

**Arquitectura e Ingeniería de Computadores. Examen Extraordinario (Teoría – parte primer cuatrimestre).  
17/09/2008**

**Instrucciones.-** Cada pregunta consta de cinco respuestas, y cada una de las respuestas puede ser cierta o falsa. Marque con un aspa las respuestas que considere ciertas y deje en blanco las que considere falsas. Si considera que alguna respuesta es ambigua y, por tanto, podría considerarse cierta o falsa en función de la interpretación, ponga una llamada y explique sus argumentos al dorso de la hoja. No se permite la utilización de calculadora.

**Puntuación.-** Pregunta con todas las respuestas acertadas: 1 punto. Pregunta con un fallo: 0,6 puntos. Pregunta dos fallos 0,2 puntos. Pregunta con más de dos fallos 0 puntos. La teoría del primer cuatrimestre supone la mitad de la nota del primer cuatrimestre. Tanto la nota de teoría como la de problemas se normalizarán para que el primer cuatrimestre tenga un peso del 65% en la nota final de la asignatura.

1. Supongamos la arquitectura básica del DLX (sin planificación dinámica), pero segmentado en siete etapas, donde el acceso a la memoria de instrucciones consume dos ciclos de reloj y el acceso a la memoria de datos consume dos ciclos de reloj. La máquina posee anticipación de operandos (forwarding). Las instrucciones de salto se resuelven en la etapa DE, con predicción estática de "salto no tomado". Además la etapa EX incluye un multiplicador, un sumador y un divisor de punto flotante, todos ellos segmentados, con tiempos de cálculo 7, 4 y 20 ciclos de reloj, respectivamente. Marque cuáles de las siguientes afirmaciones son correctas.

- ☒ a) La ejecución de una instrucción LOAD puede provocar un máximo de dos ciclos de reloj de penalización debido a las dependencias LDE.
- ☒ b) Si la mitad de los saltos se toman, entonces la penalización media por cada salto es 1.
- ☐ c) La presencia en un programa de dos instrucciones consecutivas de la forma:  
DIVD F2, F4, F6  
DIVD F6, F8, F10  
provocará penalización.
- ☐ d) En esta arquitectura es preciso detectar y resolver los riesgos EDL.
- ☒ e) El intervalo de iniciación del multiplicador es 1.

2. Supongamos una Unidad de Ejecución que utiliza el algoritmo de Tomasulo con especulación basada en el uso del buffer de reordenamiento (ROB). Marque cuáles de las siguientes afirmaciones son correctas.

- ☐ a) En la fase WRITE, los resultados generados por las UFs se escriben en las estaciones de reserva y en los registros.
- ☒ b) Cuando una instrucción genera una excepción se debe registrar en el ROB y la excepción será atendida cuando la instrucción finaliza (COMMIT).
- ☒ c) Cuando una instrucción realiza la fase ISSUE puede recibir sus operandos desde los registros o desde el ROB.
- ☐ d) Cuando una instrucción de salto incorrectamente predicha alcanza la cabecera del ROB es suficiente con eliminar la instrucción del ROB.
- ☒ e) Los datos que se transmiten a través del Bus de Datos Común (CDB) van acompañados por el número de entrada del ROB a la que se dirige.

3. Para ejecutar un cierto programa en un DLX segmentado en 5 etapas se han ejecutado  $10^{12}$  instrucciones en 1 hora. La frecuencia de reloj es de 1 GHz. Marque cuáles de las siguientes afirmaciones son correctas:

- ☐ a) La penalización media por instrucción es 3,6 ciclos
- ☐ b) La duración media de una instrucción es 1 ns.
- ☐ c) El rendimiento es 1000 MIPS
- ☒ d) Si modificamos la frecuencia de reloj, los ciclos de penalización media por instrucción no se alteran.
- ☒ e) Supongamos que introducimos un coprocesador matemático que ejecuta las operaciones en PF 6 veces más rápido que el DLX. Con el coprocesador instalado el tiempo de ejecución del programa se ha reducido a la mitad. Con estos datos se puede afirmar que en el DLX sin coprocesador los cálculos en PF consumen un 60% del tiempo total de ejecución.

**EXAMEN DE ARQUITECTURA E INGENIERÍA DE COMPUTADORES**  
**SEPTIEMBRE DE 2008. EXAMEN FINAL**  
**PARTE CORRESPONDIENTE AL PRIMER PARCIAL**

Sea un procesador segmentado con planificación dinámica mediante el algoritmo de Tomasulo

- Los datos que se escriben en la etapa de escritura no se pueden usar para la ejecución de una instrucción hasta el ciclo siguiente
- Las instrucciones SGTI, BNEZ y NOP tienen tratamiento de instrucciones enteras.
- Hay un solo bus de datos común (CDB)
- La estructura del procesador tiene las siguientes características:

UF	CANTIDAD	LATENCIA	SEGMENTADA
FP ADD	1	2	SI
FP DIV	1	5	SI
FP MUL	1	3	SI
INT ALU	1	1	SI
LOAD/STORE	1	2	SI

ESTACIONES RESERVA	CANTIDAD
FP ADD	2
FP DIV	2
FP MUL	2
INT ALU	2
LOAD	1
STORE	1

Dado el siguiente fragmento de programa:

```

      ADDI R1,R0,#DIR
      ADDI R3,R0,#100
LOOP  LD F0, 400(R1)
      LD F2, 0(R1)
      LD F8, 500(R1)
      DIVD F4,F2,F0
      MULF F4,F8,F4
      SD 300(R1), F4
      SUBI R1, R1, #8
      SUBI R3,R3,,#1
      SGTI R5,R3,DONE
      BNEZ R5,LOOP
      NOP

```

- a) Representar el diagrama instrucción – tiempo para la primera iteración, indicando en cada caso el tipo de parada que se produce (2 puntos)
- b) Suponiendo que se le añade especulación, Representar el diagrama instrucción – tiempo para la primera iteración, indicando en cada caso el tipo de parada que se produce (2 puntos)
- c) Representar el diagrama instrucción – tiempo para la primera iteración suponiendo un procesador sin especulación (como el del apartado a) que dispone de todas las estaciones de reserva que se necesiten (1 punto)

## SOLUCIONES

a)

R3 contiene el número de elementos del array que se quieren procesar y que son 100

R1 contiene la dirección base

La instrucción sgti set r5:=1 si r3≥done

	PROGRAMA	ISSUE	EJECUCIÓN	WRITE
1	ADDI R1,R0,#DIR	1	2	3
2	ADDI R3,R0,#100	2	3	4
3	LD F0, 400(R1)	3	4-5	6
4	LD F2, 0(R1)	7 ESTRUCTURAL SIN ER DE LOAD	8-9	10
5	LD F8, 500(R1)	11 ESTRUCTURAL SIN ER DE LOAD	12-13	14
6	DIVD F4,F2,F0	12	13-17	18
7	MULD F4,F8,F4	13	19-20-21 LDE CON 6	22
8	SD 300(R1), F4	14	23-24 LDE CON 7	
9	SUBI R1, R1, #8	15	16	17
10	SUBI R3,R3,#1	16	17	19 BCD OCUPADO POR 6
11	SGTI R5,R3,DONE	18 ESTRUCTURAL SIN ER ENTEROS	20 LDE CON 10	21
12	BNEZ R5,LOOP	20 ESTRUCTURAL SIN ER ENTEROS	22 LDE CON 11	23 NO USA BCD
13	NOP	22 ESTRUCTURAL SIN ER ENTEROS	23	24

b.- Recordar que con ROB no existen estaciones de reserva de store sino que se incluyen en el ROB. De manera que en el ROB se pueden realizar dos escrituras simultáneas, y en las instrucciones de store con dependencias del tipo LDE el dato se envía directamente al ROB. Esta es la razón por la que las instrucciones de DIVD y store escriben en el ROB en el mismo ciclo de reloj. La información y la etiqueta que se envía es la misma para ambas.

	PROGRAMA	ISSUE	EJECUCIÓN	WRITE	
1	ADDI R1,R0,#DIR	1	2	3	4
2	ADDI R3,R0,#100	2	3	4	5
3	LD F0, 400(R1)	3	4-5	6	7
4	LD F2, 0(R1)	7 ESTRUCTURAL SIN ER DE LOAD	8-9	10	11
5	LD F8, 500(R1)	11 ESTRUCTURAL SIN ER DE LOAD	12-13	14	15
6	DIVD F4,F2,F0	12	13-17	18	19
7	MULD F4,F8,F4	13	19-20-21 LDE CON 6	22	23
8	SD 300(R1), F4	14		22 NO HAY BCD	24
9	SUBI R1, R1, #8	15	16	17	25
10	SUBI R3,R3,#1	16	17	19 BCD OCUPADO POR 6	26
11	SGTI R5,R3,DONE	18 ESTRUCTURAL SIN ER ENTEROS	20 LDE CON 10	21	27
12	BNEZ R5,LOOP	20 ESTRUCTURAL SIN ER ENTEROS	22 LDE CON 11	23 NO USA BCD	28
13	NOP	22 ESTRUCTURAL SIN ER ENTEROS	23	24	29

C

	PROGRAMA	ISSUE	EJECUCIÓN	WRITE
1	ADDI R1,R0,#DIR	1	2	3
2	ADDI R3,R0,#100	2	3	4
3	LD F0, 400(R1)	3	4-5	6
4	LD F2, 0(R1)	4	5-6	7
5	LD F8, 500(R1)	5	6-7	8
6	DIVD F4,F2,F0	6	8-12 LDE CON 4	13
7	MULD F4,F8,F4	7	14-15-16 LDE CON 6	17
8	SD 300(R1), F4	8	18-19 LDE CON 7	
9	SUBI R1, R1, #8	9	10	11
10	SUBI R3,R3,#1	10	11	12
11	SGTI R5,R3,DONE	11	13 LDE CON 10	14
12	BNEZ R5,LOOP	12	15 LDE CON 11	16 NO USA BCD
13	NOP	13	14	15