

Arquitectura de Computadoras

2do Parcial Buceo Turno Nocturno 2013

1) Sea una CPU de 16 bits con las siguientes características que se describen a continuación:

- Tiene 8 registros de uso general de 16 bits (*R0 a R7*)
- Posee las siguientes instrucciones (se indica el código y la semántica):

0. LOAD REG1, REG2 > Instrucción que guarda en REG2 el contenido de la dirección de memoria almacenada en REG1.

1. STORE REG1, REG2 > Instrucción que escribe el contenido de REG1 en la dirección de memoria almacenada en REG2.

2. NOP > Instrucción que solo consume ciclos de reloj (no hace nada).

3. SETLO INM, REG > Instrucción para cargar el valor constante INM de 8 bits en la parte baja de REG.

4. SETHI INM, REG > Instrucción para cargar el valor constante INM de 8 bits en la parte alta de REG.

5. ADD REG1, REG2, REG3 > Suma (ADD). Esta operación trabaja sobre REG1 y REG2 como operandos y el resultado lo guarda en REG3.

6. SUB REG1, REG2, REG3 > Resta (SUB). Esta operación trabaja sobre REG1 y REG2 como operandos y el resultado lo guarda en REG3.

7. AND REG1, REG2, REG3 > Operación lógica AND bit a bit. Trabaja sobre REG1 y REG2 como operandos y el resultado lo guarda en REG3.

8. OR REG1, REG2, REG3 > Operación lógica OR bit a bit. Trabaja sobre REG1 y REG2 como operandos y el resultado lo guarda en REG3.

9. XOR REG1, REG2, REG3 > Operación lógica XOR bit a bit. Trabaja sobre REG1 y REG2 como operandos y el resultado lo guarda en REG3.

A. SL REG1, INM, REG2 > Instrucción que mueve a la izquierda tantos bits de REG1 como indique el inmediato INM (de 4 bits) y el resultado lo pone en REG2.

B. SR REG1, INM, REG2 > Instrucción que mueve a la derecha tantos bits de REG1 como indique el inmediato INM (de 4 bits) y el resultado lo pone en REG2.

C. CMP REG1, REG2 > Instrucción de comparación que realiza la resta REG1-REG2, actualizando las banderas de condición.

D. JZ INM > Instrucción de salto condicional, si la bandera de Zero está en 1 salta INM (INM es un número con signo de 12 bits) instrucciones desde la posición actual

E. JN INM > Instrucción de salto condicional, si la bandera de Negative está en 1 salta INM (INM es un número con signo de 12 bits) instrucciones desde la posición actual

F. JMP INM > Instrucción de salto incondicional, salta INM instrucciones (INM es un número con signo de 12 bits) desde la posición actual.

a) Defina un formato de *instrucción* para esta CPU e indique la codificación de cada una de las *instrucciones*.

b) Implemente en lenguaje *ensamblador* un programa que sume los valores que se encuentran en las direcciones de memoria 0xF911 y 0xFA11, y guarde el resultado en la dirección de memoria 0xFB11 si el resultado es diferente de cero. Si el resultado es cero debe guardar todo uno (FFFF) en esa dirección.

nota: El programa estará cargado en memoria a partir de la dirección 0x0000.

- 2) a)** Explique la técnica de *pipelining*.
- b)** Enumere las etapas del *ciclo de instrucción* visto en el curso, explicando cada una.
- c)** ¿Qué es un *hazard*?
- d)** Enumere los diferentes tipos de *hazard* detallando en que consiste cada uno.
- e)** Describa las principales características de las *memorias DDR SDRAM*.
- 3) a)** ¿Qué es una *ROM*? Explique como es su circuito interno.
- b)** ¿Qué es la *Memoria Cache*?
- c)** Describa las distintas *Funciones de Correspondencia* (*Diferentes formas de mapear los bloques de memoria a las líneas de la memoria cache*).
- d)** Defina *Interrupción*.
- e)** Explique las diferentes técnicas de identificación de *interrupciones* y en que consiste el *enmascaramiento* de estas.