

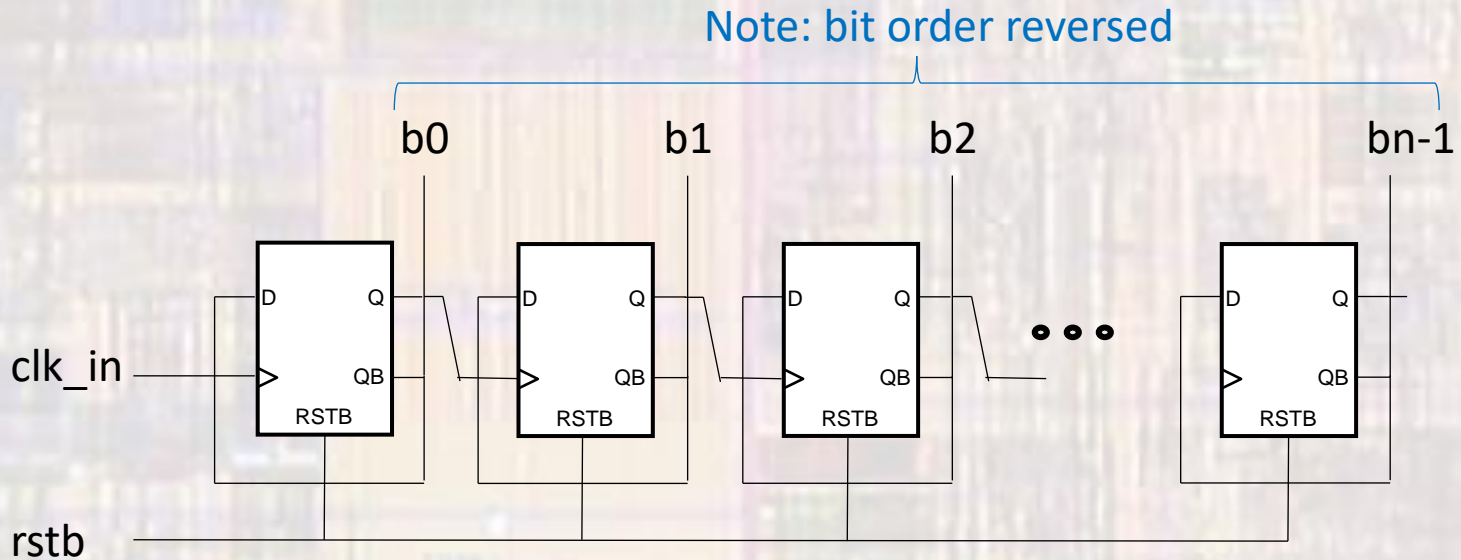
# Counters

Last updated 1/9/25

# Counters

- Binary Counter

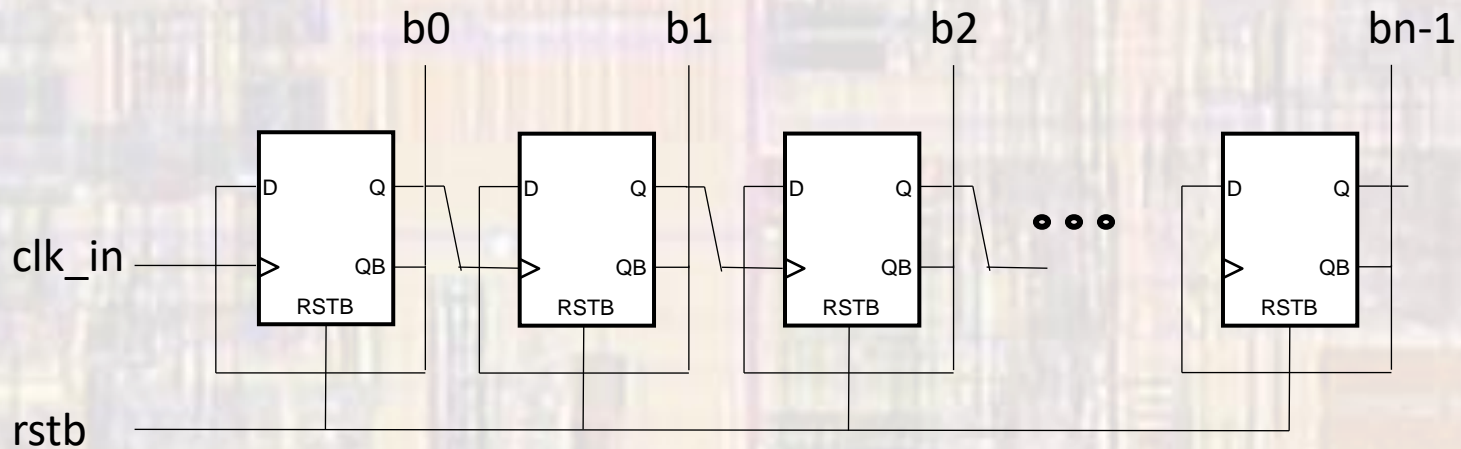
- 0000 → 0001 → 0010 → 0011 → 0100 ... 1111 → 0000 ...



# Counters

- Binary Counter

	0	0	0	0
clk ↑	1	0	0	0
clk ↑ Q1 ↑	0	1	0	0
clk ↑	1	1	0	0
clk ↑ Q1 ↑ Q2 ↑	0	0	1	0

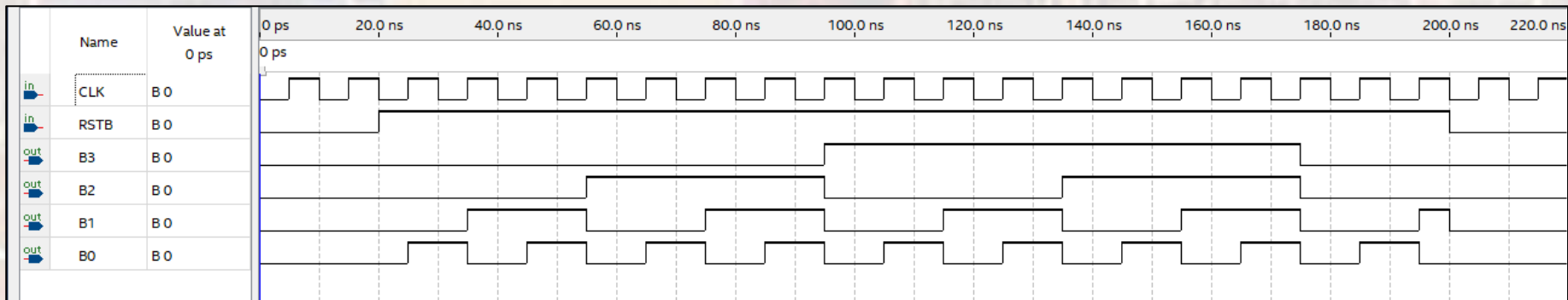
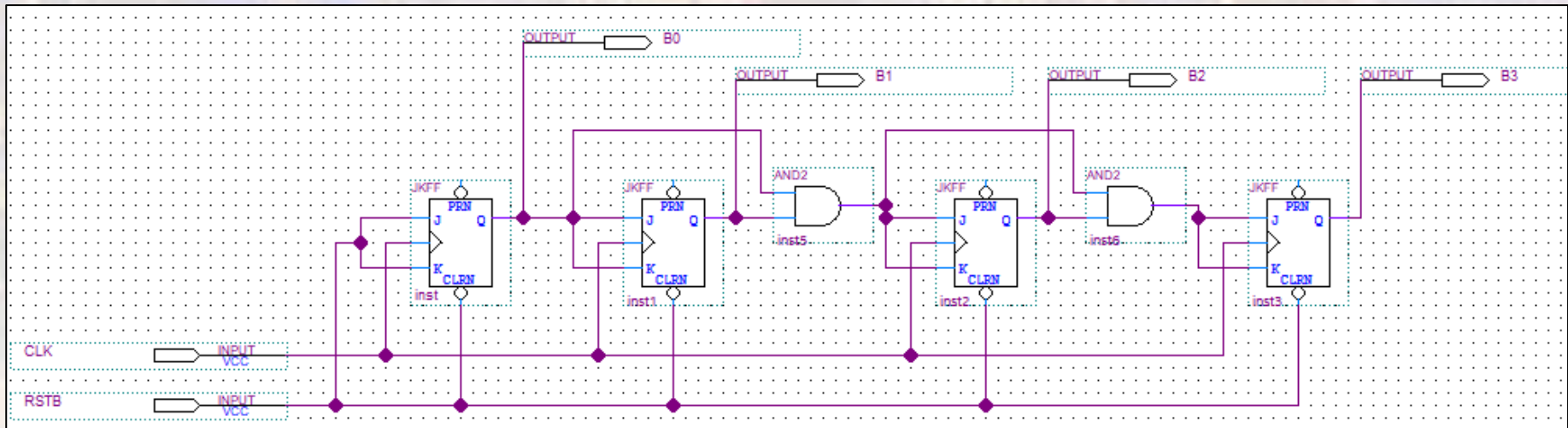


**What's wrong with this solution**

# Counters

- Binary Counter - improved

- $0000 \rightarrow 0001 \rightarrow 0010 \rightarrow 0011 \rightarrow 0100 \dots 1111 \rightarrow 0000 \dots$

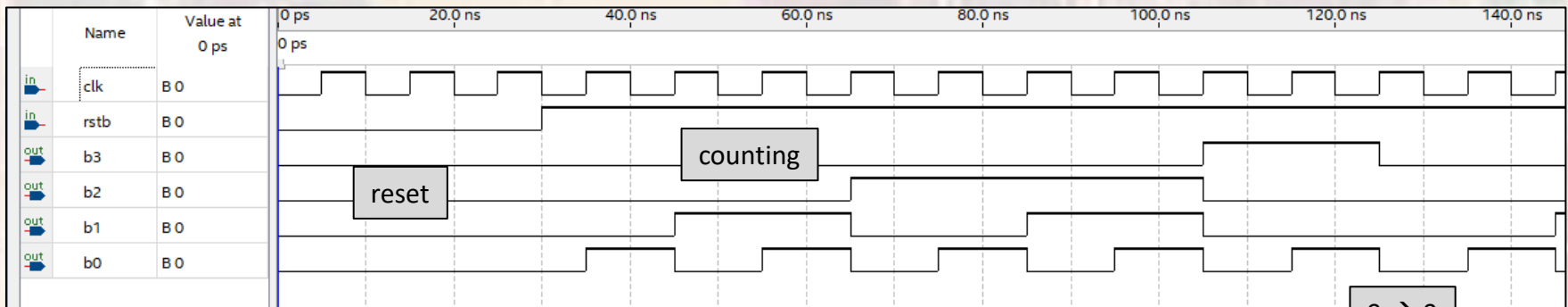
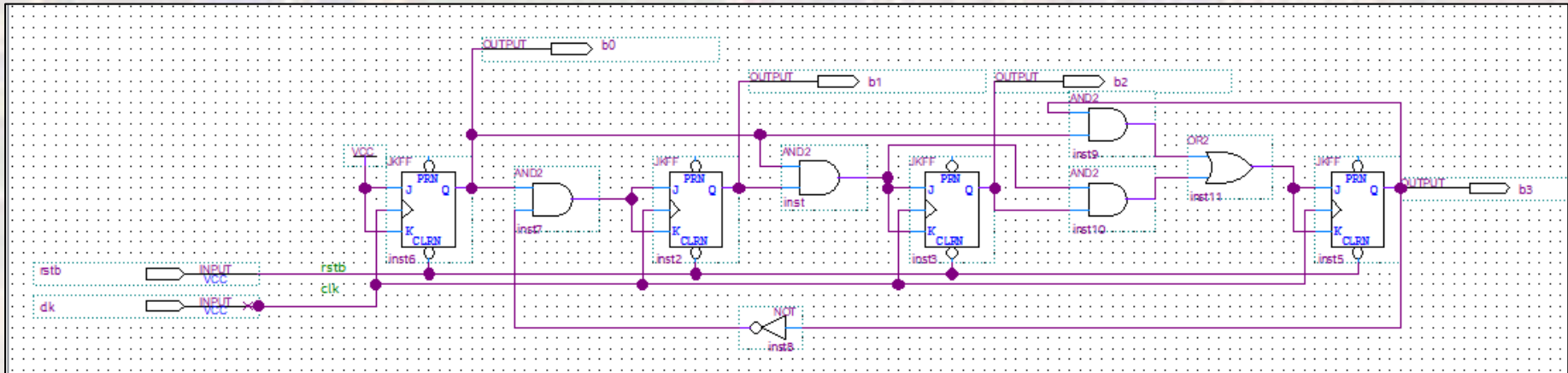


# Counters

- Modulo 10 Counter
    - Mod 10
    - Counts from 0 to 9, then back to 0, ...
  - If output is 9 (1001), want next value to be 0 (0000)
    - 1001  $\rightarrow$  1010 but we want 1001  $\rightarrow$  0000
      - no change in LSB
      - don't allow bit 1 to change (force JK to 0 – *and* with B3 *not*)
        - this works since no B1 change between 1000 and 1001
      - no change in bit 2
      - force bit 3 to zero (force JK to 1 – add in a path with B0 *and* B1)
- once in 0 state – progresses normally

# Counters

- Modulo 10 Counter
  - Mod 10
  - Counts from 0 to 9, then back to 0, ...



9 → 0

# Counters

- Adder Based Counter

