

**EXAMEN DE ARQUITECTURA DE COMPUTADORES. La Rábida, 22 de Junio de 2018**

Apellidos ..... Nombre .....

Valor de cada: Respuesta correcta + 0.1  
 Respuesta incorrecta - 0.05  
 No respuesta - 0.05

TIEMPO REALIZACIÓN EXAMEN: 45 minutos

Espacio para respuestas

1	<b>Sea una máquina A que ejecuta un programa en 20 segundos. Si se dice que una máquina B es el 10% más lenta la máquina A:</b>		A
	A) La máquina B tarda 2 segundos más en ejecutar el mismo programa	C) La máquina B tarda 30 segundos en ejecutar el mismo programa	
	B) La máquina B tarda 1 segundo más en ejecutar el mismo programa	D) La máquina B tarda menos en ejecutar el mismo programa	
2	<b>Sea un sistema computador al que se le aplica la siguiente mejora: se modifica la CPU, que se utiliza el 60%, haciendo que ésta sea 2 veces más rápida, se puede afirmar que:</b>		B
	A) La mejora aplicada no merece la pena	C) La aceleración global es de 0,76	
	B) La aceleración global es de 1,43	D) La aceleración global es de 2	
3	<b>Sea un sistema computador sobre el que se aplica una mejora que resulta en una aceleración global de 1. Esto implica que:</b>		C
	A) El sistema mejorado es el 100% más rápido que antes	C) No se ha conseguido reducir el tiempo de ejecución	
	B) La mejora se utiliza el 100% del tiempo total	D) El tiempo de ejecución es 1 sg	
4	<b>Sea un sistema con arquitectura serie Von Neumann que emplea 18 ciclos en ejecutar 3 instrucciones:</b>		D
	A) Significa que tiene una frecuencia = 6 MHz	C) Significa que tiene un rendimiento del 18%	
	B) Significa que tiene un ciclo de reloj = 18 ns	D) Significa que tiene un $CPI_{medio} = 6$	
5	<b>Sea una máquina con arquitectura DLX capaz de ejecutar 20 MFLOPS y frecuencia de procesamiento = 2 GHz</b>		A
	A) Se necesitan 100 ciclos por FLOP	C) Se necesitan 400 ciclos por FLOP	
	B) Se necesitan 20 ciclos por FLOP	D) Se necesita 1 ciclo por FLOP	
6	<b>Sea una memoria caché organizada en conjuntos, con 1 bloque/conjunto:</b>		B
	A) El bloque tendrá tantas palabras como contiene la Memoria Principal	C) Equivale a tener una función de correspondencia totalmente asociativa	
	B) Equivale a tener una función de correspondencia directa	D) Un conjunto nunca puede tener un único bloque	
7	<b>De los algoritmos de reemplazo utilizados en la memoria caché, el que penaliza el bloque que acaba de entrar es:</b>		C
	A) El algoritmo FIFO	C) El algoritmo LFU	
	B) El algoritmo LRU	D) Ninguno penaliza el bloque que acaba de entrar	
8	<b>En relación a los tipos de fallo de caché, indica cuál de las siguientes afirmaciones es cierta:</b>		D
	A) Los fallos de capacidad sólo se dan en correspondencia asociativa por conjuntos	C) La correspondencia directa no presenta fallos de conflicto	
	B) Los fallos de conflicto sólo se dan en correspondencia totalmente asociativa	D) Pueden existir fallos forzosos aun cuando la caché esté llena	
9	<b>En relación al tamaño de caché:</b>		A
	A) Los fallos forzosos son independientes del tamaño	C) Los fallos de conflicto aumentan con la asociatividad	
	B) Los fallos de capacidad aumentan con el tamaño	D) Ninguno de los tipos de fallos depende del tamaño	
10	<b>Sea una memoria caché de 2 niveles, sabiendo que de 1000 referencias a memoria hay 30 fallos en la caché de primer nivel y 9 fallos en la de segundo nivel. ¿Cuáles son las respectivas frecuencias locales de fallos?</b>		B
	A) 3% para L1 y L2	C) 33% para L1 y 30% para L2	
	B) 3% para L1 y 30% para L2	D) 0.3% para L1 y 3% para L2	
11	<b>Atendiendo a la arquitectura serie Von Neumann estudiada, si el registro RF tiene un tamaño de 4 bits:</b>		C
	A) El firmware tendrá como máximo 16 microprogramas	C) Los microprogramas tendrán un máximo de 16 instrucciones cada uno	
	B) Cualquier instrucción se ejecutará en 4 ciclos	D) Ninguna de las afirmaciones anteriores es correcta	
12	<b>Sea un sistema con Unidad de Control Microprogramada, la Memoria de Control:</b>		D
	A) Este tipo de sistemas no tienen memoria de control	C) Contiene la dirección de la siguiente instrucción	
	B) Contiene el contenido de los registros PC y SR	D) Contiene el valor de las señales de control para todos los microprogramas asociados a la máquina	
13	<b>Sea un sistema con Unidad de Control Microprogramada con secuenciamiento implícito, y etapa traductora ROM de 256 posiciones x 10 bits:</b>		A
	A) El sistema podrá ejecutar hasta 256 instrucciones	C) El ancho de palabra de memoria de control son 10 bits	
	B) El sistema podrá ejecutar hasta 8 instrucciones	D) La memoria de control tiene 10 posiciones	
14	<b>Sea un sistema con capacidad de memoria de 64 Kposiciones, si durante el arranque del sistema se ha predefinido que PC = F000h:</b>		B
	A) La pila está implementada a partir de la dirección F000h	C) Existe un error en la definición del valor de PC	
	B) El programa cargador se ubica a partir de la dirección F000h	D) Ninguna de las afirmaciones anteriores es correcta	

15	<b>De los siguientes eventos, indica el que corresponde a interrupción interna no programada:</b>			C
	A) INT 21 h B) Interrupción de periférico TIMER i8255	C) DIV .2,.3; con R3 = 0 provocando TRAP D) Corte de suministro eléctrico por batería descargada		
16	<b>Sea la arquitectura DLX estudiada, la ejecución de la instrucción (carga media palabra con extensión de signo) LH R1, 10(R2) con M(10+R2) = 5 implica:</b>			D
	A) Completar con 1s los 16 bits más significativos de R1 B) Completar con el valor 5 la parte alta de R1	C) Completar con el valor 5 la parte alta de R2 D) Completar con 0s los 16 bits más significativos de R1		
17	<b>En base a la arquitectura DLX estudiada, ¿qué tipo de registros nos garantizarían mayor precisión a la hora de representar el número <math>\pi = 3,141592653589.....</math>?</b>			A
	A) Los registros de coma flotante combinados en doble precisión ( $F_i - F_{i+1}$ ) B) Dos registros de coma fija ( $R_i - R_{i+1}$ )	C) Un registro de coma flotante de simple precisión ( $F_i$ ) D) El registro R0		
18	<b>Sea la secuencia de instrucciones: ADD R1, R2, R3 y LW R2, 10(R4) implementada sobre la arquitectura escalar DLX estudiada:</b>			B
	A) Implica detención porque supone riesgo de tipo RAW B) No supone riesgo	C) Implica detención porque supone riesgo de tipo WAR D) Implica detención porque supone riesgo de tipo WAW		
19	<b>Sea la secuencia de instrucciones: ADD R1, R2, R3 y LW R2, 10(R4) implementada sobre una arquitectura DLX súper-escalar de dos cauces:</b>			C
	A) Las dos instrucciones deben ir por el mismo cauce debido a la dependencia de datos que tienen B) No es posible ejecutar las instrucciones en una arquitectura súper-escalar	C) Cada instrucción puede ir por un cauce diferente ya que no existe dependencia de datos D) Para ejecutar las instrucciones es necesario que la arquitectura sea vectorial		
20	<b>Sea el extracto de código:</b>	<b>ADDF F4, F0, F2 si se convierte en: ADDF F4, F0, F2 SF 10(R2), F4 SUBI R1, R1, #4 BNEZ R1, loop</b>	<b>estamos:</b>	D
	A) Aumentando el paralelismo por planificación de traza B) Aumentando el paralelismo por desenrollado de bucle	C) Aumentando el paralelismo por técnica hardware D) Aumentando el paralelismo por segmentación software		
21	<b>Sean la instrucción vectorial ADDV V1, V2, V3 con longitud de vector = 64 elementos, tiempo de arranque = 6 ciclos y tiempo de iniciación = 1 ciclo, ¿cuántos ciclos se necesitan como mínimo para obtener los primeros 32 elementos del resultado?</b>			A
	A) 38 ciclos B) 224 ciclos	C) 7 ciclos D) 71 ciclos		
22	<b>Sea un sistema con la arquitectura DLX vectorial estudiada y memoria organizada en bancos de memoria (4 bancos), si la CPU solicita la dirección 32 (y siguientes), ésta se encuentra:</b>			B
	A) En el banco 4 B) En el banco 0	C) En el banco 1 D) En el banco 2		
23	<b>Sea un vector de 200 elementos sobre el que se realiza una operación vectorial en una arquitectura con MVL = 64:</b>			C
	A) El vector se divide en 3 secciones de 64 elementos B) El vector se opera en su totalidad sin seccionamiento	C) El vector se divide en 3 secciones de 64 elementos y otra de 8 elementos D) El vector se divide en 2 secciones de 64 elementos		
24	<b>Sean las sentencias de un bucle S1: B(i) = A(i-1)*K y S2: C(i) = B(i) + A(i), con i = 1...64:</b>			D
	A) La sentencia S1 es vectorizable B) Las dos sentencias son vectorizables	C) Ninguna sentencia es vectorizable D) La sentencia S2 es vectorizable		
25	<b>Sean las sentencias de un bucle S1: B(i) = A(i)/K y S2: C(i) = B(i) + A(i), con i = 1...64:</b>			A
	A) Existe riesgo de tipo RAW en S2 respecto a S1 B) Existe riesgo de tipo RAR en S2 respecto a S1	C) Existe riesgo de tipo WAR en S2 respecto a S1 D) Existe riesgo de tipo WAW en S2 respecto a S1		
26	<b>Atendiendo al esquema de la arquitectura Harvard estudiada:</b>			B
	A) Presenta una única memoria de instrucciones y de datos B) Diferencia una memoria de instrucciones y otra de datos	C) Presenta dos unidades operativas D) Presenta dos unidades de control		
27	<b>En base a la clasificación de Flynn sobre los sistemas computadores, en una arquitectura de tipo MISD:</b>			C
	A) El flujo de instrucciones es compartido B) Existen tantos flujos de datos como procesadores	C) La unidad de memoria es compartida D) Ninguna de las afirmaciones anteriores es correcta		
28	<b>Sea el polinomio <math>R(x) = x^2 + bx</math>, si se desea calcular empleando una arquitectura súper-escalar tipo array sistólico:</b>			D
	A) Se necesitan 1 ciclo y 2 elementos de proceso B) Se necesitan 3 ciclos y 3 elementos de proceso	C) Se necesitan 1 ciclo y 1 elemento de proceso D) Se necesitan 2 ciclos y 3 elementos de proceso		
29	<b>Considerando una arquitectura superescalar multihebra con entrelazado:</b>			A
	A) Después de cada ciclo, conmuta a otra hebra B) Después de cada detención, conmuta a otra hebra	C) Conmuta a otra hebra cuando finaliza con la anterior D) Debe tener tantos cauces como hebras		
30	<b>Dado un terminal de smartphone con procesador Exynos a 2,3 GHz y 6 GB de RAM:</b>			B
	A) Se incluye dentro de los sistemas denominados minicomputadores B) Se incluye dentro de los sistemas denominados microcomputadores	C) Se incluye dentro de los sistemas denominados computadores de tamaño medio D) Se incluye dentro de los sistemas denominados supercomputadores		

## EXAMEN DE ARQUITECTURA DE COMPUTADORES

La Rábida, 22 de Junio de 2017

**PROBLEMA 1.** (2,5 ptos.). Un sistema computador (sin memoria virtual) tiene una memoria principal con ancho de palabra de 16 bits y formada por 16 bloques de 256 Kpalabras/bloque. La memoria caché la componen 4 bloques.

- Justificar e indicar la **capacidad** total (en bytes) que tiene la **Memoria Principal** y la **Memoria Caché**.
- Después de haber estado la memoria caché “vacía”, se encuentran en el *instante 1* los bloques de Memoria Principal B0, B7 y B6, que en base a la función de correspondencia empleada, se ubican respectivamente en los bloques 0, 3 y 2 de Memoria Caché. Justificar la **función de correspondencia** empleada y el **formato de la dirección** que define la unidad central de proceso, especificando cada uno de los campos en los que se divide.
- Teniendo en cuenta que los bloques B0, B7, B6 han sido leídos ordenadamente en todas sus direcciones una sola vez, indicar cuántos fallos y de qué tipo se han producido hasta el instante 1. Indicar las direcciones que producen dichos fallos.

A continuación en el *instante 2*, la CPU lee la secuencia de direcciones de memoria: 0, 1,  $10 \times 2^{18}$ ,  $15 \times 2^{18}$ ,  $(7 \times 2^{18} + 2)$ ,  $2^{18}$ ,  $(2^{18} + 4)$  y  $(7 \times 2^{18} + 200)$ , según el orden marcado en la misma. Mostrar el contenido de la memoria caché en el *instante 2*.

- Identificar cuántos fallos de caché se han producido y de qué tipo en el *instante 2*. ¿Y cuántos aciertos?
- Calcular justificadamente la frecuencia de uso de todos los bloques que han pasado por la caché (tanto los que se encuentran en la memoria caché en el instante 2 como los que han salido fuera), teniendo en cuenta el número total de referencias que han tenido desde que fueron introducidos en caché por primera vez.
- A continuación, la CPU genera la dirección 10 1111 1111 1111 1111. Justificar la palabra y los bloques de M. Principal y de M. Caché al que corresponde.

**PROBLEMA 2.** (2,5 ptos.). Sea el sistema computador serie basado en la arquitectura Von Neumann, representado en la figura, y para la secuencia de instrucciones siguiente:

Dirección de Memoria (en hexadecimal)	Instrucción (en ensamblador)
B0000h	MOVE .2, 00120h(++.1)
B0001h	ADD 0000Ah, .2
B0002h	RET
B0003h	NOP

1. Mostrar los formatos de las instrucciones.
2. Definir la secuencia de operaciones elementales y el solapamiento posible de las mismas. Especificar la secuencia según el flujo marcado por el programa propuesto.
3. Definir el valor de las señales de control correspondiente a la secuencia de operaciones elementales:  $D \leftarrow 0000Ah$ ,  $RM \leftarrow R2$  y  $M(D) \leftarrow RM$ , realizadas en el menor tiempo posible. Mostrar únicamente el valor de las señales que intervienen directamente en dichas operaciones.

Teniendo en cuenta:

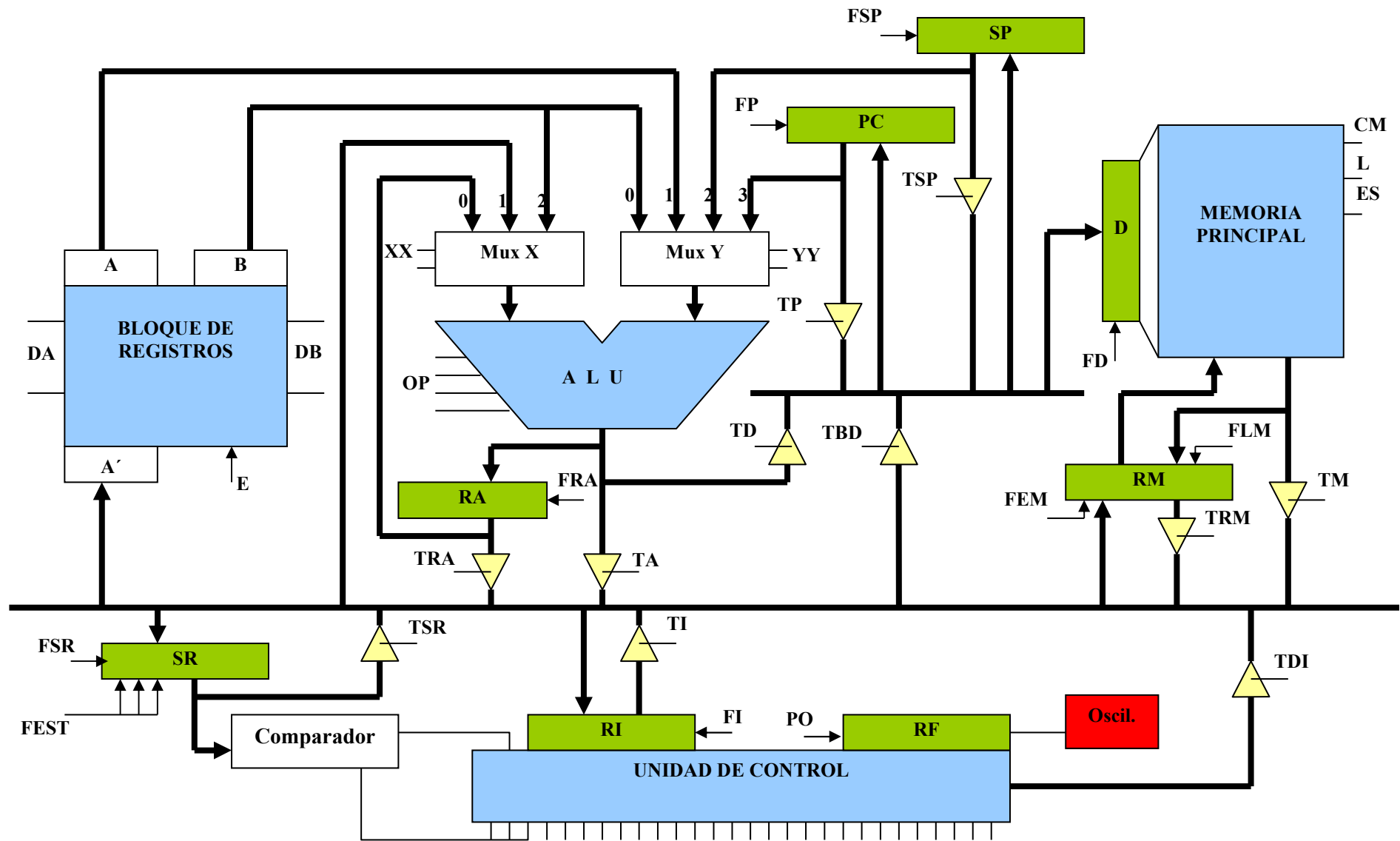
- a) PRIMERA INSTRUCCIÓN: La instrucción de transferencia MOVE, con direccionamiento directo absoluto a registro para el primer operando y directo relativo a registro índice con auto-preincremento para el segundo operando.
- b) SEGUNDA INSTRUCCIÓN: La instrucción aritmética ADD, con modo de direccionamiento directo absoluto a memoria para el primer operando y directo absoluto a registro para el segundo operando.
- c) TERCERA INSTRUCCIÓN: La instrucción RET de retorno de subrutina. El valor que se recupera de la pila no está definido.
- d) CUARTA INSTRUCCIÓN: La instrucción de no operación NOP.
- e) Las operaciones con la unidad aritmético-lógica se realizan en un periodo.
- f) Los cálculos de las direcciones de los operandos en memoria especificados con direccionamiento relativo, se realizarán de manera automática durante el periodo de decodificación.
- g) El resultado de una operación o el operando destino va a ser siempre el especificado en primer lugar en la instrucción de ensamblador.
- h) Las operaciones de L/E en memoria se realizan durante dos periodos.
- i) La pila, que se implementa en el sistema con el registro SP (puntero de pila), funciona de la siguiente forma: 1) La pila crece hacia direcciones decrecientes. 2) El registro puntero de pila indica la dirección de memoria principal en la que se introdujo el último dato en la pila.

- j) Para cuando hay que salvar el contenido de SR y el de PC, siempre se almacenará primero el de SR.

**PROBLEMA 3.** (2 ptos.). El siguiente fragmento de código se ejecuta en un procesador con arquitectura DLX. Las latencias (en ciclos de reloj) de las unidades funcionales son las siguientes: Sumador/restador entero: 1; Sumador/restador flotante: 2; y Multiplicador/Divisor: 4 (las unidades funcionales para operaciones en coma flotante **no** están **segmentadas**).

```
LF F2, 20(R1)
LF F3, 10(R2)
ADDI R1, R1, #4
SUBI R2, R2, #4
FMULT F6, F7, F8
FDIV F1, F2, F3
FADD F1, F4, F5
SF 30(R0), F1
```

- a) Suponiendo que la **memoria** tiene un **único puerto de acceso** y hay **adelantamiento generalizado**, indicar el estado de cada instrucción durante los ciclos de ejecución del código; calcular el número de ciclos necesarios para ejecutar este código; indicar si existen bloqueos en la cadena y a qué se deben, indicar los adelantamientos que se realizan y entre qué etapas actúan.
- b) Justificar qué modificaciones hardware y técnicas software se podrán aplicar para aumentar el paralelismo a nivel de instrucción. A continuación, con estas modificaciones indicar el estado de cada instrucción durante los ciclos de ejecución del código; calcular el número de ciclos necesarios para ejecutar este código; indicar si existen bloqueos en la cadena y a qué se deben e identificar los adelantamientos así como los datos que se adelantan y entre qué etapas actúan. Considerar las mismas condiciones que en el apartado anterior (**memoria de un único puerto de acceso y adelantamiento generalizado**).



<b>Etap</b>	<b>Instrucción ALU</b>	<b>Carga/Almacenamiento</b>	<b>Salto/Bifurcación</b>
<b>IF</b>	$IR \leftarrow M[PC]$ $PC \leftarrow PC+4$	$IR \leftarrow M[PC]$ $PC \leftarrow PC+4$	$IR \leftarrow M[PC]$ $PC \leftarrow PC+4$
<b>ID</b>	$A \leftarrow Rs1; B \leftarrow Rs2$ $PC1 \leftarrow PC$ $IR1 \leftarrow IR$	$A \leftarrow Rs1; B \leftarrow Rs2$ $PC1 \leftarrow PC$ $IR1 \leftarrow IR$	$A \leftarrow Rs1; B \leftarrow Rs2$ $PC1 \leftarrow PC$ $IR1 \leftarrow IR$
<b>EX</b>	$ALU_{output} \leftarrow A \text{ op } (B \text{ or } (IR1_{16})^{16}###IR1_{16..31})$	$MAR \leftarrow A+(IR1_{16})^{16}###IR1_{16..31}$ $SMDR \leftarrow B$	<div> <b>Salto</b>  <math>ALU_{output} \leftarrow PC1+IR1_{16})^{16}###IR1_{16..31})</math>  <math>Cond \leftarrow (Rs1 \text{ op } 0);</math> </div> <div> <b>Bifurcación</b>  <math>ALU_{output} \leftarrow PC1+ IR1_{26})^6###IR1_{6..31})</math> </div>
<b>MEM</b>	$ALU_{output1} \leftarrow ALU_{output}$	$LMDR \leftarrow M[MAR]$ o $M[MAR] \leftarrow SMDR$	If (cond): $PC \leftarrow ALU_{output}$
<b>WB</b>	$Rd \leftarrow ALU_{output1}$	$Rd \leftarrow LMDR$	

**Tabla.** Eventos en cada etapa de la segmentación de DLX diferenciados según el tipo de instrucción.

**Apellidos:** ..... **Nombre:** .....

[illegible]

## 1) FORMATOS


### 3) SEÑALES DE CONTROL

A full page of graph paper with a uniform grid of small squares. The grid consists of 20 columns and 20 rows, creating a total of 400 squares. The lines are thin and black, forming a continuous pattern across the entire page.



Apellidos:

Nombre:

Apartado a)	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	26	27	28	29	30
LF F2, 20 (R1)																														
LF F3, 10 (R2)																														
ADDI R1, R1, #4																														
SUBI R2, R2, #4																														
FMULT F6, F7, F8																														
FDIV F1, F2, F3																														
FADD F1, F4, F5																														
SF 30 (R0), F1																														
Detenciones							Detenciones										Adelantamientos							Adelantamientos						

Apartado b)	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	26	27	28	29	30
Detenciones							Detenciones										Adelantamientos							Adelantamientos						