

EJERCICIOS DE CLASE

OBJETIVOS DEL TEMA

Objetivo general

- Conocer las características básicas de un lenguaje ensamblador y su correspondiente lenguaje máquina
- Conocer el repertorio de instrucciones de un procesador real y las metodologías para la programación de bajo nivel con dicho repertorio.

Objetivos específicos

- Enumerar las características de un repertorio de instrucciones
- Calcular la ubicación de un dato para un modo de direccionamiento dado
- Escribir secuencias de instrucciones en lenguaje ensamblador de diferentes máquinas
- Explicar los diferentes campos que aparecen en el formato máquina de una determinada instrucción
- Traducir instrucciones de lenguaje ensamblador del Z80 a lenguaje máquina y viceversa
- Interpretar el significado y simular la ejecución de programas en ensamblador del Z80
- Enunciar las diferencias entre los enfoques CISC y RISC para el diseño de repertorios de instrucciones
- Clasificar un repertorio de instrucciones como CISC o RISC

BIBLIOGRAFÍA

- Capítulos 10 y 11 del libro "Organización y Arquitectura de Computadores", de W. Stallings
- Capítulo 6 del libro "Fundamentos de Computadores", de R. Hermida
- Material de Z80 disponible en Aula Virtual

Ejercicios

Comparación de repertorios de instrucciones

1. Comparar los computadores de 0, 1, 2 y 3 direcciones escribiendo los programas para calcular la expresión $X = (A+B*C) / (D-E*F-G*H)$, suponiendo los siguientes repertorios de instrucciones (donde M es una posición de memoria y X, Y, Z son direcciones de memoria o registros):

<u>0 direcciones</u>	<u>1 dirección</u>	<u>2 direcciones</u>	<u>3 direcciones</u>
PUSH M	LOAD M	MOVE X, Y	ADD X, Y, Z
POP M	STORE M	ADD X, Y	SUB X, Y, Z
ADD	ADD M	SUB X, Y	MUL X, Y, Z
SUB	SUB M	MUL X, Y	DIV X, Y, Z
MUL	MUL M	DIV X, Y	
DIV	DIV M		

Modos de direccionamiento

2. El campo de dirección de una instrucción contiene el valor decimal 14. Indica dónde se ubica el operando correspondiente en cada uno de los siguientes casos:

- direccionamiento inmediato
- direccionamiento directo
- direccionamiento indirecto
- direccionamiento de registro
- direccionamiento indirecto de registro

3. Un computador presenta los siguientes contenidos en registros y memoria:

$$R1 = 98 \quad R2 = 2$$

Memoria:

Dirección	96	97	98	99	100	101	102	103	104	198	199	200	201	202
Contenido	99	102	101	104	102	106	107	109	110	45	100	34	96	201

¿Cuál sería el valor del operando con los siguientes modos de direccionamiento?

- Directo*, conteniendo el campo correspondiente de la instrucción el valor 100
- Registro*, conteniendo el campo correspondiente de la instrucción el valor 1
- Indirecto de registro*, conteniendo el campo correspondiente de la instrucción el valor 1
- Indirecto de memoria*, conteniendo el campo correspondiente de la instrucción el valor 100
- Desplazamiento*, conteniendo los campos correspondientes de la instrucción los valores 1 (para el registro) y 100 (para el desplazamiento)
- Desplazamiento*, conteniendo los campos correspondientes de la instrucción los valores 2 (para el registro) y 100 (para el desplazamiento)

¿Cuántos bits harían falta para codificar los anteriores modos de direccionamiento si suponemos que el procesador tiene 16 registros de propósito general y una memoria de 16MB?

Si suponemos que T_{MEM} es el tiempo de acceso a memoria para lectura, T_{REG} el de acceso a un registro para lectura y T_+ el tiempo necesario para realizar una suma, ¿cuánto tiempo se emplea en conseguir el dato con cada uno de los modos?

Tema 2: Repertorios de instrucciones

4. Dados los valores de memoria siguientes, y suponiendo una máquina con instrucciones de una sola dirección, y con un acumulador, ¿qué valores cargan a las siguientes instrucciones en el acumulador?

MEM[20] = 40 MEM[30] = 50 MEM[40] = 60 MEM[50] = 70

- a) Carga inmediata 20
- b) Carga directa 20
- c) Carga indirecta 20
- d) Carga inmediata 30
- e) Carga directa 30
- f) Carga indirecta 30

Programación en ensamblador

- 5. Escribe un programa en lenguaje ensamblador del Z80 que sume dos números enteros (X e Y) y deje el resultado en Z. X, Y y Z se han de corresponder con las direcciones de memoria, respectivamente, 0xC000, 0xC001h y 0xC002h.
- 6. Intenta el programa anterior con modos de direccionamiento distintos: Registro, Directo, Indirecto Registro, Desplazamiento.
- 7. Escribe un programa en lenguaje ensamblador que sume dos vectores de enteros (A y B) y deje el resultado en otro vector C. El tamaño de los vectores será de 3 componentes. A, B y C se han de corresponder con las direcciones de memoria donde comienza cada vector y serán, respectivamente, 00h, 03h y 06h.
- 8. Escribe un programa en ensamblador que dados dos números enteros, A y B, determine cuál es el mayor de ambos y cuál el menor.
- 9. Escribe un algoritmo en pseudo-código que calcule el producto de dos números enteros utilizando sumas. Traduce dicho algoritmo a lenguaje ensamblador.
- 10. Escribe un algoritmo que realice la suma de dos vectores de 10 componentes mediante un bucle de tipo FOR. Tradúcelo a lenguaje ensamblador.
- 11. Escribe un algoritmo que calcule la suma de los 10 elementos de un vector. Tradúcelo a lenguaje ensamblador.
- 12. Escribe un algoritmo que calcule el elemento máximo de un vector de 10 componentes. Tradúcelo a lenguaje ensamblador.

Conversión entre lenguaje ensamblador y lenguaje máquina

13. Pasa a notación binaria y hexadecimal las siguientes instrucciones escritas en lenguaje ensamblador del Z80:

a) ADD A, (IX-15) b) LD A, (\$4000) c) DJNZ -2
d) JR Z, -24 e) AND C f) XOR \$FE

(Nota: aprovecha el ejercicio para ver las instrucciones no documentadas (en rojo en la tabla) del Z80)

14. Considera las siguientes instrucciones

- | | |
|------------------------|----------------------------------|
| a) 0111 1110 | b) 1101 1101 0111 1110 |
| c) 1111 1101 0010 0011 | d) 1101 1101 0010 0011 |
| e) 0010 0011 | f) 1100 0010 0000 0000 1000 0000 |

¿Qué instrucciones del Z80 representan? Pásalas a formato ensamblador.

Observa si existen prefijos y su relación con las instrucciones sin prefijo

No olvides que el Z80 es LITTLE ENDIAN