

Ampliación de Arquitecturas de Computadoras

Examen de junio (26/06/06) Teoría y problemas

Durante el examen no se permiten apuntes ni nada encima de la mesa. El tiempo para la realización de este ejercicio es de 2 horas. En caso de duda escribe en el examen tu suposición y sigue haciendo el ejercicio.

1. (3 puntos) Supongamos una máquina vectorial con registros vectoriales de 64 elementos, una unidad funcional de cada tipo (ADDV, SUBV, MULTV y DIVV) y un cauce de lectura (LV) y otro de escritura (SV) con la memoria. Cada registro vectorial tiene dos cauces de lectura y uno de escritura. Los tiempos de arranque son 12 para LV y SV; 6 para ADDV y SUBV; 7 para MULTV; y 16 para DIVV. El tiempo de bucle es de 15 ciclos y la frecuencia de reloj es de 350 MHz. La máquina soporta encadenamiento. Dado el siguiente código vectorial (parte interna de un bucle):

```
LV      V1, R1
ADDV    V2, V1, V1
LV      V3, R2
SUBV    V4, V2, V1
MULTV   V5, V3, V1
SV      R1, V4
SV      R2, V5
```

Calcula R_{∞} (MFLOPS) y N_v (tamaño del vector tal que el procesador vectorial tiene el mismo rendimiento que uno escalar). Para calcular N_v se supondrá que el procesador escalar tarda 20 ciclos en realizar las operaciones anteriores sobre cada elemento del vector. Debe incluirse el código con la representación gráfica de la separación de las instrucciones en convoyes y los tiempos de arranque.

2. (2 puntos) A continuación se presentan dos códigos. Ambos realizan la misma operación dentro de un bucle que se ejecuta para vectores de longitud n . Suponiendo que f es la fracción del vector A cuyos elementos son distintos de cero, explicad y calculad para qué valor de f es más rápido uno u otro código. Suponed para ello que el tiempo de bucle y los tiempos de arranque de todas las unidades funcionales son despreciables y que todas ellas se ejecutan en un único ciclo de reloj.

LV	V1, Ra	; Carga vector A en V1	LV	V1, Ra	; Carga vector A en V1
LV	V2, Rb	; Carga vector B en V2	LD	F0, #0	; Carga F0 con 0
LD	F0, #0	; Carga F0 con 0	SNESV	F0, V1	; VM(i)=1 si V1(i)!=0
SNESV	F0, V1	; VM(i)=1 si V1(i)!=0	CVI	V2, #8	; Genera indices en V2
SUBV	V1, V1, V2	; Resta	POP	R1, VM	; Numero de unos en VM
MULTV	V1, V1, V2	; Multiplica	MOVI2S	VLR, R1	
ADDV	V1, V1, V2	; Suma	CVM		; Mascara a todo unos
CVM		; Pone la mascara a unos	LVI	V3, (Ra+V2)	; A distinto de 0
SV	Ra, V1	; Almacena resultado.	LVI	V4, (Rb+V2)	; Los mismos de B
			SUBV	V3, V3, V4	; Resta
			MULTV	V1, V1, V2	; Multiplica
			ADDV	V1, V1, V2	; Suma
			SVI	(Ra+V2), V3	; Guarda resultado

3. (1.5 puntos) Da la definición de redes de tipo malla, n-cubo k-arias e hipercubos. Dibuja una malla de dos dimensiones y 6 nodos. Dibuja un 2-cubo ternario. Dibuja una red que sea a la vez malla, n-cubo k-aria e hipercubo.
4. (1.5 puntos) Explica en qué consiste el mecanismo de conmutación de exploración. Dibuja el diagrama de tiempos de este mecanismo suponiendo que un paquete atraviesa tres enlaces. ¿Qué dos mecanismos de conmutación combina este mecanismo de exploración?
5. (2 puntos) Explica y describe en pseudo-código el algoritmo de encaminamiento basado en salto negativo para una malla 2-D. Para ello supondremos que $X(i)+$, $X(i)-$, $Y(i)+$, $Y(i)-$ son los enlaces derecha, izquierda, arriba y abajo y que i es la clase por la que se debe inyectar el paquete. El

algoritmo, además de los nodos actual y destino (X_{act} , Y_{act} , X_{dest} , Y_{dest}) acepta un parámetro adicional *Clase* para indicar en qué clase se encuentra el paquete.