far
mov ah, 02H
mov bh, 00H
int 10H
ret
SetMousePosition endp
```

Figure 4: Ejemplo de procedimientos en ensamblador.

Es importante entender que el uso de procedimientos requiere de dos instrucciones

básicas, **call**, que produce la bifurcación. **ret**, instrucción de retorno. El CPU debe de guardar la dirección de retorno, puede estar normalmente en tres lugares, un registro, la pila o la principio del procedimiento. Solamente que tiene el limitante que no puede hacer uso de procedimientos reentrantes, en otras palabras. **Procedimientos recursivos.**

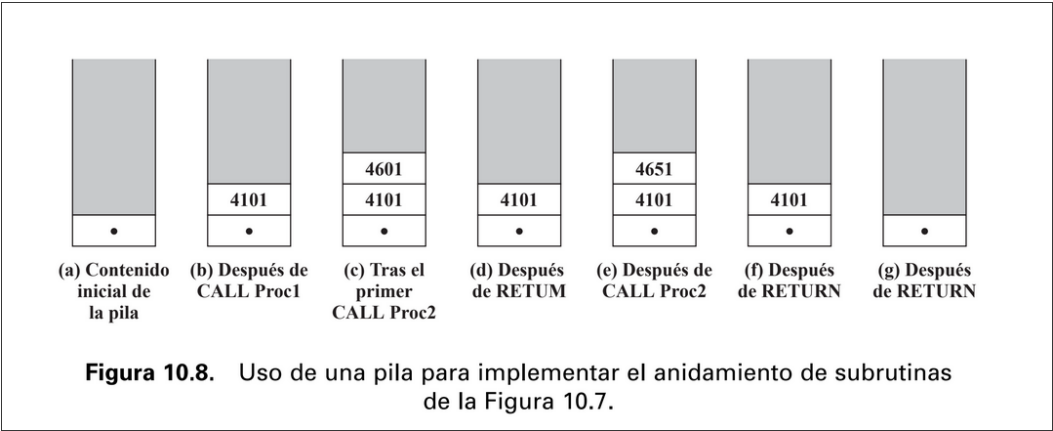


Figure 5: Uso de la pila. [1]

Códigos de condición

Son bits de reigstros especiales que pueden ser activados por ciertas operaciones, y ser utilizados en bifurcaciones condicionales.

Tabla 10.9. Códigos de condición del Pentium.

Bit de estado	Nombre	Descripción
C	Acarreo	Indica un acarreo o un acarreo negativo en la posición de bit más significativa, tras una operación aritmética. También se modifica por algunas operaciones de desplazamiento y de rotación.
P	Paridad	Paridad del resultado de una operación aritmética o lógica. Un 1 indica paridad par, y el 0 paridad impar.
A	Acarreo auxiliar	Representa un acarreo o un acarreo negativo entre las dos mitades de una operación aritmética o lógica de ocho bits, utilizando el registro AL.
Z	Cero	Indica que el resultado de una operación aritmética o lógica es 0.
S	Signo	Indica el signo del resultado de una operación aritmética o lógica.
O	Desbordamiento	Indica un desbordamiento aritmético después de una suma o una resta.

Figure 6: Códigos de condición. [1]

References

[1] W. Stallings. *Organización y arquitectura de computadores*. Fuera de colección Out of series. Pearson Educación, 2006.