

Ejemplo Instrucciones

Alberto Ruiz

IES Virgen de la Paloma

Ejercicio

- Vamos a comparar cómo se comportan distintos procesadores a la hora de ejecutar un determinado programa
- El programa será:
- $M[1A] = M[1A] * M[2B] + M[3C] * M[4D] * M[5E]$
- $M[2B]$ quiere decir el valor contenido en la posición de memoria 2B, expresada en hexadecimal

Memoria

- Pregunta 1: ¿Cuántos bits necesitaremos para representar estas direcciones de memoria?
- Pregunta 2: Y con esos bits, ¿qué rango de direcciones reconocería el procesador? Es decir, ¿cuál será la dirección de memoria más baja y la más alta en la que se podrá leer o escribir?

Memoria

- Supongamos que las primeras 32 posiciones de memoria se reservan para almacenar valores temporales
- Pregunta 3: ¿Cuál será el rango de direcciones que podremos utilizar para almacenar resultados temporales de los programas?

- Para cada uno de los siguientes procesadores se indica su número de direcciones y su juego de instrucciones
- Tu trabajo para cada uno será:
 - Debes determinar su formato de instrucción mínimo, incluyendo los bits dedicados a código de operación, dirección de operando/s y dirección de resultado, según el caso.
 - Asignar un código de operación a cada instrucción del juego de instrucciones
 - Realizar el programa expresándolo como un conjunto de instrucciones (por ejemplo $1A \leftarrow 2B + 2C$)
 - Estimar el coste del programa (medido en número de accesos a memoria)
 - Pasar las dos primeras instrucciones a binario

Procesador P1

- Procesador de 3 direcciones
- Instrucciones:
 - ADD $M[A] + M[B] \rightarrow M[C]$
 - MUL $M[A] * M[B] \rightarrow M[C]$

Procesador P2

- Procesador de 2 direcciones
- Instrucciones:
 - ADD $M[A] + M[B] \rightarrow M[A]$
 - MUL $M[A] * M[B] \rightarrow M[A]$
 - MOVE $M[A] \rightarrow M[B]$

Procesador P3

- Procesador de 1 dirección con acumulador
- Instrucciones:
 - LOAD $M[A] \rightarrow Ac$
 - STORE $Ac \rightarrow M[A]$
 - ADD $M[A] + Ac \rightarrow Ac$
 - MUL $M[A] * Ac \rightarrow Ac$

Procesador P4

- Procesador de 0 direcciones
- Pila implementada en memoria (se reserva una zona de la memoria para simular la pila)
- Instrucciones:
 - PUSH M[A] (el contenido de la dirección de memoria A se lleva a la cima de la pila)
 - POP M[A] (el valor de la cima de la pila se almacena en la dirección de memoria A)
 - ADD
 - MUL

Procesador P5

- Procesador de 0 direcciones
- Pila cableada (la pila es hardware, no se necesitan accesos a memoria para manipularla)
- Instrucciones:
 - PUSH M[A] (el contenido de la dirección de memoria A se lleva a la cima de la pila)
 - POP M[A] (el valor de la cima de la pila se almacena en la dirección de memoria A)
 - ADD
 - MUL