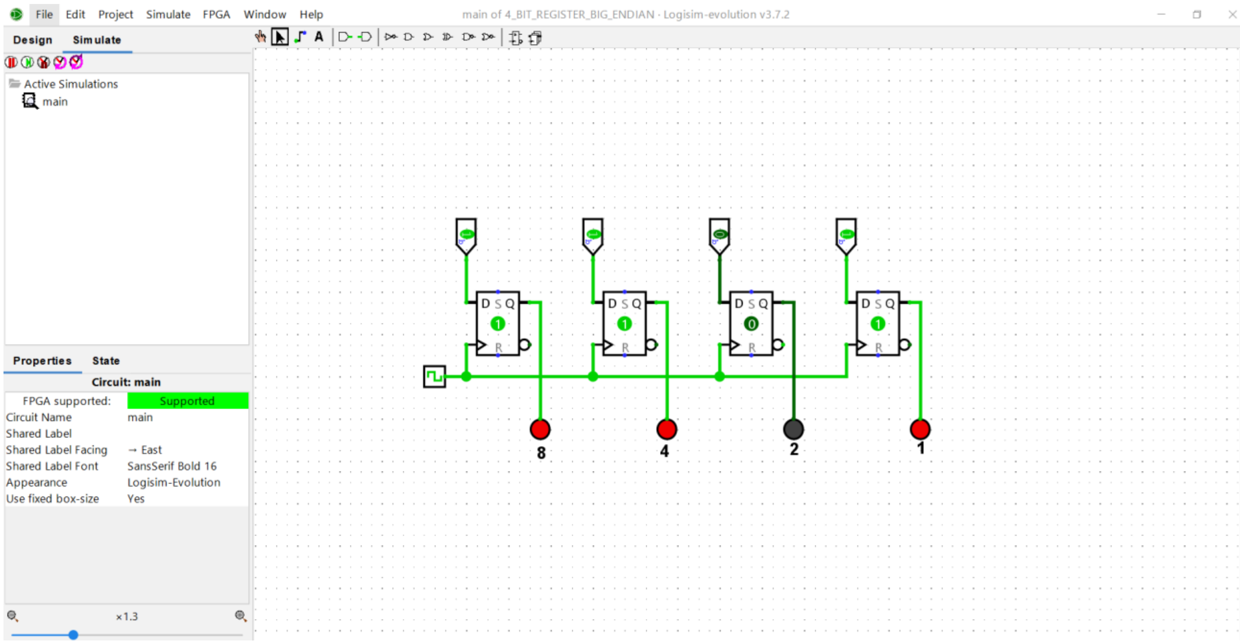
I – Big-endian 4-bit register:

1-5:



6: Test schedule

0x	Input binary	Output binary
0	0000	0000
1	0001	0001
2	0010	0010
3	0011	0011
4	0100	0100
5	0101	0101
6	0110	0110
7	0111	0111
8	1000	1000
9	1001	1001
A	1010	1010
B	1011	1011
C	1100	1100
D	1101	1101
E	1110	1110
F	1111	1111

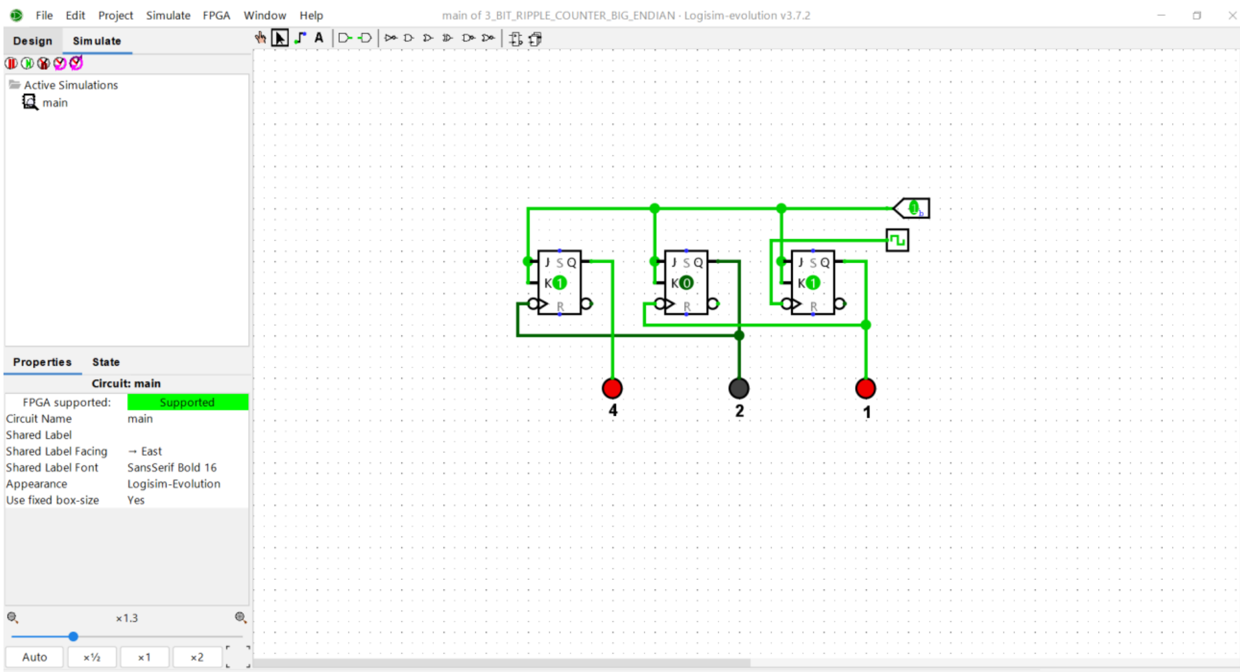
7:

7.1: In modern computing architectures, counters can help keep track of the number of times an event has happened.

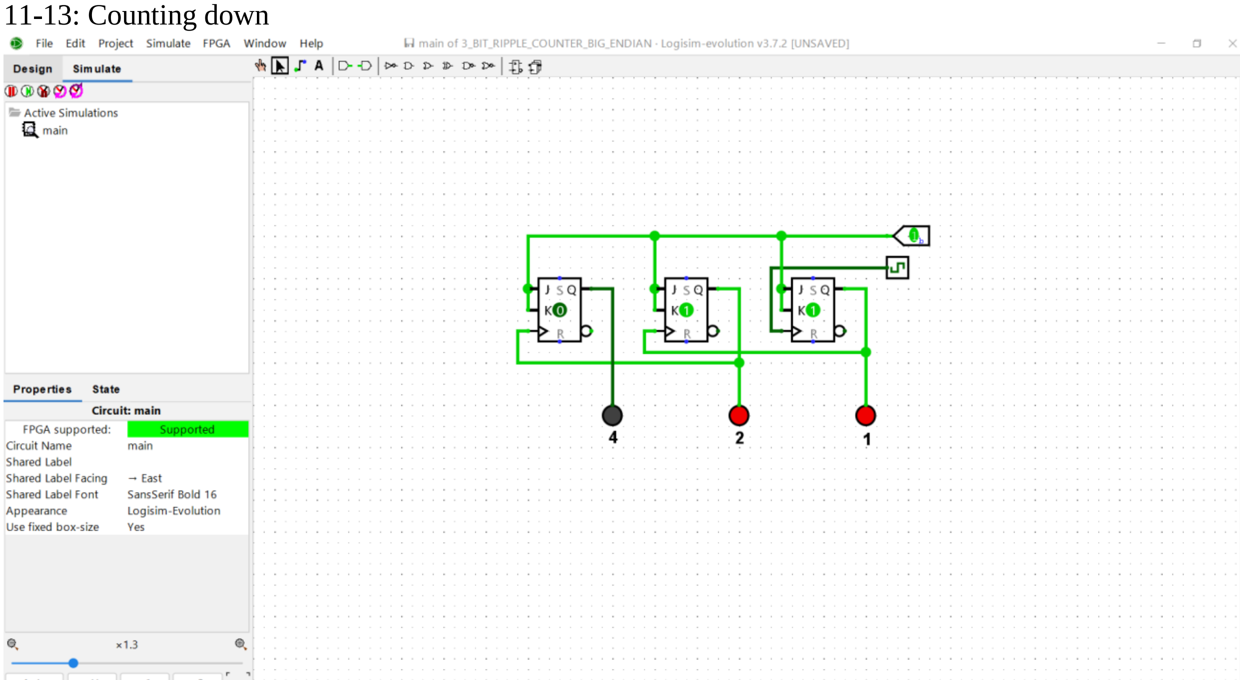
7.2: A ripple counter works by combining multiple toggle JK flip-flops together, with the output of one JK flip-flop used as the clock input of the next JK flip-flop. The toggle frequency of the first JK flip-flop is half the frequency of the clock, the toggle frequency of the second JK flip-flop is half the frequency of the first, and so on. The clock pulse can be thought of as rippling through the circuit, halving its frequency for every pass through a JK flip-flop, which imitates the act of counting in binary.

II – Ripple counter

8-10: Counting up

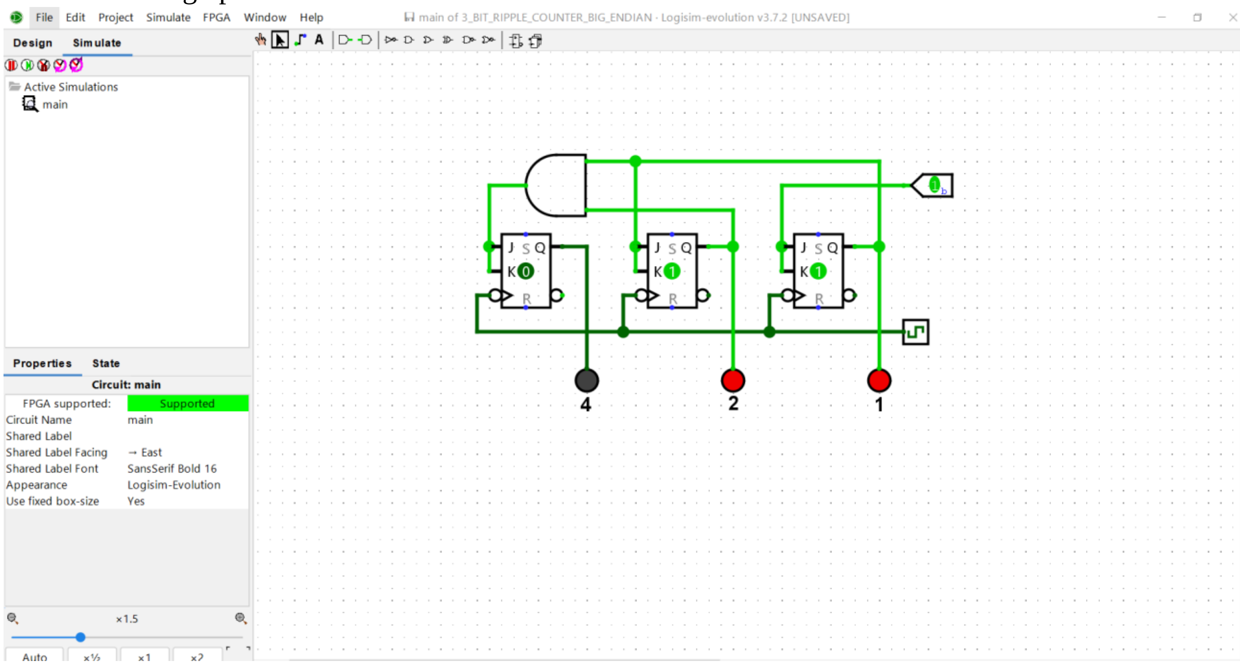


11-13: Counting down

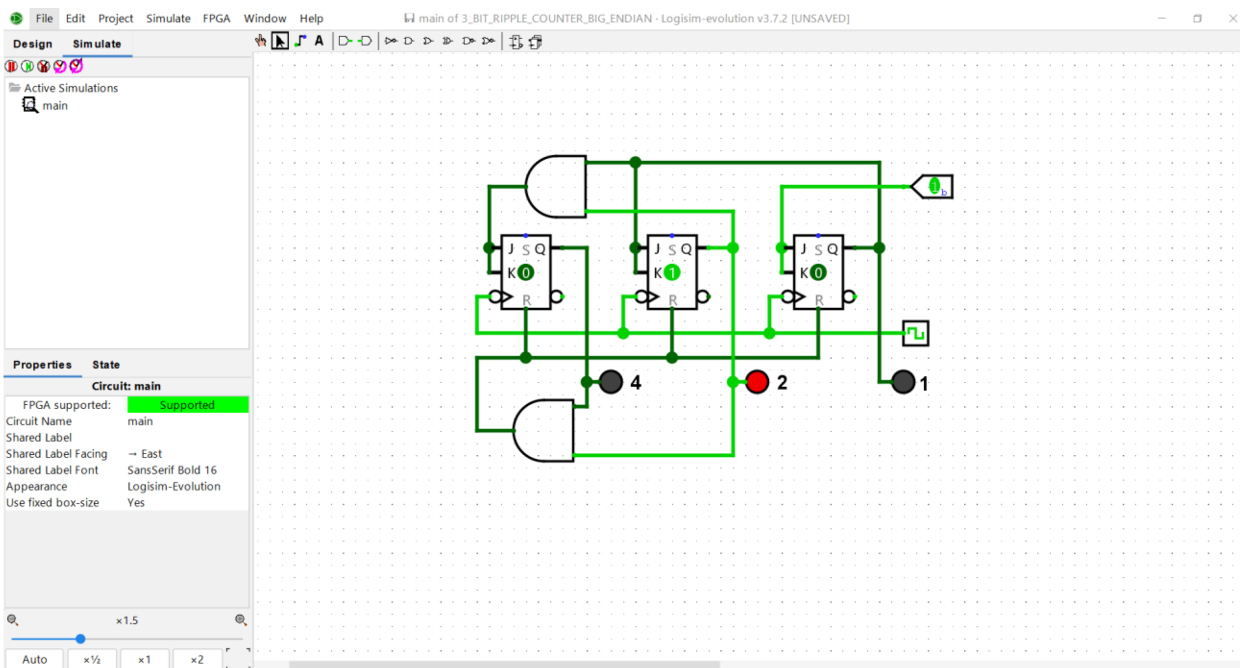


III – Common clock counter

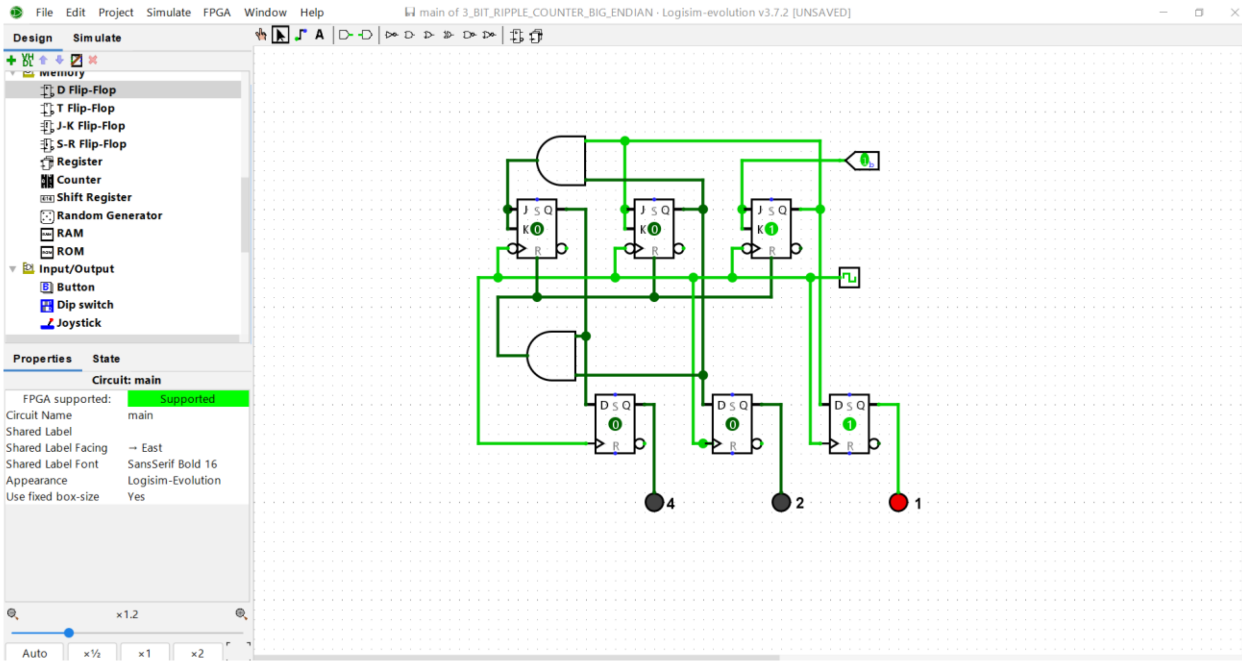
14-15: Counting up



16: Counting from 0 to 5 only (mod 6)



17: Counting from 0 to 5 only (mod 6) with D flip-flop buffers



17.2: Handling the illegal state with D flip-flop buffers will make the circuit more stable and predictable, which can be important if the counter is used in critical applications.

18: Counting from 0 to 5 with HEX display: