

Módulo II: Arquitectura del procesador

Problemas

Problema 1

Un computador tiene las siguientes características:

- una memoria M
- un acumulador A
- un registro de estado de un solo bit: el bit N
- tres instrucciones de longitud fija y direccionamiento directo:

STA D : $M[D] \leftarrow \langle A \rangle$

SUB D : $A \leftarrow \langle A \rangle - M[D]$

JNB D : IF $N=1$ THEN $CP \leftarrow D$ ELSE $CP \leftarrow \langle CP \rangle + 1$

Codificar tres programas que realicen las siguientes funciones:

- Carga del acumulador con el contenido de la posición de memoria X .
- Suma del contenido de la posición de memoria X al contenido del acumulador.
- Implementación de un salto incondicional a la posición Y .

Problema 2

Un computador tiene el siguiente repertorio de instrucciones:

LDA D : $A \leftarrow M[D]$

LDAC N : $A \leftarrow N$

STA D : $M[D] \leftarrow \langle A \rangle$

ADD D : $A \leftarrow \langle A \rangle + M[D]$

SUB D : $A \leftarrow \langle A \rangle - M[D]$

TEST_Z : Activa el bit Z y desactiva el bit N.

TEST_N : Activa el bit N y desactiva el bit Z.

FLIP : Complementa bit a bit el registro de códigos de condición.

JMP_C D : IF bit activo = 1 THEN $CP \leftarrow D$ ELSE $CP \leftarrow \langle CP \rangle + 1$

JMP D : $CP \leftarrow D$

Diseñar los patrones de compilación de las siguientes construcciones de control de alto nivel:

- IF <condición> THEN <sentencias_1> ELSE <sentencias_2>
- WHILE <condición> DO <sentencias>
- REPEAT <instrucciones> UNTIL <condición>

Problema 3

Un vector se encuentra almacenado a partir de la dirección 300 relativa a un área de memoria cuya dirección inicial es 1000. Cada elemento del vector ocupa 4 bytes y se pretende acceder al tercer elemento del vector utilizando direccionamiento indexado: a)

¿Qué información debe contener el operando de la instrucción?

b) ¿Cual debe ser el contenido del registro índice?

c) ¿Qué habrá que modificar si cambia la dirección inicial del área que contiene al vector?

Problema 4

Si en el ejercicio anterior se quiere utilizar *direccionamiento base + desplazamiento indexado* para acceder al tercer elemento del vector:

- ¿Qué información debe contener el operando de la instrucción?
- ¿Qué información debe contener el registro índice?
- ¿Qué información debe contener el registro base?
- ¿Qué habrá que modificar si cambia la dirección inicial del área que contiene al vector?
- ¿Cuales son las ventajas de utilizar este direccionamiento?

Problema 5

El siguiente registro:

```
Var R : record
    a : integer;           (1 palabra)
    b : real;              (2 palabras)
    c : array [0..11] of integer (12 palabras)
    d : char               (1 palabra)
end
```

se halla ubicado en el registro de activación de un procedimiento a partir del desplazamiento -25.

- ¿Cual es el desplazamiento de R.b?
- ¿Cual es el desplazamiento de R.d?
- ¿Qué tipo de direccionamiento sería más conveniente utilizar para la sentencia $R.a := R.c[10]$?

Problema 6

Un programa escrito en un lenguaje de alto nivel tiene la siguiente estructura:

```
program MAIN
...
    procedure A
    ...
        procedure C
        ...
        end C
        ...
        procedure D
        ...
        end D
    ...
    end A
...
    procedure B
    procedure E
    ...
        procedure H
        ...
        end H
    ...
    procedure F
    ...
        Procedure I
        ...
        end I
    ...
    procedure G
    ...
    end G
...
end MAIN
```

y se produce la siguiente secuencia de llamada a procedimientos:

MAIN → A → B → B → F → E → H → H

Si suponemos que:

- a) la dirección inicial del registro de activación de MAIN es 1000
- b) el registro de activación de MAIN ocupa 50 posiciones
- c) el registro de activación de A ocupa 50 posiciones
- d) el registro de activación de B ocupa 100 posiciones
- e) el registro de activación de F ocupa 10 posiciones
- f) el registro de activación de E ocupa 10 posiciones
- g) el registro de activación de H ocupa 50 posiciones
- h) la pila crece hacia posiciones decrecientes de memoria

Calcula la dirección de los elementos del *display* local de H y sus contenidos.

Problema 7

Si suponemos que en el problema anterior existe una variable escalar en la tercera posición después del *display* local en el registro de activación del procedimiento H:

- a) ¿Cual sería el mejor modo de direccionamiento para acceder a él?
- b) ¿Cual es el contenido de registro Puntero al Frame Local (PFL) ?
- c) ¿Cual es el desplazamiento de la variable en el registro de activación?

Problema 8

Un computador tiene las siguientes características: longitud de palabra 24 bits, 8 registros, 16 instrucciones diferentes y 3 modos de direccionamiento incluido el indexado.

- a) Diseñar un único formato de instrucción que especifique el modo de direccionamiento, un registro y un desplazamiento.
- b) ¿Cual es el rango de valores del desplazamiento en magnitud y en c2?

Problema 9

Codificar con formato de longitud fija de 36 bits un conjunto de instrucciones compuesto por los siguientes grupos:

- a) 7 instrucciones de dos direcciones de 15 bits y una dirección de 3 bits
- b) 500 instrucciones de una dirección de 15 bits y una de 3 bits
- c) 50 instrucciones sin dirección

Problema 10

Probar que en un computador con arquitectura de acumulador que disponga únicamente de la siguiente instrucción:

SUB D cuya semántica es: $A \leftarrow \langle A \rangle - M[D]$, $M[D] \leftarrow \langle A \rangle$

se pueden codificar las siguientes instrucciones:

- | | | | | |
|--------|------------|---|---|------------|
| a) CLA | semántica: | A | ← | 0 |
| b) NEG | semántica: | A | ← | - <A> |
| c) TAD | semántica: | D | ← | <A> |
| d) TDA | semántica: | A | ← | M[D] |
| e) ADD | semántica | A | ← | <A> + M[D] |

Problema 11

Un computador tiene instrucciones de longitud fija de 11 bits con campo de operando de 4 bits. ¿Es posible codificar en este formato 5 instrucciones de 2 operandos, 45 instrucciones de 1 operando y 32 instrucciones sin operando?

Problema 12

Un computador tiene las siguientes características:

- longitud de palabra 16 bits (instrucciones, memoria y registros)
 - 8 registros generales
 - 14 instrucciones de referencia a memoria (1 operando en memoria) con direccionamiento directo e indirecto.
 - 31 instrucciones con 2 operandos direccionables por registro directo e indirecto.
 - 32 instrucciones sin operando explícito.
- a) Especificar la codificación de las instrucciones
b) Proponer una estructura para el decodificador de instrucciones
c) Especificar la zona de memoria accesible con cada tipo de direccionamiento y rango posible de valores de los operandos

Problema 13

Considerando los siguientes datos obtenidos de un compilador de C:

Tipo de instrucción

	UAL	Carga/Almacenamiento	Salto
<i>Ciclos por instrucción</i>	1	3	5
<i>Frecuencia de ejecución</i>			
Versión optimizada	0.46	0.36	0.18
Versión no optimizada	0.45	0.39	0.16

Calcular los CPI (ciclos medios por instrucción) para las versiones optimizada y no optimizada.

Problema 14

Supongamos que se han realizado las siguientes medidas:

- ☐ Frecuencia de ejecución de las instrucciones de PF (punto flotante) = 25%
- ☐ CPI medio de las instrucciones de PF = 4.0
- ☐ CPI medio de las restantes instrucciones = 1.33
- ☐ Frecuencia de ejecución de la instrucción de PF SQRT = 2%
- ☐ CPI de la instrucción SQRT = 20

Comparar las dos alternativas de diseño siguientes:

- 1) reducir a 2 el CPI de SQRT,
- 2) reducir a 2 el CPI de todas las operaciones de PF.

¿Cuál es la de mejor rendimiento?

Problema 15

Codificar con instrucciones SIMD del Pentium (MMX) el siguiente segmento de programa:

```
for i = 0 to 3
    if a[i] = p then
        s[i] = (a[i] - c[i])/2
    else if a[i] = q then
        s[i] = 4*c[i]
    else
        s[i] = s[i]*c[i]
```

Problema 16

Un computador tiene un rendimiento de 3 MFLOPS normalizados al ejecutar una determinada tarea con cálculo intensivo en coma flotante. Si la operación de multiplicación equivale a cuatro operaciones en coma flotante sencillas y la de exponenciación se equipara con 8:

- a) Calcular el tiempo que se tarda en realizar la tarea sabiendo que consta de las siguientes operaciones en coma flotante: 4000 sumas, 1500 multiplicaciones y 2200 operaciones de exponenciación.
- b) Para el computador anterior se ofrecen tres alternativas de diseño de igual coste:
- 1) Introducir una nueva ALU que realiza la operación de exponenciación en 5 ciclos
 - 2) Introducir una nueva ALU que realiza la operación de multiplicación en 1 ciclo
 - 3) Introducir algunas mejoras estructurales que permitan aumentar la frecuencia de reloj del procesador de forma que su rendimiento sea 3.5 MFLOPS

Analizar la rentabilidad de las tres alternativas sabiendo que una operación sencilla en coma flotante se realiza en un ciclo.

Problema 17

Sabiendo que un computador tiene una longitud de palabra de 32 bit y que el repertorio de instrucciones tiene 16 instrucciones diferentes, calcula la máxima memoria direccionable en los siguientes casos:

- a) Instrucciones con dos operandos, ambos con direccionamiento directo a memoria
- b) Instrucciones con dos operandos, uno con acceso directo a un banco de 32 registros y otro con direccionamiento directo a memoria
- c) Instrucciones con tres operandos, todos con direccionamiento directo a memoria
- d) Instrucciones con un solo operando con direccionamiento directo a memoria