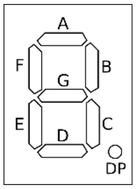
Lawrence Naidu
Collaborated with Sergio Zavala
HW2

1. This circuit takes 4-bit binary input and has a 7-bit binary output. From the basic analysis, it seems like it has a 7-segment display. This circuit is known as **BCD *(Binary Coded Decimals)* to 7-segment display.** It lights up each LED segment whenever the output is '1' and it gives out the decimal value. BCD: 4-bits to 16-bits ($2^4 = 16$), used for the 10 digits *0 to 9* to be displayed.

Display	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	0	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	1
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	0	0	1	1



Example: Decimal value 0 has output abcdef 'HIGH'.
HIGH = LED ONE
LOW = LED OFF
Although the output is recorded in binary, the 7-segment display shows decimal value.

2. Circuit A is a JK Flip Flop. It functions like a SR Flip Flop except when all the inputs are HIGH.
SR Flip Flop is best expressed using **OR gates** whereas JK Flip Flop is best described using **AND gate**

- DFF has one input (data) and 2 possible combo
  When a positive rising edge, the value remembered by the FF becomes the value of the D input
- JKFF has 2 inputs and 4 possible combo
  When positive rising edge:
  - the value remembered by the FF toggles when J = 1 & K = 1.
  - the value remains the same when J & K = 0.
  - the value equals to K or J when J =1 & K = 0 or K = 1 & J = 0
- TFF functions with JKFF with same inputs
  When the positive rising edge, the value remembered by FF either toggles or remains the same depending on whether the T inputs is 1 or 0

Truth Table for SR Flip Flop

clk	S	R	Output	
0	0	0	Memory	
1	0	0	Memory	
1	1	0	$Q = 1$	$\bar{Q} = 0$
1	0	1	$Q = 0$	$\bar{Q} = 1$
1	1	1	Not in use	

Truth table for JK Flip Flop

clk	J	K	Output	
0	0	0	Memory	
1	0	0	Memory	
1	1	0	$Q = 1$	$\bar{Q} = 0$
1	0	1	$Q = 0$	$\bar{Q} = 1$
1	1	1	Toggles	

Notice the first 4 inputs to be similar! The difference occurs when all inputs are HIGH. JK Flip Flop, toggles its output

Truth Table for T Flip Flop

clk	T	Output

Lawrence Naidu
Collaborated with Sergio Zavala
HW2

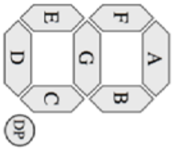
Display	X_1	X_2	X_3	X_4	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	1	0	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	1
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	0	0	1	1
A	1	0	1	0	1	1	1	0	1	1	1
b	1	0	1	1	0	0	1	1	1	1	1
C	1	1	0	0	1	0	0	1	1	1	0
d	1	1	0	1	0	1	1	1	1	0	1
E	1	1	1	0	1	0	0	1	1	1	1
F	1	1	1	1	1	0	0	1	1	1	1

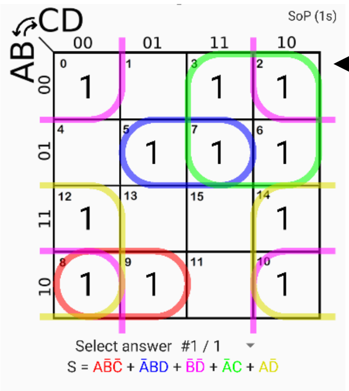
0	X	Memory
1	0	Memory
1	1	Toggles

TFF uses a **XOR gate**

It is implemented using a JKFF – or at least it functions similarly as it. The output will depend on previous states.

3. Truth table for the 7 segment display



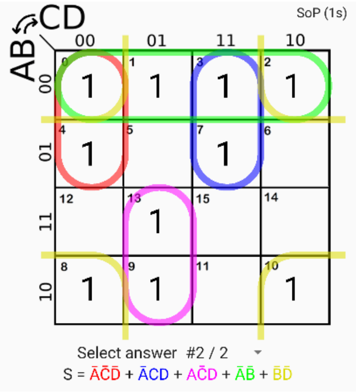


AB \ CD	00	01	11	10
00	1		1	1
01		1	1	1
11	1			1
10	1	1		1

Select answer #1 / 1

S = $A\bar{B}\bar{C} + \bar{A}BD + \bar{B}\bar{D} + \bar{A}C + A\bar{D}$

$$A = X_1\overline{X_2X_3} + \overline{X_1}X_2X_4 + \overline{X_2X_4} + \overline{X_1}X_3 + X_1\overline{X_4}$$



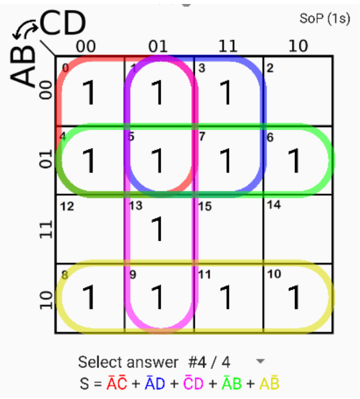
AB \ CD	00	01	11	10
00	1	1	1	1
01	1		1	
11		1		
10	1	1		1

Select answer #2 / 2

S = $\bar{A}\bar{C}\bar{D} + \bar{A}CD + A\bar{C}D + \bar{A}\bar{B} + \bar{B}\bar{D}$

$$B = \overline{X_1X_3X_4} + \overline{X_1}X_3X_4 + X_1\overline{X_3}X_4 + \overline{X_1X_2} + \overline{X_2X_4}$$

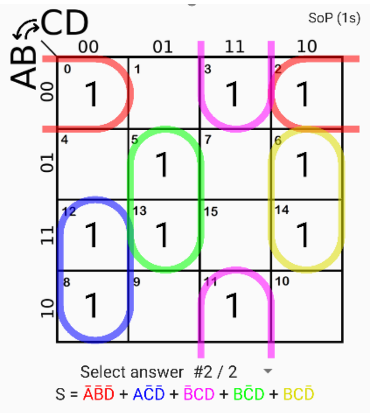
$$C = \overline{X_1X_3} + \overline{X_1}X_4 + \overline{X_3}X_4 + \overline{X_1}X_2 + X_1\overline{X_2}$$



AB \ CD	00	01	11	10
00	1	1	1	
01	1	1	1	1
11		1		
10	1	1	1	1

Select answer #4 / 4

S = $\bar{A}\bar{C} + \bar{A}D + \bar{C}D + \bar{A}B + A\bar{B}$



AB \ CD	00	01	11	10
00	1		1	1
01		1		1
11	1	1		1
10	1		1	

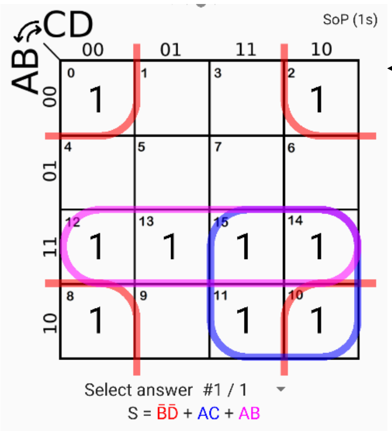
Select answer #2 / 2

S = $\bar{A}\bar{B}\bar{D} + A\bar{C}\bar{D} + \bar{B}CD + B\bar{C}D + BC\bar{D}$

$$D = \overline{X_1X_2X_4} + X_1\overline{X_3X_4} + \overline{X_2}X_3X_4 + X_2\overline{X_3}X_4 + X_2X_3\overline{X_4}$$

Lawrence Naidu
Collaborated with Sergio Zavala
HW2

$$E = \overline{X_2 X_4} + X_1 X_3 + X_1 X_2$$



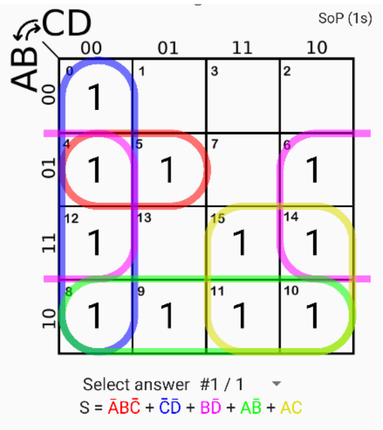
AB \ CD — SoP (1s)

AB \ CD	00	01	11	10
00	0 1	1	3	2 1
01	4	5	7	6
11	12 1	13 1	15 1	14 1
10	8 1	9	11 1	10 1

Select answer #1 / 1

S = $\overline{B}\overline{D}$ + AC + AB

$$F = \overline{X_1} X_2 \overline{X_3} + \overline{X_3 X_4} + X_1 \overline{X_2} + X_1 X_3$$



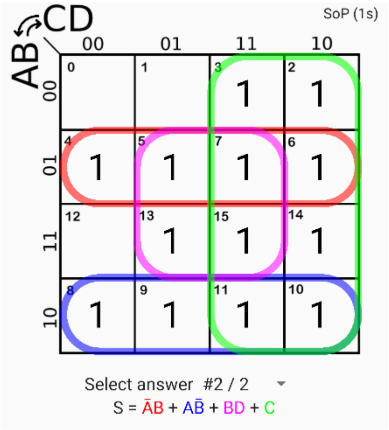
AB \ CD — SoP (1s)

AB \ CD	00	01	11	10
00	0 1	1	3	2
01	4 1	5 1	7	6 1
11	12 1	13	15 1	14 1
10	8 1	9 1	11 1	10 1

Select answer #1 / 1

S = $\overline{A}B\overline{C}$ + $\overline{C}\overline{D}$ + $B\overline{D}$ + $A\overline{B}$ + AC

$$G = \overline{X_1} X_2 + X_1 \overline{X_2} + X_2 X_4 + X_3$$

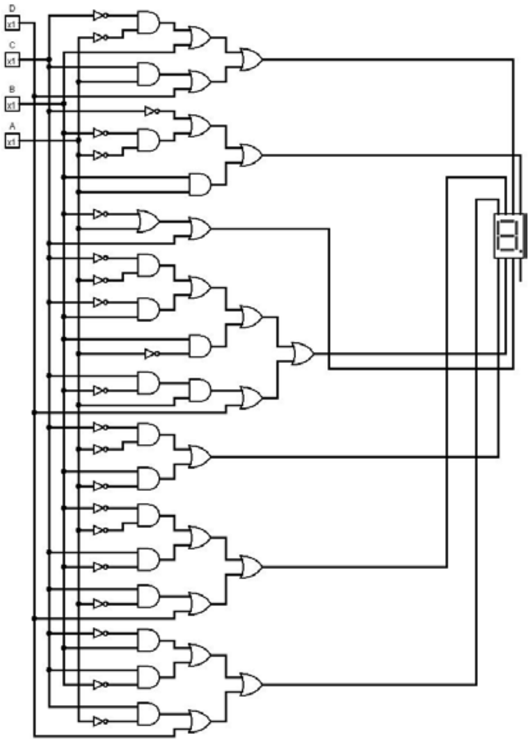


AB \ CD — SoP (1s)

AB \ CD	00	01	11	10
00	0	1	3 1	2 1
01	4 1	5 1	7 1	6 1
11	12	13 1	15 1	14 1
10	8 1	9 1	11 1	10 1

Select answer #2 / 2

S = $\overline{A}B$ + $A\overline{B}$ + BD + C



This was the closest PLA Design I could find that is an replica of each Boolean expression.