Assignment#4 CS207 Fall 2023

Ben Chen(12212231)

December 18, 2023

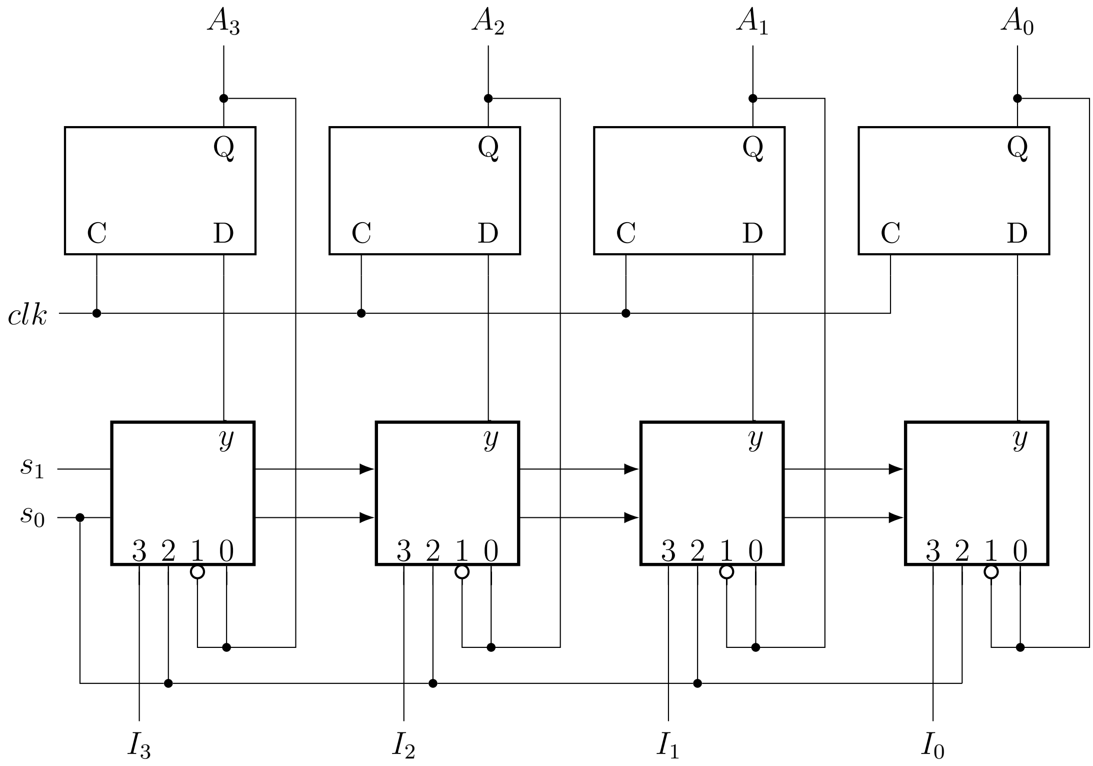
Problem 1. Given the function table, design a 4-bit register with four DFF and 4X1 MUX whose mode selection inputs are S_1 and S_0.

Solution. According to the description of operations, we can derive the function table as follows

Table 1: Function Table of 4-bit Register

Mode		**Output**				Register
S_1	S_0	A_3	A_2	A_1	A_0	Operation
0	0	A_3	A_2	A_1	A_0	No change
0	1	A_3'	A_2'	A_1'	A_0'	Complement
1	0	0	0	0	0	Reset
1	1	I_3	I_2	I_1	I_0	Load Parallel

And to design the circuit, we first connect the input wires to the port 3 of each multiplexer to load the data in parallel, and then connect the s_0 to the port 2 of each multiplexer to have the register reset, and put the output wire back to the port 1 with inverter to get the complement and port 0 to have the register unchanged of each multiplexer.



PROBLEM 2. Design a sequence generator to generate the sequence 1011110.

SOLUTION. The bit length is 7, so we need 4 *DFF*s since 3 *DFF*s will lead to duplicate state. And we can derive the state table as follows

CLK	Z	Q_3	Q_2	Q_1	Q_0
↑	0	1	0	1	1
↑	1	0	1	0	1
↑	1	1	0	1	0
↑	1	1	1	0	1
↑	1	1	1	1	0
↑	0	1	1	1	1
↑	1	0	1	1	1

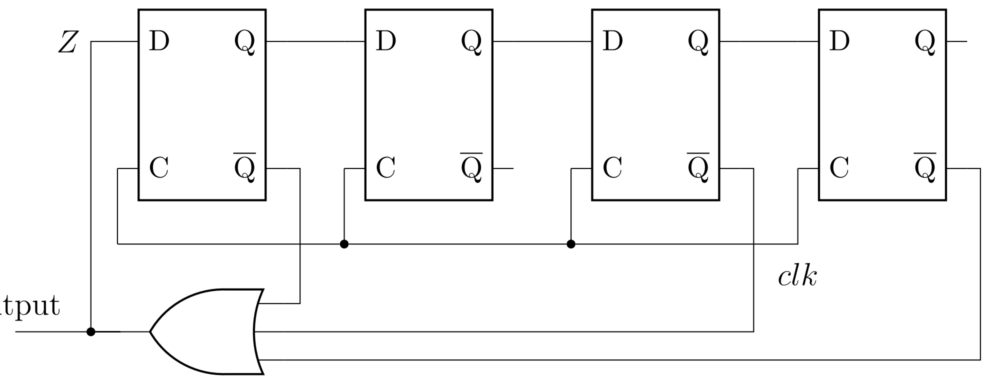
And we can derive the Karnaugh map as follows

$Q_3Q_2 \backslash Q_1Q_0$	00	01	11	10
00	X	X	X	X
01	X	1	1	X
11	X	1	0	1
10	X	X	0	1

so we can derive the simplified equations as follows

$$Z = Q_3' + Q_1' + Q_0'$$

And the logic diagram will be

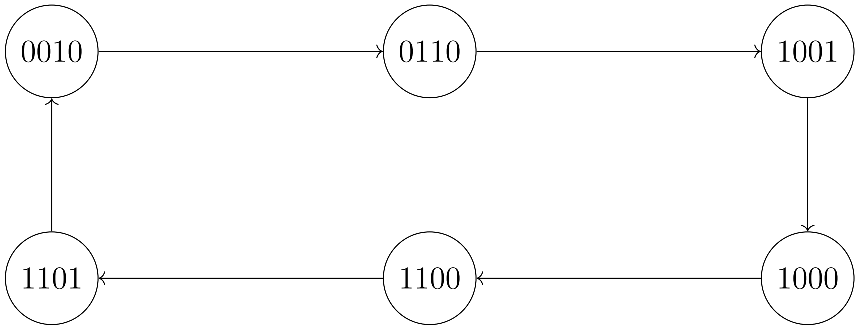


Problem 3. Design a synchronous counter using DFFs that has the following sequence:

$$0010, 0110, 1001, 1000, 1100, 1101$$

From the undesired states the counter must go back on the next clock pulse.

SOLUTION. From the sequence, we can derive the state diagram as follows



and then we can derive the state table as follows

Present State				Next State			
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0	0	1	0
0	0	0	1	0	0	1	0
0	0	1	0	0	1	1	0
0	0	1	1	0	0	1	0
0	1	0	0	0	0	1	0
0	1	0	1	0	0	1	0
0	1	1	0	1	0	0	1
0	1	1	1	0	0	1	0
1	0	0	0	1	1	0	0
1	0	0	1	1	0	0	0
1	0	1	0	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0	0	1	1	0	1
1	1	0	1	0	0	1	0
1	1	1	0	0	0	1	0
1	1	1	1	0	0	1	0

And we can derive the Karnaugh maps for each DFF as follows

$Q_3Q_2 \backslash Q_1Q_0$	00	01	11	10
00	0	0	0	0
01	0	0	0	1
11	1	0	0	0
10	1	1	0	0

$Q_3Q_2 \backslash Q_1Q_0$	00	01	11	10
00	0	0	0	1
01	0	0	0	0
11	1	0	0	0
10	1	0	0	0

$Q_3Q_2 \backslash Q_1Q_0$	00	01	11	10
00	1	1	1	1
01	1	1	1	0
11	0	1	1	1
10	0	0	1	1

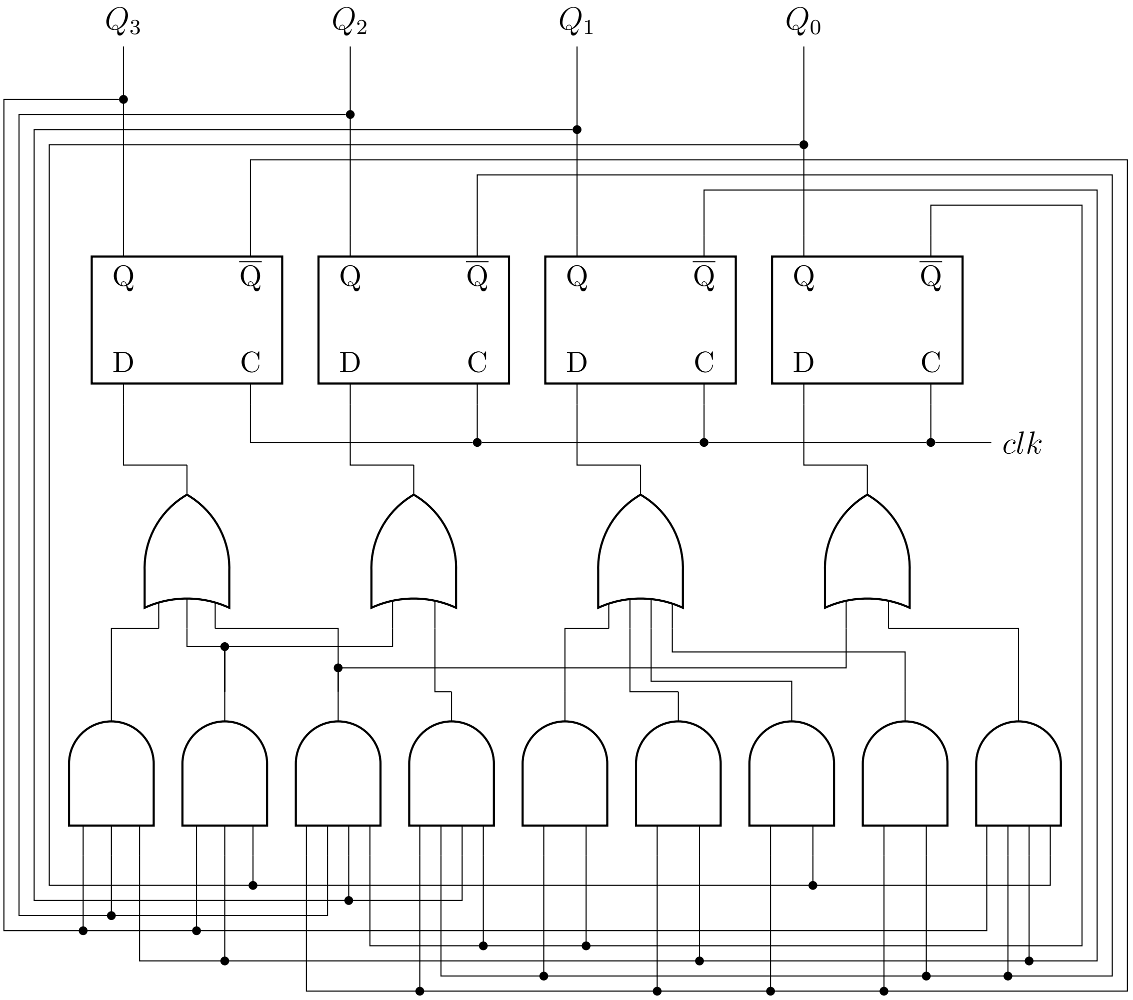
$Q_3Q_2 \backslash Q_1Q_0$	00	01	11	10
00	0	0	0	0
01	0	0	0	1
11	1	0	0	0
10	0	0	0	0

so the simplified state equations for each DFF will be

$$
\begin{aligned}
Q_3(t+1) &= Q_3Q_1'Q_0' + Q_3Q_2'Q_1' + Q_3'Q_2Q_1Q_0' \\
Q_2(t+1) &= Q_3Q_1'Q_0' + Q_3'Q_2'Q_1Q_0' \\
Q_1(t+1) &= Q_3'Q_1' + Q_2Q_0 + Q_3'Q_2' + Q_3Q_1 \\
Q_0(t+1) &= Q_3'Q_2Q_1Q_0' + Q_3Q_2Q_1'Q_0'
\end{aligned}
$$

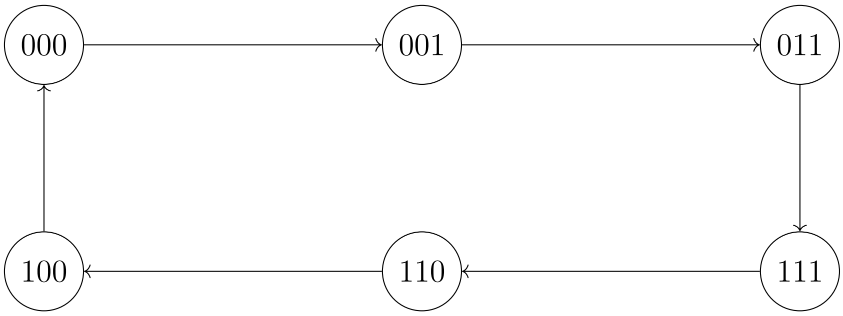
Since in DFF the excitation is equivelant to the state equation, we skip the input equations. To draw the logic diagram, we connect the input components based on the excitation to the D port of each DFF.

And the logic diagram will be like



Problem 4. Design a counter with T flip-flops that goes through the following binary repeated sequence: 0, 1, 3, 7, 6, 4. Derive the input equations for the FFs. Show that when binary states 010 and 101 are considered as don't care conditions, the counter may not operate properly.

Solution. From the sequence, we can derive the state diagram as follows.



and then we can derive the state table as follows

Present State			**Next State**			**JKFF Inputs**		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	T_2	T_1	T_0
0	0	0	0	0	1	0	0	1
0	0	1	0	1	1	0	1	0
0	1	0	X	X	X	X	X	X
0	1	1	1	1	1	1	0	0
1	0	0	0	0	0	1	0	0
1	0	1	X	X	X	X	X	X
1	1	0	1	0	0	0	1	0
1	1	1	1	1	0	0	0	1

And we can derive the Karnaugh maps for each TFF as follows

$Q_2 \backslash Q_1Q_0$	00	01	11	10
0	0	0	1	X
1	1	X	0	0

$Q_2 \backslash Q_1Q_0$	00	01	11	10
0	0	1	0	X
1	0	X	0	1

$Q_2 \backslash Q_1Q_0$	00	01	11	10
0	1	0	0	X
1	0	X	1	0

If we count the don't care conditions, the simplified input equation will be

$$Q_2(t+1) = Q_2Q_1' + Q_2'Q_1 = Q_2 \oplus Q_1$$
$$Q_1(t+1) = Q_1Q_0' + Q_1Q_0' = Q_1 \oplus Q_0$$
$$Q_0(t+1) = Q_2Q_0 + Q_2'Q_0' = (Q_2 \oplus Q_0)'$$

and the state table will be turned into

Present State			**Next State**			**JKFF Inputs**		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	T_2	T_1	T_0
0	0	0	0	0	1	0	0	1
0	0	1	0	1	1	0	1	0
0	1	0	1	0	1	1	1	1
0	1	1	1	1	1	1	0	0
1	0	0	0	0	0	1	0	0
1	0	1	0	1	0	1	1	1
1	1	0	1	0	0	0	1	0
1	1	1	1	1	0	0	0	1

From the state table we could see clearly that if we treat 010 and 101 as don't care condition, the counter will be trapped in a infinite loop between 010 and 101 when we encounter either of the states. Therefore we shall assign 011 to the next state of state 010, and 110 to that of 101 to enable the counter with self-correction. So the state table, Kmaps and simplified equation will be

Present State			**Next State**			**JKFF Inputs**		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	T_2	T_1	T_0
0	0	0	0	0	1	0	0	1
0	0	1	0	1	1	0	1	0
0	1	0	0	1	1	0	0	1
0	1	1	1	1	1	1	0	0
1	0	0	0	0	0	1	0	0
1	0	1	1	1	0	0	1	1
1	1	0	1	0	0	0	1	0
1	1	1	1	1	0	0	0	1

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

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

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

$$Q_2(t+1) = Q_2Q_1'Q_0' + Q_2'Q_1Q_0$$
$$Q_1(t+1) = Q_1'Q_0 + Q_2Q_1Q_0'$$

$$Q_0(t+1) = Q_2Q_0 + Q_2'Q_0' = (Q_2 \oplus Q_0)'$$

and finally we can draw the logic diagram based on the equations