

Examen Parcial de FCO – Temas 5 al 7

21 de enero de 2013

APELLIDOS: _____

NOMBRE: _____

DNI: _____

FIRMA: _____

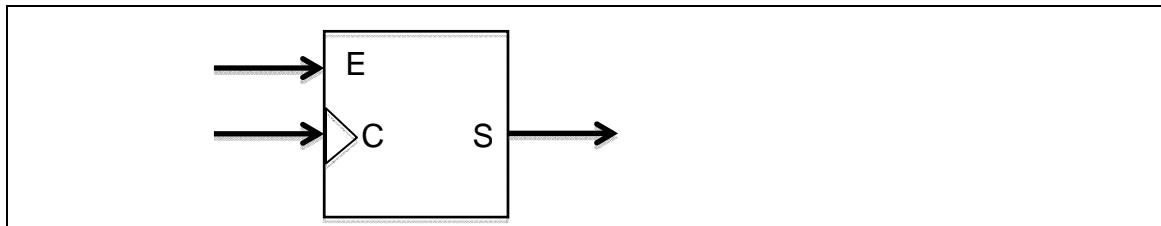
Normativa:

- La duración del examen es de 2h.
- 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.)

1. **(1 punto)** Se pretende diseñar un circuito que detecte la secuencia **1-0-1** por su entrada **E**. Cuando haya detectado dicha secuencia activará su salida **S** a 1 durante un ciclo de reloj. Considere que no puede ocurrir solapamiento en la entrada (el final de la secuencia anterior no puede ser considerado el principio de la siguiente). Se pide:

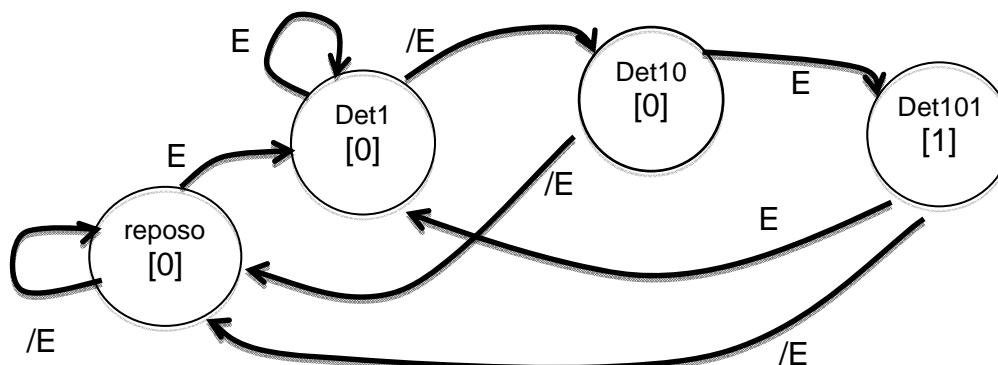
a) Interfaz con las entradas y salidas del sistema secuencial síncrono. **(0,25 puntos)**

Solución:

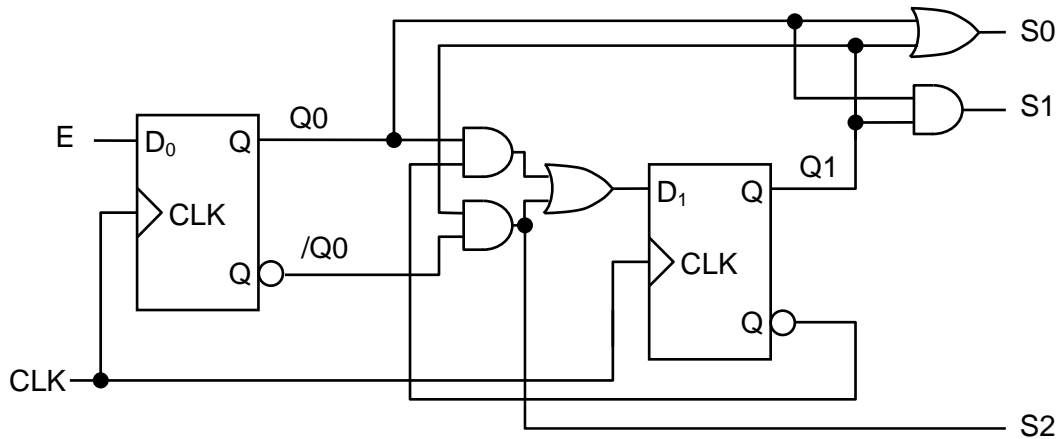


b) Diagrama de estados. **(0,75 puntos)**

Solución:



2. **(2 puntos)** Dado el circuito siguiente, que implementa un SSS, resuelva los siguientes apartados:



- a) Indique las expresiones algebraicas para D₁, D₀, S₂, S₁ y S₀ **(0,2 puntos)**

Solución:

$$\begin{aligned} D_1 &= (Q_0 \cdot /Q_1) + (/Q_0 \cdot Q_1) \\ D_0 &= E \\ S_2 &= /Q_0 \cdot Q_1 \\ S_1 &= Q_0 \cdot Q_1 \\ S_0 &= Q_0 + Q_1 \end{aligned}$$

- b) Obtenga la tabla de verdad de la función de transición/excitación **(0,4 puntos)**

Solución:

Estado actual	Entrada	Estado siguiente
Q ₁ (t) Q ₀ (t) E		Q ₁ (t+1) Q ₀ (t+1) = D ₁ D ₀
0 0 0		0 0
0 0 1		0 1
0 1 0		1 0
0 1 1		1 1
1 0 0		1 0
1 0 1		1 1
1 1 0		0 0
1 1 1		0 1

Examen Parcial de FCO – Temas 5 al 7

21 de enero de 2013

APELLIDOS: _____ NOMBRE: _____

DNI: _____

FIRMA: _____

c) Obtenga la tabla de verdad de la función de salida **(0,4 puntos)**

Solución:

Estado actual $Q_1(t) Q_0(t)$	Salida $S_2 S_1 S_0$
0 0	0 0 0
0 1	0 0 1
1 0	1 0 1
1 1	0 1 1

d) Obtenga la tabla de estados **(0,6 puntos)**

Solución:

	Estado siguiente $Q_1(t+1) Q_0(t+1)$		
Estado actual	Entrada E		Salida
$Q_1(t) Q_0(t)$	0	1	$S_2 S_1 S_0$
0 0	0 0	0 1	0 0 0
0 1	1 0	1 1	0 0 1
1 0	1 0	1 1	1 0 1
1 1	0 0	0 1	0 1 1

e) Indique cuál es la salida del sistema para la secuencia de entrada $E = 0 - 0 - 1 - 1 - 0 - 0$ considerando que el estado inicial es $Q_0=Q_1=0$ **(0,4 puntos)**

Solución:

Estado = 00 – 00 – 00 – 01 – 11 – 00 – 00
Salidas = 000 – 000 – 000 – 001 – 011 – 000 – 000

3.- (2 puntos) Dados los siguientes números, $A=0x63$ y $B=0x40$, que representan valores **enteros en complemento a dos**, realice las operaciones en complemento a 2 que se indican a continuación. Para ello, utilice 8 bits, detalle claramente el procedimiento empleado e indique si el resultado es correcto o no, justificándolo en ambos casos.

a) $A+B$ **(0.6 puntos)**

Solución:

1_ Pasamos los números a binario (y ya estarán representados en Ca2)

$$A = 0x63 = 01100011$$

$$B = 0x40 = 01000000$$

2_ Como ya están representados en Ca2, realizamos la operación de suma directamente.

3_ Para detectar si el resultado es correcto, nos fijamos en los últimos dos acarreos; son distintos, esto indica que se produce desbordamiento y por lo tanto el resultado no es válido.

(También podemos observar que sumamos dos números positivos y el resultado es negativo.)

	0	1	0	0	0	0	0	0	Acarreos
	0	1	1	0	0	0	1	1	A
+	0	1	0	0	0	0	0	0	B
	1	0	1	0	0	0	1	1	Resultado

b) A-B (0.7 puntos)**Solución:**

1_ Pasamos los números a binario (y ya estarán representados en Ca2)

$$A = 0x63 = 01100011$$

$$B = 0x40 = 01000000$$

2_ Como se trata de una operación en Ca2, debemos tener en cuenta que:

$$A-B = A + \mathbf{Ca2}(B), \text{ hacemos por lo tanto el } \mathbf{Ca2}(B) = 11000000$$

3_ Realizamos la operación de suma teniendo en cuenta que según las reglas en Ca2, si existe un acarreo final se desprecia.

4_ Para detectar si el resultado es correcto, nos fijamos en los dos últimos acarreos; como son iguales no se produce desbordamiento y el resultado es por lo tanto válido.

	1	1	0	0	0	0	0	0	Acarreos
	0	1	1	0	0	0	1	1	A
+	1	1	0	0	0	0	0	0	Ca2(B)
	4	0	0	1	0	0	1	1	Resultado

c) B-A (0.7 puntos)**Solución:**

1_ Pasamos los números a binario (y ya estarán representados en Ca2)

$$A = 0x63 = 01100011$$

$$B = 0x40 = 01000000$$

2_ Como se trata de una operación en Ca2, debemos tener en cuenta que:

$$B-A = B + \mathbf{Ca2}(A), \text{ hacemos por lo tanto el } \mathbf{Ca2}(A) = 10011101$$

3_ Realizamos la operación de suma teniendo en cuenta que según las reglas en Ca2 si existe un acarreo final se desprecia.

4_ Para detectar si el resultado es correcto, nos fijamos en los dos últimos acarreos; como son iguales no se produce desbordamiento y el resultado es por lo tanto válido.

	0	0	0	0	0	0	0	0	Acarreos
	0	1	0	0	0	0	0	0	B
+	1	0	0	1	1	1	0	1	Ca2(A)
	1	1	0	1	1	1	0	1	Resultado

Examen Parcial de FCO – Temas 5 al 7

21 de enero de 2013

APELLIDOS: _____

NOMBRE: _____

DNI: _____

FIRMA: _____

4.- (1 punto) Represente el número real -520,8125 en el formato IEEE754 de simple precisión. Detalle todos los pasos realizados y exprese el resultado final en binario y en hexadecimal.

Solución:

En primer lugar se convierte la cantidad a binario, por un lado la parte entera con divisiones sucesivas (o sabiendo que el número es igual a $512 + 8 = 2^9 + 2^3$)

$$520_{10} = 1000001000_2$$

y por otro lado, la parte fraccionaria con multiplicaciones sucesivas

$$0,8125 \quad \times 2 = \underline{1},625$$

$$0,625 \quad \times 2 = \underline{1},25$$

$$0,25 \quad \times 2 = \underline{0},5$$

$$0,5 \quad \times 2 = \underline{1},0$$

por lo que $-520,8125 = -1000001000,1101_2$

Se reescribe en forma $\pm 1, M \times 2^E$:

$$-1000001000,1101_2 = -1000001000,1101_2 \times 2^0 = -1,0000010001101_2 \times 2^9$$

Expresamos el exponente en exceso 127:

Exponente = 9, expresado en exceso 127, se representa mediante binario ($9 + 127$) = binario (136) = 10001000

Campo S (signo): 1 (negativo)

Campo E (exponente): 10001000

Campo M (parte fraccionaria de la mantisa normalizada):

000001000110100000000000 (teniendo en cuenta que el bit de la parte entera es el bit implícito y no se almacena y rellenando hasta completar los 23 bits de este campo)

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

1 10001000 000001000110100000000000

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

1100 0100 0000 0010 0011 0100 0000 0000 0xC4023400

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

```

        .globl __start
        .data 0x10000000
datos_i: .half 1,-1,2,-2
        num: .byte 4
        .data 0x10001000
datos_f: .space 16

        .text 0x00400020
__start:
        la $8, datos_i
        la $9, num
        lb $10,0($9)
        la $11, datos_f
        li $12, -1
bucle:
        beq $10,$0,fin
        lh $13, 0($8)
        mult $13,$12
        mflo $14
        sw $14,0($11)
        addi $10, $10,-1
        addi $8,$8,2
        addi $11,$11,4
        j bucle
fin:
        .end

```

5.- (0,5 puntos) Indique el contenido del segmento de datos antes de iniciarse la ejecución del código, teniendo en cuenta que los datos se almacenan en formato “little endian”. El contenido debe ponerse en hexadecimal para cada byte de memoria.

Solución:

31	...	24	23	...	16	15	...	8	7	...	0	Dirección
0xff			0xff			0x00			0x01			0x10000000
0xff			0xfe			0x00			0x02			0x10000004
0x00			0x00			0x00			0x04			0x10000008
0x00			0x00			0x00			0x00			0x10001000
0x00			0x00			0x00			0x00			0x10001004
0x00			0x00			0x00			0x00			0x10001008
0x00			0x00			0x00			0x00			0x1000100c

Examen Parcial de FCO – Temas 5 al 7

21 de enero de 2013

APELLIDOS: _____ NOMBRE: _____

DNI: _____

FIRMA: _____

6.- (1,25 puntos) 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 cada byte de memoria.

Solución

31	...	24	23	...	16	15	...	8	7	...	0	Dirección
0xff			0xff			0x00			0x01			0x10000000
0xff			0xfe			0x00			0x02			0x10000004
0x00			0x00			0x00			0x04			0x10000008
0xff			0xff			0xff			0xff			0x10001000
0x00			0x00			0x00			0x01			0x10001004
0xff			0xff			0xff			0xfe			0x10001008
0x00			0x00			0x00			0x02			0x1000100c

7.- (1,25 puntos) Determine el contenido de los siguientes registros cuando haya finalizado la ejecución del programa.

Solución:

Registro	Contenido
\$8	0x10000008
\$9	0x10000008
\$10	0
\$11	0x10001010
\$12	-1
\$13	-2
\$14	2

8.- (1 punto) Codifique la instrucción `j bucle`. Indique el resultado en binario y hexadecimal. Indique además la dirección en hexadecimal que representa la etiqueta `bucle`.

Solución:

Determinación de la dirección que representa la etiqueta `bucle`:

```
__start:0x00400020 lui $8, 0x1000 (la $8, datos_i) datos_i=0x10000000
0x00400024 lui $1, 0x1000 (la $9, num) num=0x10000008
0x00400028 ori $9, $1, 0x0008
0x0040002c lb $10, 0($9)
0x00400030 lui $1, 0x1000 (la $11, datos_f) datos_f=0x10001000
0x00400034 ori $11, $1, 0x1000
0x00400038 lui $1, 0xffff (li $12, -1)
0x0040003c ori $12, $1, 0xffff
bucle: 0x00400040 beq $10, $0, fin
```

La etiqueta `bucle` representa la dirección `0x00400040`

Codificación con formato J.

CO = $0x02 = 000010_2$

destino contiene los bits 2 a 27 de la dirección de salto `0x00400040`:
(0000)0000010000000000000000010000(00)

CO	destino
000010	0000010000000000000000010000

En hexadecimal: `0x08100010`