Name: **Samrathpreet Singh Randhawa**

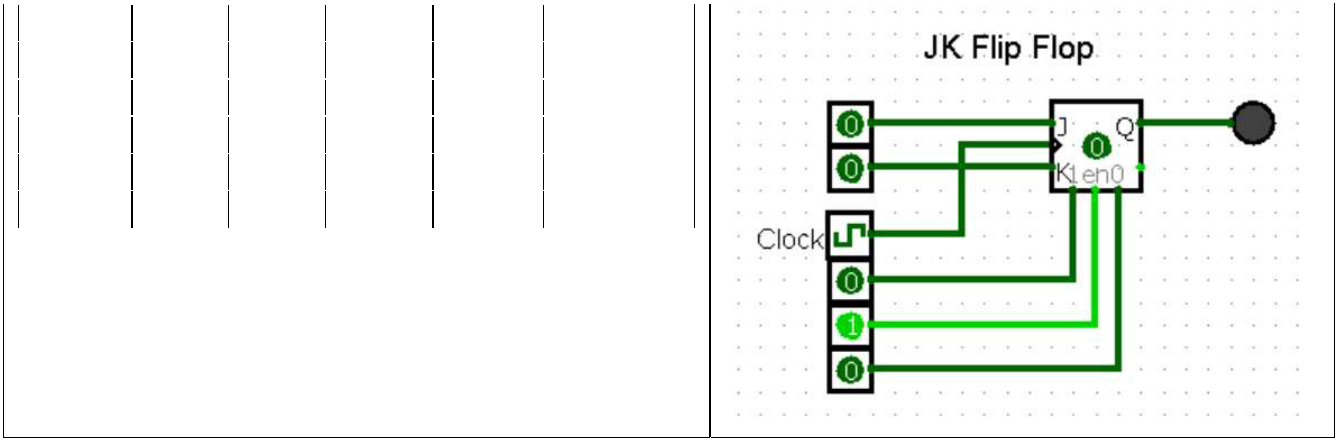
Roll No.: **1801CS43**

**CS-226 (Switching Theory Lab)**

**Lab-8**

**Question 0)** Basic sequential elements J-K, Q and D flip flops.

**JK Flip Flop**

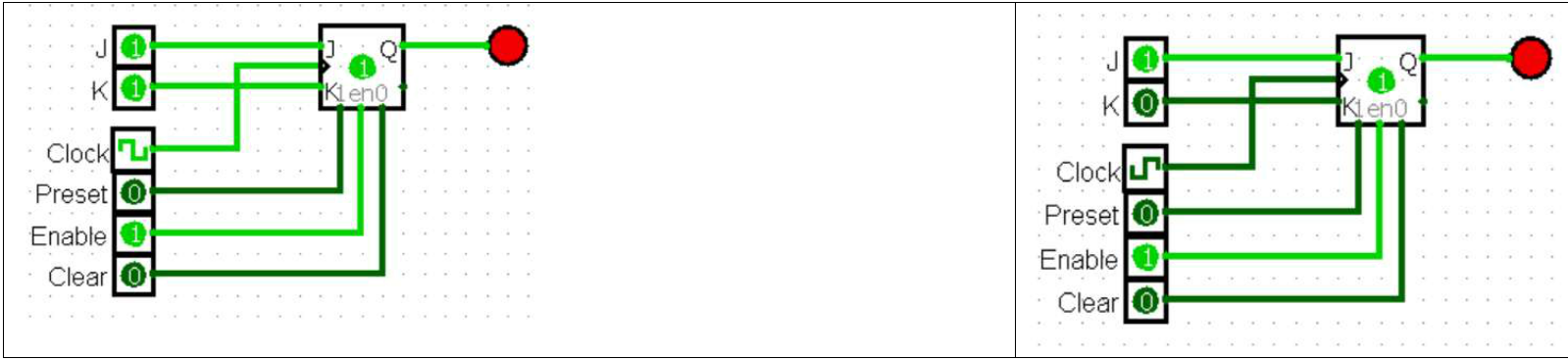


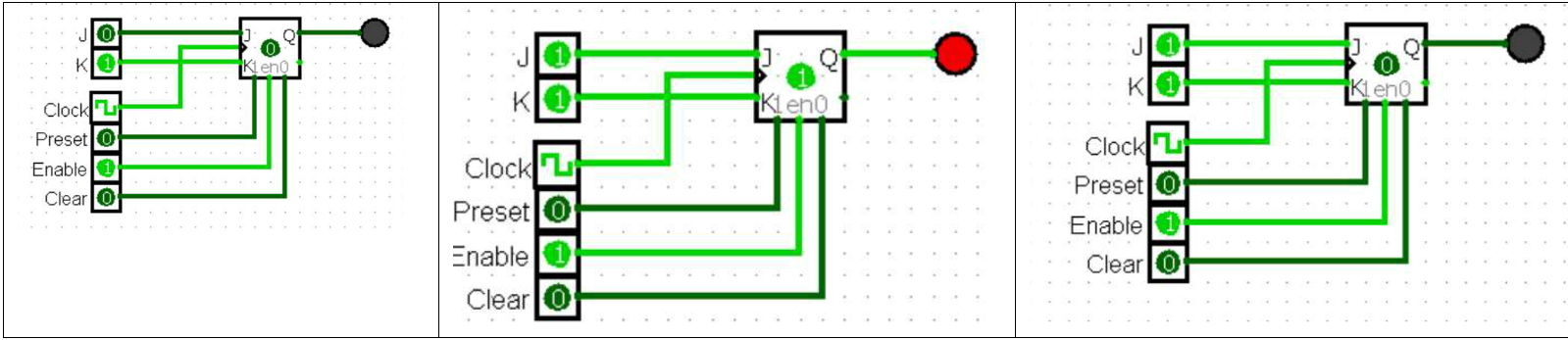
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Clock** |  | **J** |  | **K** | **Q(t)** |  | **Q’(t)** | **State** |  |
|  | **0** |  | **X** |  | **X** | **Q(t-1)** |  | **Q’(t-1)** | **Memory** |  |
|  | **1** | **0** | | **0** | | **Q(t-1)** |  | **Q’(t-1)** | **Memory** |  |
|  | **1** | **0** | | **1** | | **0** | **1** | | **Reset** |  |
|  | **1** | **1** | | **0** | | **1** | **0** | | **Set** |  |
|  | **1** |  | **1** |  | **1** | **Q’(t-1)** |  | **Q(t-1)** | **Toggle** |  |

**Truth Table**

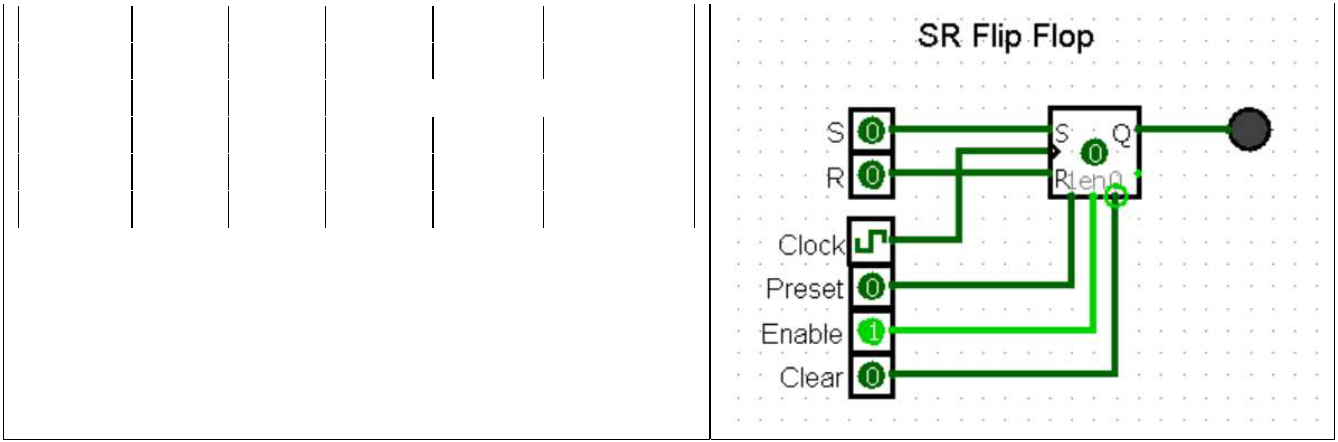
Test Case:

1. Set J = 1, K = 1 followed by Clock = 1, notice Value stored = 1 (Set State)
2. Set Clock = 0, change J and K randomly, no change in Value stored (Memory State)
3. Set J = 0, K = 1, then Clock = 1, notice Value stored = 0 (Reset State)
4. Set Clock = 0
5. Set J = 1, K = 1, and keep toggling the clock, notice value stored also toggles (Toggle State)





**SR Flip Flop**

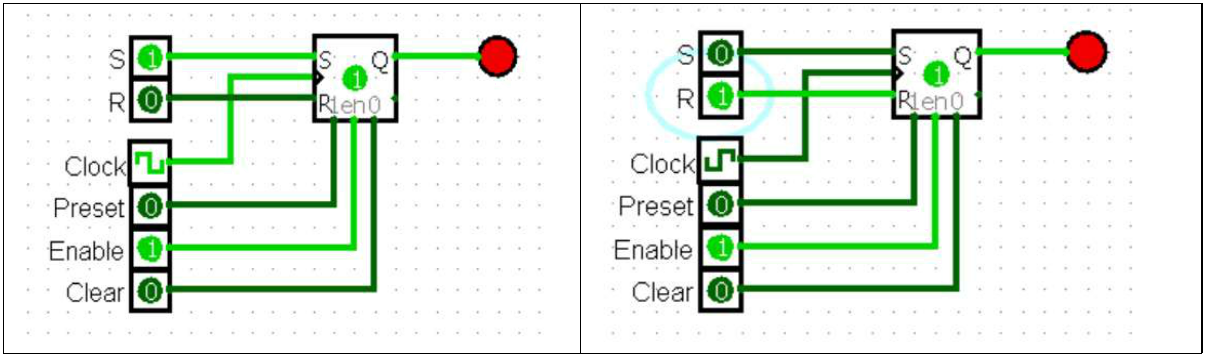


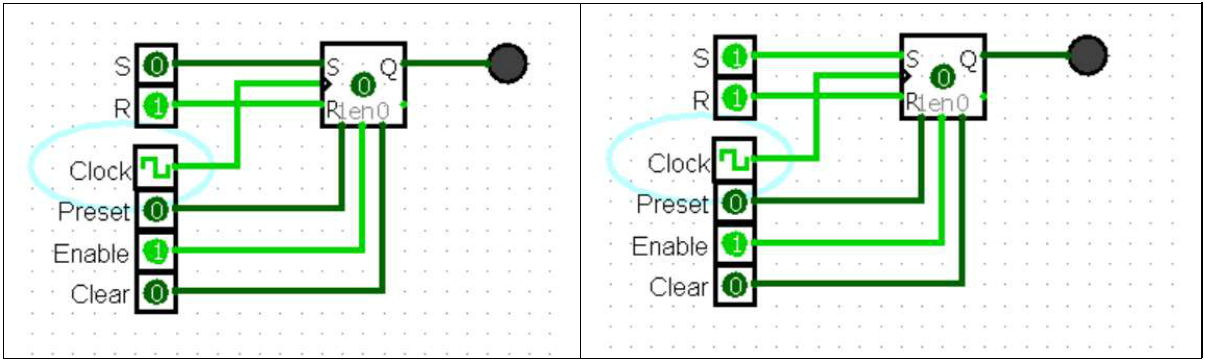
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Clock** |  | **S** |  | **R** | **Q(t)** |  | **Q’(t)** | **State** |  |
|  | **0** |  | **X** |  | **X** | **Q(t-1)** |  | **Q’(t-1)** | **Memory** |  |
|  | **1** | **0** | | **0** | |  |  | **Forbidden** | |  |
|  | **1** | **0** | | **1** | | **0** | **1** | | **Reset** |  |
|  | **1** | **1** | | **0** | | **1** | **0** | | **Set** |  |
|  | **1** |  | **1** |  | **1** | **Q(t-1)** |  | **Q’(t-1)** | **Memory** |  |

**Truth Table**

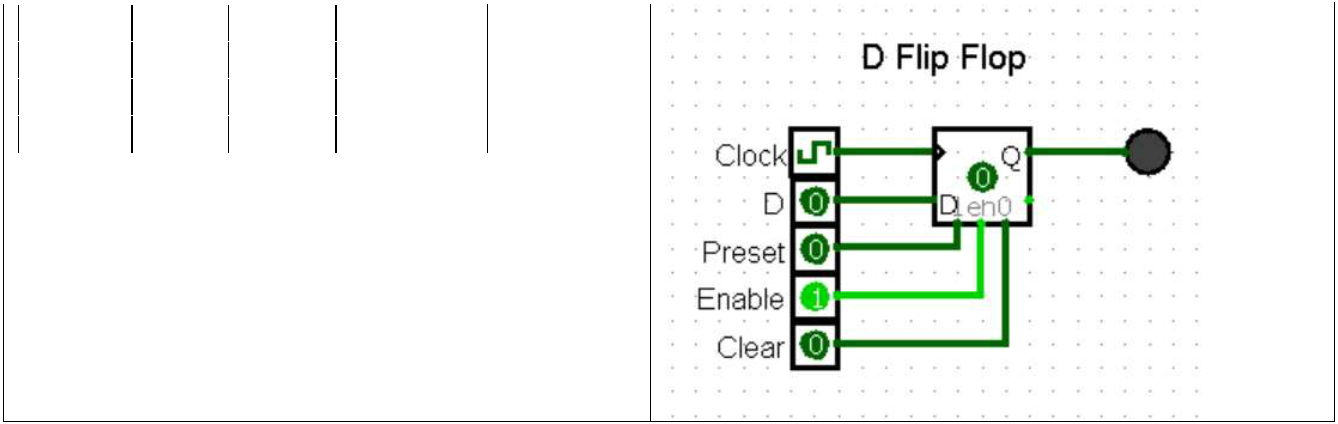
Test Case:

1. Set S = 1, followed by Clock = 1, notice Value stored = 1 (Set State)
2. Set Clock = 0, change S and R randomly, no change in Value stored (Memory State)
3. Set S = 0, R = 1, then Clock = 1, notice Value stored = 0 (Reset State)
4. Set Clock = 0
5. Set S = 1, R = 1, then Clock = 1, no change in Value stored (Memory State)





**D Flip Flop**

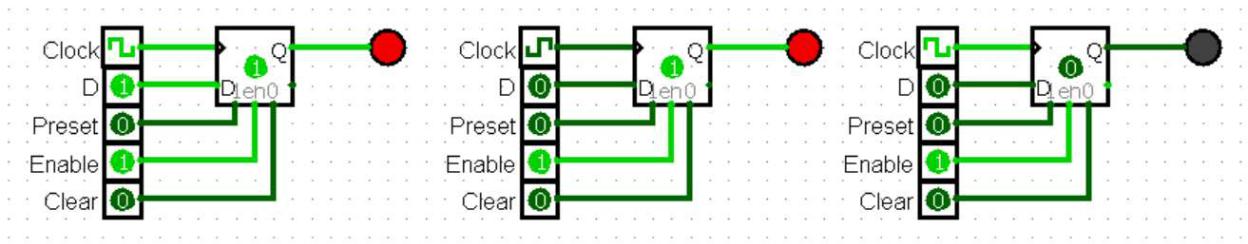


|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Clock** |  | **D** |  | **Q(t)** | **State** |  |
|  | **0** |  | **X** |  | **Q(t-1)** | **Memory** |  |
|  | **1** | **0** | | **0** | | **Reset** |  |
|  | **1** |  | **1** |  | **1** | **Set** |  |

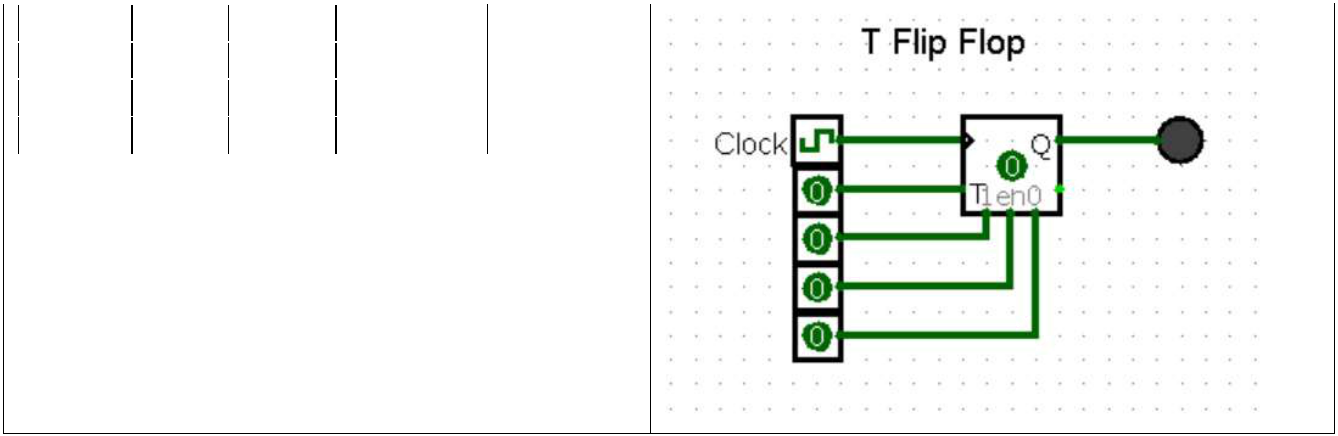
**Truth Table**

Test Case:

1. Set Clock = 1, D = 1, notice Value Stored = 1
2. Set Clock = 0, Value stored doesn’t change on changing D (Memory)
3. Set D = 1, Clock = 1, notice Value Stored = 1



T Flip Flop

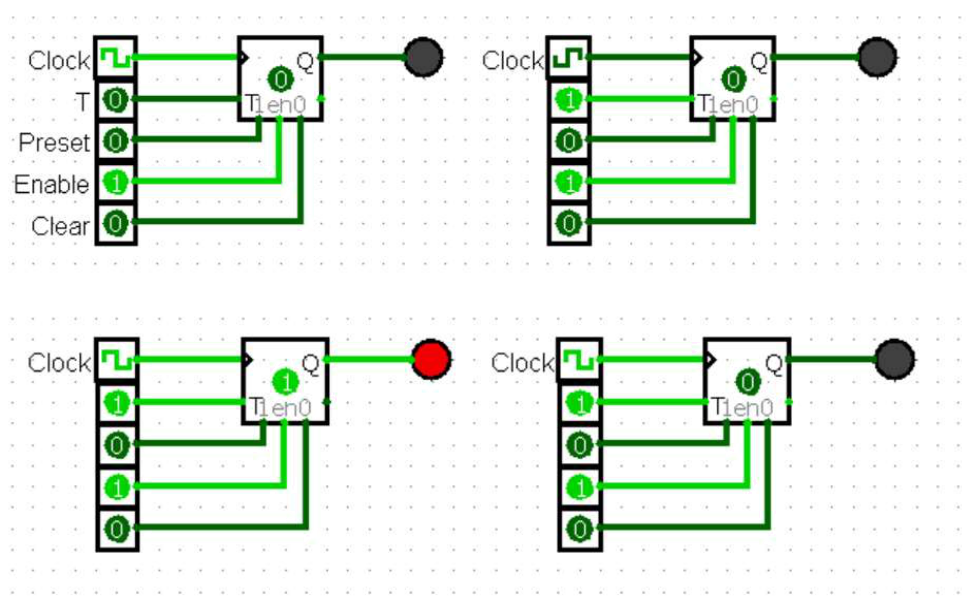


|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Clock** |  | **T** |  | **Q(t)** | **State** |  |
|  | **0** |  | **X** |  | **Q(t-1)** | **Memory** |  |
|  | **1** | **0** | |  | **Q(t-1)** | **Memory** |  |
|  | **1** |  | **1** |  | **Q’(t-1)** | **Toggle** |  |

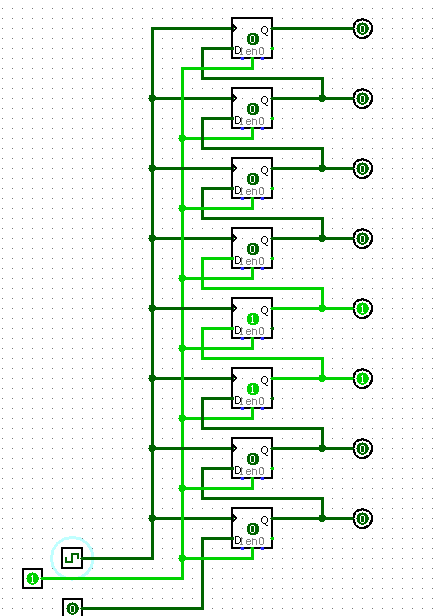
**Truth Table**

Test Case:

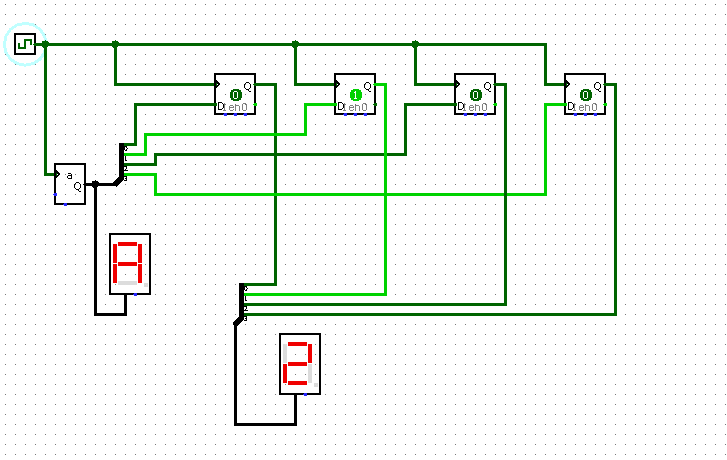
1. Set Clock = 1, T = 0, notice no change in Value Stored
2. Set Clock = 0, Value stored doesn’t change on changing T (Memory)
3. Set T = 1, Clock = 1, and keep toggling the clock, notice Value stored also toggles



**Question 1)** 8 Bit Shift Register using flipflop is implemented in file p1. circ

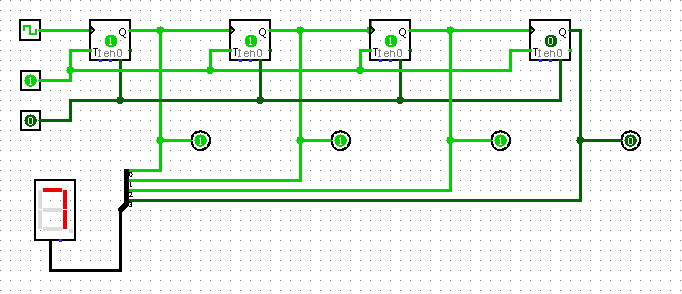


**Question 2)** Simulated 4-bit Parallel Input Parallel Output (PIPO) register and submitted in p2. circ

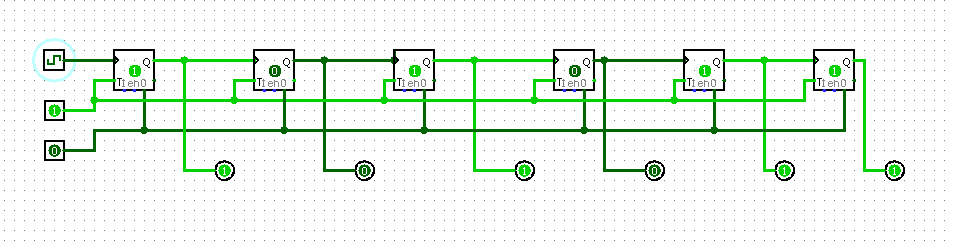


**Question 3)** Submitted in p3. circ

4 Bit counter using T-Flip Flops

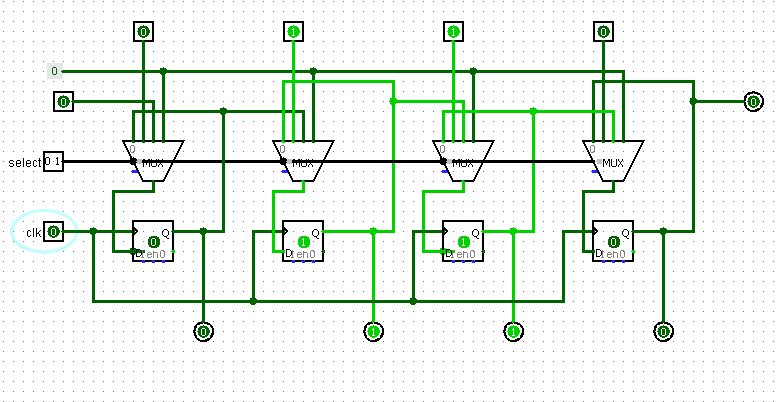


6 Bit counter using T-Flip Flops

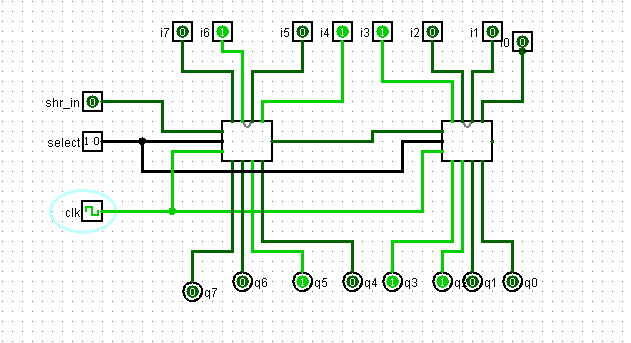


**Question 4)** Simulated 8-bit Multifunction Shift register using Logisim submitted in p4. circ

4bit multi-function register



8bit version multi-function register



**Question 5)** Implemented a 16 x 16 register file submitted in p5. circ

1. 