

**23 de enero de 2012**

NOMBRE: \_\_\_\_\_

**FIRMA:**

- La duración del examen es de 2h30min.
- Escriba el nombre y los apellidos 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.)

[illegible]

Obtenga sus tablas de excitación y salida, tabla de estados, diagrama de estados y calcule la secuencia de valores de la salida **Y** para la secuencia de valores en la entrada **Cnt** : 1 – 1 – 1 – 0 – 0 – 0 – 1 – 1 – 1 teniendo en cuenta que el estado inicial del sistema es **Q<sub>1</sub> Q<sub>0</sub> = 00**.

Tablas de excitación y salida:

Funciones de excitación

$$D_1 = Q_1 / Cnt + /Q_1 Q_0 Cnt + Q_1 / Q_0 Cnt$$

$$D_0 = Cnt \oplus Q_0$$

Función de salida

$$Y = Q_1 Q_0$$

Tabla de excitación

$Q_1 Q_0 Cnt$	$D_1 D_0$
0 0 0	0 0
0 0 1	0 1
0 1 0	0 1
0 1 1	1 0
1 0 0	1 0
1 0 1	1 1
1 1 0	1 1
1 1 1	0 0

Tabla de salida

$Q_1 Q_0$	Y
0 0	0
0 1	0
1 0	0
1 1	1

Tabla de estados:

	<i>Estado siguiente</i>		
<i>Estado actual</i>	<i>Entrada Cnt</i>		<i>Salida</i>
$Q_1 Q_0$	$Cnt = 0$	$Cnt = 1$	Y
0 0	0 0	0 1	0
0 1	0 1	1 0	0
1 0	1 0	1 1	0
1 1	1 1	0 0	1

Diagrama de estados:

## Examen Parcial de FCO – Temas 5 al 7

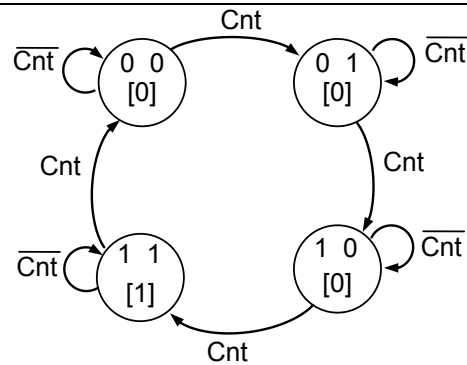
23 de enero de 2011

APELLIDOS: \_\_\_\_\_

NOMBRE: \_\_\_\_\_

DNI: \_\_\_\_\_

FIRMA: \_\_\_\_\_



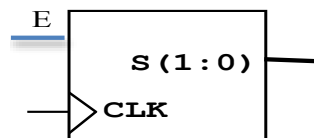
Salida del sistema para la secuencia de valores 1 – 1 – 1 – 0 – 0 – 0 – 1 – 1 – 1 en la entrada Cnt:

*Asumiendo que el sistema se encuentra en el estado inicial 00 la secuencia de valores de la salida para esos valores en la entrada sería: 0 - 0 - 0 - 1 - 1 - 1 - 1 - 0 - 0 - 0*

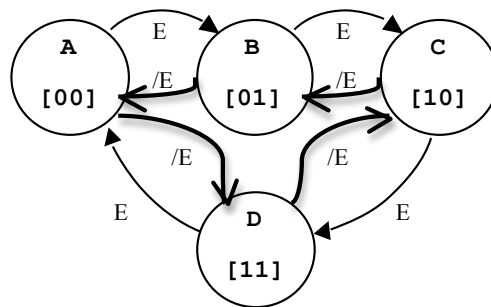
**2.- (2 puntos)** Se pide diseñar un SSS que implemente un contador binario de 2 bits reversible gobernado por una entrada de selección del sentido de la cuenta llamada E. Cuando E=1, el contador realiza una cuenta ascendente, cuando E=0, el contador realiza una cuenta descendente. Se pide:

- a) **(0,25 puntos)** Interfaz del SSS, entrada y salidas.
- b) **(1 punto)** Diagrama de estados
- c) **(0,75 puntos)** Tabla de estados con la codificación de estados

a) El circuito es un contador sencillo que tiene como entradas externas la de reloj, y la del control del sentido de la cuenta



b)



c) Tabla de estados.

	Q(t+1)			
	E			
E.A.	0	1	S1	S0
00	11	01	0	0
10	00	10	0	1
10	01	11	1	0
11	10	00	1	1

# Examen Parcial de FCO – Temas 5 al 7

23 de enero de 2011

APELLIDOS: \_\_\_\_\_

NOMBRE: \_\_\_\_\_

DNI: \_\_\_\_\_

FIRMA: \_\_\_\_\_

**3.- (2 puntos)** Dados los números decimales  $A = 125$ ,  $B = -56$ .

- A) **(0,5 puntos)** Representélos mediante 8 bits en el convenio de representación  $Ca_2$ , detallando todas las operaciones necesarias para llegar al resultado.
- B) **(0,75 puntos)** Realice la operación  $A+B$  en el convenio de representación  $Ca_2$  utilizando 8 bits, e indique claramente si hay desbordamiento, justificándolo.
- C) **(0,75 puntos)** Realice la operación  $A-B$  en el convenio de representación  $Ca_2$  utilizando 8 bits, e indique claramente si hay desbordamiento, justificándolo.

**A)**

Representación de A y B en  $Ca_2$  utilizando 8 bits

$$125_{10} = 01111101_2 = A (+125)$$

$$56_{10} = 00111000_2 \quad Ca_2 (00111000) = 11001000_2 = B (-56)$$

**B)**

Resultado de  $R = A+B$  utilizando 8 bits

$$A \rightarrow 01111101_{Ca_2} = 64+32+16+8+4+1=125_{10}$$

$$B \rightarrow 11001000_{Ca_2} = -128+64+8 = -56_{10}$$

$$R \rightarrow \overline{1}01000101_{Ca_2} = 64+4+1=69_{10}$$

**C)**

Resultado de  $R = A-B$  utilizando 8 bits

$$A-B = A + Ca_2(B)$$

$$A \rightarrow 01111101_{Ca_2} = 64+32+16+8+4+1=125_{10}$$

$$Ca_2(B) \rightarrow 00111000_{Ca_2} = 32+16+8=56_{10}$$

$$R \rightarrow \overline{1}0110101_{Ca_2} = 128+32+16+4+1 = 181 \quad ; \text{!excede el rango!!}$$

La operación genera desbordamiento, dado que al sumar 2 positivos, el resultado es negativo. Podemos además comprobarlo porque los dos últimos acarreos son diferentes:  $V = C_{n-1} \oplus C_{n-2} = 0 \oplus 1 = 1$

**4.- (1,0 puntos)** Represente el número -1024,3125 en el formato de simple precisión del estándar IEEE754. Detalle todos los pasos realizados y exprese el resultado final en hexadecimal.

*En primer lugar se convierte la cantidad a binario, por un lado la parte entera con divisiones sucesivas (o sabiendo que  $1024 = 2^{10}$ )*

$$1024_{10} = 10000000000_2$$

*y por otro lado, la parte fraccionaria con multiplicaciones sucesivas*

$$0,3125 \quad \times 2 = 0,625$$

$$0,625 \quad \times 2 = 1,25$$

$$0,25 \quad \times 2 = 0,5$$

$$0,5 \quad \times 2 = 1,0$$

*por lo que  $-1024,3125_{10} = -10000000000,0101_2$*

*En segundo lugar se reescribe la cantidad en forma  $\pm 1, M \times 2^E$ :*

$$-10000000000,0101_2 = -10000000000,0101_2 \times 2^0 = -1,00000000000101_2 \times 2^{10}$$

*Y ahora, para que la cantidad pueda ser representada mediante un número normalizado (esto es, un número en forma  $\pm 1, M \times 2^E$ , con la mantisa normalizada) es necesario que el exponente se pueda representar en exceso 127 con 8 bits y que la secuencia de bits resultante no coincida con las secuencias reservadas para las excepciones (casos 00000000 y 11111111).*

*El exponente ( $10_{10}$ ) en exceso 127 se representa mediante binario ( $10 + 127$ ) = binario (137) = 10001001*

*Como la secuencia de bits obtenida para el campo exponente no es ninguno de los casos mencionados para las excepciones a la norma, la cantidad sí se puede representar mediante un número normalizado, por lo que sólo falta asignar los bits a cada campo (S, E, M)*

*Campo S (signo): 1 (negativo)*

*Campo E (exponente): 10001001*

*Campo M (parte fraccionaria de la mantisa normalizada): 00000000000101 (teniendo en cuenta que el bit de la parte entera es el bit implícito y no se almacena); rellenando (por la derecha) hasta completar los 23 bits de este campo: 00000000000101000000000*

*Los 32 bits juntos en el orden S, E, M:*

*1 10001001 00000000000101000000000*

*En hexadecimal (reuniendo todos los bits y agrupando de cuatro en cuatro):*

*1100 0100 1000 0000 0000 1010 0000 0000 0xC4800A00*

**Examen Parcial de FCO – Temas 5 al 7**  
**23 de enero de 2011**

APELLIDOS: \_\_\_\_\_

NOMBRE: \_\_\_\_\_

DNI: \_\_\_\_\_

FIRMA: \_\_\_\_\_

El siguiente código, escrito en lenguaje ensamblador del MIPS R2000, intercala los caracteres de dos cadenas, guardando el resultado en otra cadena.

```
.globl __start
.data 0x10000000
cad1:      .asciiz "AB"
cad2:      .asciiz "123"
.data 0x10001000
long:      .word 1
result:    .space 8

.text 0x00400000
__start:
    la $8, cad1
    la $9, cad2
    li $10, 1
    la $11, result
bucle:
    lb $12, 0($8)
    sb $12, 0($11)
    beq $12, $0, fin
    lb $12, 0($9)
    sb $12, 1($11)
    beq $12, $0, fin
    addi $8, $8, 1
    addi $9, $9, 1
    addi $10, $10, 2
    addi $11, $11, 2
    j bucle
fin:
    la $8, long
    sw $10, 0($8)
.end
```

Basándose en el anterior código, responder a las siguientes preguntas.

- 5.- (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 especificarse por cada byte, en

hexadecimal para los datos numéricos, y con los caracteres correspondientes entre comillas en el caso de las cadenas.

31 ... 24	23 ... 16	15 ... 8	7 ... 0	Dirección
'1'	NULL	'B'	'A'	0x10000000
	NULL	'3'	'2'	0x10000004
				...
0x00	0x00	0x00	0x01	0x10001000
0x00	0x00	0x00	0x00	0x10001004
0x00	0x00	0x00	0x00	0x10001008

**6.- (0,5 puntos)** Codificar, mostrando el resultado en hexadecimal, la instrucción `lb $12,0($9)`. Detalle todos los pasos realizados para obtener el resultado final en hexadecimal.

La instrucción es de tipo I, por lo que tiene cuatro campos (CO de 6 bits, rs de 5 bits, rt de 5 bits y Desp/Inm de 16 bits).

Como es una instrucción `lb`, el código de operación (CO) es `0x20` (6 bits), en binario: `100000`

rs se corresponde con el registro `$9`, que en cinco bits es: `01001`

rt se corresponde con el registro `$12`, que en cinco bits es: `01100`

El desplazamiento es `0`, por lo que en 16 bits se corresponde con: `0000000000000000`

Uniendo los campos binarios tenemos la secuencia:

`100000 01001 01100 0000000000000000`

Para pasarla a hexadecimal agrupamos de cuatro a cuatro bits:

`1000 0001 0010 1100 0000 0000 0000 0000`

**Resultado: `0x812c0000`**

**7.- (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 especificarse por cada byte, en hexadecimal para los datos numéricos, y con los caracteres correspondientes entre comillas en el caso de las cadenas.

31 ... 24	23 ... 16	15 ... 8	7 ... 0	Dirección
'1'	NULL	'B'	'A'	0x10000000
	NULL	'3'	'2'	0x10000004
				...



**Examen Parcial de FCO – Temas 1 al 4**  
**28 de Noviembre de 2011**

APELLIDOS: \_\_\_\_\_ NOMBRE: \_\_\_\_\_

DNI: \_\_\_\_\_

FIRMA: \_\_\_\_\_

0x00	0x00	0x00	0x05	0x10001000
'2'	'B'	'1'	'A'	0x10001004
0x00	0x00	0x00	NULL	0x10001008

- 8.- (1 punto)** Determine el contenido de los siguientes registros cuando haya finalizado la ejecución del programa. Exprese el contenido en hexadecimal en el caso de las direcciones, entre comillas en el caso de los caracteres y en decimal en el resto de los casos.

Registro	Contenido
\$8	0x10001000
\$9	0x10000005
\$10	5
\$11	0x10001008
\$12	0x00 (NULL)

- 9.- (0,25 puntos)** ¿De qué tipo es la cadena resultante “result”?
- a) De tipo ascii, ya que no se almacena un carácter 0x00 (NULL) al final del resultado.
  - b) De tipo ascii, ya que las cadenas originales también son ascii.
  - c) De tipo asciiiz, ya que se almacena un carácter 0x00 (NULL) al final del resultado justo antes de salir del bucle.
  - d) De tipo asciiiz, ya que se toma la precaución de almacenar siempre, una vez finalizado el bucle, un carácter 0x00 (NULL).
  - e) Ninguna de las anteriores.

Respuesta: c

- 10.- (0,25 puntos)** ¿Cuál es la condición de salida del bucle?
- a) Cuando se localiza el carácter 0x00 (NULL) de cualquiera de las dos cadenas (“cad1” y “cad2”).
  - b) Cuando se localiza el carácter 0x00 (NULL) de la primera cadena (“cad1”).
  - c) Cuando se localiza el carácter 0x00 (NULL) de la segunda cadena (“cad2”).
  - d) El bucle no tiene ninguna condición de salida.
  - e) Ninguna de las anteriores

Respuesta: a