

Investigación 3

Angel Jesús Zorrilla Cuevas

Instituto Tecnológico de Oaxaca
Ingeniería en Sistemas Computacionales

Lenguajes de interfaz

Prof: Ing. Oscar Arturo López Córdoba

19 de jul. de 22

Instrucciones lógicas.

Ensamblador cuenta con un grupo de cuatro instrucciones lógicas a nivel de bit, las cuales con excepción de la primera requieren de dos operandos, estas son las siguientes:

NOT:

La instrucción NOT o negación requiere un solo operando y su función es cambiar el estado de los bits del mismo, es decir cambiar los ceros por unos y los unos por ceros.

Ejemplo:

NOT ax

Se aplica la negación al valor de los registros ax.

AND:

Esta instrucción también conocida como producto lógico requiere de dos operandos y su valor será igual a uno cuando los bits que se comparen ambos sean uno.

Ejemplo:

AND ax, bx

Se aplica el producto lógico a los valores de ax y bx.

OR:

Las instrucciones OR también conocida como suma lógica requiere de dos operandos y su valor será uno si alguno de los bits que se compara es uno.

Ejemplo:

OR ax, bx

Se aplica la suma lógica a los valores de ax y bx.

XOR:

La instrucción XOR o suma lógica exclusiva requiere dos operandos, los cuales se comparan y el resultado obtenido es uno cuando uno de los bits es uno y el otro es cero.

Ejemplo:

XOR ax, bx

Se aplica la instrucción XOR a los valores de ax y bx.

Desplazamiento y rotación.

Las instrucciones de desplazamiento son cuatro: SHL, SHR, SAR, SAL; y su objetivo es desplazar los bits de un operando un determinado número de posiciones a la izquierda o a la derecha. La estructura de los operandos manejados por estas instrucciones y su significado es idéntico para las cuatro instrucciones.

SHL (Shift left, desplazamiento a la izquierda).

Se desplazan a la izquierda los bits del operando destino tantas posiciones como indique el operando fuente. El desplazamiento de una posición se realiza de la siguiente forma, el bit de mayor peso del operando se desplaza al bit CF del registro del estado, el resto de los bits se una posición hacia la izquierda, y la posición de menor peso se rellena con un 0

SAL (Shift Arithmetic left, desplazamiento aritmético a la izquierda).

El objetivo de un desplazamiento aritmético a la izquierda es multiplicar un operando, interpretado con signo, por una potencia de 2.

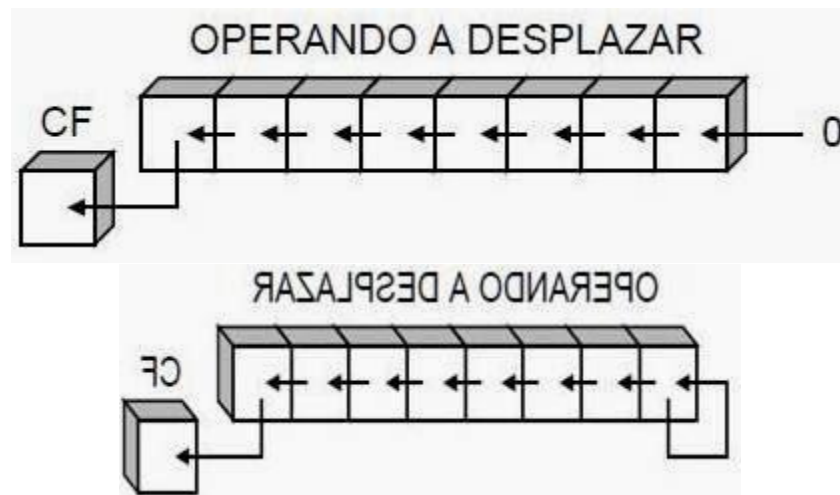
Para llevar a cabo este tipo de desplazamiento, hay que desplazar los bits del operando hacia la izquierda introduciendo ceros por su derecha.

SHR (Shift Right, desplazamiento a la derecha).

La instrucción SHR funciona de la misma forma que SHL, peros desplazando los bits a la derecha en lugar de a la izquierda.

SAR (Shift Arithmetic Right, desplazamiento aritmético a la derecha).

Esta instrucción desplaza los bits del operando destino a la derecha tantos bits como indique el operando fuente. Esta forma de funcionamiento es similar a la instrucción SHR; sin embargo, ambas instrucciones se diferencian en que SAR, en vez de introducir ceros por la izquierda del operando, replica el bit de mayor peso (bit de signo) en cada desplazamiento.





Referencias.

- 2.13 *Instrucciones lógicas*. (2018, noviembre 12). Lenguajes de interfaz.
<https://ittlenguajesdeinterfaz.wordpress.com/2-13-instrucciones-logicas/>
- 2.14 *Desplazamiento y rotación*. (2018, noviembre 12). Lenguajes de interfaz.
<https://ittlenguajesdeinterfaz.wordpress.com/2-14-desplazamiento-y-rotacion/>