

Examen Parcial de FCO – Temas 5 al 7

21 de enero de 2011

APELLIDOS: _____ NOMBRE: _____

DNI: _____

FIRMA: _____

Normativa:

- La duración del examen es de 1h30min.
- Escriba el nombre y los apellidos en y firme en TODAS las hojas.
- DEBE responder en el espacio asignado.
- No se permiten calculadoras ni apuntes.
- Debe permanecer en silencio durante la realización del examen.
- No se puede abandonar el examen hasta que el profesor lo indique.
- Debe tener una identificación en la mesa a la vista del profesor (DNI, carnet UPV, tarjeta residente, etc.)

1.- (1 punto) Dados los números enteros $A=1001001$ y $B=0111011$ representados en complemento a dos, realice las siguientes operaciones, **siempre en complemento a dos y con 7 bits**. No olvide indicar si se produce desbordamiento y justifique su respuesta.

A+B

acarreo	111011
A	1001001
B	+ 0111011
	<u>10000100</u>

Como el último y el penúltimo bit de acarreo son los dos 1, $1 \text{ xor } 1 = 0$, por lo que no hay desbordamiento y el resultado es: 0000100

B-A

$B-A = B + (-A) = B + C'2(A)$
 $C'2(A) = C'2(1001001) = 0110111$

acarreo	111111
B	0111011
-A	+ 0110111
	<u>01110010</u>

Como el último acarreo es 0 y el penúltimo es 1, $0 \text{ xor } 1 = 1$, lo que indica que hay desbordamiento y no hay resultado.

2.- (1 punto) Dados los siguientes números enteros, represéntelos en el criterio indicado, utilizando siempre 7 bits

Entero	Signo y magnitud	Complemento a dos	Exceso 63
+45	0101101	0101101	1101100
-45	1101101	1010011	0010010

Puede realizar aquí las operaciones necesarias.

Para representar las cantidades deseadas en Signo y Magnitud, sólo hay que representar la magnitud o valor absoluto en binario natural con 6 bits, y añadir, a la izquierda, el bit de signo (0 positivo; 1 negativo)

Para representar las cantidades deseadas en Complemento a 2, se representa en primer lugar la magnitud en binario natural, y se le añade, a la izquierda, un cero. Si la cantidad a representar es positiva, el proceso a finalizado. Si por el contrario es negativa, es necesario aplicarle el complemento a 2 a la secuencia de bits.

Para representar las cantidades deseadas en exceso 63, simplemente sumamos 63 a dichas cantidades y representamos en binario natural.

Examen Parcial de FCO – Temas 5 al 7
21 de Enero de 2011

APELLIDOS: _____

NOMBRE: _____

DNI: _____

FIRMA: _____

3.- (1 punto)) Represente el número -10,875 en el formato de simple precisión de IEEE754. Detalle todos los pasos realizados y exprese el resultado final en hexadecimal.

Convertimos a binario

$$10 = 1010_2$$

$$0,875 \times 2 = 1,75 \rightarrow 0,75 \times 2 = 1,5 \rightarrow 0,5 \times 2 = 1,0$$

$$-10,875 = -1010,111_2$$

$$\text{Representamos en coma flotante: } -1010,111 = -1010,111 \times 2^0$$

$$\text{Normalizamos la mantisa: } 1010,111 \times 2^0 = 1,010111 \times 2^3$$

$$\text{Representamos el exponente en exceso } 127 = 3 + 127 = 00000011 + 01111111 = 10000010$$

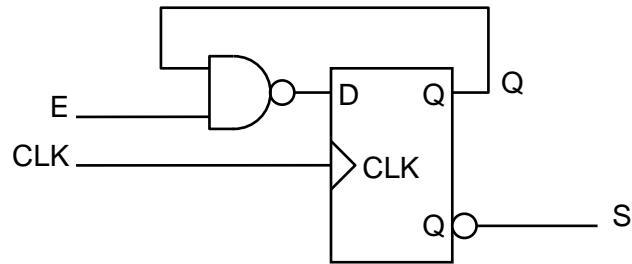
$$\text{Signo} = 1 \text{ (negativo)}$$

Todo junto de la forma signo exp mantisa con bit implícito:

$$1 \quad 10000010 \quad 010111000000000000000000$$

En hexadecimal: 1100 0001 0010 1110 0000 0000 0000 0000
0xC12E0000

4.- (1,5 puntos) Dado el circuito de la figura siguiente



Obtenga su diagrama de estados y calcule (a partir del diagrama de estados obtenido) la secuencia de valores de la salida S para la secuencia de valores en la entrada E: 0 – 0 – 1 – 1 – 0 teniendo en cuenta que el estado inicial del sistema es $Q = 0$.

Indique los pasos seguidos y detalle todas las tablas intermedias obtenidas para llegar al resultado.

Tablas de excitación y salida:

En primer lugar se obtienen las tablas de verdad de las funciones de excitación y salida del sistema, aplicando las combinaciones de valores en las entradas.

Función de excitación

Q	E	D
0	0	1
0	1	1
1	0	1
1	1	0

Función de salida

Q	S
0	1
1	0

La función de excitación coincide con el estado siguiente ya que se trata de biestables D. Es sencillo obtener la tabla de estados (con los estados codificados, pues disponemos del valor de la variable de estado) sin más que transportar la información de los estados siguientes (en función de la entrada E) y salida para cada estado a una única fila de la tabla de estados.

Tabla de estados:

	Estado siguiente, Q(t+1)		
Estado actual	Entrada E		Salida
Q(t)	E = 0	E = 1	S
0	1	1	1
1	1	0	0

Otra posibilidad:

Examen Parcial de FCO – Temas 5 al 7

21 de Enero de 2011

APELLIDOS: _____

NOMBRE: _____

DNI: _____

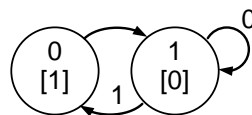
FIRMA: _____

Q(t)	E	Q(t+1)	S(t)
0	0	1	1
0	1	1	1
1	0	1	0
1	1	0	0

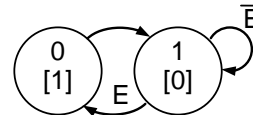
Diagrama de estados:

A partir de la tabla anterior, y utilizando los valores de estado como nombres para los estados es sencillo derivar el diagrama de estados del sistema. A continuación se muestran los dos posibles diagramas de estado, utilizando vectores o ecuaciones booleanas.

Utilizando vectores



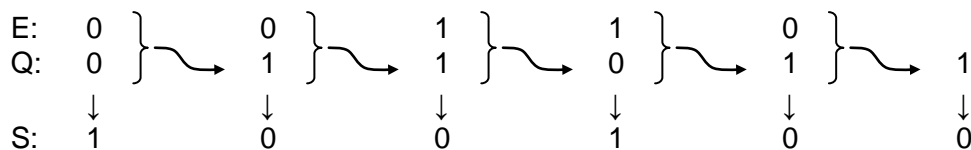
Utilizando ecuaciones



Salida del sistema para la secuencia de valores 0 – 0 – 1 – 1 – 0 en la entrada E:

Por último, a partir del estado inicial 0 (salida $S = 1$), se recorre el diagrama de estados empleando la secuencia de valores de la entrada E (0 – 0 – 1 – 1 – 0)

Este recorrido se puede representar, por ejemplo, en una tabla en la que se incluye el estado para facilitar su comprensión. A partir de un valor en el estado Q y un valor en la entrada E se determina el próximo valor del estado Q. Y cada valor de estado Q determina un valor en la salida S.



Con lo que se determina que la secuencia de valores de la salida S es:
1 (salida en el estado inicial cuando $Q=0$) – 0 – 0 – 1 – 0 – 0

5.- (1,5 puntos) Obtenga el diagrama de estados de un autómata de Moore que responda a las siguientes especificaciones:

- Entradas FP (freno pisado) y AA (activar sistema abs), ambas activas a nivel alto.
- Salida S, activa a nivel alto.
- Mientras ninguna de las entradas está activada, la salida debe ser 0 (desactivada).
- Mientras la entrada FP esté activada pero no AA, la salida debe ser 1 (activada).
- Mientras esté activada la entrada AA (independientemente de la activación de FP) la salida debe activarse y desactivarse de forma alternada, en ciclos de reloj consecutivos. El primer ciclo de reloj en el que la entrada AA esté activada debe producir la activación de la salida, el segundo su desactivación, el tercero su activación, etc.

Hay varias formas de abordar el problema, y cada una da lugar a un diagrama de estados diferente. Dado que no es necesario obtener un diagrama de estados con el número mínimo de estados, los diferentes diagramas son (todos ellos) representaciones válidas del sistema.

A - Solución con cuatro estados:

En primer lugar, a partir de una interpretación más o menos literal del enunciado se puede pensar que este sistema es un autómata con tres “modos” de funcionamiento:

- 1. Un modo de reposo en el que no hay ningún tipo de frenado, con la salida S desactivada. Este modo se puede representar por un único estado “reposo”.*
- 2. Un modo de funcionamiento con el tipo de frenado normal (sin abs), con la salida S activada. Este modo se puede representar por un único estado “frenado normal”*
- 3. Un modo de funcionamiento con el tipo de frenado abs, con la salida S alternando entre los valores 0 y 1 ciclo a ciclo de reloj. Necesitamos dos estados para representar este modo de funcionamiento, el primero con la salida activada y el segundo con la salida desactivada.*

Diagrama de estados:

Esta manera de entender el autómata da lugar al diagrama de estados (con cuatro estados) siguiente:

Examen Parcial de FCO – Temas 5 al 7

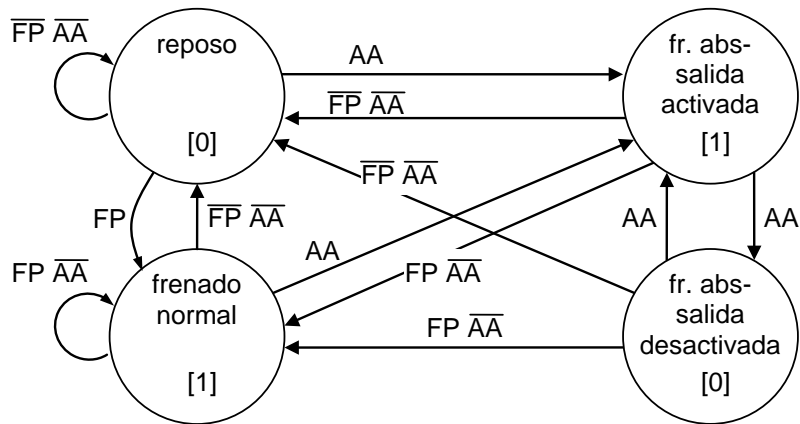
21 de Enero de 2011

APELLIDOS: _____

NOMBRE: _____

DNI: _____

FIRMA: _____

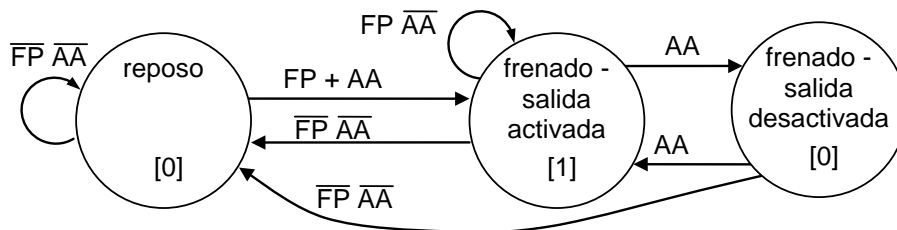


B - Solución con tres estados:

En lugar de pensar en el sistema en los términos anteriores se puede plantear el diagrama con tres estados diferentes: reposo, frenado-con salida activada (sea por frenada normal o por abs) y frenado-con salida desactivada (solamente por abs).

Diagrama de estados:

En este caso el diagrama de estados queda:

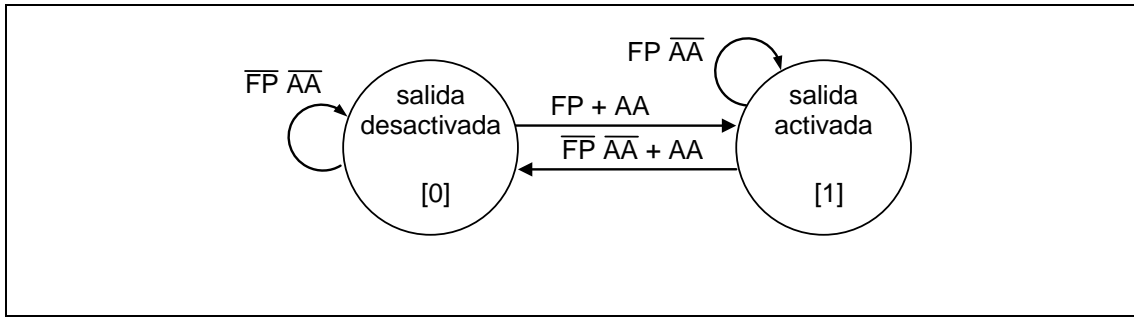


C - Solución con dos estados:

Finalmente, también se puede pensar que el autómata sólo tiene dos estados, uno en el que la salida está desactivada y otro en el que la salida está activada. Respecto de la solución con cuatro estados, el primero correspondería tanto al estado de reposo y como al de frenado abs-salida desactivada, y el segundo al estado de frenado normal y también al de frenado abs-salida activada

Diagrama de estados:

En este caso el diagrama de estados queda muy simplificado aunque se complican las ecuaciones de las transiciones:



Examen Parcial de FCO – Temas 5 al 7

21 de Enero de 2011

APELLIDOS: _____

NOMBRE: _____

DNI: _____

FIRMA: _____

A partir del siguiente código, escrito en ensamblador del MIPS R2000, responder a las siguientes preguntas.

```
.globl __start
.data 0x10000000
cadena: .asciiz "ONU"
caract: .ascii "."
.data 0x10001000
result: .space 6

.text 0x00400000
__start:
    la $8, cadena
    la $9, caract
    lb $10,0($9)
    la $11, result
bucle:
    lb $12,0($8)
    beq $12,$0,fin
    sb $12,0($11)
    sb $10, 1($11)
    addi $8, $8, 1
    addi $11,$11,2
    j bucle
fin:

.end
```

6.- (0,5 puntos) Describir brevemente qué hace el programa anterior.

El programa inserta el carácter '.' Entre cada uno de los caracteres de la cadena "ONU"

--

7.- (0,5 puntos) Indique el contenido del segmento de datos antes de iniciarse la ejecución, teniendo en cuenta que los datos se almacenan en formato “little endian”. El contenido debe ponerse en hexadecimal para los datos numéricos por cada byte de memoria, y con los caracteres correspondientes en el caso de los datos alfabéticos.

31	...	24	23	...	16	15	...	8	7	...	0	Dirección
NULL			'U'			'N'			'O'			0x10000000
									.'			0x10000004
0x00			0x00			0x00			0x00			0x10001000
						0x00			0x00			0x10001004

8.- (1 punto) Indique el contenido del segmento de datos después de finalizar la ejecución, teniendo en cuenta que los datos se almacenan en formato “little endian”. El contenido debe ponerse en hexadecimal para los datos numéricos por cada byte de memoria, y con los caracteres correspondientes en el caso de los datos alfabéticos.

31	...	24	23	...	16	15	...	8	7	...	0	Dirección
NULL			'U'			'N'			'O'			0x10000000
									.'			0x10000004
.'			'N'			.'			'O'			0x10001000
						.'			'U'			0x10001004

9.- (1 punto) Determinar el contenido de los siguientes registros cuando haya finalizado la ejecución del programa.

Registro	Contenido
\$8	0x10000003
\$9	0x10000004
\$10	.'
\$11	0x10001006
\$12	0x00000000

Examen Parcial de FCO – Temas 5 al 7
21 de Enero de 2011

APELLIDOS: _____

NOMBRE: _____

DNI: _____

FIRMA: _____

10.- (1 punto) Codificar la instrucción sb \$10, 1(\$11).

"sb" es una instrucción de tipo I. Por lo que habrá que averiguar el valor de los campos CO, rs, rt y desp, en ese orden.

La forma genérica de la instrucción es: "sb rt, desp(rs)". Por lo que el valor de cada campo será:

CO: código de operación de la instrucción "sb", que es 0x28. Codificado en 6 bits será 101000

rs: será el registro \$11. Codificado en 5 bits: 01011

rt: será el registro \$10. Codificado en 5 bits: 01010

desp: será 1. Codificado en 16 bits: 0000000000000001

Poniendo los campos en el orden especificado en el formato:

101000 01011 01010 0000000000000001

Uniéndolo todo:

10100001011010100000000000000001

Separando en grupos de 4 bits para pasar el código a hexadecimal:

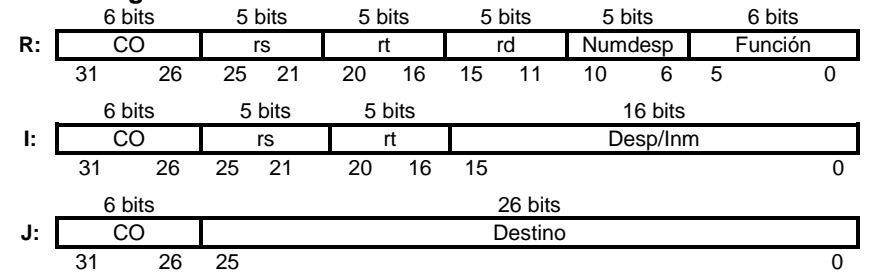
1010	0001	0110	1010	0000	0000	0000	0001
A	1	6	A	0	0	0	1

Por lo que la respuesta es: 0xA16A0001

Sintaxis y tipo de las instrucciones

Sintaxis	Formato	Descripción
add rd, rs, rt	R	$rd \leftarrow rs + rt$
addi rt, rs, inm	I	$rt \leftarrow rs + \text{inm}$, el dato inmediato es de 16 bits y se extiende el signo
sub rd, rs, rt	R	$rd \leftarrow rs - rt$
mult rs, rt	R	Multiplica rs por rt dejando los 32 bits de mayor peso en el registro HI y los 32 bits de menor peso en LO
div rs, rt	R	Divide rs entre rt dejando el cociente en el registro LO y el resto en el registro HI
and rd, rs, rt	R	$rd \leftarrow rs \text{ and } rt$, la operación lógica indicada se realiza bit a bit
nor rd, rs, rt	R	$rd \leftarrow rs \text{ nor } rt$
xor rd, rs, rt	R	$rd \leftarrow rs \text{ xor } rt$
or rd, rs, rt	R	$rd \leftarrow rs \text{ or } rt$
andi rt, rs, inm	I	$rt \leftarrow rs \text{ and } \text{inm}$, dato inmediato de 16 bits que se extiende con 16 ceros
ori rt, rs, inm	I	$rt \leftarrow rs \text{ or } \text{inm}$, dato inmediato de 16 bits que se extiende con 16 ceros
xori rt, rs, inm	I	$rt \leftarrow rs \text{ xor } \text{inm}$, dato inmediato de 16 bits que se extiende con 16 ceros
sll rd, rt, desp	R	$rd \leftarrow rt \ll \text{desp}$, desplazamiento a izquierdas. Conforme desplaza se rellena con 0
srl rd, rt, desp	R	$rd \leftarrow rt \gg \text{desp}$, desplazamiento a derechas. Conforme desplaza se rellena con 0
sra rd, rt, desp	R	$rd \leftarrow rt \gg \text{desp}$, desplazamiento a derechas. Conforme desplaza se extiende el bit de signo
Sintaxis	Formato	Descripción
lw rt, desp(rs)	I	$rt \leftarrow M[\text{desp}+rs]$, (el desplazamiento, desp, es de 16 bits y se extiende el signo), carga una palabra (32 bits)
lh rt, desp(rs)	I	$rt \leftarrow M[\text{desp}+rs]$, carga media palabra (16 bits) y extiende el signo
lb rt, desp(rs)	I	$rt \leftarrow M[\text{desp}+rs]$, carga un byte (8 bits) y extiende el signo
sw rt, desp(rs)	I	$M[\text{desp}+rs] \leftarrow rt$
sh rt, desp(rs)	I	$M[\text{desp}+rs] \leftarrow rt$, almacena la parte baja (16 bits) de rt en memoria
sb rt, desp(rs)	I	$M[\text{desp}+rs] \leftarrow rt$, almacena el byte menos significativo de rt en memoria
lui rt, inm	I	$rt31...16 \leftarrow \text{inm}$, $rt15...0 \leftarrow 0$
Sintaxis	Formato	Descripción
mfhi rd	R	$rd \leftarrow HI$
mflo rd	R	$rd \leftarrow LO$
mthi rs	R	$HI \leftarrow rs$
mtlo rs	R	$LO \leftarrow rs$
Sintaxis	Formato	Descripción
slt rd, rs, rt	R	Si $(rs < rt)$ entonces $rd \leftarrow 1$ si no $rd \leftarrow 0$
slti rt, rs, inm	I	Si $(rs < \text{inm})$ entonces $rt \leftarrow 1$ si no $rt \leftarrow 0$
Sintaxis	Formato	Descripción
beq rs, rt, etiqueta	I	Si $(rs == rt)$ $PC \leftarrow \text{etiqueta}$. Si se cumple la condición salta a la dirección etiqueta.
bne rs, rt, etiqueta	I	Si $(rs != rt)$ $PC \leftarrow \text{etiqueta}$. Si se cumple la condición salta a la dirección etiqueta.
Sintaxis	Formato	Descripción
j etiqueta	J	$PC \leftarrow \text{etiqueta}$, salta a la dirección etiqueta
jal etiqueta	J	$\$31 \leftarrow PC+4$, $PC \leftarrow \text{etiqueta}$, salta a la dirección etiqueta guardándose previamente la dirección de retorno en $\$31$
jr rs	R	$PC \leftarrow rs$, salta a la dirección contenida en el registro rs

Codificación según el formato



Códigos de operación y de función

Instrucción	CO
j	0x02
jal	0x03
beq	0x04
bne	0x05
addi	0x08
andi	0x0C
ori	0x0D
xori	0x0E
lui	0x0F
lb	0x20
lh	0x21
lw	0x23

6 bits

Instrucción	CO
sb	0x28
sh	0x29
sw	0x2B

6 bits

Instrucción	CO	Función
sll	0x00	0x00
srl	0x00	0x02
sra	0x00	0x03
jr	0x00	0x08
mfhi	0x00	0x10

6 bits

6 bits

Instrucción	CO	Función
mthi	0x00	0x11
mflo	0x00	0x12
mtlo	0x00	0x13
mult	0x00	0x18
div	0x00	0x1A
add	0x00	0x20
sub	0x00	0x22
and	0x00	0x24
or	0x00	0x25
xor	0x00	0x26
nor	0x00	0x27
slt	0x00	0x2A

6 bits

6 bits

Convenio MIPS

Nombre	Nº
\$zero	0
\$at	1
\$v0, \$v1	2,3
\$a0, ..., \$a3	4,...,7

Nombre	Nº
\$t0, ..., \$t7	8, ..., 15
\$s0, ..., \$s7	16,...,23
\$t8, \$t, 0	24,25
\$k0, \$k1	26,27

Nombre	Nº
\$gp	28
\$sp	29
\$fp	30
\$ra	31