University of New Brunswick

Computer Science

CS3853: Computer Architecture and Organization

Instructor: Joannah Nanjekye, jnanjeky@unb.ca

Due Date: July 26, 2024 —- 11:59 PM

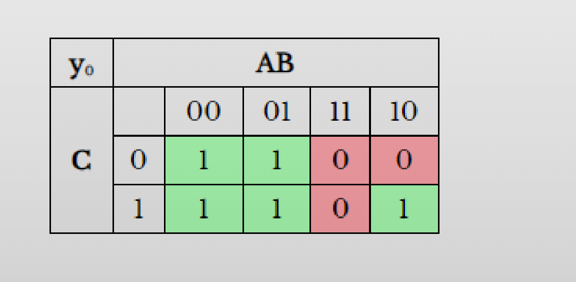
ASSIGNMENT 2

Submission instructions:

- Submit a pdf file to the Desire2Learn dropbox

Problem 1. Find the respective expressions for following Boolean functions using Karnaugh maps:

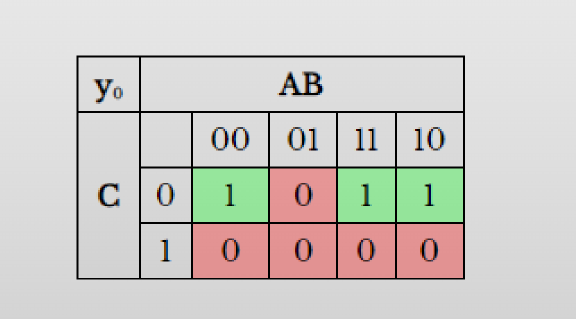
- F(A,B,C) = $\sum$m(0, 1, 2, 3, 4, 6)
  Solution: 3 points



y_0	AB				
C		00	01	11	10
	0	1	1	0	0
	1	1	1	0	1

$F = \overline{B}C + \overline{A}$

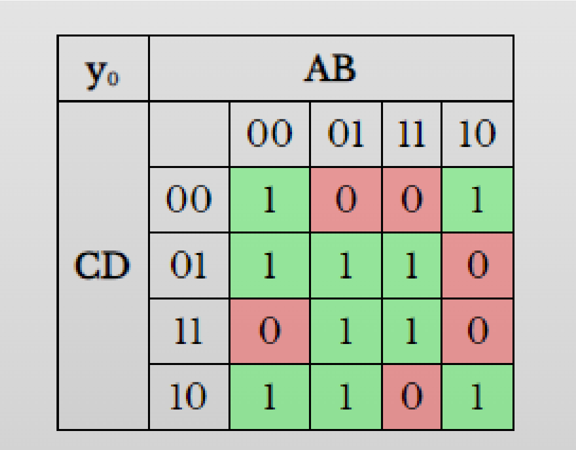
- F(A,B,C) = $\sum$m(0, 4, 6, 7) **Solution: 3 points**



y_0	AB				
C		00	01	11	10
	0	1	0	1	1
	1	0	0	0	0

$F = \overline{BC} + A\overline{C}$

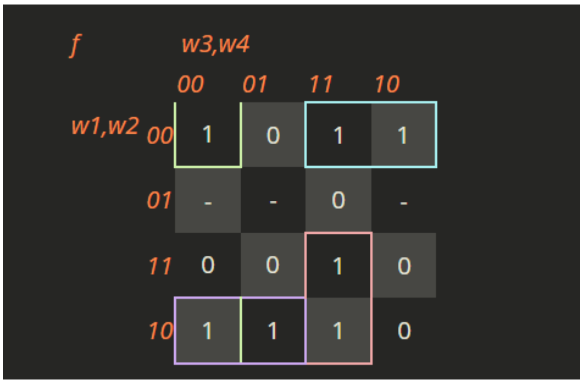
- F(A, B, C, D) = $\sum$m(0, 1, 2, 5, 6, 7, 8, 10, 13, 15)
  Solution: 3 points



y₀	AB				
		00	01	11	10
CD	00	1	0	0	1
	01	1	1	1	0
	11	0	1	1	0
	10	1	1	0	1

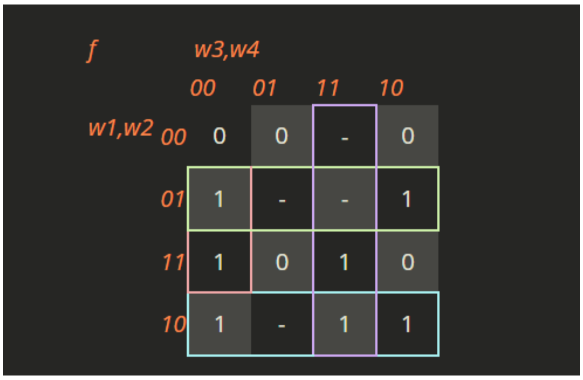
$F = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{C}D + \overline{A}C\overline{D} + \overline{A}BC + \overline{B}\,\overline{D} + BD$

- F($w_1, w_2, w3, w_4$) = $\prod$m(0, 2, 3, 8, 9, 11, 15) + $\sum$ d(4,5,6)
  Solution: 3 points



f		w3,w4			
		00	01	11	10
w1,w2	00	1	0	1	1
	01	-	-	0	-
	11	0	0	1	0
	10	1	1	1	0

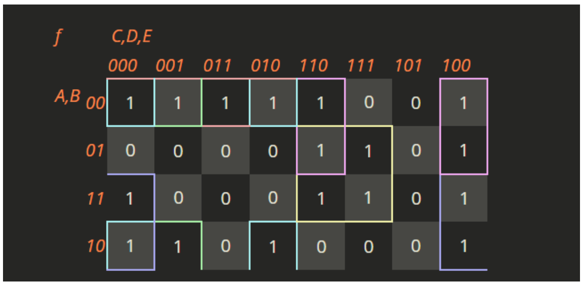
- F($w_1, w_2, w3, w_4$)= $\sum$m(4,6,8,10,11,12,15) + D(3,5,7,9)
  Solution: 3 points



f		w3,w4			
		00	01	11	10
w1,w2	00	0	0	-	0
	01	1	-	-	1
	11	1	0	1	0
	10	1	-	1	1

$F = \overline{w1}w2 + w1\overline{w2} + w3w4 + w2\overline{w3w4}$

- F(A,B,C,D,E) = $\sum$m(0,1,2, 3, 4, 6,12,14,15,16,17,18,20,24,28,30,31)
  Solution: 3 points



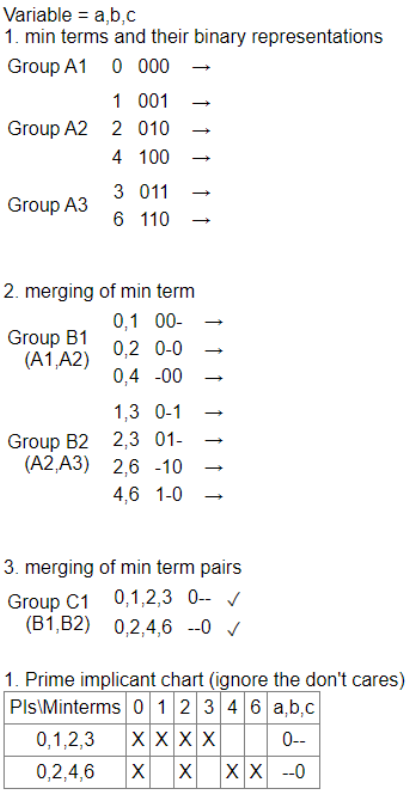
f C,D,E

A,B	000	001	011	010	110	111	101	100
00	1	1	1	1	1	0	0	1
01	0	0	0	0	1	1	0	1
11	1	0	0	0	1	1	0	1
10	1	1	0	1	0	0	0	1

$F = \overline{ABC} + BCD + \overline{BCD} + B\overline{CE} + A\overline{DE} + \overline{ACE}$

Problem 2. Find the respective expressions for following Boolean functions using the Quine-McCluskey tabular algorithm:

- F(A, B, C) = $\sum$m(0, 1, 2, 3, 4, 6)
  Solution: 8 points



Variable = a,b,c

1. min terms and their binary representations

Group A1	0	000	→
	1	001	→
Group A2	2	010	→
	4	100	→
Group A3	3	011	→
	6	110	→

2. merging of min term

Group B1 (A1,A2)	0,1	00-	→
	0,2	0-0	→
	0,4	-00	→
Group B2 (A2,A3)	1,3	0-1	→
	2,3	01-	→
	2,6	-10	→
	4,6	1-0	→

3. merging of min term pairs

Group C1 (B1,B2)	0,1,2,3	0--	✓
	0,2,4,6	--0	✓

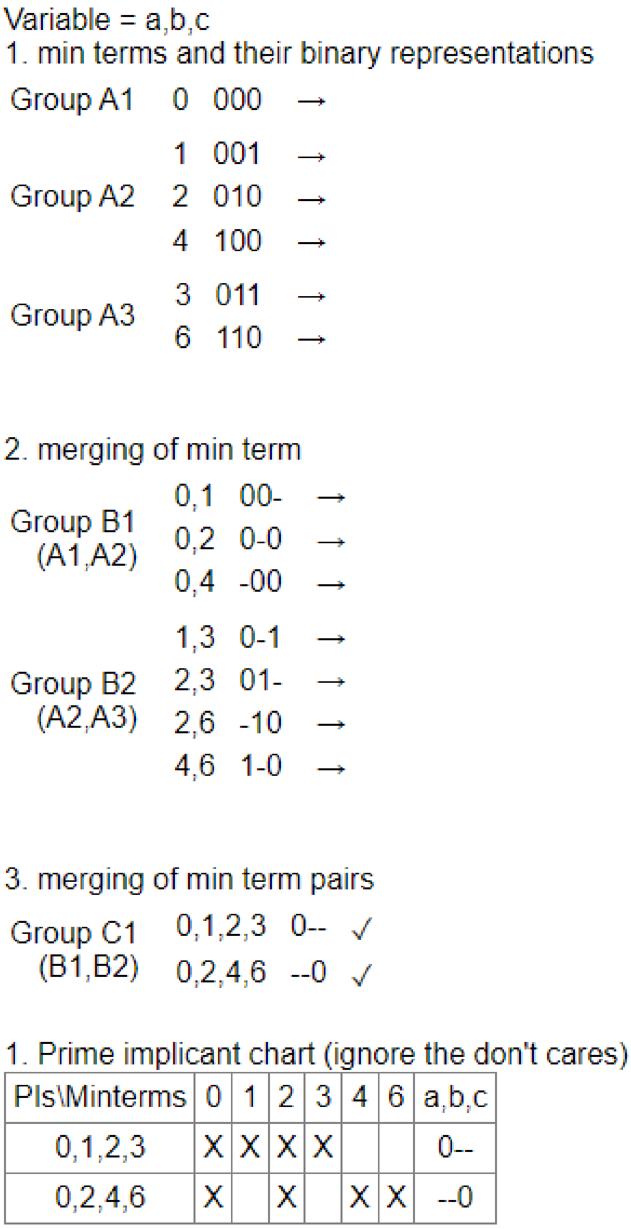
1. Prime implicant chart (ignore the don't cares)

PIs\Minterms	0	1	2	3	4	6	a,b,c
0,1,2,3	X	X	X	X			0--
0,2,4,6	X		X		X	X	--0

$F = \overline{A} + \overline{C}$

- F($w_1, w_2, w3, w_4$)= $\sum$m(4,6,8,10,11,12,15) + D(3,5,7,9)

  Solution: 8 points



Variable = a,b,c

1. min terms and their binary representations

Group A1	0	000	→
Group A2	1	001	→
	2	010	→
	4	100	→
Group A3	3	011	→
	6	110	→

2. merging of min term

Group B1 (A1,A2)	0,1	00-	→
	0,2	0-0	→
	0,4	-00	→
Group B2 (A2,A3)	1,3	0-1	→
	2,3	01-	→
	2,6	-10	→
	4,6	1-0	→

3. merging of min term pairs

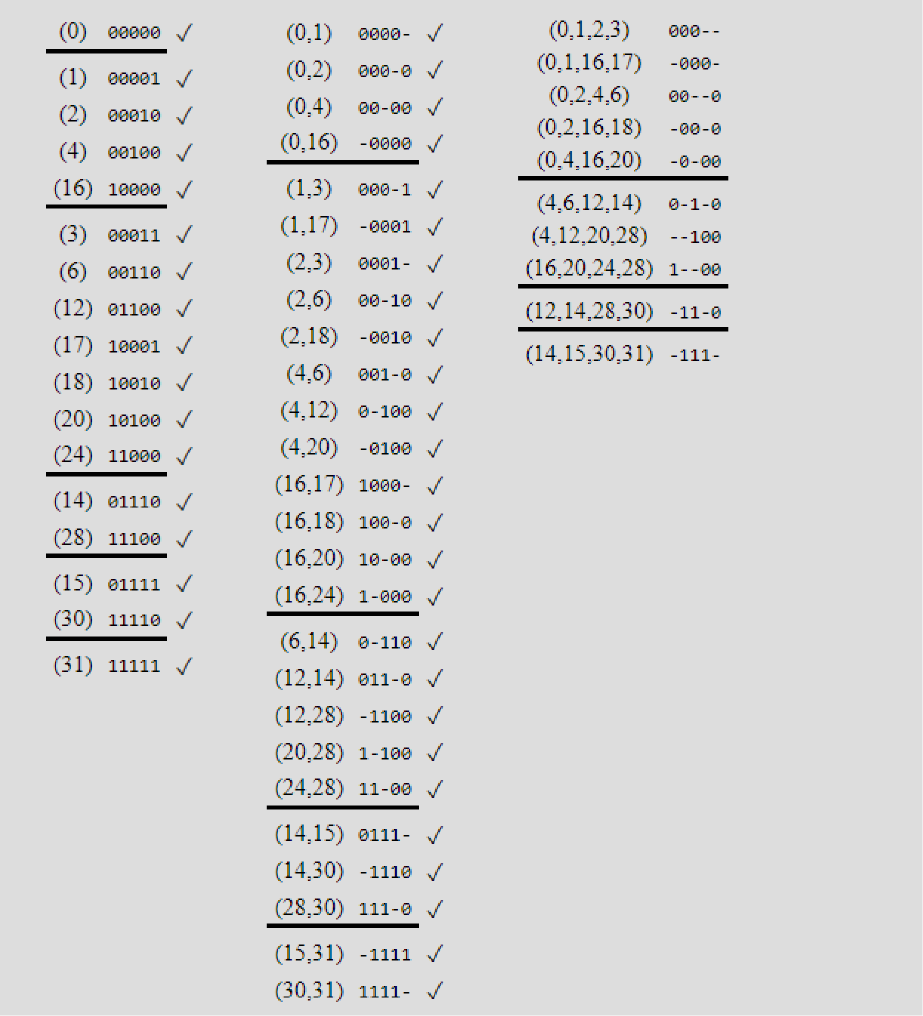
Group C1 (B1,B2)	0,1,2,3	0--	✓
	0,2,4,6	--0	✓

1. Prime implicant chart (ignore the don't cares)

PIs\Minterms	0	1	2	3	4	6	a,b,c
0,1,2,3	X	X	X	X			0--
0,2,4,6	X		X		X	X	--0

$F = \overline{w_1}w_2 + w_1\overline{w_2} + w_3w_4$

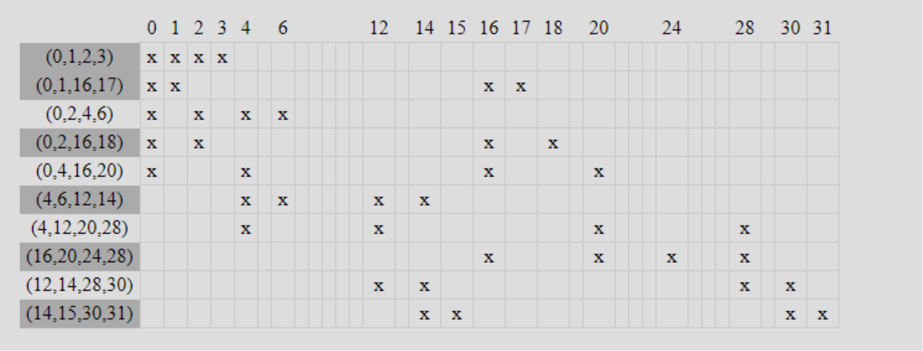
- F(A,B,C,D,E)= $\sum$m(0,1,2, 3, 4, 6,12,14,15,16,17,18,20,24,28,30,31)
  Solution: 12 points



(0)	00000	✓
(1)	00001	✓
(2)	00010	✓
(4)	00100	✓
(16)	10000	✓
(3)	00011	✓
(6)	00110	✓
(12)	01100	✓
(17)	10001	✓
(18)	10010	✓
(20)	10100	✓
(24)	11000	✓
(14)	01110	✓
(28)	11100	✓
(15)	01111	✓
(30)	11110	✓
(31)	11111	✓

(0,1)	0000-	✓
(0,2)	000-0	✓
(0,4)	00-00	✓
(0,16)	-0000	✓
(1,3)	000-1	✓
(1,17)	-0001	✓
(2,3)	0001-	✓
(2,6)	00-10	✓
(2,18)	-0010	✓
(4,6)	001-0	✓
(4,12)	0-100	✓
(4,20)	-0100	✓
(16,17)	1000-	✓
(16,18)	100-0	✓
(16,20)	10-00	✓
(16,24)	1-000	✓
(6,14)	0-110	✓
(12,14)	011-0	✓
(12,28)	-1100	✓
(20,28)	1-100	✓
(24,28)	11-00	✓
(14,15)	0111-	✓
(14,30)	-1110	✓
(28,30)	111-0	✓
(15,31)	-1111	✓
(30,31)	1111-	✓

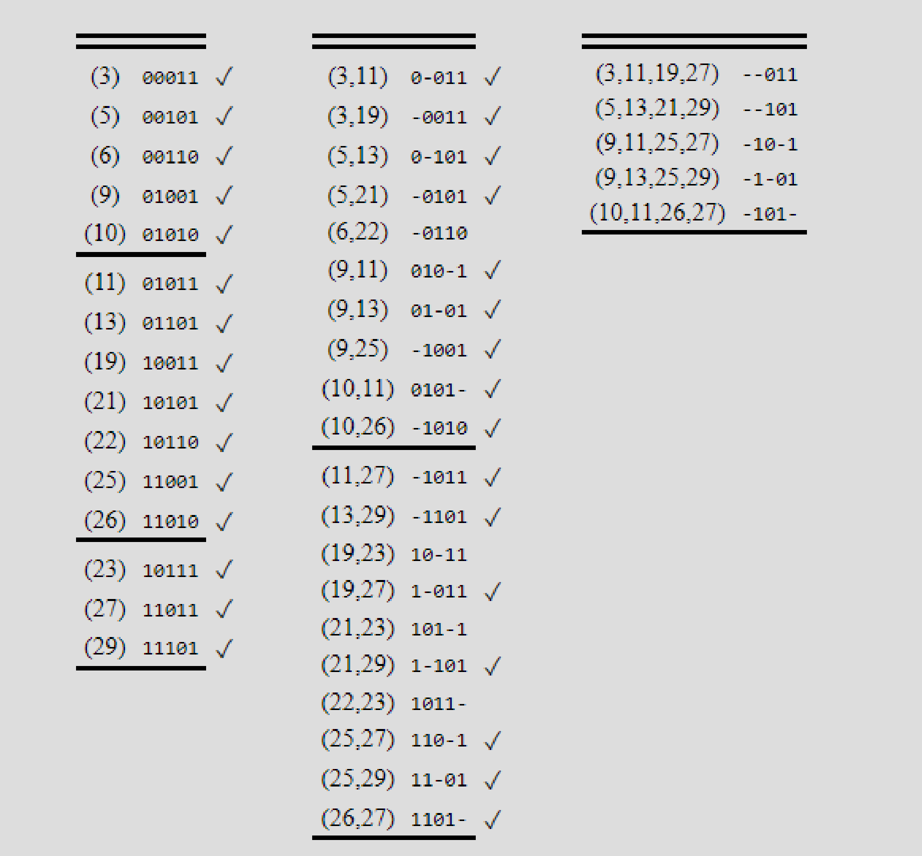
(0,1,2,3)	000--
(0,1,16,17)	-000-
(0,2,4,6)	00--0
(0,2,16,18)	-00-0
(0,4,16,20)	-0-00
(4,6,12,14)	0-1-0
(4,12,20,28)	--100
(16,20,24,28)	1--00
(12,14,28,30)	-11-0
(14,15,30,31)	-111-



	0	1	2	3	4	6	12	14	15	16	17	18	20	24	28	30	31
(0,1,2,3)	x	x	x	x													
(0,1,16,17)	x	x								x	x						
(0,2,4,6)	x		x		x	x											
(0,2,16,18)	x		x							x		x					
(0,4,16,20)	x				x					x			x				
(4,6,12,14)					x	x	x	x									
(4,12,20,28)					x		x						x		x		
(16,20,24,28)										x			x	x	x		
(12,14,28,30)							x	x							x	x	
(14,15,30,31)								x	x							x	x

$F = \overline{ABC} + \overline{BCD} + A\overline{CE} + BCD + AD\overline{E}$

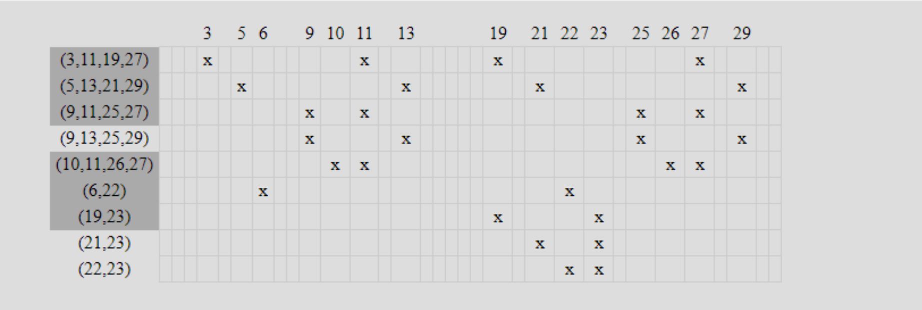
- F(A,B,C,D,E) = $\sum$m(3,5,6,9,10,11,13,19,21,22,23,25,26,27,29)
  Solution: 12 points



(3)	00011	✓
(5)	00101	✓
(6)	00110	✓
(9)	01001	✓
(10)	01010	✓
(11)	01011	✓
(13)	01101	✓
(19)	10011	✓
(21)	10101	✓
(22)	10110	✓
(25)	11001	✓
(26)	11010	✓
(23)	10111	✓
(27)	11011	✓
(29)	11101	✓

(3,11)	0-011	✓
(3,19)	-0011	✓
(5,13)	0-101	✓
(5,21)	-0101	✓
(6,22)	-0110	
(9,11)	010-1	✓
(9,13)	01-01	✓
(9,25)	-1001	✓
(10,11)	0101-	✓
(10,26)	-1010	✓
(11,27)	-1011	✓
(13,29)	-1101	✓
(19,23)	10-11	
(19,27)	1-011	✓
(21,23)	101-1	
(21,29)	1-101	✓
(22,23)	1011-	
(25,27)	110-1	✓
(25,29)	11-01	✓
(26,27)	1101-	✓

(3,11,19,27)	--011
(5,13,21,29)	--101
(9,11,25,27)	-10-1
(9,13,25,29)	-1-01
(10,11,26,27)	-101-



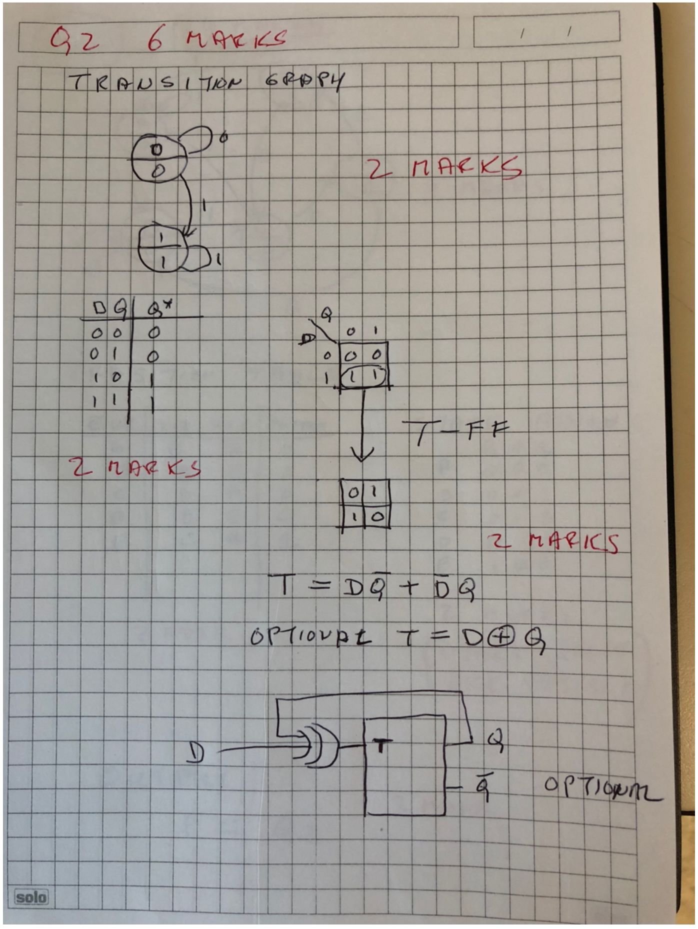
	3	5	6	9	10	11	13	19	21	22	23	25	26	27	29
(3,11,19,27)	x					x		x						x	
(5,13,21,29)		x					x		x						x
(9,11,25,27)				x		x						x		x	
(9,13,25,29)				x			x					x			x
(10,11,26,27)					x	x							x	x	
(6,22)			x							x					
(19,23)								x			x				
(21,23)									x		x				
(22,23)										x	x				

$F = \overline{C}DE + C\overline{D}E + B\overline{C}E + B\overline{C}D + \overline{B}CD\overline{E} + A\overline{B}DE$

Problem 3. Consult the class notes (slides) and discussion on flip-flops to implement the following. In each case use any additional logic gates that are required.

- A T-FF using a D-FF

  Solution: 6 points — 2 points for transition diagram, 2 points for state table, 2 points for the kmap and final expressions



Q2 6 MARKS

TRANSITION GRAPH

2 MARKS

D	Q	Q*
0	0	0
0	1	0
1	0	1
1	1	1

2 MARKS

$$\begin{array}{c|cc} D\backslash Q & 0 & 1 \\ \hline 0 & 0 & 0 \\ 1 & 1 & 1 \end{array}$$

T-FF

$$\begin{array}{cc} 0 & 1 \\ 1 & 0 \end{array}$$

2 MARKS

$T = D\bar{Q} + \bar{D}Q$

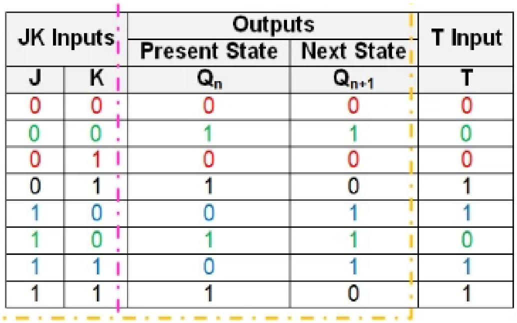
OPTIONAL $T = D \oplus Q$

0
1
0
1
1
D
T
Q
$\bar{Q}$

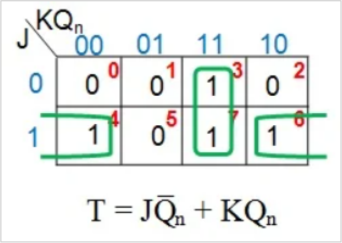
OPTIONAL

- A JK-FF using a T-FF[1]

  Solution: 6 points — 2 points for transition diagram, 2 points for state table, 2 points for the kmap and final expressions



JK Inputs		Outputs		T Input
		Present State	Next State	
J	K	Q_n	Q_{n+1}	T
0	0	0	0	0
0	0	1	1	0
0	1	0	0	0
0	1	1	0	1
1	0	0	1	1
1	0	1	1	0
1	1	0	1	1
1	1	1	0	1



J \ KQ_n	00	01	11	10
0	0 [0]	0 [1]	1 [3]	0 [2]
1	1 [4]	0 [5]	1 [7]	1 [6]

$$T = J\bar{Q}_n + KQ_n$$

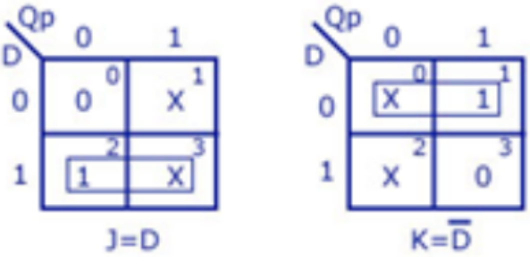
- A D-FF from a JK-FF[2]

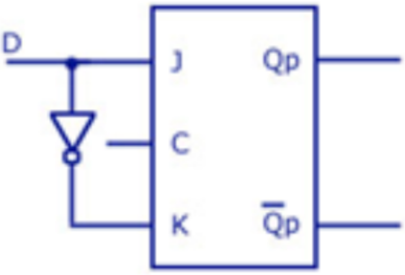
  Solution: 6 points — 2 points for transition diagram, 2 points for state table, 2 points for the kmap and final expressions

[1] https://www.allaboutcircuits.com/technical-articles/conversion-of-t-flip-flops-part-v/

[2] https://www.electronics-tutorial.net/sequential-logic-circuits/toggle-flip-flop/

D	q	q^*	J	K
0	0	0	0	x
0	1	0	x	1
1	0	1	1	x
1	1	0	x	0





- A JK-FF using a D-FF

Solution: 6 points — 2 points for transition diagram, 2 points for state table, 2 points for the kmap and final expressions

J	K	q	q^*
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

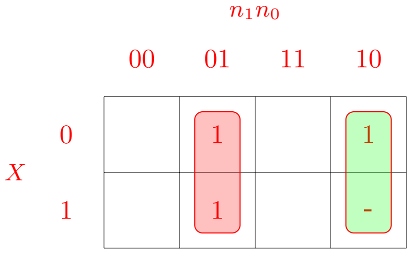
		Kq			
		00	01	11	10
J	0		1		
	1	1	1		1

$F(J, K, q) = \overline{K}q + J\overline{q}$

Problem 4. Design a synchronous counter using D-FFs and one input x. If x = 0 it counts 1,2,3, 0, 1,2 . . .; if x = 1 it counts 1, 3, 0, 1, 3, ... Assume that x only changes in 1 or 3 (in which case there is one combination that will never occur – state 2 and x = 1).

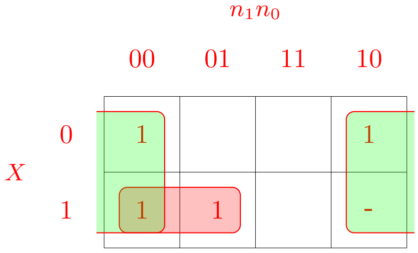
Solution: 12 points — 3 points for transition diagram, 3 points for state table, 3 points for the kmap and final expressions for D_1, and 3 points for the kmap and final expressions for D_2

Current		**x = 0**				**x = 1**			
n_1	n_0	n_1	n_0	D_1	D_0	n_1	n_0	D_1	D_0
0	0	0	1	0	1	0	1	0	1
0	1	1	0	1	0	1	1	1	1
1	0	1	1	1	1	x	x	x	x
1	1	0	0	0	0	0	0	0	0



	n_1n_0	00	01	11	10
X	0		1		1
	1		1		-

$D_1 = \overline{n_1}n_0 + n_1\overline{n_0}$

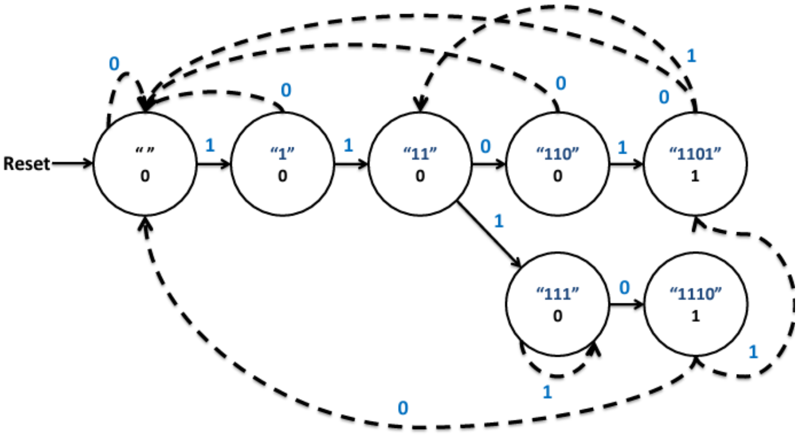


	n_1n_0	00	01	11	10
X	0	1			1
	1	1	1		-

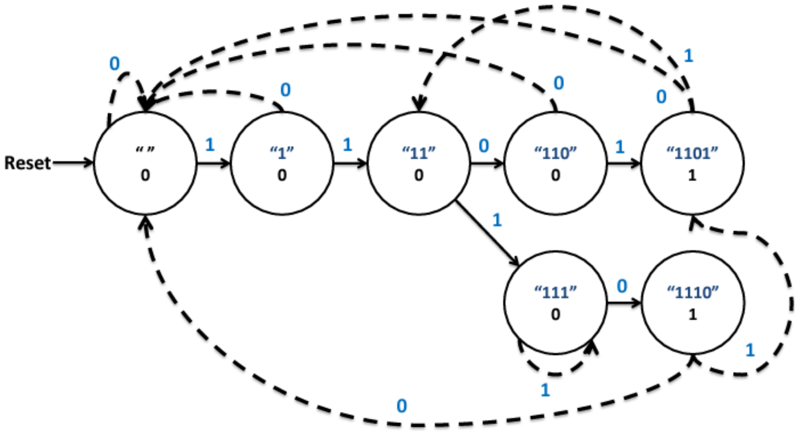
$D_0 = X\overline{n_1} + \overline{n_0}$

Problem 5. Design an FSM that recognizes 10111 or 10101.

- Draw state transition diagram (use as few states as possible).
- Choose state encodings.
- Write state transition and output table using the encodings.
- Write next state equations and output equations.



State encoding:

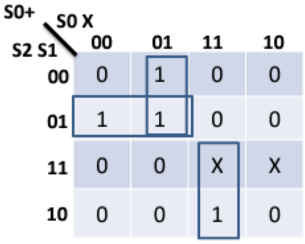


A = 000 ; B = 001 ; C = 010 ; D = 011; E = 100 ; F = 101 ; G = 110

State transition and output table:

Present State S2 S1 S0	X = 0	X = 1	F
	NS S2+ S1+ S0+	NS S2+ S1+ S0+	
0 0 0	0 0 0	0 0 1	0
0 0 1	0 0 0	0 1 0	0
0 1 0	0 1 1	1 0 1	0
0 1 1	0 0 0	1 0 0	0
1 0 0	0 0 0	0 1 0	1
1 0 1	1 1 0	1 0 1	0
1 1 0	0 0 0	1 0 0	1

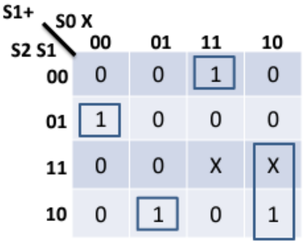
K-Maps:



S0+

S2 S1 \ S0 X	00	01	11	10
00	0	1	0	0
01	1	1	0	0
11	0	0	X	X
10	0	0	1	0

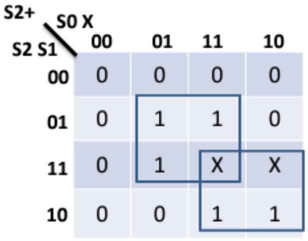
S0+ = S0'.S1.S2' + X.S0'.S2' + X.S0.S2



S1+

S2 S1 \ S0 X	00	01	11	10
00	0	0	1	0
01	1	0	0	0
11	0	0	X	X
10	0	1	0	1

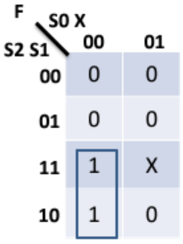
S1+ = X.S0.S1'.S2' + X'.S0'.S1.S2' + X.S0'.S1'.S2 + X'.S0.S2



S2+

S2 S1 \ S0 X	00	01	11	10
00	0	0	0	0
01	0	1	1	0
11	0	1	X	X
10	0	0	1	1

S2+ = S0.S2 + X.S1



F

S2 S1 \ S0 X	00	01
00	0	0
01	0	0
11	1	X
10	1	0

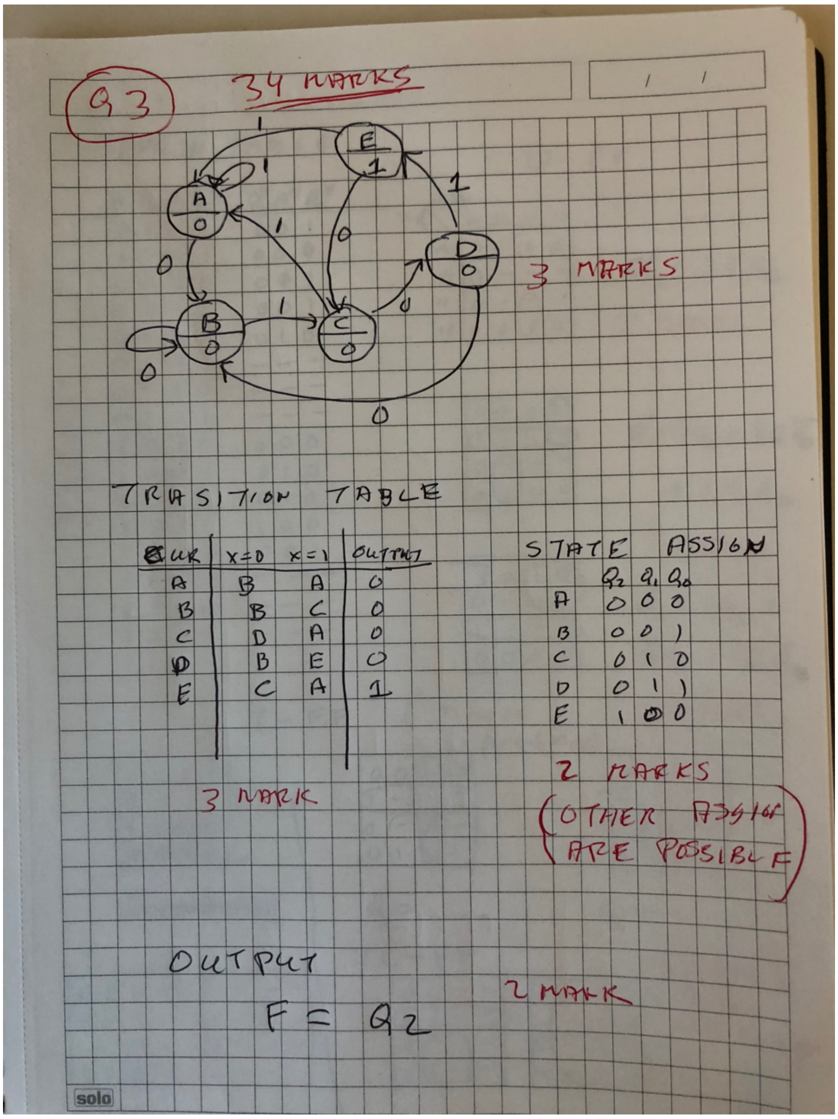
F = S0'.S2

Problem 6. Design a circuit that recognizes an input sequence that has at least two consecutive 1's or two consecutive 0's. The recognizer has a single output Y. There also should be asynchronous reset. For example:

```
input:   0011111010100110000011101
Y:       0010111100000101011110110
```

- Devise the state diagram
- Encode the states
- Obtain the equations for D flip-flops
- Simulate the circuit using LogicWorks (this is done in Lab 2—nothing needs to be handed in here.)

NOTE: just show the equations, no circuit needs to be drawn.


Q3
34 MARKS
A
0
B
0
C
0
D
0
E
1
0
1
1
0
1
0
1
1
0
0
0
3 MARKS
solo

TRASITION TABLE

CUR	X=0	X=1	OUTPUT
A	B	A	0
B	B	C	0
C	D	A	0
D	B	E	0
E	C	A	1

3 MARK

STATE ASSIGN

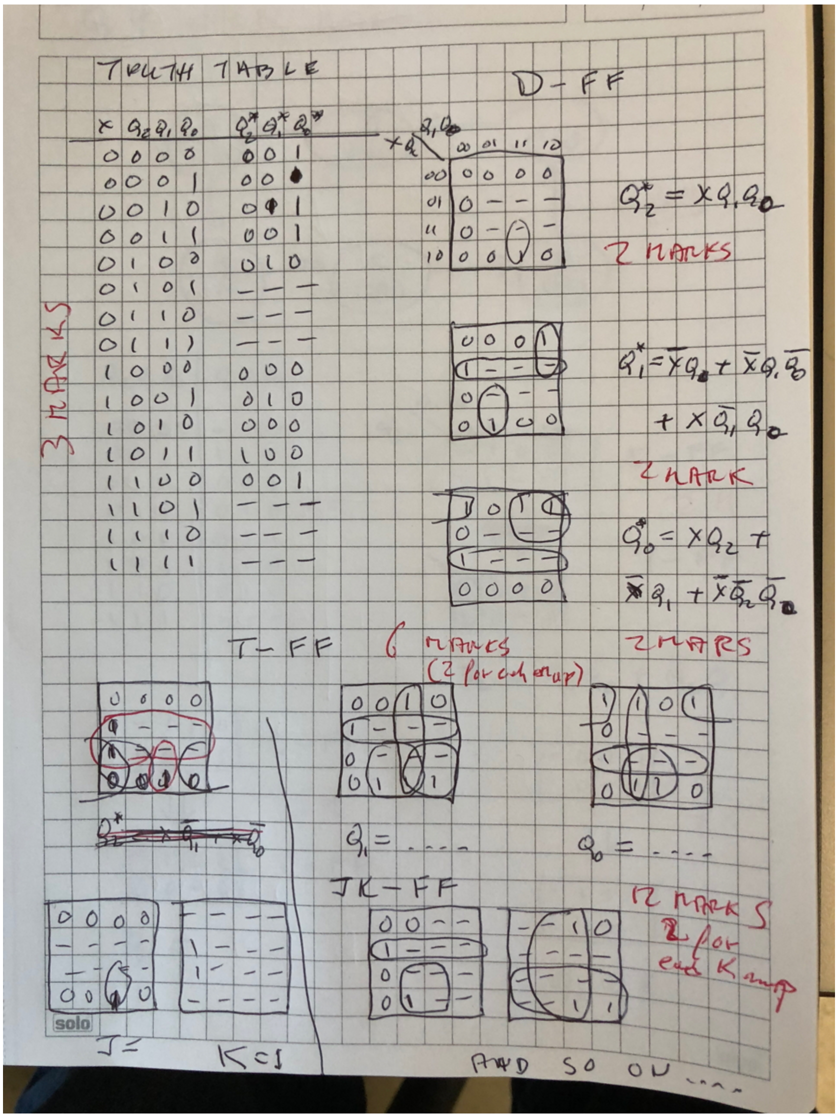
	Q_2	Q_1	Q_0
A	0	0	0
B	0	0	1
C	0	1	0
D	0	1	1
E	1	0	0

2 MARKS
(OTHER ASSIGN ARE POSSIBLE)

OUTPUT

$F = Q_2$

2 MARK



TRUTH TABLE

x	Q_2	Q_1	Q_0	Q_2^*	Q_1^*	Q_0^*
0	0	0	0	0	0	1
0	0	0	1	0	0	●
0	0	1	0	0	●	1
0	0	1	1	0	0	1
0	1	0	0	0	1	0
0	1	0	1	–	–	–
0	1	1	0	–	–	–
0	1	1	1	–	–	–
1	0	0	0	0	0	0
1	0	0	1	0	1	0
1	0	1	0	0	0	0
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	–	–	–
1	1	1	0	–	–	–
1	1	1	1	–	–	–

3 MARKS

D – FF

$xQ_2 \backslash Q_1Q_0$	00	01	11	10
00	0	0	0	0
01	0	–	–	–
11	0	–	–	–
10	0	0	1	0

$Q_2^* = xQ_1Q_0$

2 MARKS

0	0	0	1
1	–	–	–
0	–	–	–
0	1	0	0

$Q_1^* = \bar{x}Q_0 + \bar{x}Q_1\bar{Q}_0$

$+ x\bar{Q}_1Q_0$

2 MARK

1	0	1	1
0	–	–	–
1	–	–	–
0	0	0	0

$Q_0^* = xQ_2 +$

$\bar{x}Q_1 + \bar{x}\bar{Q}_2\bar{Q}_0$

2 MARS

T – FF

6 MARKS (2 for each map)

0	0	0	0
0	–	–	–
1	–	–	–
0	0	1	0

~~$Q_2^* = x\bar{Q}_1 + x\bar{Q}_0$~~

0	0	1	0
1	–	–	–
0	–	–	–
0	1	1	1

$Q_1 = \dots$

1	1	0	1
0	–	–	–
1	–	–	–
0	1	1	0

$Q_0 = \dots$

0	0	0	0
–	–	–	–
–	–	–	–
0	0	1	0

–	–	–	–
1	–	–	–
1	–	–	–
–	–	–	–

J =

K = 1

JK – FF

12 MARKS 2 for each K map

0	0	–	–
1	–	–	–
0	–	–	–
0	1	–	–

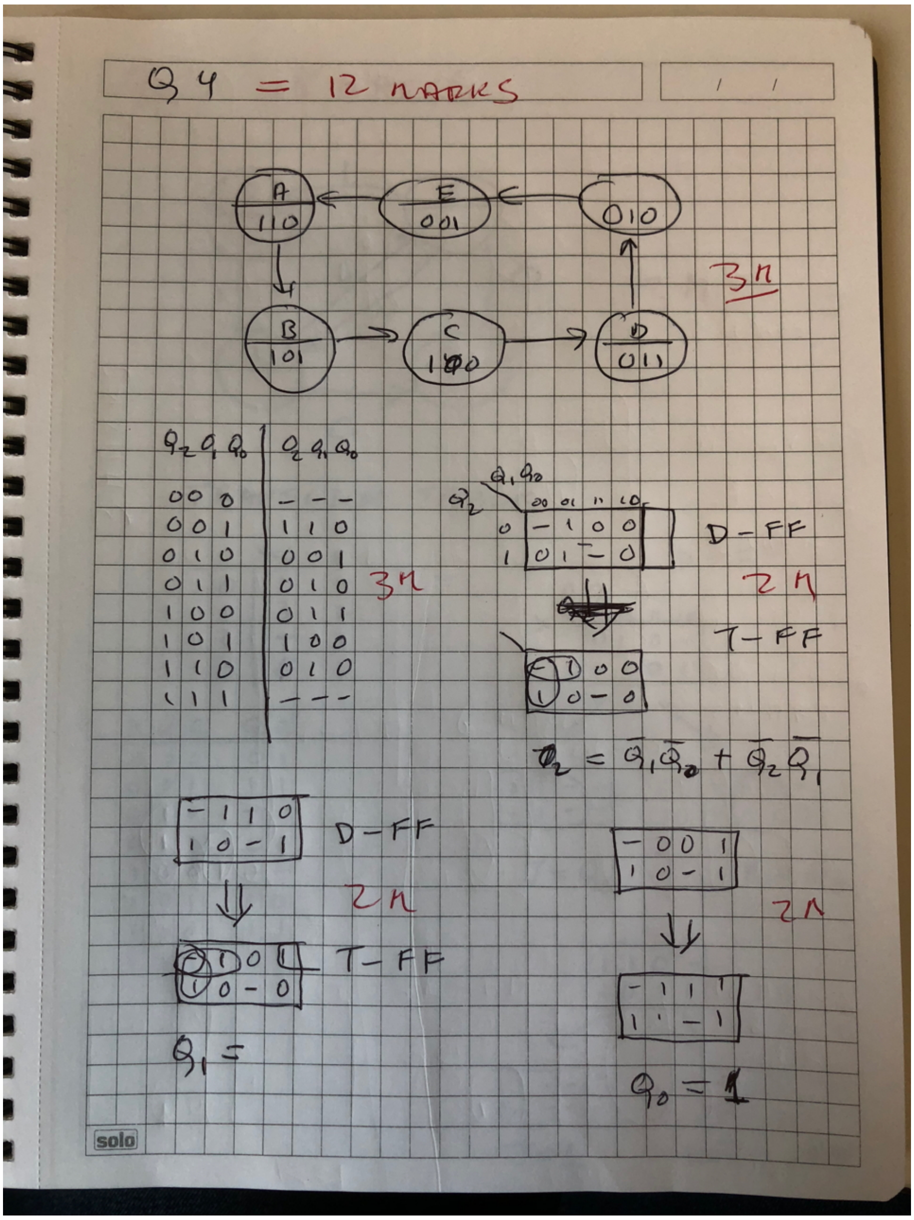
–	–	1	0
–	–	–	–
–	–	–	–
–	–	1	1

AND SO ON ....

Problem 7. Design a 3-bit up/down counter. If the input *up* = 1 the counter will count up, otherwise it will count down. Use T-FF. Show how this can be expanded to a 4-bit counter. No formal methods are needed for this problem.

Any counter solution using TFFs is fine

Make sure it makes sense



Q 4 = 12 MARKS

A
110
E
001
010
B
101
C
1~~0~~0
D
011
3M

$Q_2 Q_1 Q_0$	$Q_2 Q_1 Q_0$
0 0 0	– – –
0 0 1	1 1 0
0 1 0	0 0 1
0 1 1	0 1 0
1 0 0	0 1 1
1 0 1	1 0 0
1 1 0	0 1 0
1 1 1	– – –

3M

Q_2 \ Q_1Q_0	00	01	11	10
0	–	1	0	0
1	0	1	–	0

D – FF

2M

T – FF

–	1	0	0
1	0	–	0

$\overline{Q}_2 = \overline{Q}_1\overline{Q}_0 + \overline{Q}_2\overline{Q}_1$

–	1	1	0
1	0	–	1

D – FF

2M

–	1	0	1
1	0	–	0

T – FF

$Q_1 =$

–	0	0	1
1	0	–	1

2M

–	1	1	1
1	1	–	1

$Q_0 = 1$