UNIVERSIDAD DE ANTIOQUIA
FACULTAD DE INGENIERÍA

Laboratorio No. 1

Diseño de sistemas digitales

Autores:

Omar Alberto Torres
Leidy Castaño Castaño

Profesor:

Fredy Alexander Rivera Velez
Arquitectura de Computadores y Laboratorio

Universidad de Antioquia
Facultad de Ingeniería
Ingeniería de Sistemas
Ude@

2023

Laboratorio No. 1

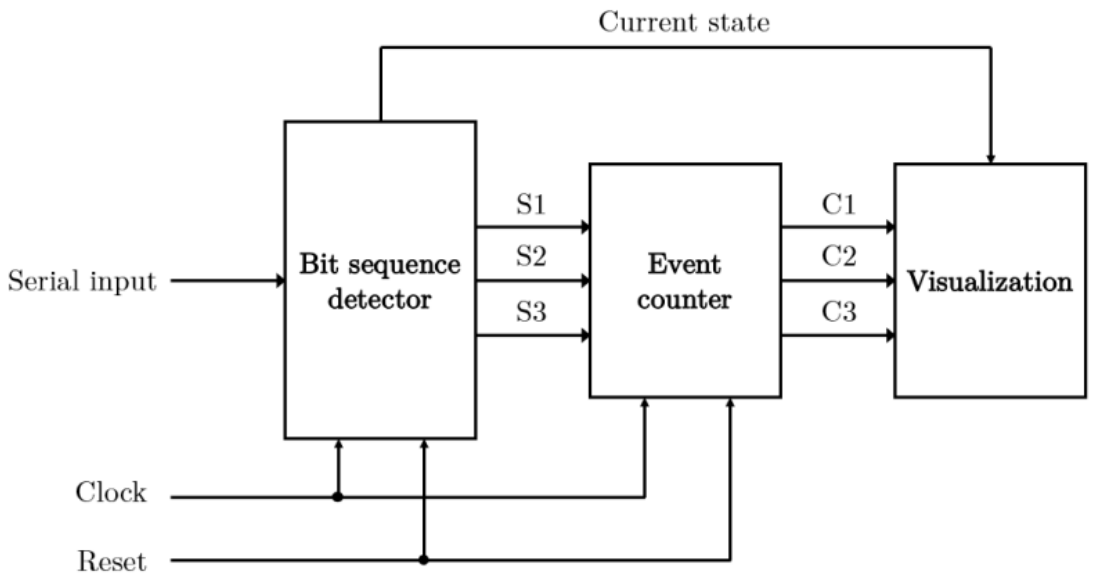
Video de sustentación:
https://drive.google.com/file/d/1bNDQGE6y1zcsZxa3AXgK1pyNX1WUCcVk/view?usp=sharing

Descripción

Diseñar, implementar y simular un sistema secuencial que detecte tres cadenas de dígitos binarios diferentes.

Equipo 4:
Secuencia #1 (4 bits): 1100
Secuencia #2 (5 bits): 10001
Secuencia #3 (6 bits): 110000
Flip flop JK

El sistema dispone de una entrada binaria a través de la cual es conformada, al ritmo de la señal de reloj (clock), la secuencia a analizar. Las secuencias de bits son identificadas sin importar donde ocurran dentro de una secuencia más larga. El sistema cuenta con una señal de reset asíncrona que permite llevar el sistema a su estado inicial, tal como se muestra a continuación:



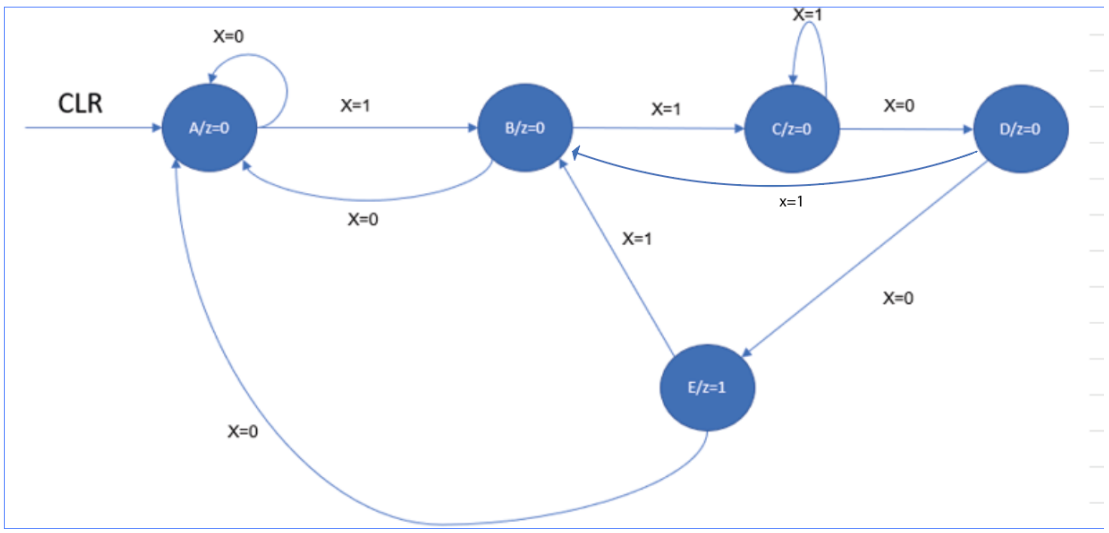
Detector de secuencias

Autómata que reconoce la secuencia #1: 1100

Tabla de estados:

Est act	Est next		
	X=0	X=1	Z
A	A	B	0
B	A	C	0
C	D	C	0
D	E	B	0
E	A	B	1

Diagrama de estados:



Autómata que reconoce la secuencia #2: 10001

Diagrama de estados:

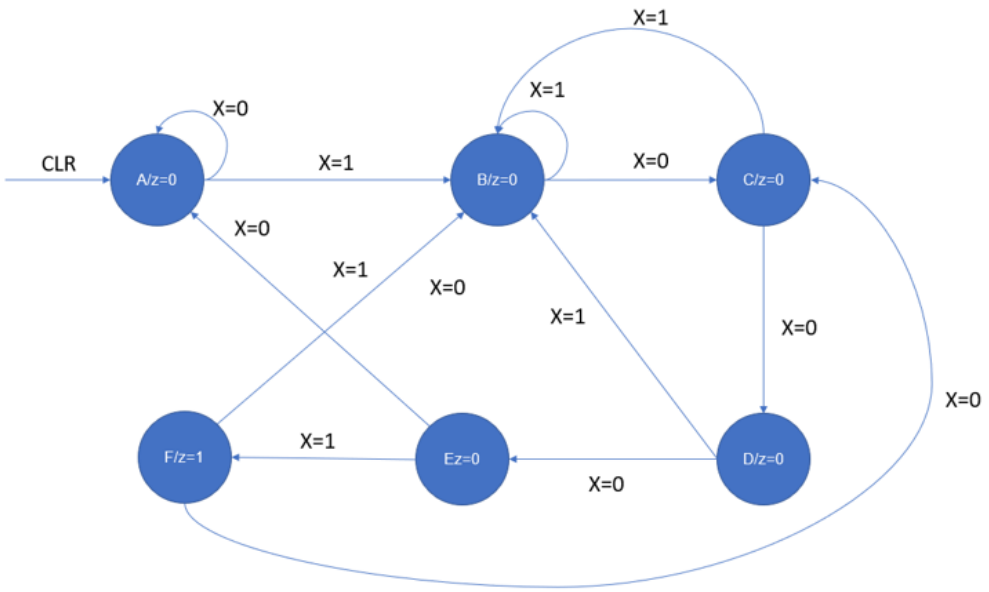
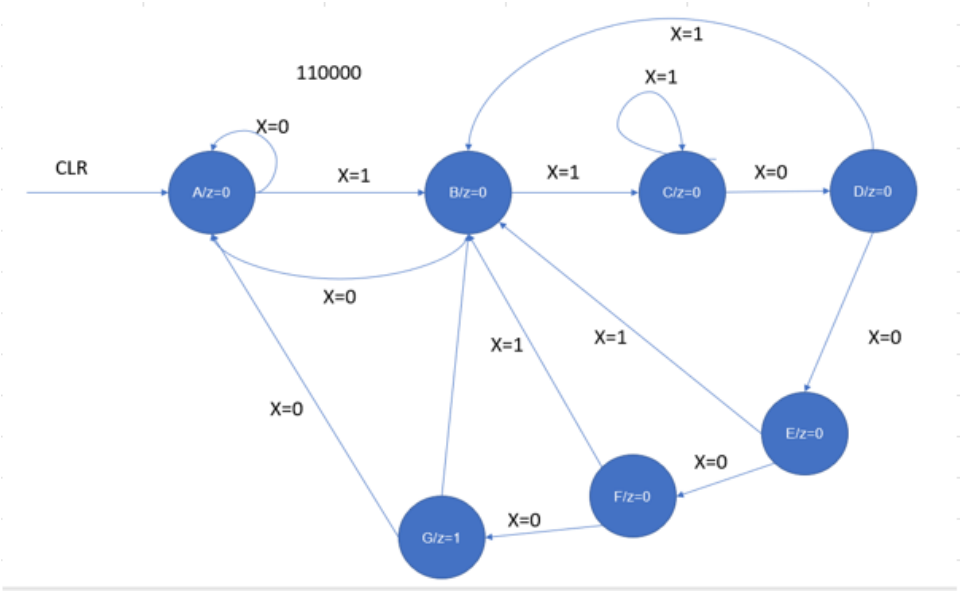


Tabla de estados:

Est act	Est next		
	X=0	X=1	Y
A	A	B	0
B	C	B	0
C	D	B	0
D	E	B	0
E	A	F	0
F	C	B	1

Autómata que reconoce la secuencia #3: 110000

Diagrama de estados:



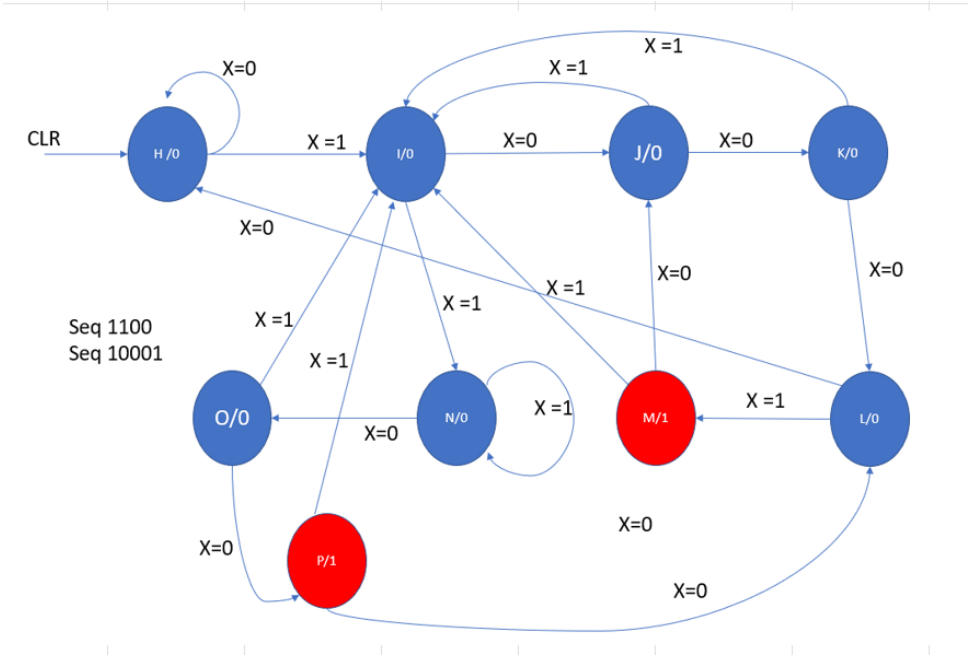
Unión de la secuencia #1 con la secuencia #2:

Estado Actua	Estado siguiente			
	X = 0	X = 1	S1	S2
AF	AF	BG	0	0
BG	AH	CG	0	0
AH	AI	BG	0	0
AI	AJ	BG	0	0
AJ	AF	BK	0	1
CG	DH	CG	0	0
DH	EI	BG	0	0
EI	AJ	BG	1	0
BK	AH	CG	0	1

Est actual	Esta sig			
	X = 0	X = 1	S1	S2
A	A	B	0	0
B	C	F	0	0
C	D	B	0	0
D	E	B	0	0
E	A	I	0	1
F	G	F	0	0
G	H	B	0	0
H	E	B	1	0
I	C	F	0	1

Renombrado los nodos quedan de la siguiente manera:

Diagrama de estados:



Unión de la secuencia anterior con la secuencia # 3:

Se obtuvo el detector de secuencias como una sola máquina de estados finitos tipo Moore.

Diagrama de estados final:

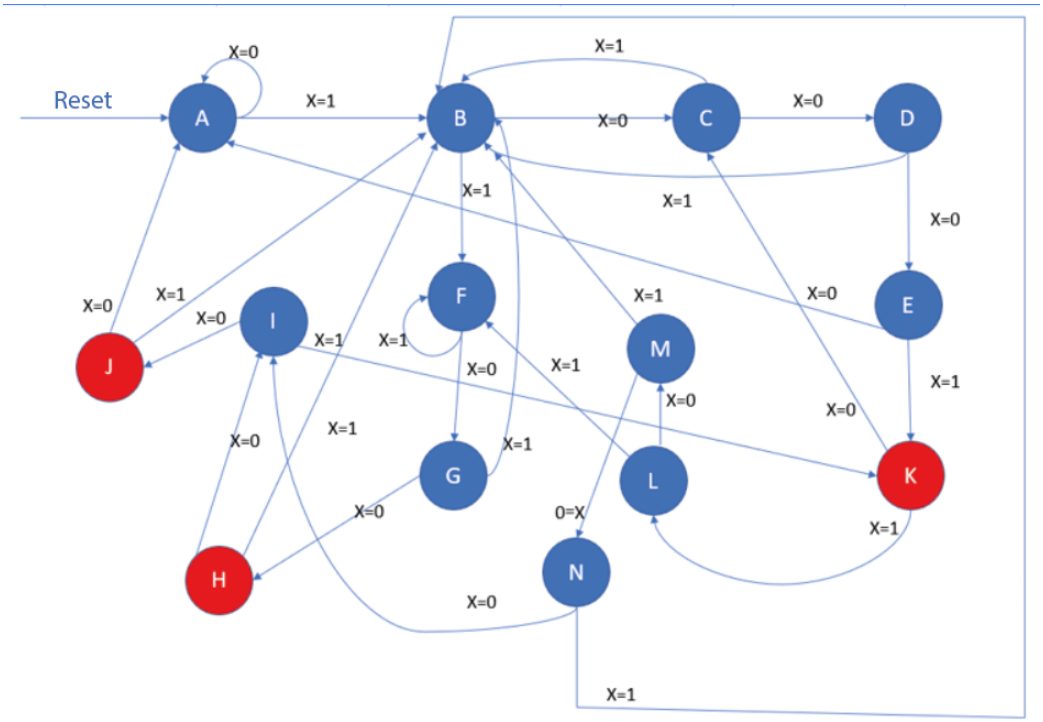


Tabla de estados:

Estado Actual	Estado siguiente				
	X = 0	X = 1	S1	S2	S3
A	A	B	0	0	0
B	C	F	0	0	0
C	D	B	0	0	0
D	E	B	0	0	0
E	A	K	0	0	0
F	G	F	0	0	0
G	H	B	0	0	0
H	I	B	1	0	0
I	J	K	0	0	0
J	A	B	0	1	0
K	C	L	0	0	1
L	M	F	0	0	0
M	N	B	0	0	0
N	I	B	0	0	0

Tabla de expansión:

La siguiente tabla de expansión se realizó usando 4 flip flop de tipo JK, debido a que se obtuvo anteriormente un autómata de 14 nodos.

	ESTADO ACTUAL					ESTADO SIGIENTE											
	Q3	Q2	Q1	Q0	X	Q3n	Q2n	Q1n	Q0n	S1	S2	S3	J3 K3	J2 K2	J1 k1	J0 K0	
A	0	0	0	0	0	0	0	0	0	0	0	0	0X	0X	0X	0X	0
A	0	0	0	0	1	0	0	0	1	0	0	0	0X	0X	0X	1X	1
B	0	0	0	1	0	0	0	1	0	0	0	0	0X	0X	1X	X1	2
B	0	0	0	1	1	0	1	0	1	0	0	0	0X	1X	0X	X0	3
C	0	0	1	0	0	0	0	1	1	0	0	0	0X	0X	X0	1X	4
C	0	0	1	0	1	0	0	0	1	0	0	0	0X	0X	X1	1X	5
D	0	0	1	1	0	0	1	0	0	0	0	0	0X	1X	X1	X1	6
D	0	0	1	1	1	0	0	0	1	0	0	0	0X	0X	X1	X0	7
E	0	1	0	0	0	0	0	0	0	0	0	0	0X	X1	0X	0X	8
E	0	1	0	0	1	1	0	1	0	0	0	0	1X	X1	1X	0X	9
F	0	1	0	1	0	0	1	1	0	0	0	0	0X	X0	1X	X1	10
F	0	1	0	1	1	0	1	0	1	0	0	0	0X	X0	0X	X0	11
G	0	1	1	0	0	0	1	1	1	0	0	0	0X	X0	X0	1X	12
G	0	1	1	0	1	0	0	0	1	0	0	0	0X	X1	X1	1X	13
H	0	1	1	1	0	1	0	0	0	1	0	0	1X	X1	X1	X1	14
H	0	1	1	1	1	0	0	0	1	1	0	0	0X	X1	X1	X0	15
I	1	0	0	0	0	1	0	0	1	0	0	0	X0	0X	0X	1X	16
I	1	0	0	0	1	1	0	1	0	0	0	0	X0	0X	1X	0X	17
J	1	0	0	1	0	0	0	0	0	0	0	1	X1	0X	0X	X1	18
J	1	0	0	1	1	0	0	0	1	0	0	1	X1	0X	0X	X0	19
K	1	0	1	0	0	0	0	1	0	0	1	0	X1	0X	X0	0X	20
K	1	0	1	0	1	1	0	1	1	0	1	0	X0	0X	X0	1X	21
L	1	0	1	1	0	1	1	0	0	0	0	0	X0	1X	X1	X1	22
L	1	0	1	1	1	0	1	0	1	0	0	0	X1	1X	X1	X0	23
M	1	1	0	0	0	1	1	0	1	0	0	0	X0	X0	0X	1X	24
M	1	1	0	0	1	0	0	0	1	0	0	0	X1	X1	0X	1X	25
N	1	1	0	1	0	1	0	0	0	0	0	0	X0	X1	0X	X1	26
N	1	1	0	1	1	0	0	0	1	0	0	0	X1	X1	0X	X0	27

Luego se minimizaron las ecuaciones de excitación para los flip flop JK, empleando mapas de Karnaugh.

J3:

J3(Q3,Q2,Q1,Q0,X) =(9,14)
Zero=(0,1,2,3,4,5,6,7,8,10,11,12,13,15)

Q2Q1 / QOX	"00"	"01"	"11"	"10"
"00"	0	0	0	0
"01"	0	0	0	1
"11"	0	0	0	0
"10"	0	0	1	0

Q3=0

Q2Q1 / QOX	"00"	"01"	"11"	"10"
"00"	X	X	X	X
"01"	X	X	X	X
"11"	X	X	X	X
"10"	X	X	X	X

Q3=1

1) IPS
Q2 Q1'Q0'X
Q2 Q1 Q0 X'

2) IPES
Q2 Q1'Q0'X
Q2 Q1 Q0 X'

3) Expresión minima
J3(Q3,Q2,Q1,Q0,X) = Q2 Q1'Q0'X + Q2 Q1 Q0 X'

K3:

K3(Q3,Q2,Q1,Q0,X) =(18,19,20,23,25,27)
Zero=(16,17,21,22,24,26)

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	X	X	X	X
"01"	X	X	X	X
"11"	X	X	X	X
"10"	X	X	X	X

Q3=0

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	0	1	X	0
"01"	0	0	X	1
"11"	1	1	X	1
"10"	1	0	X	0

Q3=1

1) IPS
Q0 X
Q2 X
Q1 Q0' X'
Q2' Q1' Q0 X'

2) IPES
Q0 X
Q2 X
Q1 Q0' X'
Q2' Q1' Q0 X'

3) Expresión minima

K3(Q3,Q2,Q1,Q0,X) = Q0 X + Q2 X + Q1 Q0' X' + Q2' Q1' Q0 X'

J2:

J2(Q3,Q2,Q1,Q0,X) =(3,6,22,23)
Zero=(0,1,2,4,5,7,16,17,18,19,20,21)

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	0	0	X	X
"01"	0	0	X	X
"11"	1	0	X	X
"10"	0	1	X	X

Q3=0

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	0	0	X	X
"01"	0	0	X	X
"11"	0	1	X	X
"10"	0	1	X	X

Q3=1

1) IPS
Q3' Q1' Q0 X
Q3 Q1 Q0
Q1 Q0 X'

2) IPES
Q3' Q1' Q0 X
Q3 Q1 Q0
Q1 Q0 X'

3) Expresión minima

J2(Q3,Q2,Q1,Q0,X) =Q3' Q1' Q0 X + Q3 Q1 Q0 + Q1 Q0 X'

K2:

K2(Q3,Q2,Q1,Q0,X) =(8,9,13,14,15,25,26,27)
Zero=(10,11,12,24)

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	X	X	0	1
"01"	X	X	1	1
"11"	X	X	1	0
"10"	X	X	1	0

Q3=0

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	X	X	X	0
"01"	X	X	X	1
"11"	X	X	X	1
"10"	X	X	X	1

Q3=1

1) IPS
Q1 X
Q3 X
Q3 Q0
Q1 Q0
Q3' Q1' Q0'
Q2 Q0' X

2) IPES
Q1 X
Q3 X
Q3 Q0
Q1 Q0
Q3' Q1' Q0'

3) Expresión minima

K2(Q3,Q2,Q1,Q0,X) =Q1 X + Q3 X + Q3 Q0 + Q1 Q0 + Q3' Q1' Q0'

J1:

J1(Q3,Q2,Q1,Q0,X) =(2,9,10,17)
Zero=(0,1,3,8,11,16,18,19,24,25,26,27)

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	0	X	X	0
"01"	0	X	X	1
"11"	0	X	X	0
"10"	1	X	X	1

Q3=0

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	0	X	X	0
"01"	1	X	X	0
"11"	0	X	X	0
"10"	0	X	X	0

Q3=1

1) IPS
Q3' Q0 X'
Q3' Q2 Q0' X
Q3 Q2' Q0' X

2) IPES
Q3' Q0 X'
Q3' Q2 Q0' X
Q3 Q2' Q0' X

3) Expresión minima

J1(Q3,Q2,Q1,Q0,X) = Q3' Q0 X' + Q3' Q2 Q0' X + Q3 Q2' Q0' X

K1:

K1(Q3,Q2,Q1,Q0,X) =(5,6,7,13,14,15,22,23)
Zero=(4,12,20,21)

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	X	0	0	X
"01"	X	1	1	X
"11"	X	1	1	X
"10"	X	1	1	X

Q3=0

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	X	0	X	X
"01"	X	0	X	X
"11"	X	1	X	X
"10"	X	1	X	X

Q3=1

1) IPS
Q0
Q3' X

2) IPES
Q0
Q3' X

3) Expresión minima

K1(Q3,Q2,Q1,Q0,X) = Q0 + Q3' X

J0:

J0(Q3,Q2,Q1,Q0,X) =(1,4,5,12,13,16,21,24,25)
Zero=(0,8,9,17,20)

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	0	1	1	0
"01"	1	1	1	0
"11"	X	X	X	X
"10"	X	X	X	X

Q3=0

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	1	0	X	1
"01"	0	1	X	1
"11"	X	X	X	X
"10"	X	X	X	X

Q3=1

1) IPS
Q3'Q2'X
Q3' Q1
Q3 Q1' X'
Q3 Q2
Q1 X

2) IPES
Q3'Q2'X
Q3' Q1
Q3 Q1' X'
Q3 Q2
Q1 X

3) Expresión minima
J0(Q3,Q2,Q1,Q0,X) = Q3'Q2'X + Q3' Q1 + Q3 Q1' X' + Q3 Q2 + Q1 X

K0:

K0(Q3,Q2,Q1,Q0,X) =(2,6,10,14,18,22,26)
Zero=(3,7,11,15,19,23,27)

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	X	X	X	X
"01"	X	X	X	X
"11"	0	0	0	0
"10"	1	1	1	1

Q3=0

Q2Q1 / Q0X	"00"	"01"	"11"	"10"
"00"	X	X	X	X
"01"	X	X	X	X
"11"	0	0	X	0
"10"	1	1	X	1

Q3=1

1) IPS
X'

2) IPES
X'

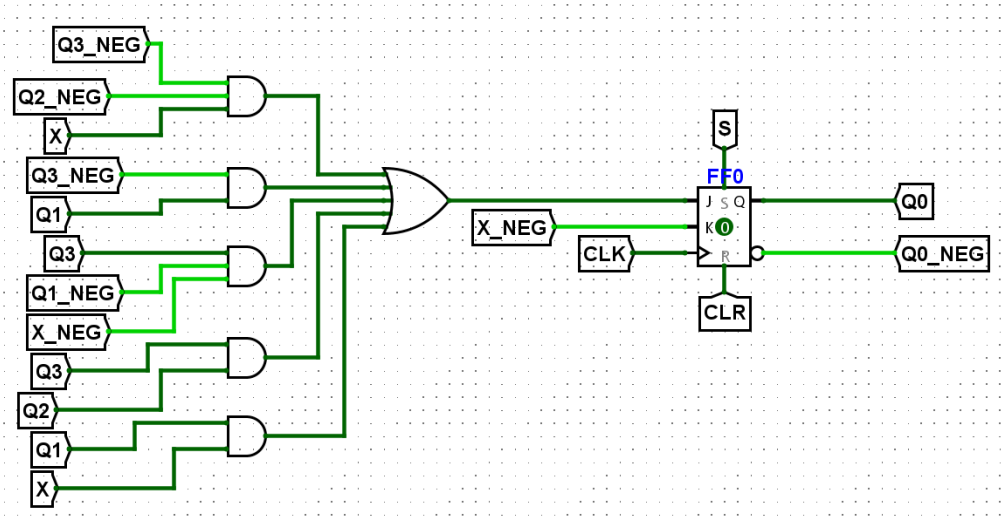
3) Expresión minima
K0(Q3,Q2,Q1,Q0,X) = X'

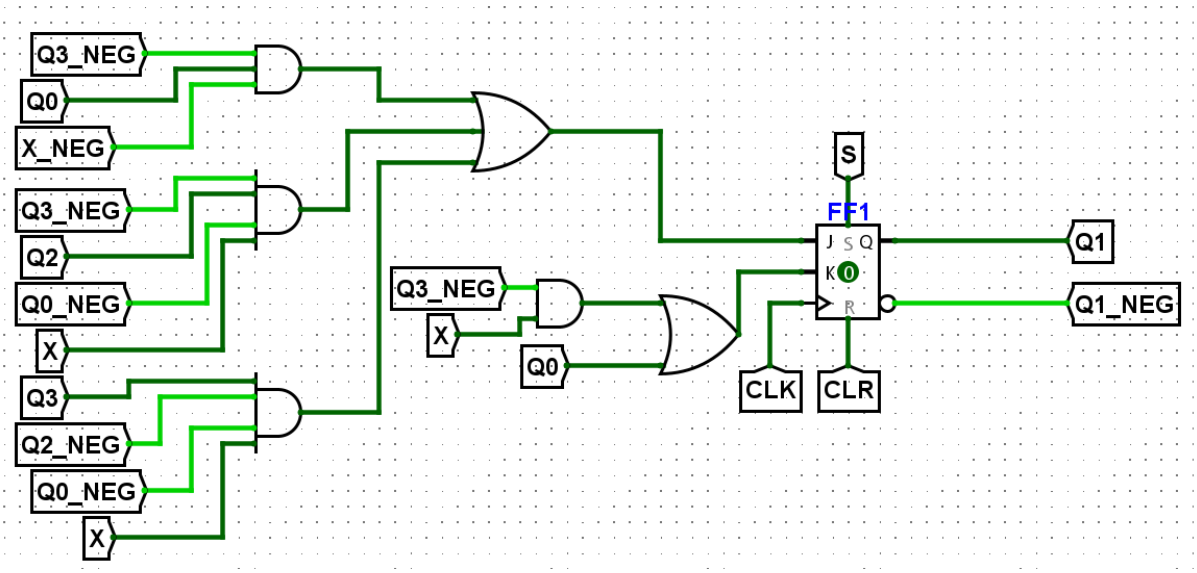
Luego se realizó un análisis de las salidas s1, s2, s3 donde se obtuvieron las siguientes ecuaciones según la tabla de expansión:

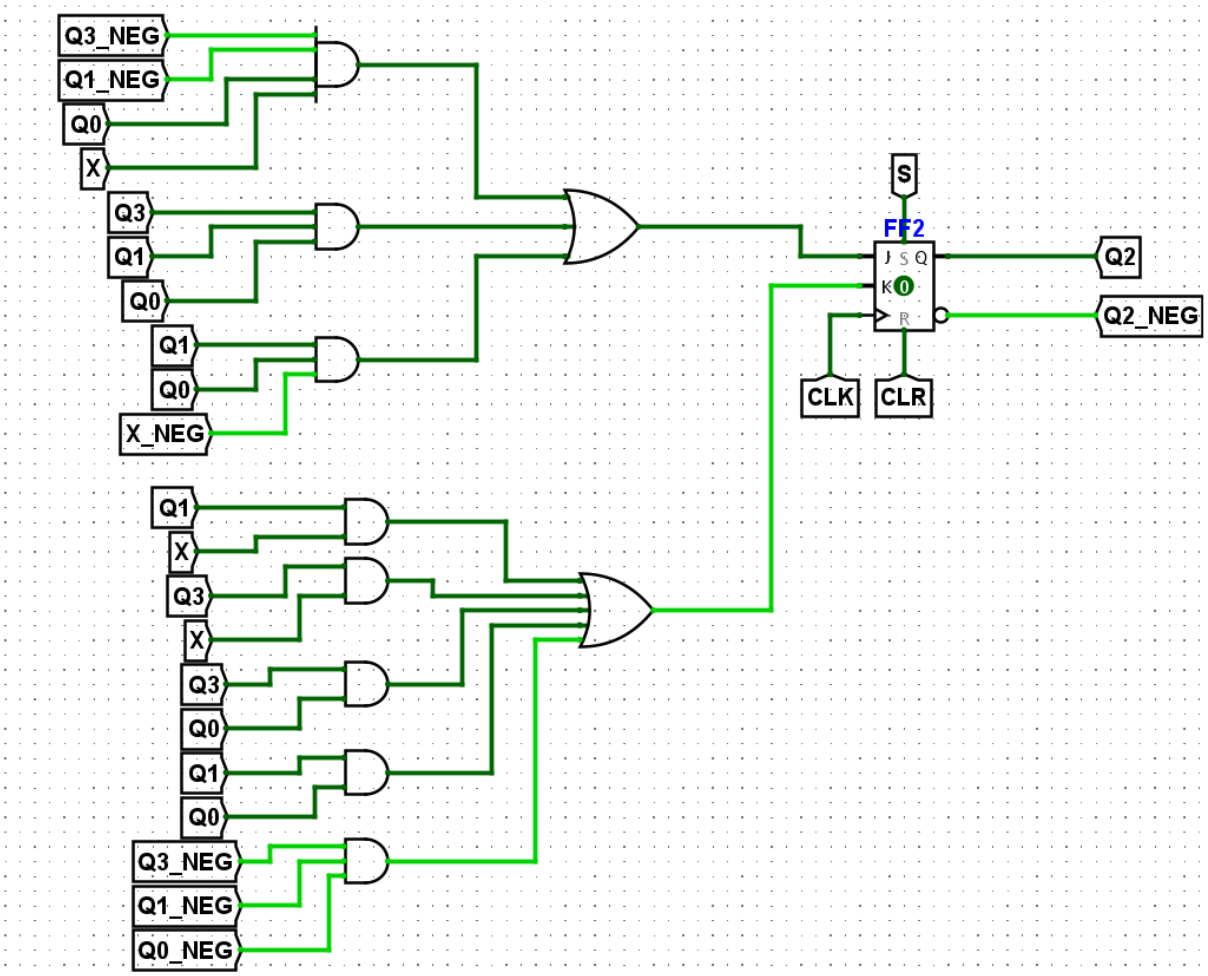
$$S1 = (Q3,Q2,Q1,Q0) = Q3'\,Q2\,Q1\,Q0$$
$$S2 = (Q3,Q2,Q1,Q0) = Q3\,Q2'\,Q1\,Q0'$$
$$S3 = (Q3,Q2,Q1,Q0) = Q3\,Q2'\,Q1'\,Q0$$

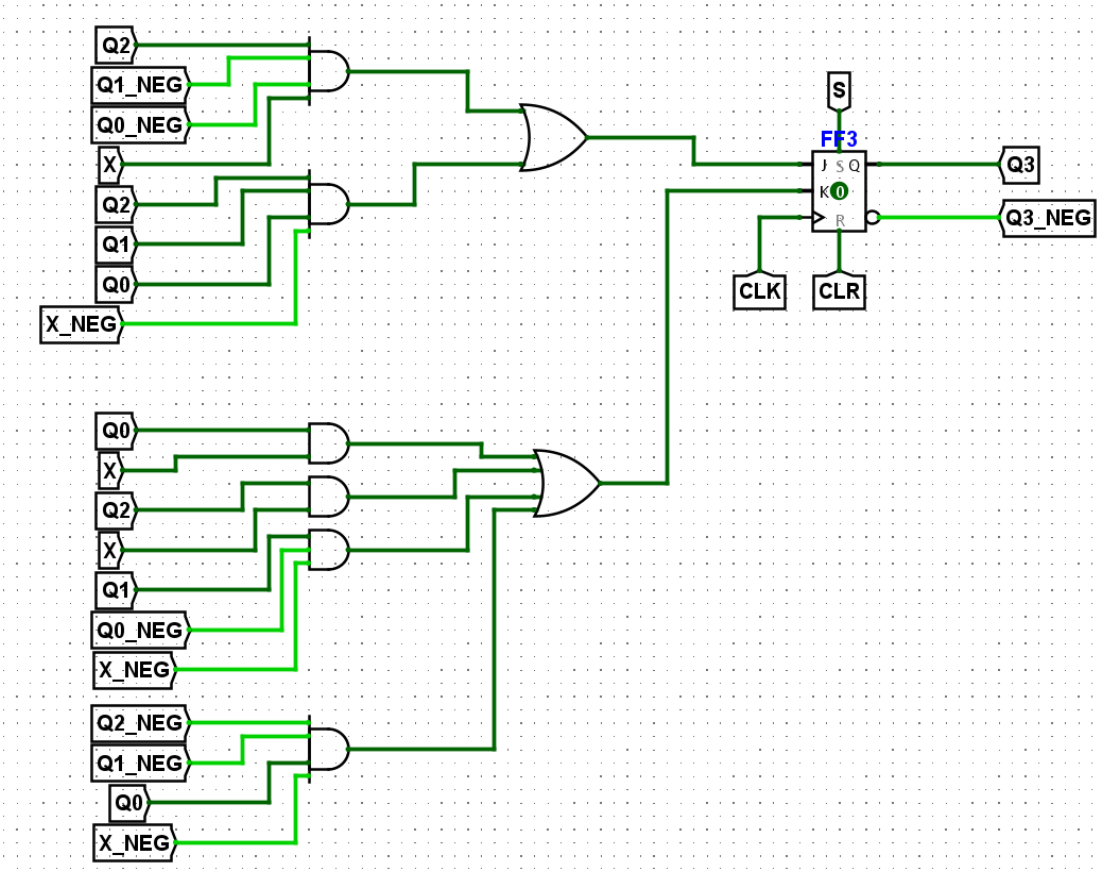
Implementación circuital empleando solo compuertas AND, OR y Not

Detector de las secuencia (1100, 10001, 110000)

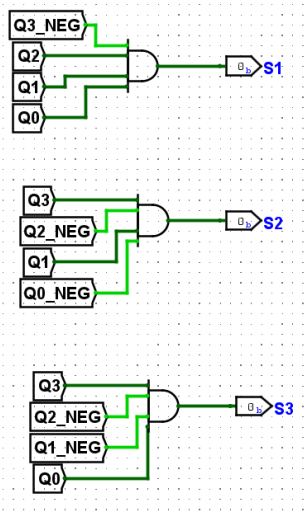




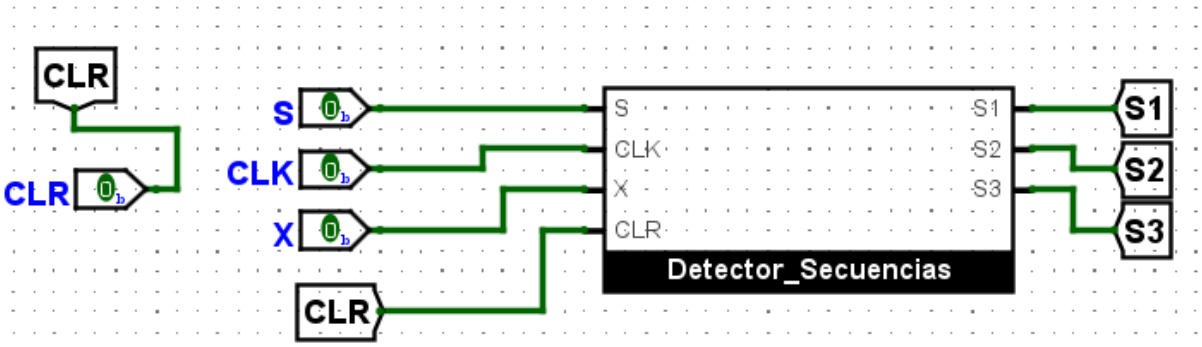

Q3_NEG
Q1_NEG
Q0
X
Q3
Q1
Q0
Q1
Q0
X_NEG
Q1
X
Q3
X
Q3
Q0
Q1
Q0
Q3_NEG
Q1_NEG
Q0_NEG
S
FF2
J S Q
K 0
R
Q2
Q2_NEG
CLK
CLR



Salidas del circuito Moore:



Detector de secuencias modularizado



Componente de visualización

Se usó un decodificador BCD a 7 segmentos para convertir un número binario a un numero decimal codificado en binario (BCD) en una señal de salida que activa los segmentos necesarios de un display de 7 segmentos para mostrar el dígito decimal correspondiente.

El decodificador consta de cuatro entradas (A, B, C, D) y siete salidas que se corresponden con los segmentos del display (a, b, c, d, e, f, g). Cada entrada BCD representa un dígito decimal (0-9) en binario, y las salidas activan los segmentos correspondientes para mostrar ese dígito en el display.

El proceso de decodificación se realiza mediante una tabla de verdad que relaciona cada combinación de entradas BCD con las salidas correspondientes.

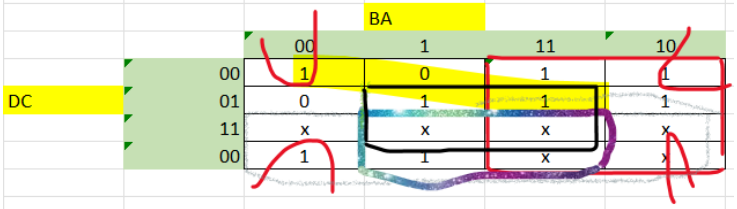
Tabla para decodificador BCD a 7 segmentos

#	D	C	B	A	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1	1
7	0	1	1	1	1	1	1		0	0	1
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1
10	x	x	x	x	x	x	x	x	x	x	

Expresiones mínimas obtenidas con mapas de Karnaugh

a(D,C,B,A)={0,2,3,5,6,7,8,9 + DONT CARE 10,11,12,13,14,15}

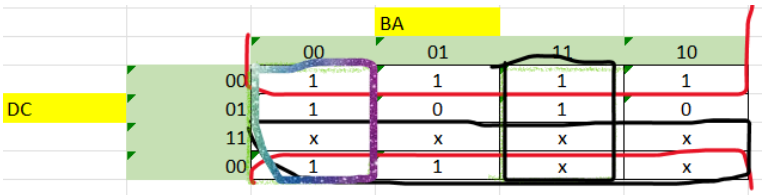
		BA			
		00	1	11	10
DC	00	1	0	1	1
	01	0	1	1	1
	11	x	x	x	x
	00	1	1	x	x



a = Sum(0,2,3,5,6,7,8,9)	IPS	IPS
	C'A'	C'A'
	DC'	a= C' A' + D + CA + B
	DC'	
	DA	
	CA	
	D'CB	
	D'B	
	B	

b(D,C,B,A)={0,1,2,3,4,7,8,9 + DONT CARE 10,11,12,13,14,15}

		BA			
		00	01	11	10
DC	00	1	1	1	1
	01	1	0	1	0
	11	x	x	x	x
	00	1	1	x	x



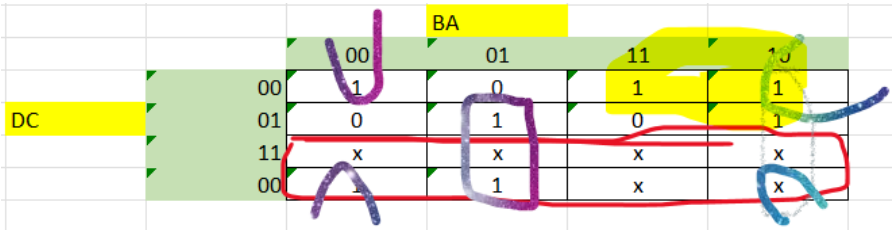
b = suma (0,1,2,3,4,7,8,9)		b = D'C' + D + BA + B'A'
	IPS	IES
	D'C'	D
	B'A'	D'C'
	D'C'	BA
	B'A'	B'A'
	BA	
	C'A'	

c(D,C,B,A)={0,1,3,4,5,6,7,8,9 + DONT CARE 10,11,12,13,14,15}

		BA			
		00	01	11	10
DC	00	1	1	1	0
	01	1	1	x	1
	11	x	x	x	x
	00	1	1	x	x

C = suma (0,1,3,4,5,6,7,8,9)	IPS	IPE	c = B' +A + D'C
	B'	B'	
	A	A	
	B'C	B'C	

d(D,C,B,A) ={0,2,3,5,6,8,9 + DONT CARE 10,11,12,13,14,15}



		BA			
		00	01	11	10
DC	00	1	0	1	1
	01	0	1	0	1
	11	x	x	x	x
	00	1	1	x	x

d = suma(0,2,3,5,6,8,99	IPS	IPE
	C1A'	C'A'
	D	CB'A
	DB'	BA'
	CB'A	
	BA'	
	D'C'	

d = C'A' + D + CB' A + D'C'B + BA'

e(D,C,B,A)={0,2,6,8 + DONT CARE 10,11,12,13,14,15}

DC \ BA	00	01	11	10
00	1	0	0	1
01	0	0	0	1
11	x	x	x	x
00	1	0	x	x

e = suma(0,2,6,8)	IPS	IPE
zeros (1,3,4,5,7,9)	C'A'	C'A'
	BA'	BA'

e= C'A' + BA'

f(D,C,B,A)={0,4,5,6,8,9 + DONT CARE 10,11,12,13,14,15}

DC \ BA	00	01	11	10
00	1	0	0	0
01	1	1	0	1
11	x	x	x	x
00	1	1	x	x

f = (0,4,5,6,8,9) Zeros(1,2,3,7)

IPS	IPE
B'A'	CB'
D	BA
DB'	´CBA'
CBA'	D
CB'	

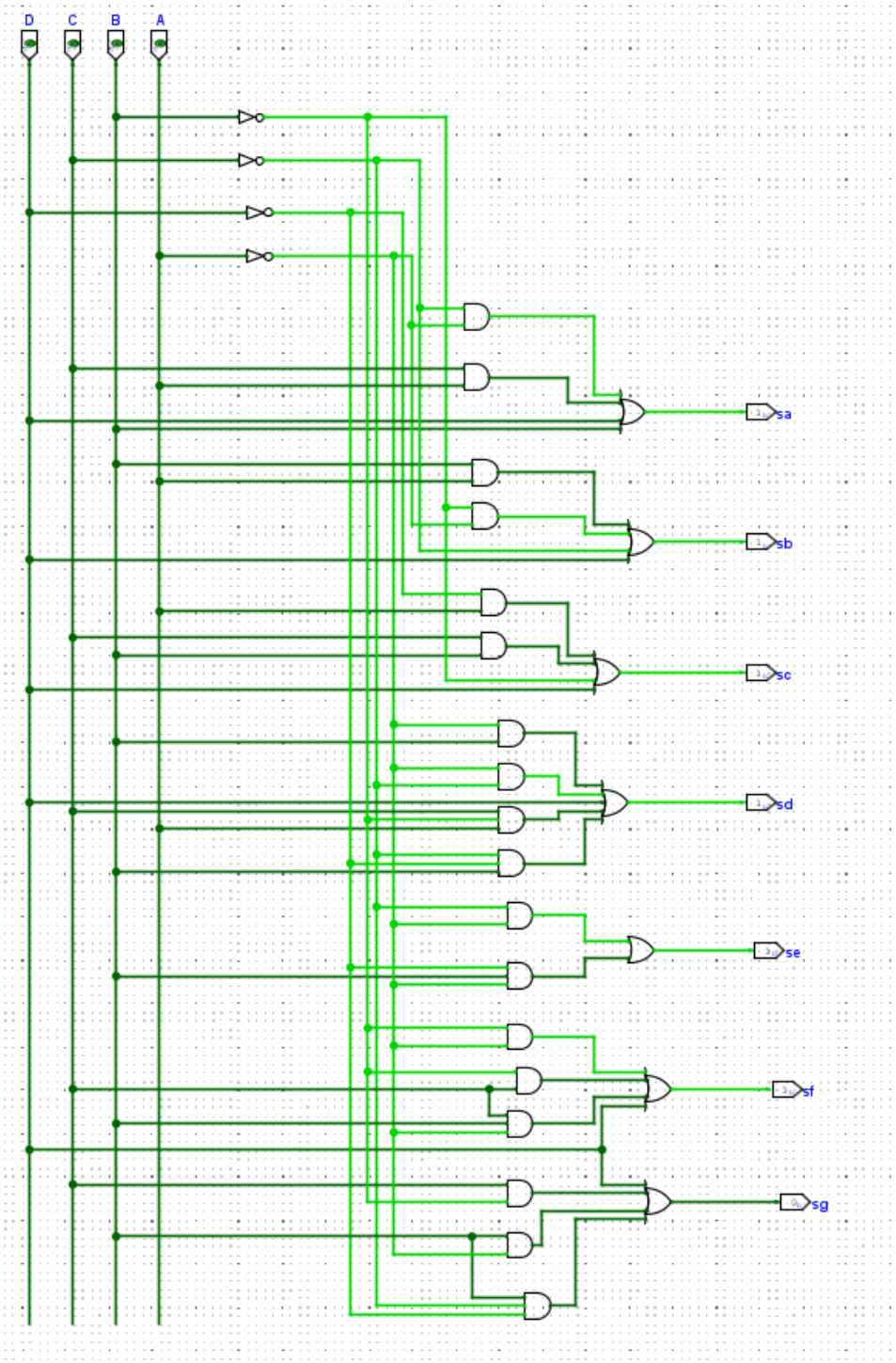
f = B'A' + CBA' + CB´+D

g(D,C,B,A)={2,3,4,5,6,8,6,8,9 + DONT CARE 10,11,12,13,14,15}

DC \ BA	00	01	11	10
00	0	0	1	1
01	1	1	1	1
11	x	x	x	x
00	1	1	x	x

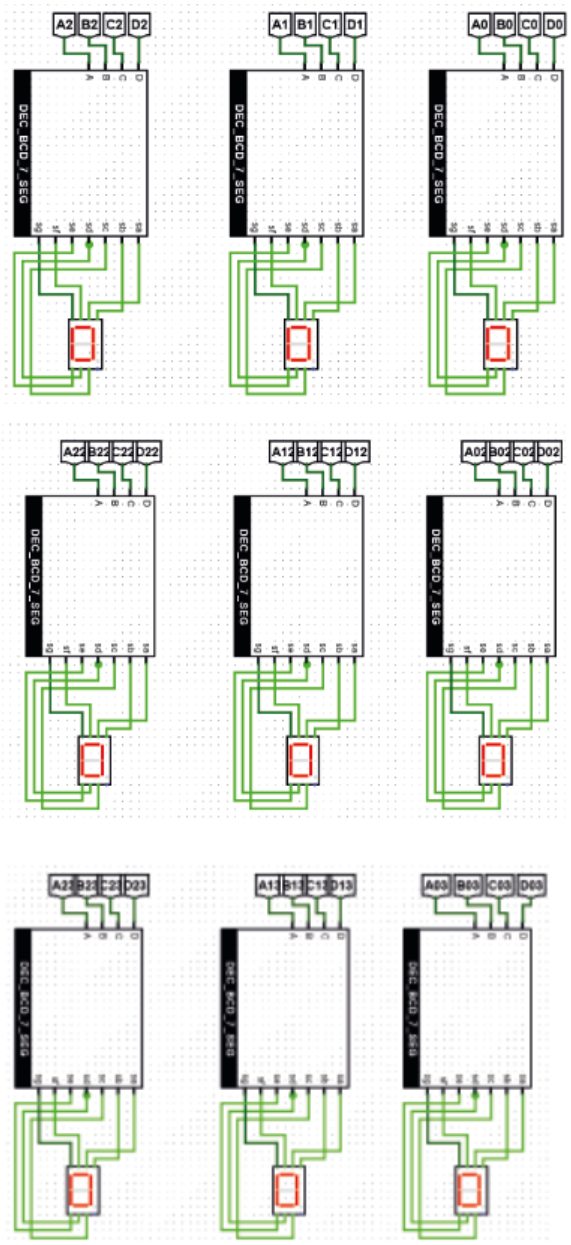
IPS	IPE	g = C+D+B
C	C	
D	D	
B	B	

Decodificador BCD a 7 segmentos



Visualización

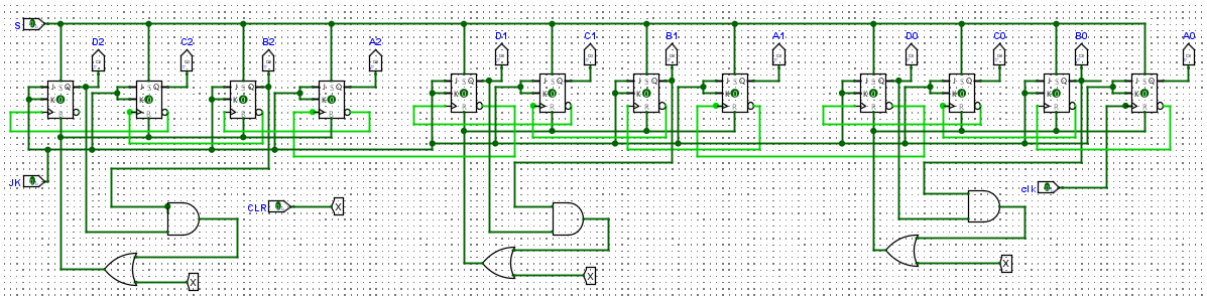
Se usó un display de siete segmentos, donde cada segmento está formado por un diodo led, que se activa con un ‘1’ lógico y se apaga con un ‘0’ lógico.



Contador binario:

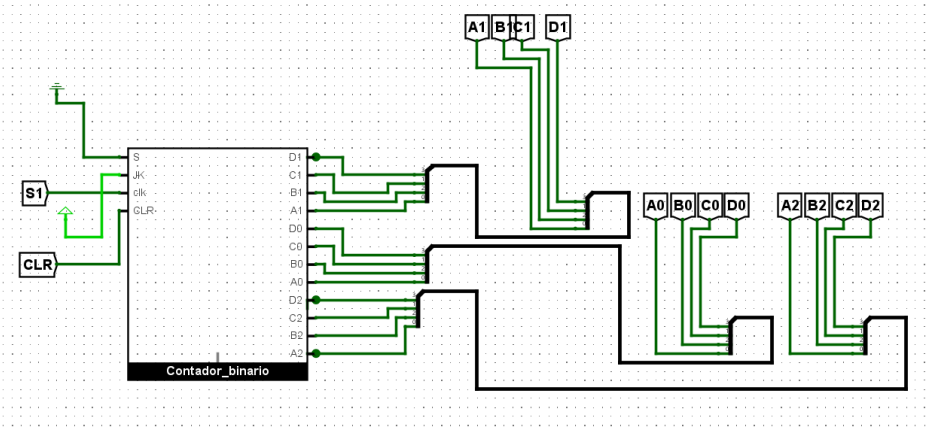
El contador de décadas, es un contador binario, construido a partir de los flip flop jk,
La entrada de reloj se conecta al primer flip flop del contador, y la salida más alta se conecta a la entrada del siguiente flip flop de la cascada, de esta manera construimos el contador de décadas. Como en realidad esta conexión nos lleva es a un contador binario se implementó un circuito auxiliar que detecta cuándo una etapa contadora llegó a la combinación ‘ 1010 ‘, de tal forma que el circuito auxiliar forza al contador a reiniciar la cuenta.

Obsérvese que para que el contador sea ascendente se conecta la salida negada de cada flip flop a la entrada del reloj del subsiguiente flip flop.

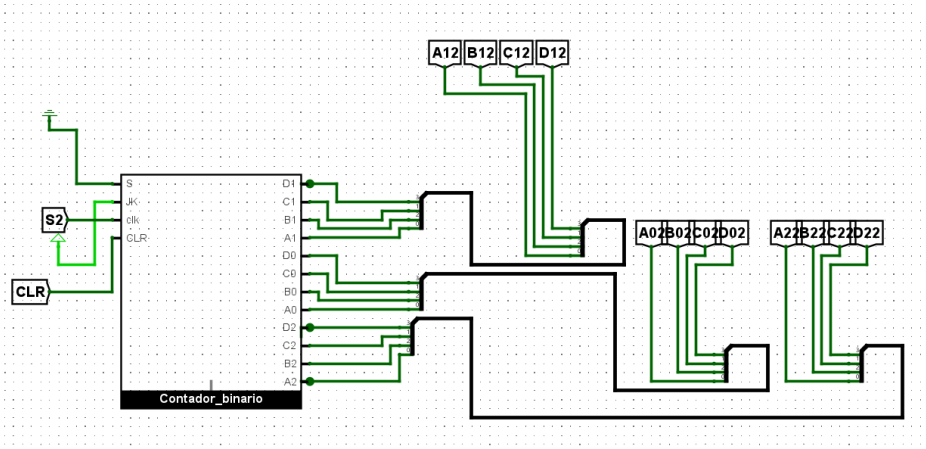


Contadores modularizados:
Como se debe llevar la cuenta de cada ocurrencia de secuencias se usa un módulo contador por cada secuencia detectada.

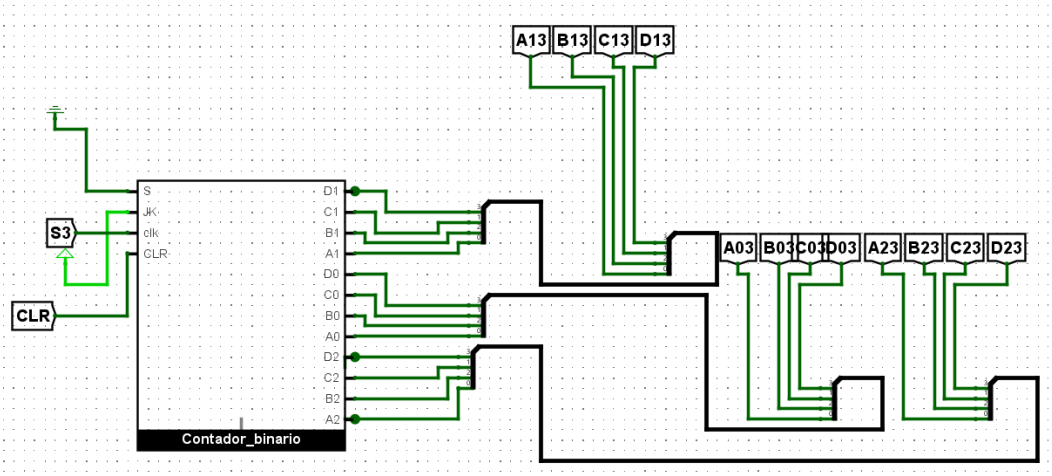
Módulo contador 1



Módulo contador 2:



Módulo contador 3:



Conclusiones

En este laboratorio pudimos poner en práctica los conceptos de diseño de circuitos lógico combinacionales y secuenciales estudiados en las clases magistrales.

El laboratorio nos exigió trabajo de investigación y pudimos cimentar los conocimientos.

Herramientas utilizadas

Simulador Logisim
Apuntes de clase
Documentos de soporte de la asignatura presentes en ude@.