

1 [2 puntos] Describa las fases en las que se divide la ejecución de la instrucción RET, retorno de subrutina, indicando las acciones que se realizan en cada una de ellas. ¿Qué registros del procesador se utilizan en cada una de las fases? ¿Cuántas veces se accede a memoria en cada fase? Justifique su respuesta.

SOLUCIÓN

a) Fases de ejecución de la instrucción

En primer lugar está la fase de Fetch, que realiza un acceso a memoria a la dirección donde apunta el contador de programa e incrementa éste. Lo que viene de memoria (la instrucción) se mete en el registro de instrucción.

Se realiza la decodificación.

La ejecución consta de:

- Un segundo acceso a la memoria, de lectura a la dirección donde apunta el puntero de pila, previamente incrementado. Lo que se lee (que se carga en el registro DR) es la dirección de retorno.
- Carga del contador de programa con el contenido del registro DR, lo que significa que continúa la ejecución del programa principal en la instrucción siguiente a la llamada a subrutina.

2 [5 puntos] Un computador dispone del siguiente formato de representación: Coma flotante de 16 bits con el primer bit para el signo, los 6 bits siguientes para el exponente, en exceso 32, y los 9 últimos bits para la magnitud de la mantisa, con bit implícito y la coma situada a la izquierda de éste.

a) Determine el rango y la resolución de este formato.

b) Siendo las cadenas de bits $A = H'D523$ y $B = H'4D7F$ representaciones en este formato de coma flotante, determine su valor decimal.

c) Represente los valores de A y B en el formato de coma flotante IEEE754 de 32 bits.

d) Realice la operación $A + B$ en el formato de 16 bits utilizando dos bits de guarda, un bit retenedor y redondeo al más próximo. Expresé el resultado en el formato de partida.

e) Determine el error absoluto que se ha producido en la operación.

SOLUCIÓN

a) Rango y resolución del formato.

Números normalizados.

Exponente: El exponente está representado en exceso a 32. Así, su rango es $[-32, 31]$. Como se reserva el exponente mínimo (-32) para la representación del cero, el rango de exponente queda: $[-31, 31]$.

La mantisas normalizadas en este formato se representan en signo-magnitud, con bit implícito y la coma situada a la izquierda del bit implícito:

$$\text{Mantisa: } \pm \begin{cases} ,1000000000 & \rightarrow 2^{-1} \\ ,1111111111 & \rightarrow 1 - 2^{-10} \end{cases}$$

El rango para números normalizados es: $\pm [2^{-1} \cdot 2^{-31}, (1 - 2^{-10}) \cdot 2^{31}]$

La resolución depende del exponente y es: $2^{-10} \cdot 2^E$

b) Valor decimal de las cadenas.

$$A = H'D523 = 1 \ 101010 \ 100100011 = -,1100100011 \cdot 2^{10} = -1100100011 = -803$$

$$B = H'4D7F = 0 \ 100110 \ 101111111 = +,1101111111 \cdot 2^6 = -110111,1111 = +55,9375$$

c) Representación de los números en el formato IEEE754 de 32 bits.

Este formato usa un bit para el signo, 8 para el exponente, representado en exceso a 127 y 23 para la mantisa, con bit implícito y la coma situada a la derecha de éste. Así:

$$A = -1,1001\ 0001\ 1000\ 0000\ 0000\ 000 \cdot 2^9 = 1\ 10001000\ 100\ 1000\ 1100\ 0000\ 0000\ 0000 = \text{H}'\text{C448C000}$$

$$B = +1,1001\ 1111\ 1000\ 0000\ 0000\ 000 \cdot 2^5 = 0\ 10000100\ 101\ 1111\ 1100\ 0000\ 0000\ 0000 = \text{H}'\text{425FC000}$$

d) Suma $A + C$.

Este formato utiliza dos bits de guarda y un bit retenedor para la suma. Los números a sumar son:

$$A = -,1100100011 \cdot 2^{10} \quad y \quad B = +,1101111111 \cdot 2^6$$

Se restan los exponentes: $E_A - E_B = 10 - 6 = 4$. Hay que desplazar la mantisa de B cinco lugares a la derecha. Así, teniendo en cuenta los dos bits de guarda y el bit retenedor queda:

$$B = +0,000110111\ 111 \cdot 2^{10}$$

Como tienen signos distintos la operación a realizar es una resta de la mantisa de A menos la mantisa de B desplazada. El signo del resultado es el signo de A

$$\begin{array}{r} M_A \quad \quad \quad ,1100100011\ 000 \\ M_B \quad \quad \quad - ,0000110111\ 111 \\ \hline \quad \quad \quad ,1011101011\ 001 \cdot 2^{10} \text{ Normalizado} \\ \text{Redondeo} \quad \quad \quad ,0000000000\ 100 \\ \hline \quad \quad \quad ,1011101011\ 101 \cdot 2^{10} \end{array}$$

$$A + B = -,1011101011 \cdot 2^{10} = 1\ 101010\ 011101011 = \text{H}'\text{D4EB}$$

e) Error absoluto del resultado

Poniendo en decimal el resultado obtenido queda:

$$A + B = -1011101011 = -747$$

Sumando los valores decimales tenemos

$$A + B = -803 + 55,9375 = -747,0625$$

El error absoluto:

$$e_a = |747,0625 - 747| = 0,0625$$

3 [3 puntos] Sea un procesador con palabras y direcciones de 32 bits y acceso a memoria a nivel de byte. Indique qué hacen las instrucciones que se muestran a continuación, los registros, biestables de estado y posiciones de memoria que se modifican en su ejecución y sus valores finales. Suponga para ello que R1 tiene el valor H'00020304, R3 H'00FFFFFF y que las direcciones de memoria H'00020304, H'00020308 y H'0002030C contienen los valores decimales 50, 60 y 70 respectivamente.

a) XOR .R1, .R1

b) AND .R1, H'0000000F

c) BR #4[.R1]

d) LD .R2, #4[.R1++]

e) ST .R3, #0[- .R1]

SOLUCIÓN

a) Realiza una operación *or exclusivo* del valor contenido en R1 consigo mismo.

En R1 queda el valor 0 y el biestable de estado Z toma el valor 1

b) Realiza la operación *and* del valor contenido en R1 con el dato inmediato H'0000000F.

En R1 queda el valor H'00000004 y el biestable de estado Z toma el valor 0

c) Salta a la instrucción contenida en la dirección a la que apunta R1+4.

Se modifica el PC que queda con el valor H'00020308

d) Lee el contenido de la dirección de memoria a la que apunta $R1+4$ ($H'00020308$), que se almacena en $R2$. A continuación se incrementa $R1$ en 4 (número de bytes que ocupa cada palabra).

$R2 = 60$ y $R1 = H'00020308$

e) Se decrementa el valor de $R1$ en 4 (número de bytes que ocupa cada palabra) y se almacena el contenido del registro $R3$ en la dirección de memoria apuntada por el nuevo valor de $R1$ ($H'00020300$).

$R1 = H'00020300$ y $Mem(H'00020300) = H'00FFFFFF$