

Arquitectura e Ingeniería de Computadores
Examen final (22/06/2006). Teoría.

Preguntas correspondientes al Primer Cuatrimestre

1. Consideremos un DLX básico en cuya etapa EX se ha añadido un multiplicador segmentado en 7 etapas, un sumador en punto flotante segmentado en 4 etapas y un divisor segmentado en 24 etapas. Los saltos se resuelven en la etapa DE y se aplica una política de saltos retardados. Responda razonadamente a las siguientes preguntas (no se otorgará puntuación por respuestas sin razonar):

a) ¿Es posible que se produzcan paradas por riesgos estructurales? En caso afirmativo indique cómo. En caso negativo indique por qué. (0,25 puntos)

b) Aplicando cortocircuitos (forwarding) convenientemente ¿es posible eliminar todas las paradas del pipeline producidas por dependencias LDE entre instrucciones aritméticas? (0,25 puntos)

c) Explique cómo se pueden detectar y tratar las dependencias EDE en la fase DE. (0,25 puntos)

d) Supongamos un programa con un 15% de instrucciones de salto donde el compilador puede rellenar el delay slot con una instrucción del programa en el 60% de los casos, y en la mitad de esos casos la instrucción realiza trabajo útil. Si al ejecutar el programa sólo existen riesgos de control, determine cuál es valor del CPI. (0,5 puntos)

2. Consideremos una memoria cache virtualmente accedida y físicamente marcada:

a) Explique el esquema de acceso a esta memoria cache, indicando los pasos que tienen lugar desde que el procesador genera la dirección virtual hasta que se produce el acceso a la cache. La explicación debe incluir un diagrama en el que se muestre cómo se genera la información necesaria para acceder a la cache. (0,75 puntos)

b) Si el tamaño de página virtual es de 8K bytes, explique razonadamente cuál es tamaño máximo que puede tener la cache, suponiendo que se utiliza emplazamiento asociativo por conjuntos con dos vías. (0,5 puntos)

Arquitectura e Ingeniería de Computadores. Examen Final (Problemas).22/06/2006

Problema correspondiente al Primer Cuatrimestre

1) En un DLX segmentado en cinco etapas, sin planificación dinámica, ejecutamos el siguiente fragmento de código:

```
addi r2,r0,#10
ld f5,0(r2)
multd f6,f5,f3
ld f4,0(r3)
multd f6,f4,f6
addi r2,r2,#1
beqz r2,loop
divd f5,f2,f6
divd f6,f2,f3
```

a) Construir el diagrama de tiempo (instrucciones – ciclos de reloj) y hallar los CPI (ciclos por instrucción) resultantes en la ejecución del código suponiendo que: (1 pto)

- * Un dato se puede escribir en un registro y leer su valor en el mismo ciclo.
- * Se dispone de lógica de cortocircuito (anticipación de operandos, forwarding).
- * Los saltos se resuelven en la etapa de decodificación (escriben el PC) y se espera a que se resuelva antes de lanzar la siguiente instrucción.
- * El registro r0 contiene siempre el valor 0.
- * La etapa de ejecución es multiciclo y, además de la habitual ALU de operaciones enteras, existen las siguientes unidades funcionales en punto flotante:

UF	Cantidad	Latencia	Segmentación
FP ADD	1	2	No
FP MUL	1	3	No
FP DIV	1	5	Sí

b) Señalar los cortocircuitos activados indicando entre qué ciclos se produce la activación. (0.5 ptos)

c) Repetir el diagrama de tiempo del primer apartado bajo las mismas condiciones excepto que ahora no se dispone de lógica de cortocircuito. (1 pto)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
addi r2, r0, #10	IF	DE	EX	ME	WB																				
ld f5, 0 (r2)		IF	DE	EX	ME	WB																			
multd f6, f5, f3			IF	DE	DE	EX1	EX2	EX3	ME	WB															
ld f4, 0(r3)				IF	IF	DE	EX	ME	WB																
multd f6, f4, f6						IF	DE	DE	EX1	EX2	EX3	ME	WB												
addi r2, r2, #1							IF	IF	DE	EX	ME	WB													
beqz r2, loop									IF	DE	DE	EX	ME	WB											
divd f5, f2, f6												IF	DE	EX1	EX2	EX3	EX4	EX5	ME	WB					
divd f6, f2, f3													IF	DE	EX1	EX2	EX3	EX4	EX5	ME	WB				

CPI=21/9

b) Los cortocircuitos son los señalados en el diagrama. Los pares de ciclos entre los que se produce la activación son: 3-4, 5-6, 8-9 (2) y 10-11.

c)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25		
addi r2, r0, #10	IF	DE	EX	ME	WB																						
ld f5, 0 (r2)		IF	DE	DE	DE	EX	ME	WB																			
multd f6, f5, f3			IF	IF	IF	DE	DE	DE	EX1	EX2	EX3	ME	WB														
ld f4, 0(r3)						IF	IF	IF	DE	EX	ME	WB															
multd f6, f4, f6									IF	DE	DE	DE	DE	EX1	EX2	EX3	ME	WB									
adddi r2, r2, #1										IF	IF	IF	IF	DE	EX	ME	WB										
beqz r2, loop														IF	DE	DE	DE	EX	ME	WB							
divd f5, f2, f6																		IF	DE	EX1	EX2	EX3	EX4	EX5	ME	WB	
divd f6, f2, f3																			IF	DE	EX1	EX2	EX3	EX4	EX5	ME	WB

CPI=27/9