

Laboratório 5:



Análise e síntese de circuitos sequenciais assíncronos no modo assíncrono

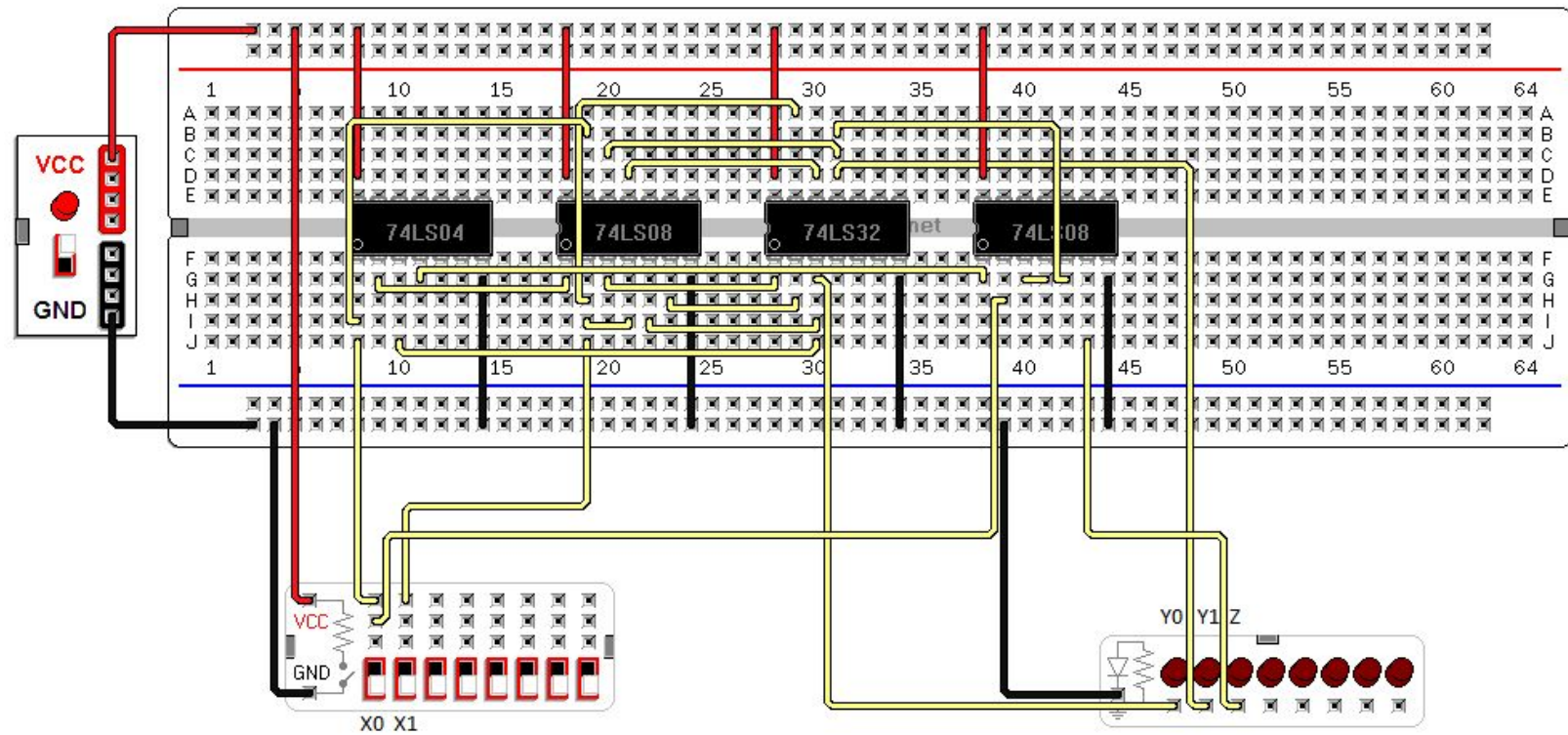
Professor:
Osamu Saotome

Alunos:
Rodrigo Alves de Almeida
(rodrigoalmeida@gmail.com)
Eduardo Menezes Moraes
(eduardomenezesm@msn.com)

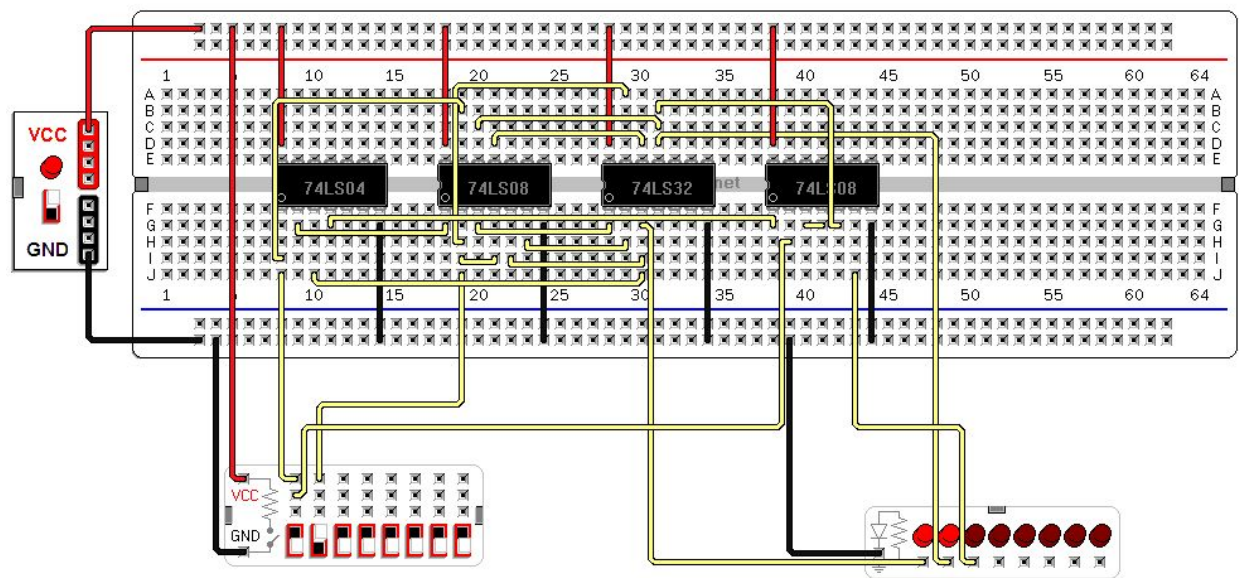
Comp 22

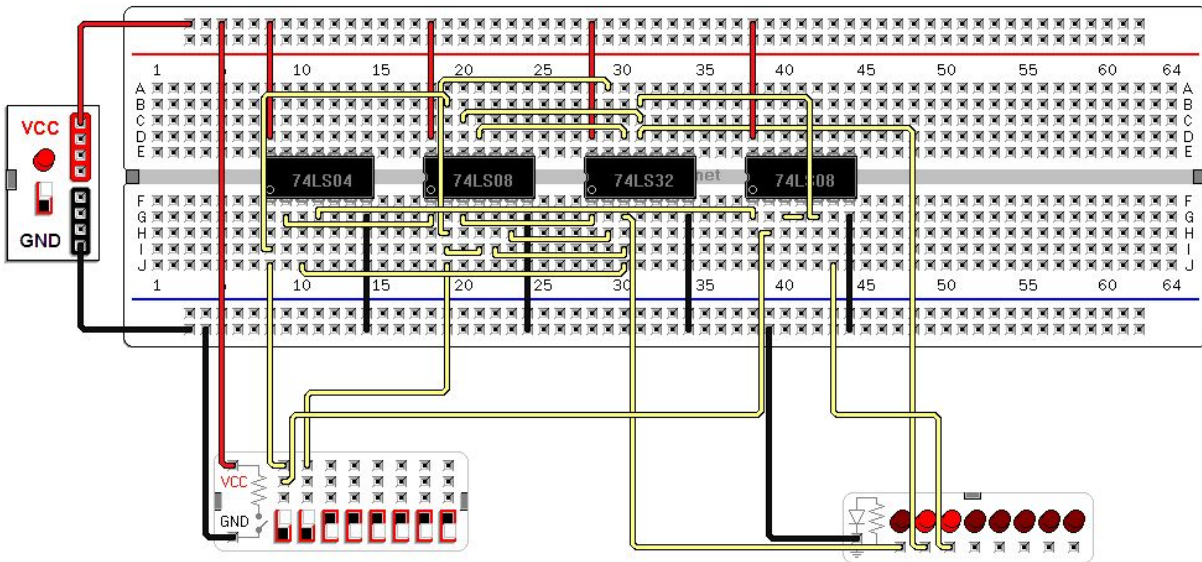
4.2) Montagem no protoboard

A montagem do circuito no protoboard fica da seguinte maneira:

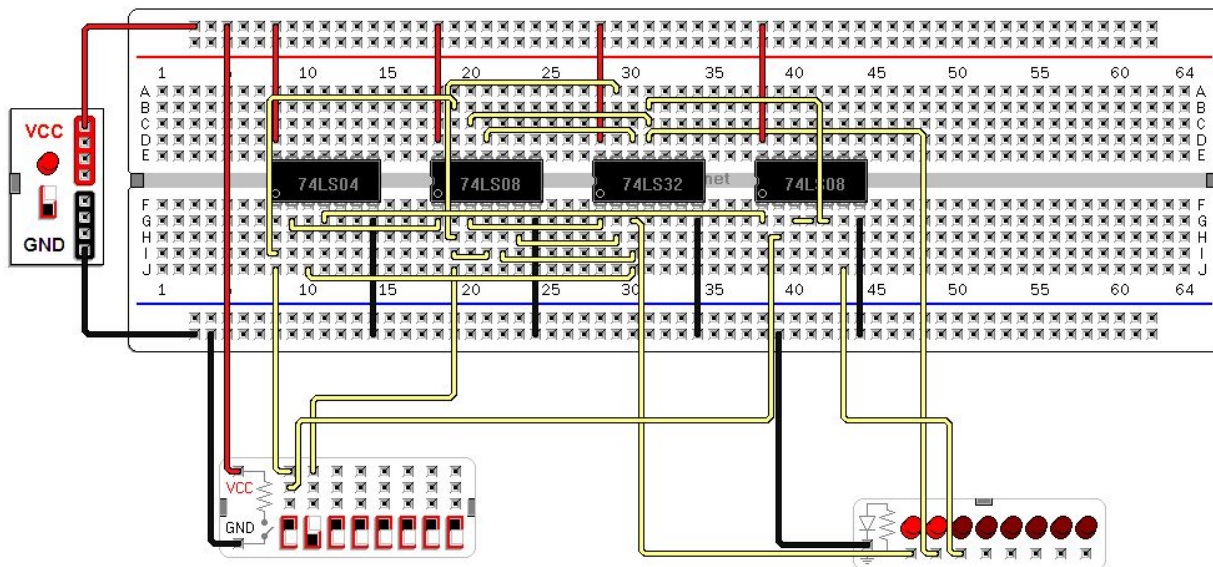


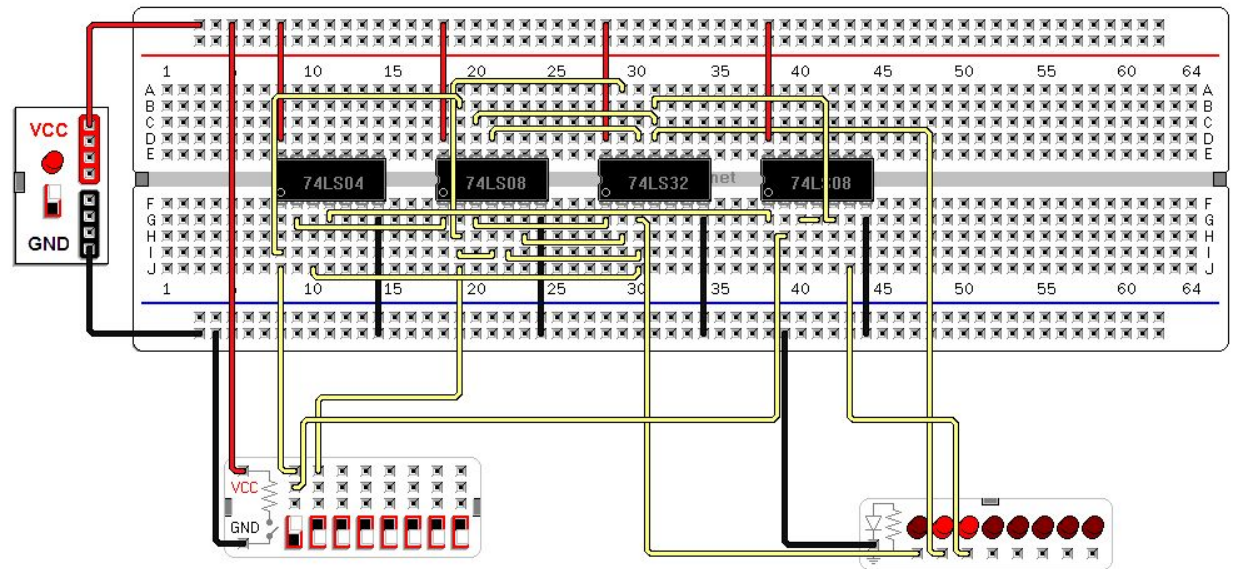
Do estado 00 ($X_0X_1 = 00$), podemos ligar X0 e continuar em 00, ligar X1 e ir para 11 ou ligar X0 depois X1 e ir para 01:



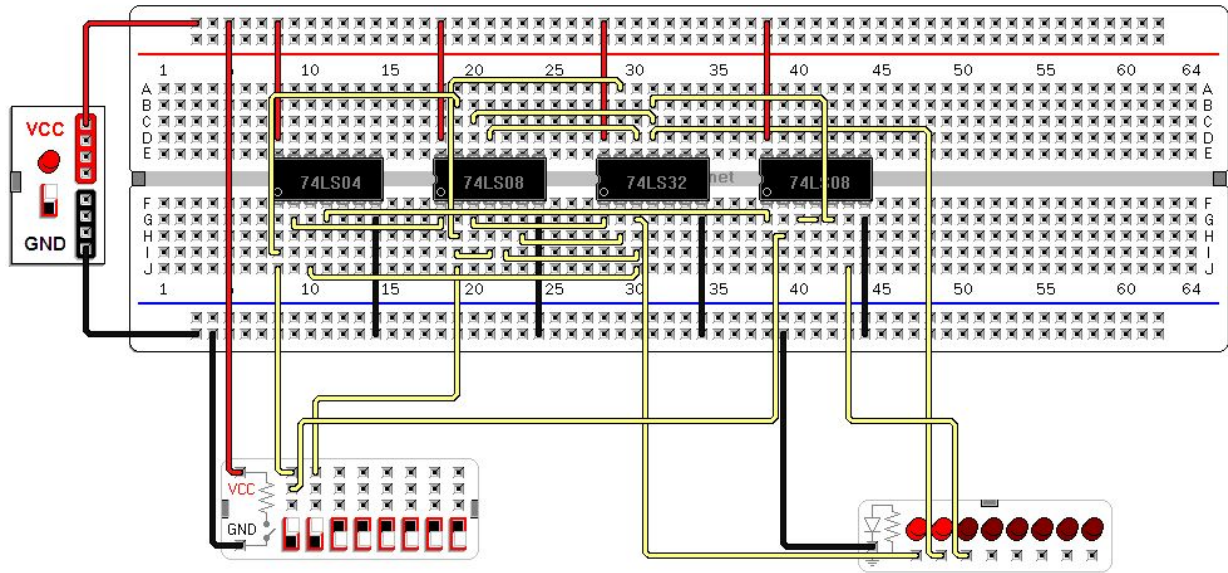
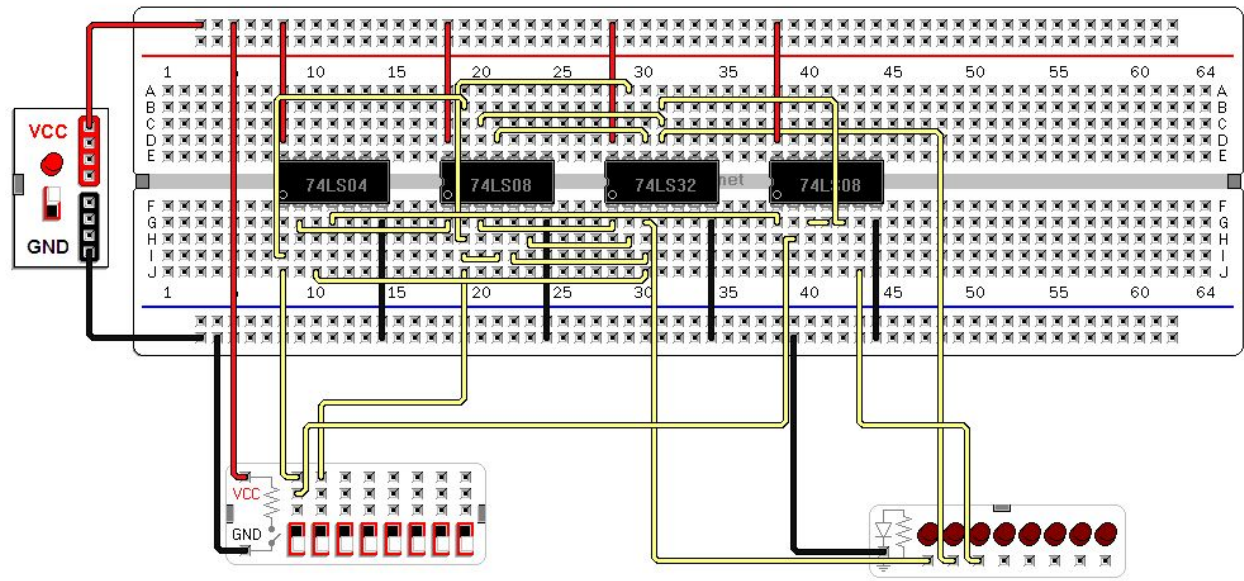


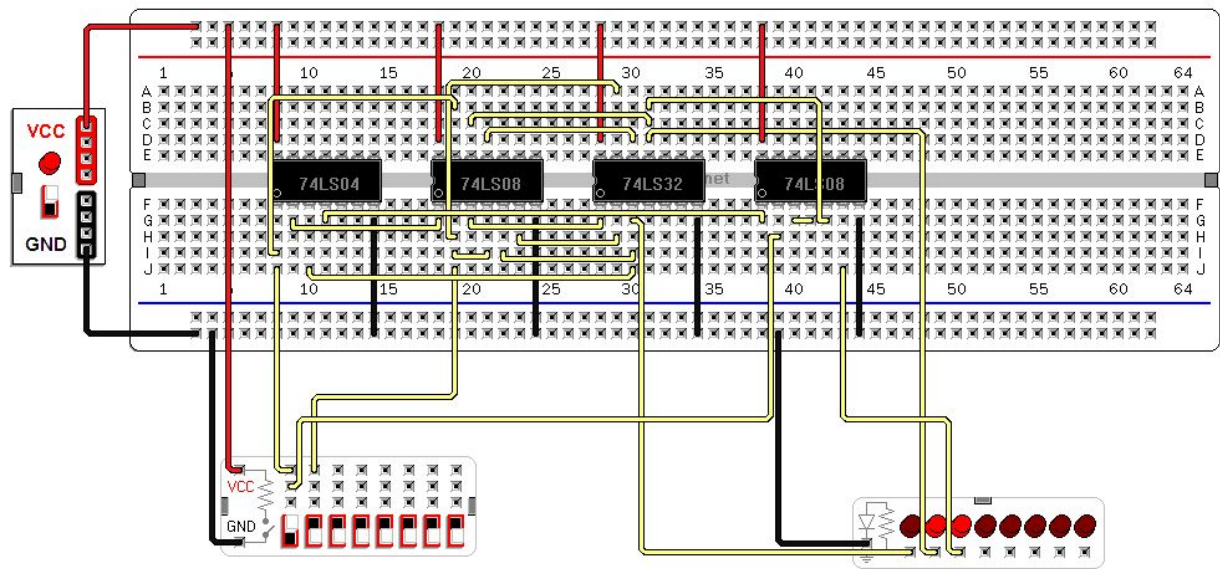
Do estado 01 ($X_0X_1 = 11$), podemos desligar X_0 e ir para 11, desligar X_1 e continuar em 01 ou desligar X_1 depois desligar X_0 e voltar para 00:





Do estado 11 ($X_0X_1 = 01$), podemos desligar X_1 e voltar para 00, ligar X_0 e continuar em 11 ou ligar X_0 depois desligar X_1 e ir para o estado 01:





5.1) Projeto e simulação de um Flip-Flop tipo D

a) Mostre a tabela primitiva de estados (Modelo Moore):

Analisando o tabela de operações fornecida para o Flip-Flop, obteve-se a tabela primitiva de estados para o modelo de Moore.

<i>Estado \ CLK D</i>	00	01	11	10	<i>z</i>
<i>a</i>	a	d	-	b	0
<i>b</i>	a	-	c	b	0
<i>c</i>	-	d	c	b	0
<i>d</i>	a	d	e	-	0
<i>e</i>	-	g	e	f	1
<i>f</i>	h	-	e	f	1
<i>g</i>	h	g	e	-	1
<i>h</i>	h	g	-	b	1

b) Mostre a tabela de estados minimizados, com assinalamento de variáveis de estado livre de corrida.

Através dessa tabela foi realizada a sua minimização. Primeiramente segue a tabela de pares de estados compatíveis parcial (TPEC parcial).

b	V						
c	V	V					
d	V	<i>c-e</i>	<i>c-e</i>				
e	X	X	X	X			
f	X	X	X	X	V		
g	X	X	X	X	V	V	
h	X	X	X	X	<i>f-b</i>	<i>f-b</i>	V

a	b	c	d	e	f	g
----------	----------	----------	----------	----------	----------	----------

Após isso, obtém-se a tabela de pares de estados compatíveis final (TPEC final).

b	V					
c	V	V				
d	V	X	X			
e	X	X	X	X		
f	X	X	X	X	V	
g	X	X	X	X	V	V
h	X	X	X	X	X	X
	a	b	c	d	e	f

Com isso, encontra-se a tabela de classes de estados de máxima compatibilidade (TECMC).

Estados	Estados Compatíveis	Classes de máxima compatibilidade
g	h	(g, h)
f	g	(f,g) (g,h)
e	g, f	(e, f, g) (g, h)
b	c	(b, c) (e, f, g) (g, h)
a	d, c, b	(a, d) (a, b, c) (e, f, g) (g, h)

Desse modo, escolheu-se as classes, para a cobertura mínima.

Estados	Classes
A	(a, b, c)
B	d
C	(f, g, e)

D	h
---	---

Com isso, temos a tabela de fluxo de estados minimizada:

<i>Estado \ CLK D</i>	00	01	11	10	z
A	A	B	A	A	0
B	A	B	C	-	0
C	B	C	C	D	1
D	B	C	-	A	1

Assim, chega-se ao assinalamento livre de corrida crítica, aonde vamos utilizar a variável Q_n para ser variável de estado, e a P_n para também codificar os estados. Note que a convenção a seguir e as adjacências observadas na tabela acima fazem com que estejamos livre de corrida crítica.

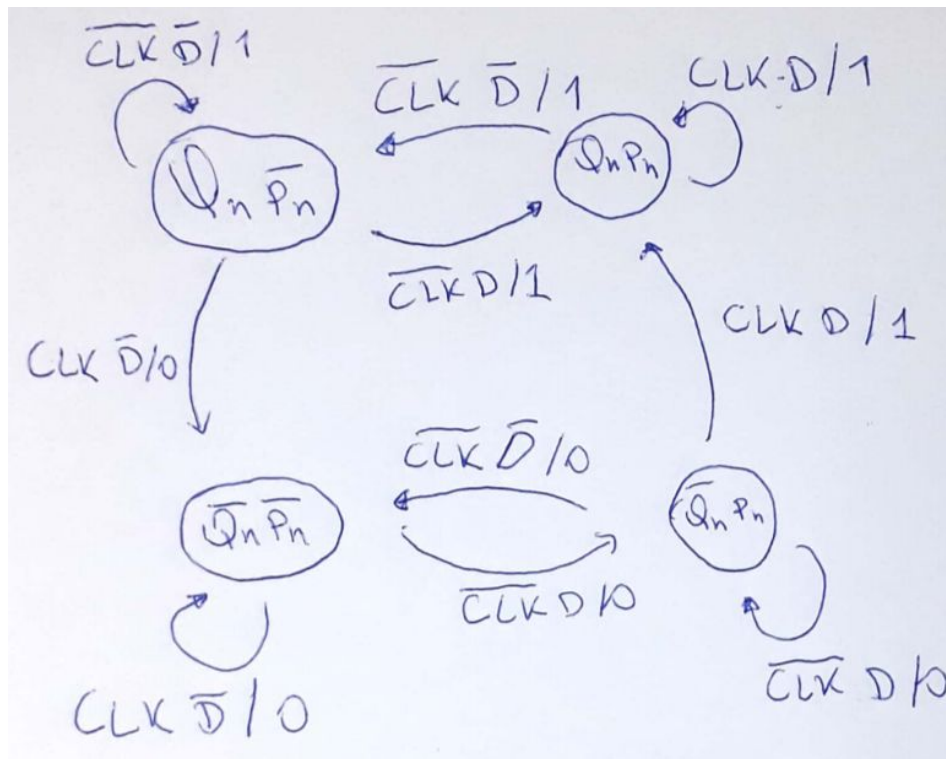
$Q_n \setminus P_n$	0	1
0	A	B
1	C	D

Finalmente, chega-se a tabela de fluxo de estados minimizada com assinalamento livre de corrida crítica.

$Q_n P_n \setminus CLK D$	00	01	11	10
00	00	01	00	00
01	00	01	11	-
11	10	11	11	11
10	10	11	-	00

c) Mostre o diagrama de estados.

Finalmente, chegamos ao diagrama de transição de estados.



d) Verifique se há Hazard essencial do tipo estável.

De acordo com a regra de *Unger*, pode haver hazard estável na transição entre alguns estados. Como, por exemplo, do estado 00 para 01, o qual pode ir erroneamente para o estado 11.

e) Verifique se há Hazard essencial do tipo transiente.

Não há. Nota-se, da tabela de fluxo de estados, que todos os estados estáveis adjacentes a dois determinados estados totais possuem a mesma saída.

f) Obtenha as funções minimizadas de próximo estado e de saída que estejam livres de Hazard lógico combinacional (estático e dinâmico), considerando SIC (single input change).

Segue as tabelas minimizadas, respectivamente, para o próximo estado e para o estado de saída, conforme pedido.

$Q_n P_n \setminus \text{CLK D}$	00	01	11	10
00	0	0	0	0
01	0	0	1	-
11	1	1	1	1
10	1	1	-	0

$Q_n P_n \setminus \text{CLK D}$	00	01	11	10
00	0	1	0	0
01	0	1	1	-
11	0	1	1	1
10	0	1	-	0

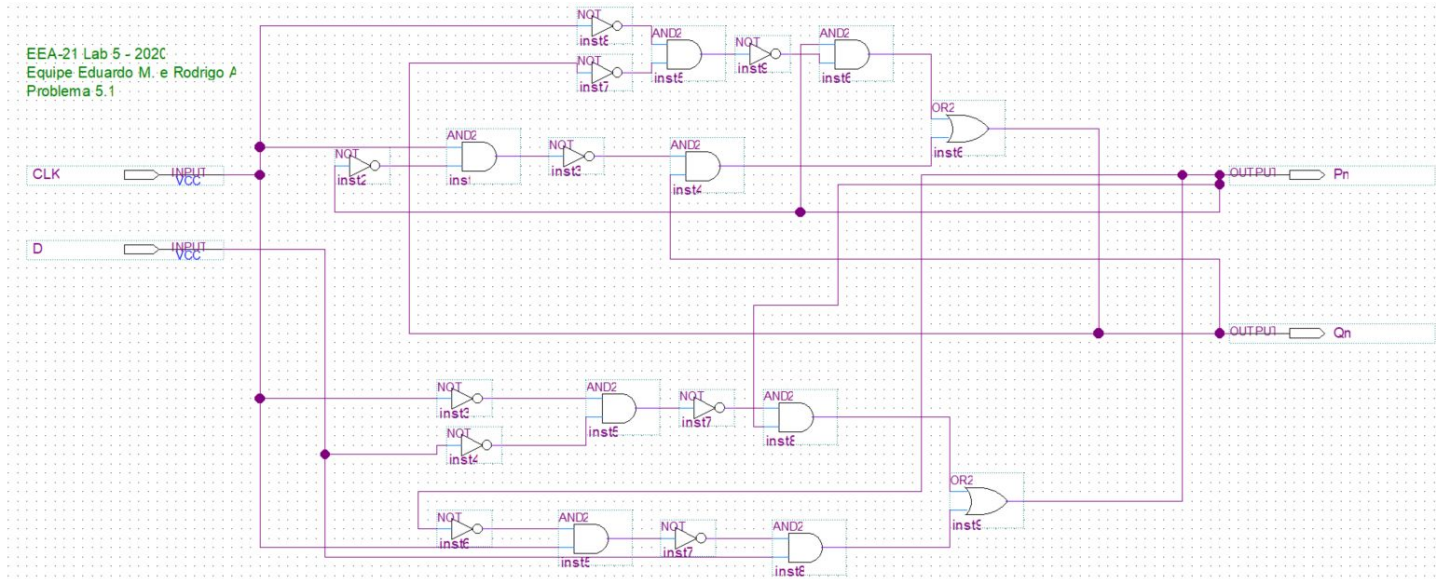
Desse modo, chegamos às funções minimizadas do próximo estado e da saída, conforme pedido:

$$Q_{n+1} = Q_n(P_n'CLK)' + P_n(Q_n'(CLK))'$$

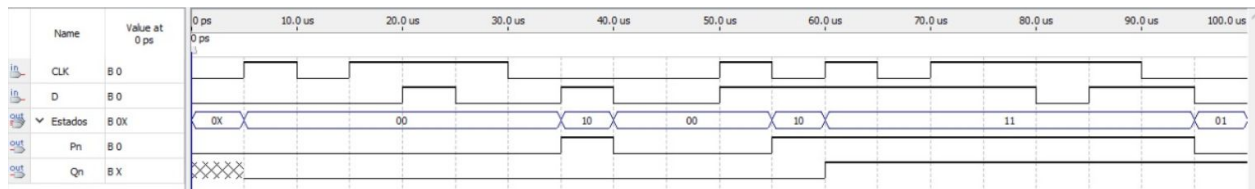
$$P_{n+1} = P_n(D'(CLK))' + D(P_n'CLK)'$$

g) Obtenha o diagrama esquemático.

Através das funções mínimas, obteve-se o diagrama esquemático por meio do software *Quartus*.



h) Obtenha o diagrama de temporização, testando todas as mudanças de estado.



5.2) Projeto e simulação de um Flip-Flop tipo JK

a) Mostre a tabela primitiva de estados (Modelo Moore):

Analisando o tabela de operações fornecida para o Flip-Flop, obteve-se a tabela primitiva de estados para o modelo de Moore.

<i>Estado \ JK</i>	CLK = 0				CLK = 1				Q_n
	00	01	11	10	00	01	11	10	
a	a	b	-	c	d	-	-	-	0
b	a	b	e	-	-	f	-	-	0
c	a	-	e	c	-	-	-	g	0
d	a	-	-	-	d	f	-	n	0
e	-	b	e	c	-	-	h	-	0
f	-	b	-	-	d	f	i	-	0
g	-	-	-	j	k	-	h	g	1
h	-	-	m	-	-	l	h	g	1
i	-	-	e	-	-	f	i	n	0
j	o	-	m	j	-	-	-	g	1
k	o	-	-	-	k	l	-	g	1
l	-	p	-	-	k	l	h	-	1
m	-	p	m	j	-	-	i	-	1
n	-	-	-	c	d	-	i	n	0
o	o	p	-	j	k	-	-	-	1
p	o	p	m	-	-	f	-	-	1

b) Mostre a tabela de estados minimizados, com assinalamento de variáveis de estado livre de corrida.

Através dessa tabela foi realizada a sua minimização. Primeiramente segue a tabela de pares de estados compatíveis (TPEC).

b	V														
c	V	V													
d	V	V	X												
e	V	V	V	V											
f	V	V	V	V	X										
g	X	X	X	X	X	X									
h	X	X	X	X	X	X	V								
i	V	V	X	V	X	V	X	X							
j	X	X	X	X	X	X	V	V	X						
k	X	X	X	X	X	X	V	V	X	V					
l	X	X	X	X	X	X	V	V	X	V	V				
m	V	X	X	X	X	X	X	X	X	V	V	X			
n	X	V	X	V	X	V	X	X	X	X	X	X	X		
o	X	X	X	X	X	X	V	V	X	V	V	V	V	X	
p	o	X	X	X	X	X	V	X	X	V	X	X	V	X	V
	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o

Similarmente ao item 5.1, foi feita a tabela de classes de estados de máxima compatibilidade (TECMC), através dela, para a cobertura mínima escolheu-se as classes:

Estados	Classes
A	(a, b, d, f, i, n)
B	(c, e)
C	(m, p)
D	(g, h, j, k, l, o)

<i>Estado \ JK</i>	CLK = 0				CLK = 1				Q_n
	00	01	11	10	00	01	11	10	
A	A	A	B	B	A	A	A	A	0
B	A	A	B	B	-	-	D	D	0
C	D	C	C	D	-	A	A	-	1
D	D	C	C	D	D	D	D	D	1

Utilizando da mesma convenção da questão 5.1,

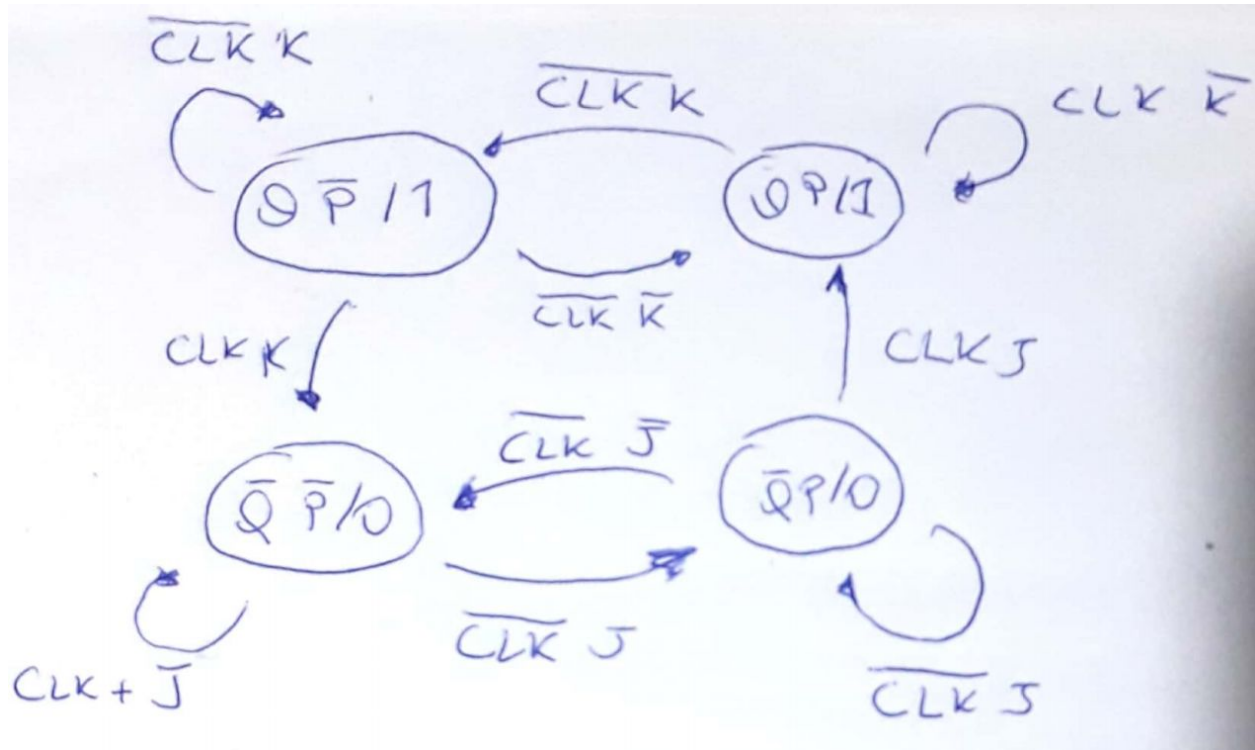
$Q_n \setminus P_n$	0	1
0	A	B
1	C	D

Chega-se a tabela de fluxo de estados minimizada com assinalamento livre de corrida crítica.

<i>Estado \ JK</i>	CLK = 0				CLK = 1				Q_n
	00	01	11	10	00	01	11	10	
00	00	00	01	01	00	00	00	00	0
01	00	00	01	01	-	-	11	11	0
11	11	10	10	11	11	11	11	11	1
10	11	10	10	11	-	00	00	-	1

c) **Mostre o diagrama de estados.**

Segue o esboço para o diagrama de estados com base no obtido anteriormente.



d) **Verifique se há Hazard essencial do tipo estado estável.**

De acordo com a regra de Unger, há hazard essencial do tipo estável. Basta observar que há várias sequências de transições que confirmam, conforme a regra, a presença do hazard essencial de estado estável.

e) **Verifique se há Hazard essencial do tipo transiente.**

Não há hazard essencial do tipo transiente. Isso porque, segundo a regra de Unger, ele só ocorre quando o circuito sofre trocas durante o caminho até seu estado de destino final correto, o que não é o caso, como pode-se observar na tabela de fluxo de estados do Flip Flop JK.

f) **Obtenha as funções minimizadas de próximo estado e de saída que estejam livres de Hazard lógico combinacional (estático e dinâmico), considerando SIC (single input change).**

Através das tabelas obtidas anteriormente, e com a utilização dos mapas de Karnaugh, obteve-se as funções mínimas. Vale ressaltar que o desenvolvimento foi muito parecido com o da questão 5.1, por isso, não achou-se necessário demonstrar o mesmo procedimento aqui no relatório.

Vale lembrar, no entanto, que devido aos hazards presentes, foi dificultoso para nós acharmos qual era a função mínima, visto que, nas primeiras tentativas, através da simulação foi verificado que elas não estavam funcionando adequadamente.

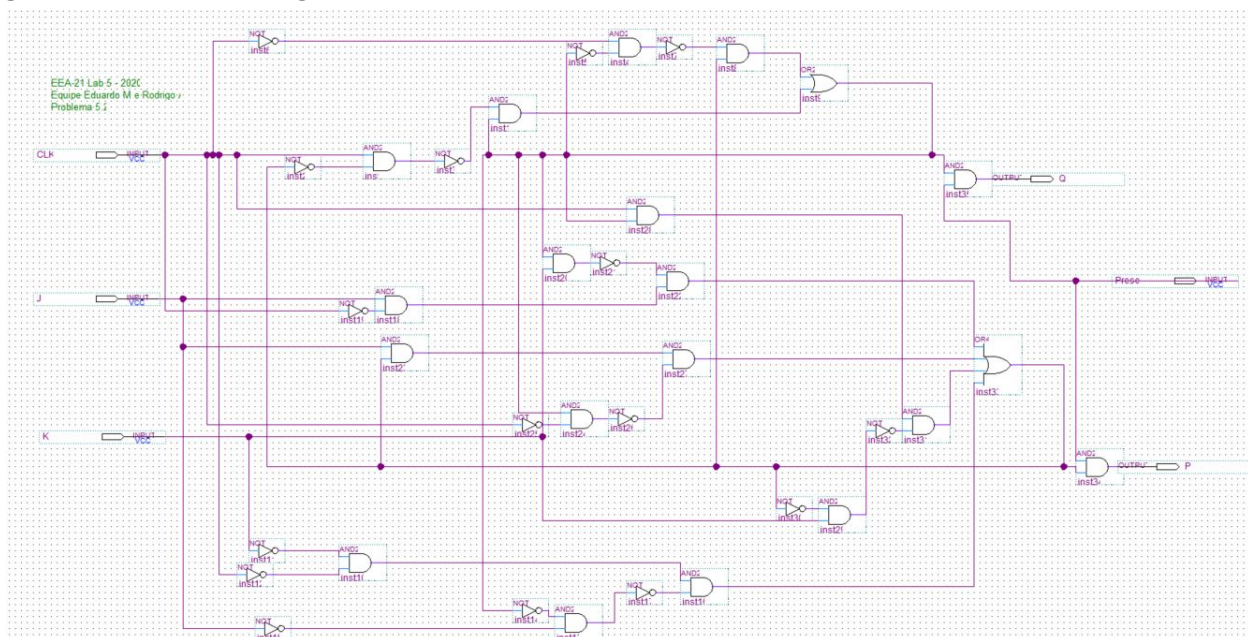
Após algumas tentativas, conseguiu-se encontrar as seguintes funções minimizadas:

$$Q_{n+1} = Q_n(P_n'CLK)' + P_n(Q_n'CLK)'$$

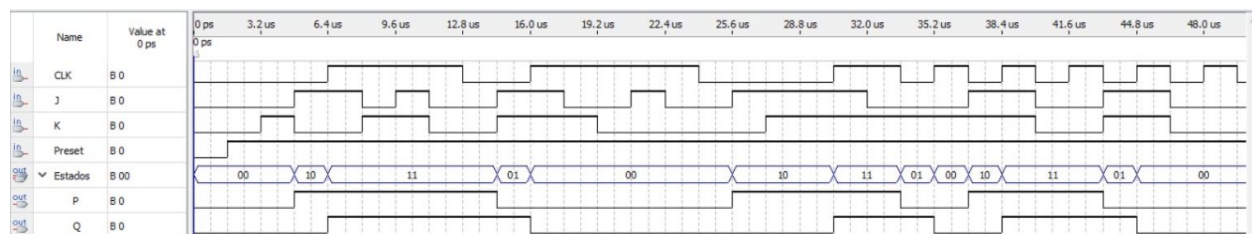
$$P_{n+1} = K'(CLK)'(Q_n'J)' + Q_nCLK(P_n'K)' + JCLK'(Q_nK)' + P_nJ(Q_nCLK)'$$

As quais foram verificadas na simulação via *Quartus*.

g) Obtenha o diagrama esquemático.



h) Obtenha o diagrama de temporização, testando todas as mudanças de estado.



5.3) Simulação de uma ULA de 8 bits

Para montar a unidade básica da ULA, considera-se a seguinte tabela de operações:

- Para F_i

$M \setminus S_1 S_0$	00	01	11	10
0	A_i	$\overline{A_i}$	$\overline{A_i \oplus B_i}$	$A_i \oplus B_i$
1	$A_i \oplus C_{in}$	$\overline{A_i \oplus C_{in}}$	$\overline{A_i \oplus B_i \oplus C_{in}}$	$A_i \oplus B_i \oplus C_{in}$

- Para C_{out}

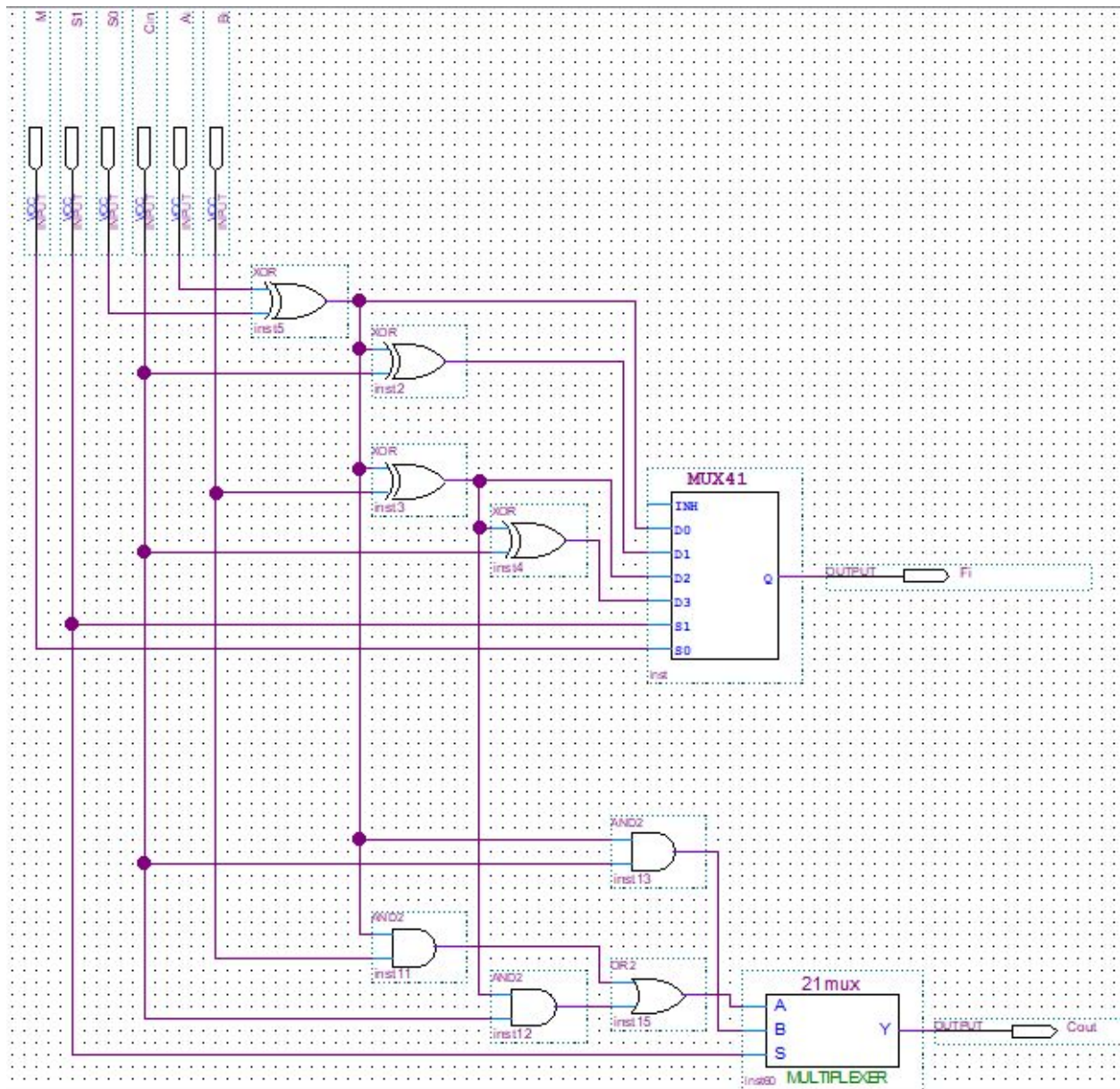
$M \setminus S_1 S_0$	00	01	11	10
0	X	X	X	X
1	$A_i C_{in}$	$\overline{A_i} C_{in}$	$\overline{A_i} B_i + C_{in} (\overline{A_i \oplus B_i})$	$A_i B_i + C_{in} (A_i \oplus B_i)$

Assim, as equações booleanas da unidade básica:

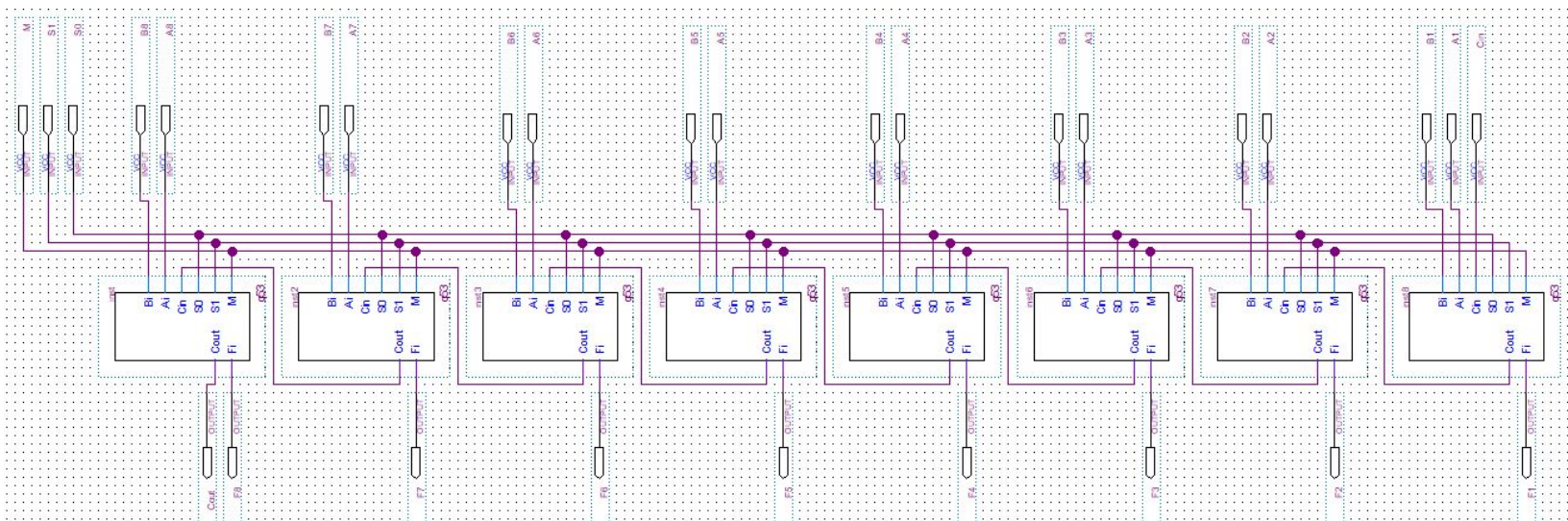
$$F_i = \overline{M} \overline{S_1} (A_i \oplus S_0) + \overline{M} S_1 (A_i \oplus B_i \oplus S_0) + M \overline{S_1} (A_i \oplus C_{in} \oplus S_0) + M S_1 (A_i \oplus B_i \oplus C_{in} \oplus S_0)$$

$$C_{out} = \overline{S_1} C_{in} (A_i \oplus S_0) + S_1 (B_i (A_i \oplus S_0) + C_{in} (A_i \oplus S_0 \oplus B_i))$$

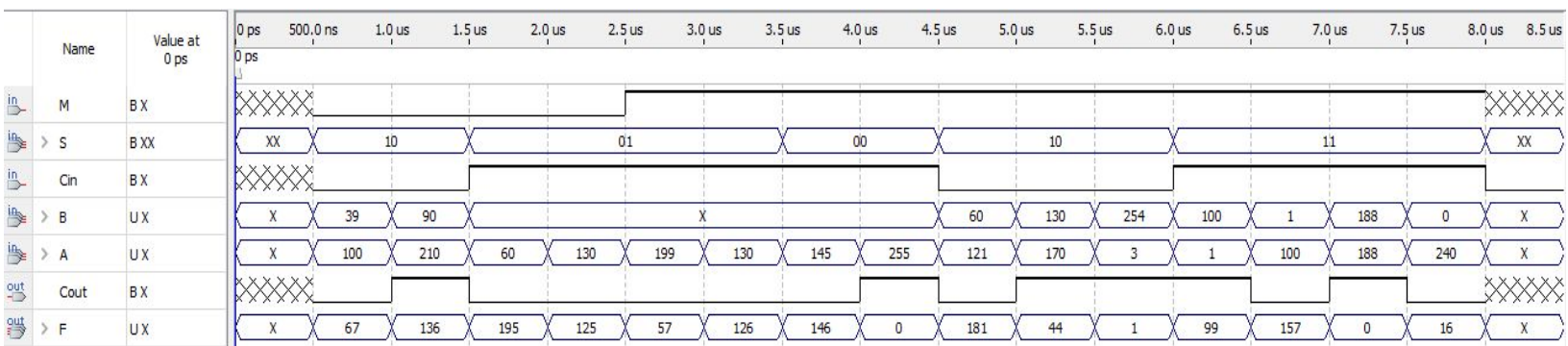
Faz-se então o diagrama esquemático utilizando MUXs:



Montando a ULA a partir da célula básica:



Assim, é possível realizar o diagrama de temporização a partir da simulação proposta no enunciado:



Pelos valores de *output* de F, verifica-se a realização das operações:

- Nas operações lógicas, é realizada a operação bit a bit
- A operação de complemento de 2 é utilizada para obter a representação do número negativo (no diagrama isso não fica evidente pois não há bit de sinal)
- A operação de incremento de A excede os 8 bits quando A = 255, fazendo com que Cout seja 1
- A operação de soma de A com B também mostra Cout = 1 quando o resultado excede o limite de 8 bits
- Por fim, o resultado de B menos A (realizado por complemento de 2) mostra Cout = 0 quando A é maior que B