Bangladesh University of Engineering and Technology

Course: CSE 206

Digital Logic Design Sessional

Experiment 7
Flip-Flops and Registers

Group No: 05

Section: B2

Department: CSE

Group Members: 1805111

1805112

1805113

1805114

1805115

1405040

Submission Date: 12/06/2021

Problem-1

Problem Specification:

In this problem, we have to design and implement a **master-slave JK** flip-flop using only **NAND** gates.

Required Instrument:

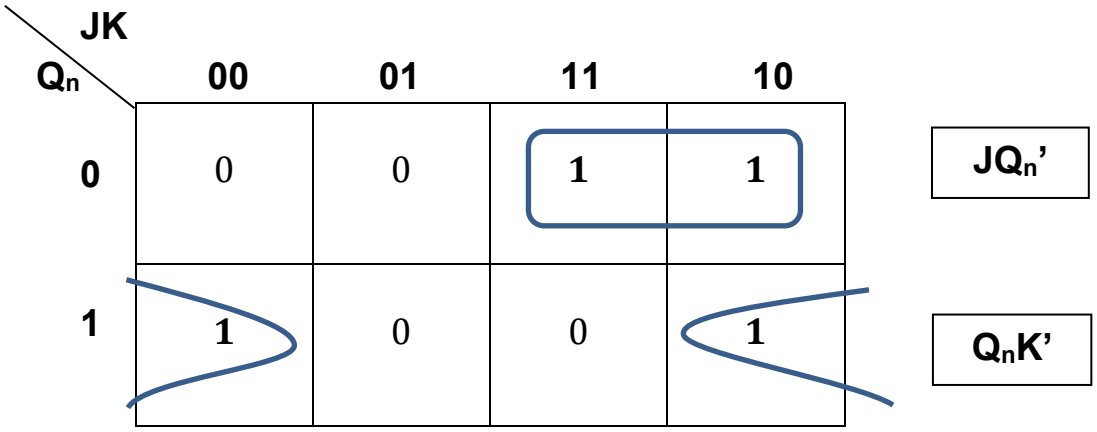
1. Logisim software
2. 7400 IC (4 pieces)
3. 1 clock, 2 input pins, and 2 output pins
4. Electric wires

Excitation Table:

J	K	Q_n	Q_{n+1}
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

K-Map for Master Slave J-K Flip Flop:



From K-Map,
The characteristic equation is,

Q_{n+1}

$= JQ_n' + K'Q_n$

$= (JQ_n' + K'Q_n)''$
$=((JQ_n')' (K'Q_n)')'$

Circuit Diagram:

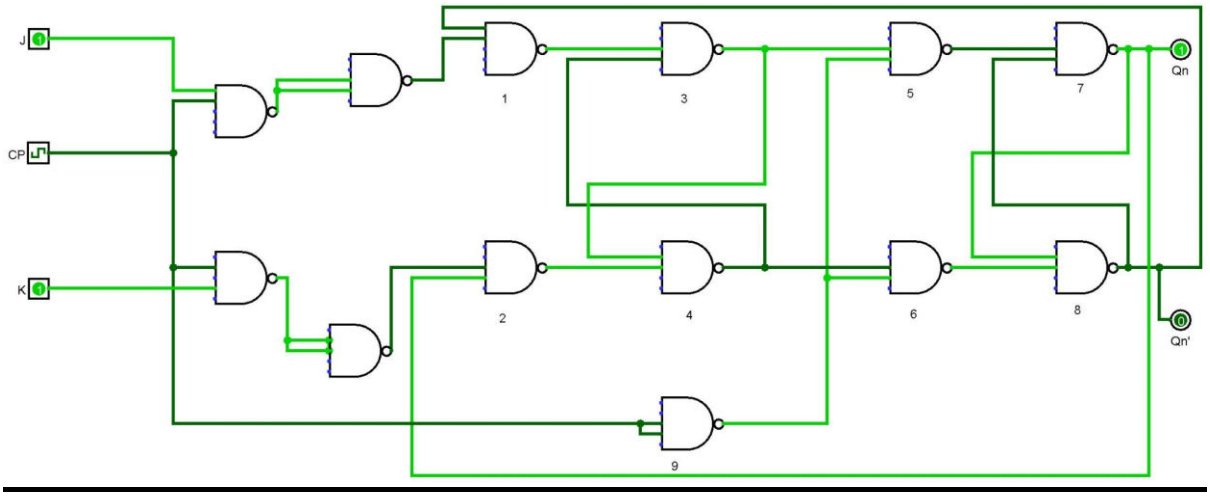


Fig: Master-Slave JK Flip flop

Observation:

The J-K flip flop circuit has a problem regarding the output: when both the excitation inputs are set, the flip flop produces a continuous transition of state. That means, in this case, for a single clock pulse the state is being toggled continuously until the triggering of the clock pulse is being stopped. This phenomenon is known as the **'Race Around' condition**. To solve this problem, **a master-slave JK** flip flop is introduced. This is a negative edge-triggered flip flop. In this configuration, two similar consecutive flip flops are used. The first one is called the master flip flop and the latter one is called the slave flip flop. For a single clock pulse, one of them becomes active and the other one remains to stop. That's why, when both excitation input is set (1), the state is being toggled once at a time for a single clock pulse.

Problem- 2

Problem Specification:

In this problem, we have to design and implement a 4-bit universal shift register.

Required Instruments:

1. Trainer Board	-	1 piece
2. IC 7474	-	2 pieces
3. IC 74153	-	2 pieces
4. Input pins	-	8 pieces
5. Output pins	-	4 pieces
6. Button	-	1 piece
7. Clock	-	1 piece
8. Wires	-	a lot

Excitation Table:

In this table, I_3, I_2, I_1, I_0 represent the parallel inputs. SRI represents serial input for the right shift and SLI is the serial input for the left shift.

Input					Output				
Clear'	Clock	S_1	S_0	$D_{i(i=0,1,2,3)}$	A_3	A_2	A_1	A_0	Mode
0	X	X	X	X	0	0	0	0	Asynchronous Clear
1	X	0	0	X	A_3	A_2	A_1	A_0	Data hold
1	H	0	1	A_{i+1}	SRI	A_3	A_2	A_1	Shift right
1	H	1	0	A_{i-1}	A_2	A_1	A_0	SLI	Shift left
1	H	1	1	A_i	I_3	I_2	I_1	I_0	Parallel load

Circuit Diagram:

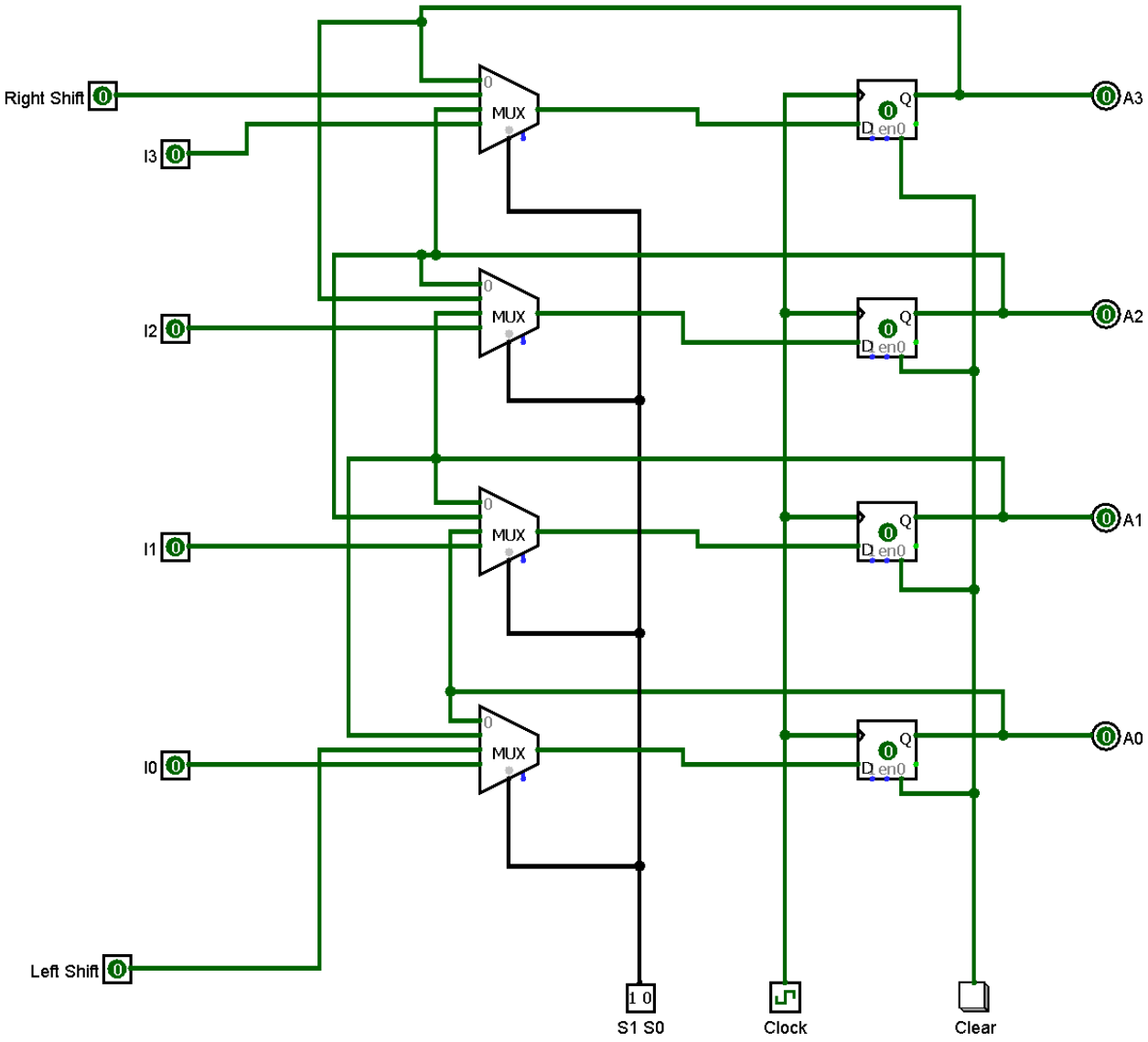


Fig: 4-bit universal shift register

Observations:

This register can perform 3 operations such as shift-left, shift-right, and parallel loading. It can be used for arithmetic operations like multiplication and division. If we perform the left shift with signal 0 in the serial input for the left shift, it is equivalent to multiplication by 2. And when we perform the right shift with signal 0 in the serial input for the right shift, the operation is equivalent to integer division by 2. This register acts as an interface between one device to another device to transfer the data. It is also used for data transmission at a low cost. Because of the serial input feature, data can be transmitted through a single line. Otherwise, we would need n separate lines for transmitting n bits of data. This reduces the cost of data transfer**.**