EXAMEN DE FUNDAMENTO DE COMPUTADORES
INGENIERÍA TÉCNICA EN INFORMÁTICA DE SISTEMAS
Grupos A y B
FEBRERO DE 2010

Problema 1 (1 punto)
a) Expresa $(0AF13)_{16}$ en binario y octal.
b) Expresa $(139)_{10}$ en octal, binario y hexadecimal.
c) Expresa $(4573)_8$ en binario y decimal.

Problema 2 (1 punto)
Si A=10 y B=-14, haz las operaciones que aparecen a continuación operando en complemento a dos y utilizando el menor número de bits posible. Indica en cada caso si hay desbordamiento y acarreo razonándolo.
a) A + B
b) A – B
c) - A + B
d) -A - B

Problema 3
Sea un sistema que tiene como entrada las señales de datos {A,B} y la señal de control S. El sistema tiene que reconocer la secuencia de datos BAB de manera que si S=0 el reconocedor es sin solapamiento y si S=1 el reconocedor es con solapamiento.

1) Utilizando una máquina de Mealy (4 puntos)
   a) Diagrama de estados
   b) Tabla de verdad
   c) implementa la salida Z con el menor número de puertas lógicas
   d) implementa la función *siguiente estado* con decodificadores de tamaño mínimo representando el sistema secuencial final teniendo en cuenta la función obtenida en c)

2) Suponiendo una máquina de Moore (4 puntos)
   a) Diagrama de estados
   b) Tabla de verdad
   c) Implementa la salida Z mediante multiplexores que utilicen como señal de control las entradas de datos y de control del sistema reconocedor.
   d) Implementa la función siguiente estado mediante multiplexores con tantas señales de control como variables tengan las funciones
   d) implementa y representa todo el sistema utilizando una ROM

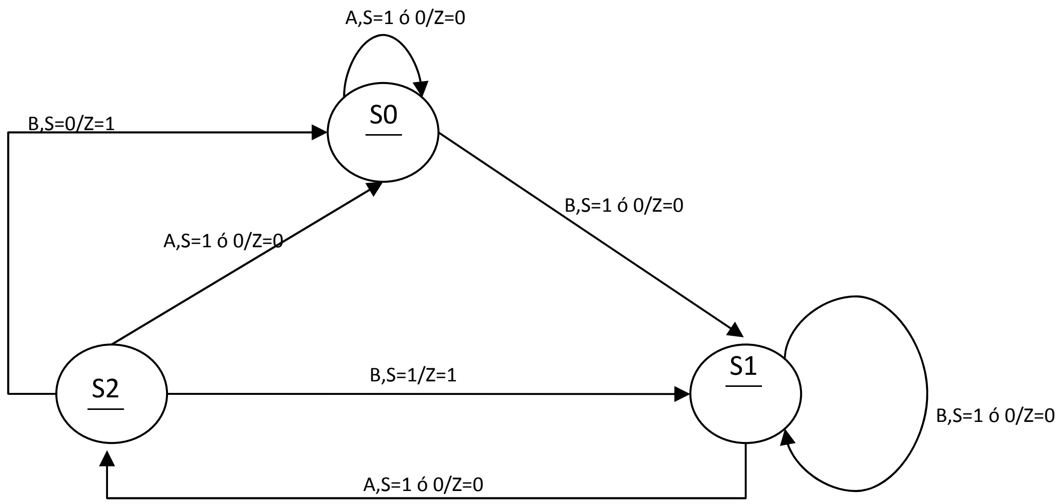
Soluciones
Problema 1

a)
$(000010101111100010011)_2$
00-001-010-111-100-010-011=$(0127423)_8$
b)
$(10001011)_2$
10-001-011=$(213)_8$
1000-1011=$(8B)_{16}$
c)
$100101111011)_2$

Problema 2
a)
10→01010; -14 →10010
A+B=01010+10010=11100; no hay acarreo, no hay desbordamiento
b)
A-B = A+(-B)
-B=01110→A+(-B)= 01010+01110=11000; no hay acarreo, pero hay desbordamiento (la suma de dos números positivos no puede dar un número negativo)
c)
–A= 10110; -A +B = 10110+10010=101000; hay acarreo y hay desbordamiento
d)
-A + (-B)= 10110+01110= 100100; hay acarreo pero no desbordamiento

Problema 3
1) MAQUINA DE MEALY
a) Estados del sistema
S0→nada
S1→ B
S2→BA

Diagrama de estados



b) Tablas de verdad
Codificación:

estados		
	E1	E0
S0	0	0
S1	0	1
S2	1	0
S3	1	1

Fijarse que en Mealy el estado S3 no se da nunca, luego provocara don't cares en las tablas de verdad.

entrada de datos	X
A	0
B	1

Tablas de verdad:

entradas		estado actual		siguiente estado		salida
S	X	E1	E0	e1	e0	z
0	0	0	0	0	0	0
0	0	0	1	1	0	0
0	0	1	0	0	0	0
0	0	1	1	d	d	d
0	1	0	0	0	1	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	d	d	d
1	0	0	0	0	0	0
1	0	0	1	1	0	0
1	0	1	0	0	0	0
1	0	1	1	d	d	d
1	1	0	0	0	1	0
1	1	0	1	0	1	0
1	1	1	0	0	1	1
1	1	1	1	d	d	d

Siendo:

- S la señal cambia el tipo de reconocedor
- E1,E0 el estado actual almacenado en los biestables
- e1,e0 el siguiente estado
- Z la salida del sistema

c) implementa la salida Z con el menor número de puertas lógicas
Para usar el menor número de puertas hay que aplicar diagramas de k.

		E1 E0			
		00	01	11	10
S X	00	(0)	(1)	D (3)	(2)
	01	(4)	(5)	D (7)	1 (6)
	11	(12)	(13)	D (15)	1 (14)
	10	(8)	(9)	D (11)	(10)

Z= X+E1

2) MAQUINA DE MOORE

a)

Estados:

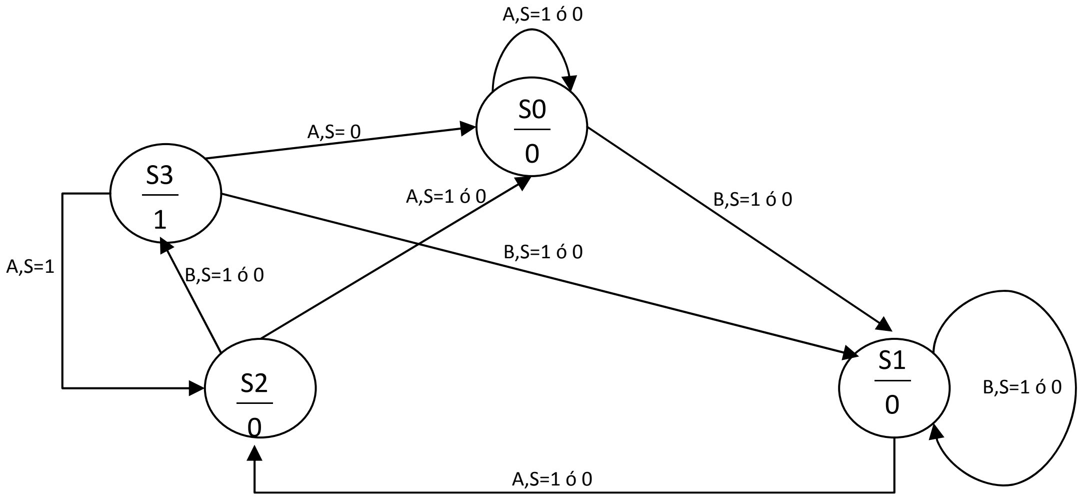
S0→NADA

S1→B

S2→BA

S3→BAB

diagrama de estados



b)

tabla de verdad

entradas		estado actual		siguiente estado		salida
S	X	E1	E0	e1	e0	z
0	0	0	0	0	0	0
0	0	0	1	1	0	0
0	0	1	0	0	0	0
0	0	1	1	0	0	1
0	1	0	0	0	1	0
0	1	0	1	0	1	0
0	1	1	0	1	1	0
0	1	1	1	0	1	1
1	0	0	0	0	0	0
1	0	0	1	1	0	0
1	0	1	0	0	0	0
1	0	1	1	1	0	1
1	1	0	0	0	1	0
1	1	0	1	0	1	0
1	1	1	0	1	1	0
1	1	1	1	1	1	1

c) En moore la salida no depende de las entradas (en este caso S y X), sólo del estado actual (E1,E0), por lo tanto no se puede implementar el circuito con las especificaciones del enunciado