

2. DISEÑO DEL REPERTORIO DE INSTRUCCIONES

2.1. [3!] Con la ayuda de la figura 2.4 [Hennessy and Patterson 2002, p. 95] responde a las siguientes preguntas: A) ¿Cuál es la principal ventaja e inconveniente de que las instrucciones aritmético-lógicas binarias tengan 2 frente a que tengan 3 operandos explícitos? B) Tanto las máquinas registro-registro como las registro-memoria pueden tener 2 ó 3 operandos explícitos pero ¿qué opción es más común en cada caso?

2.2. [?!] Los procesadores RISC dominan el mercado de los procesadores empujados. Pero en estos procesadores la memoria es escasa por lo que es importante una buena densidad de código, lo cual no es precisamente una característica de los procesadores RISC. Para conocer tres modos en que se ha solucionado este problema lee el apartado titulado “*Reduced Code Size in RISCs*” (excepto el penúltimo párrafo) [Hennessy and Patterson 2002, pp. 119-120]. ¿Qué ventaja e inconveniente tiene la solución de IBM frente a la de procesadores como ARM Thumb?

2.3. [5!] Se desea comparar la eficiencia de memoria de tres tipos de computadores:

- Basados en acumulador.
- De tipo memoria-memoria con tres operandos de memoria en las operaciones de la UAL: *destino, fuente1 y fuente2*.
- De carga/almacenamiento con tres registros como operandos en las operaciones de la UAL.

Se supone que:

- El código de operación ocupa siempre un byte.
- Todas las direcciones de memoria ocupan dos bytes.
- Todos los operandos datos son de cuatro bytes.
- Los especificadores de registros ocupan cinco bits.
- La longitud de cualquier instrucción es un número entero de bytes.

1º/ Escribir el código, para cada uno de los tipos de máquinas, correspondiente a la instrucción de alto nivel

$A := B + C$

¿En qué arquitectura resulta menor el tamaño de código? ¿Cuál es más eficiente en cuanto al ancho de banda total requerido (accesos a memoria para instrucciones y datos)?

2º/ Responde las mismas preguntas para la siguiente secuencia de código en alto nivel

$A := B + C$

$B := A + C$

$D := A - B$

¿Por qué la respuesta es diferente?

2.4. [;4!] Se quieren comparar tres máquinas A, B y C con las siguientes características:

- Sus instrucciones son de tres operandos. Cualquier operando puede ser una referencia a memoria o un registro.
- Cada uno de los operandos tiene la misma probabilidad de estar en un registro.
- El coste de acceso a un operando de memoria es de seis ciclos de reloj. El coste de acceso a un operando registro es de un ciclo.

Las diferencias entre las máquinas se detallan en la siguiente tabla:

Máquina	Nº de registros	Ciclos de ejecución por operación	Probabilidad de que un operando esté en un registro
A	0	4	0,0
B	8	5	0,5
C	16	6	0,8

Los ciclos de ejecución de una instrucción son los de ejecución por operación más los de acceso a los operandos. ¿Cuál es el número medio de ciclos por instrucción en cada máquina?

2.5. [;3!] Dados los porcentajes y número de instrucciones ejecutadas al ejecutar el programa TeX (ver transparencia 16) y suponiendo los siguientes valores de CPI:

Operación ALU: 1

Load/Store: 3

Salto: 5

A) Calcula el número medio de ciclos por instrucción para la ejecución del programa TeX compilado sin y con optimizaciones.

B) ¿Cuántas veces es más rápido el programa optimizado?

2.6. [;6!] Algunos investigadores han sugerido que puede ser útil añadir un modo de direccionamiento registro-memoria a una máquina de carga-almacenamiento. El objetivo es sustituir un par de instrucciones tales como

Load R1, Mem[Rs]

Add R2, R2, R1

por una sola, como

Add R2, Mem[Rs]

Se sabe que: a) El nuevo tipo de instrucciones exige aumentar el ciclo de reloj en un 10%; b) para el *benchmark* GCC, las frecuencias de las instrucciones en la máquina de carga-almacenamiento son

Salto	18%
Cargas	24%
Almacenamientos	12%
Operaciones ALU	46%

Suponiendo que la nueva instrucción no afecta al CPI, ¿qué porcentaje de las instrucciones de carga debe haberse eliminado para que el rendimiento sea el mismo?

2.7. [;3!] A) Relaciona las formas de instrucción de una columna, pertenecientes al computador RISC-I, con los tipos de direccionamiento de la otra. (LDL es el nombre de la instrucción *LOAD normal*; R0 es un registro de sólo lectura que contiene 0)

a) JMPR condición, Inm	Inmediato
b) CALL Rd,Inm(R0)	De registro
c) ADD R_{s1}, R_{s2}, Rd	Indexado
d) LDL (R0)Inm,Rd	Relativo al PC
e) LDL (R1),Rd	Indirecto mediante registro
f) LDL (R1)Inm, Rd	Directo

B) ¿Qué instrucción permitirá la asignación entre dos registros: $R1 := R2$?

