Sequential Logic

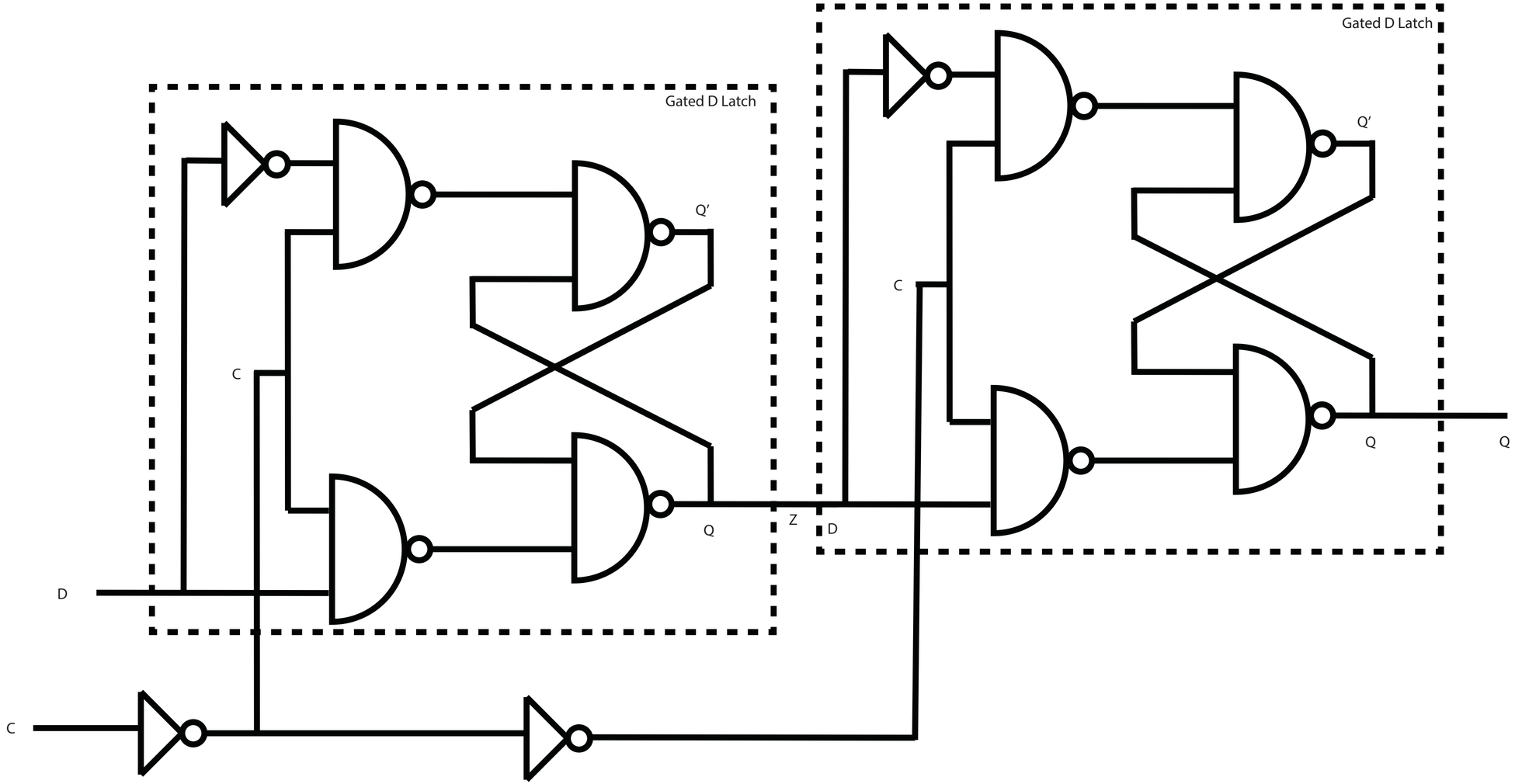
Flip-flops and Counter Implementation

Kumaravel Jagasivamani
Feb. 14, 2024

Overview
Part 2

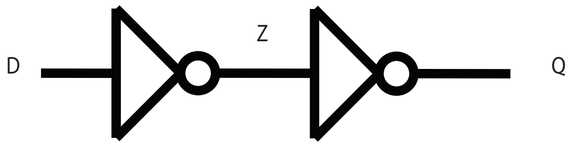
- Master-Slave D Flip Flop
  - Circuit
  - Propagation Delay
  - Timing Diagram
  - Setup/Hold Time
  - Enable input
  - Reset input
- Counter
  - Truth Table
  - Inverting bits using enable

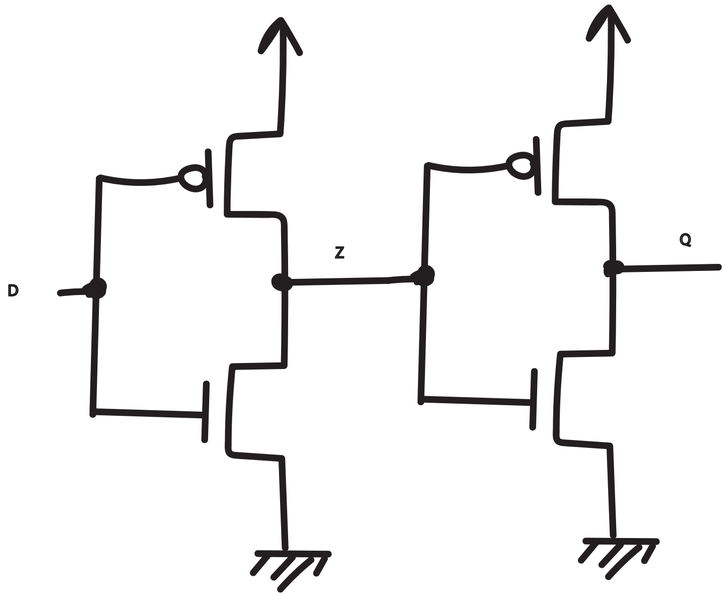
Positive-Edge Master-Slave D Flip-Flop

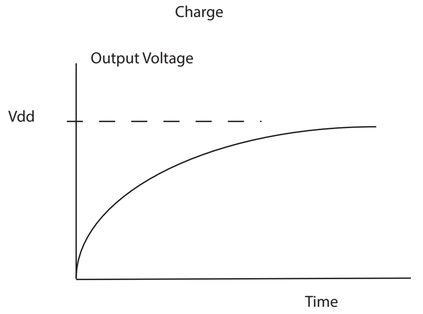


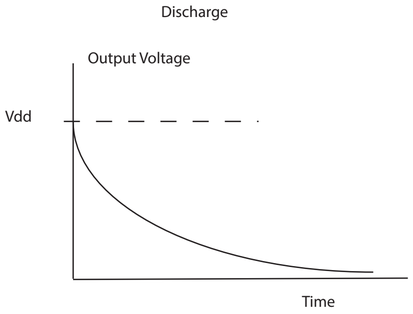
Propagation Delay

- Wire Delay
- Charge/Discharge Delay





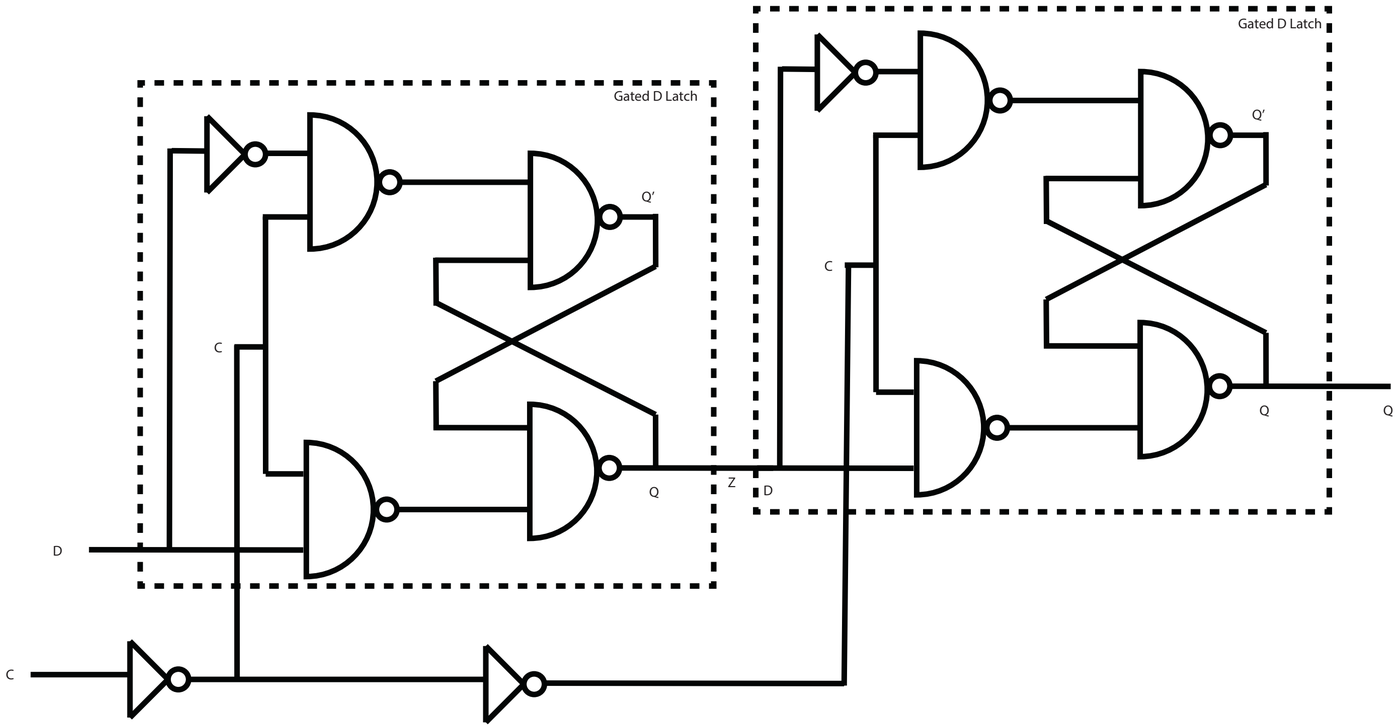


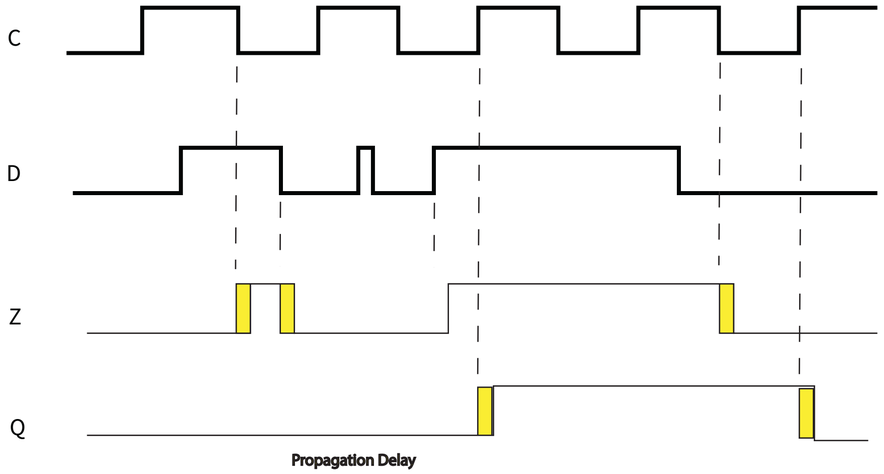


Positive-Edge Master-Slave D Flip-Flop

Timing Diagram with Propagation Delay

- Q updates take place only on rising-edge of C (clock) signal

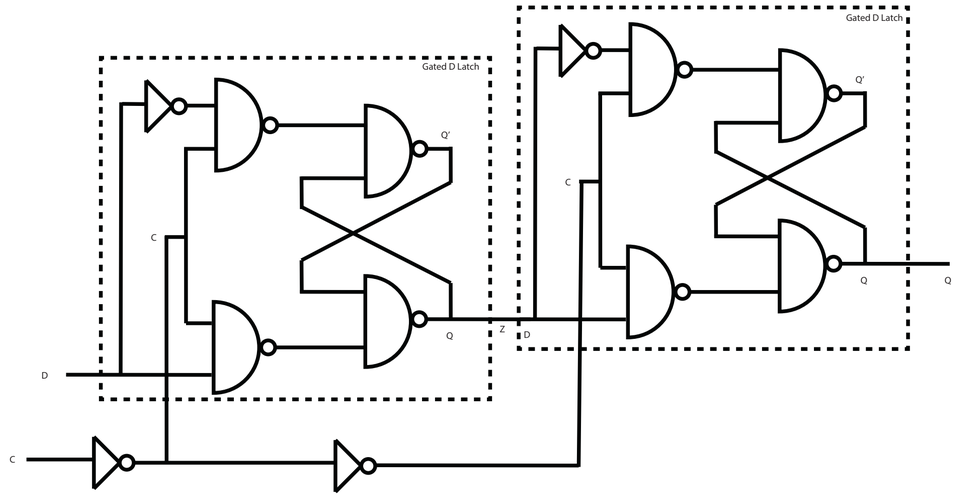


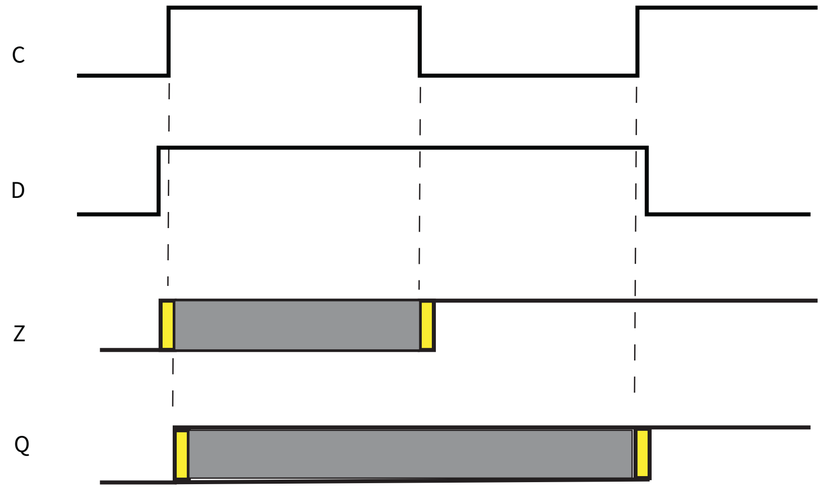


Positive-Edge Master-Slave D Flip-Flop

Setup Time

- D input need to be stable before C rising edge
- Race condition between D and C
- Avoid uncertainty of output



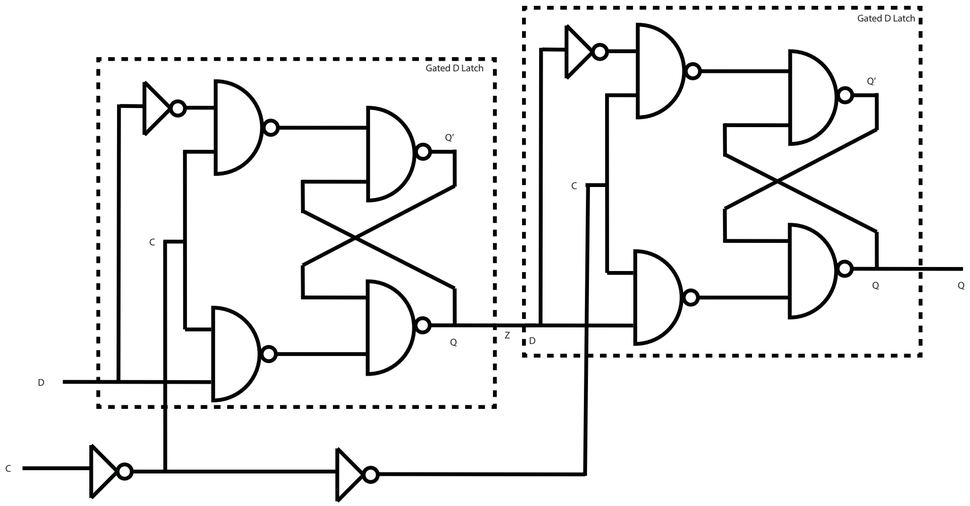


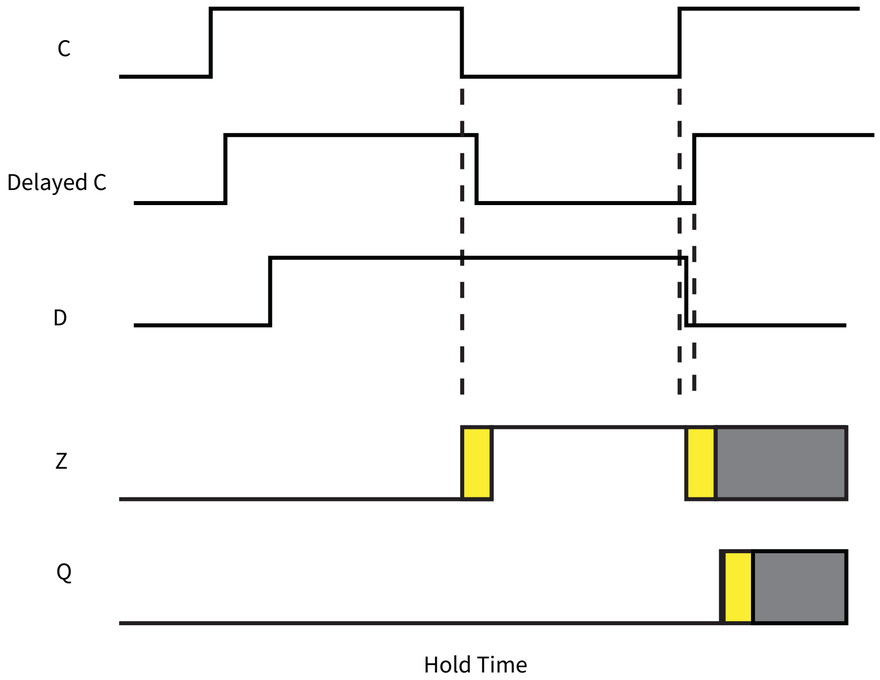
Setup Time

Positive-Edge Master-Slave D Flip-Flop

Hold Time

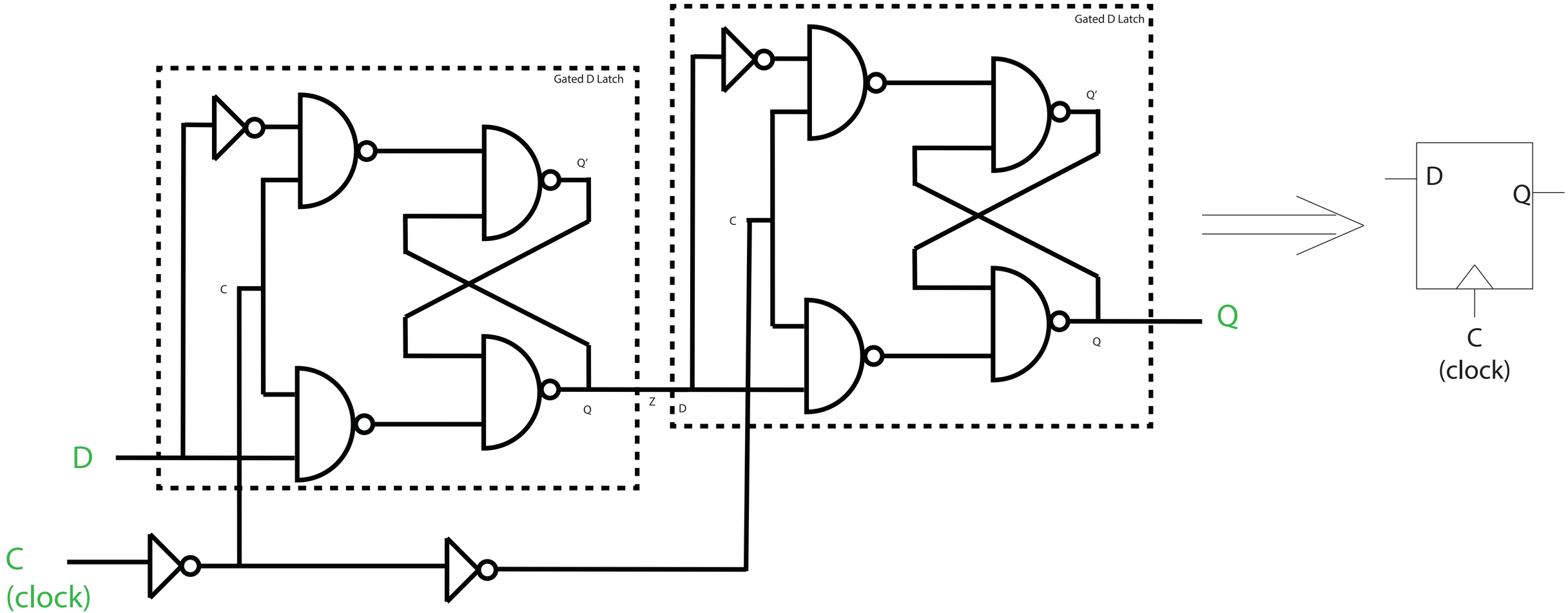
- D input need to be stable after C rising edge
- Race condition between D and C
- Avoid uncertainty of output





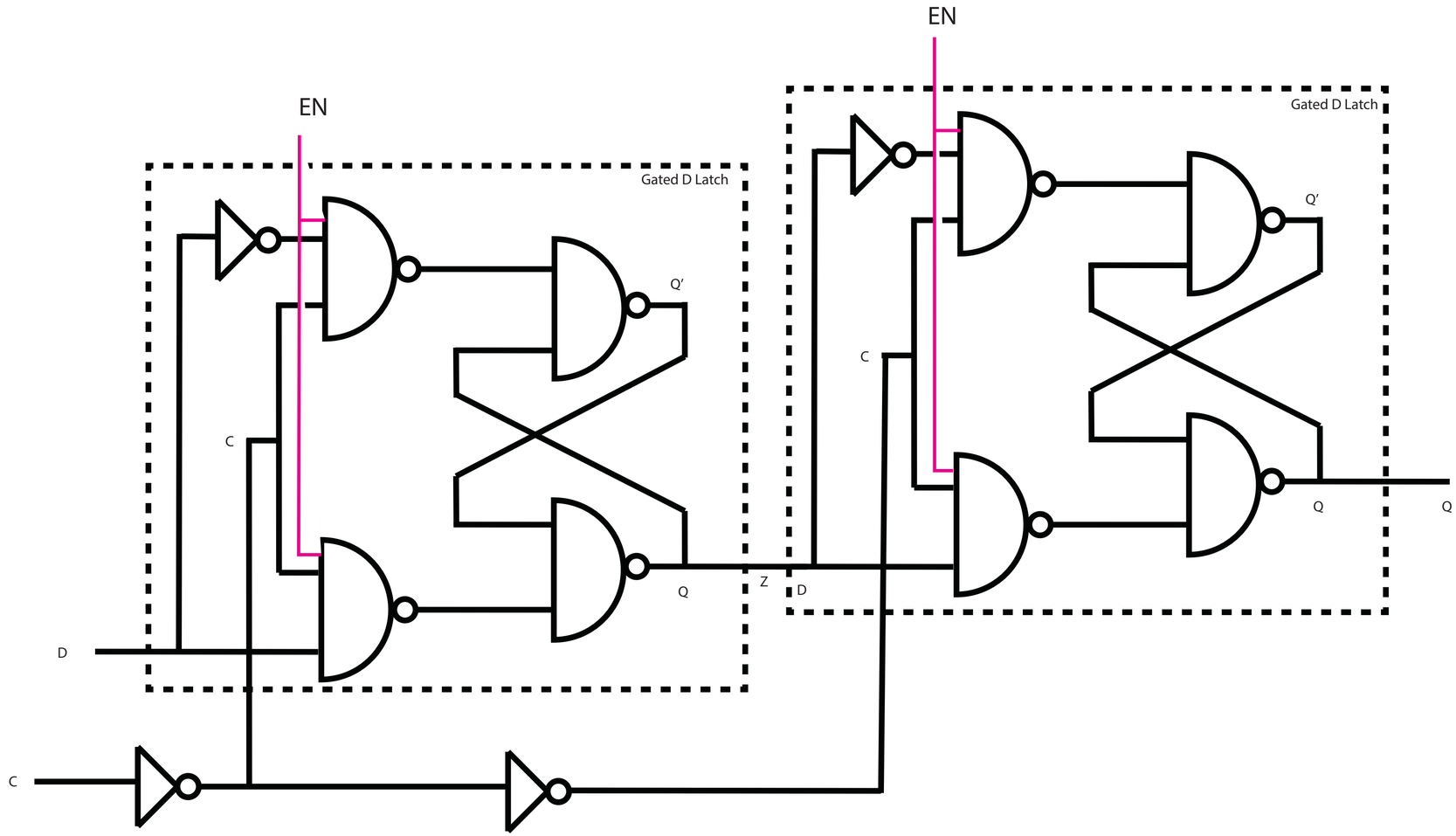
Positive-Edge Master-Slave D Flip-Flop

Standard (Simplified) Representation



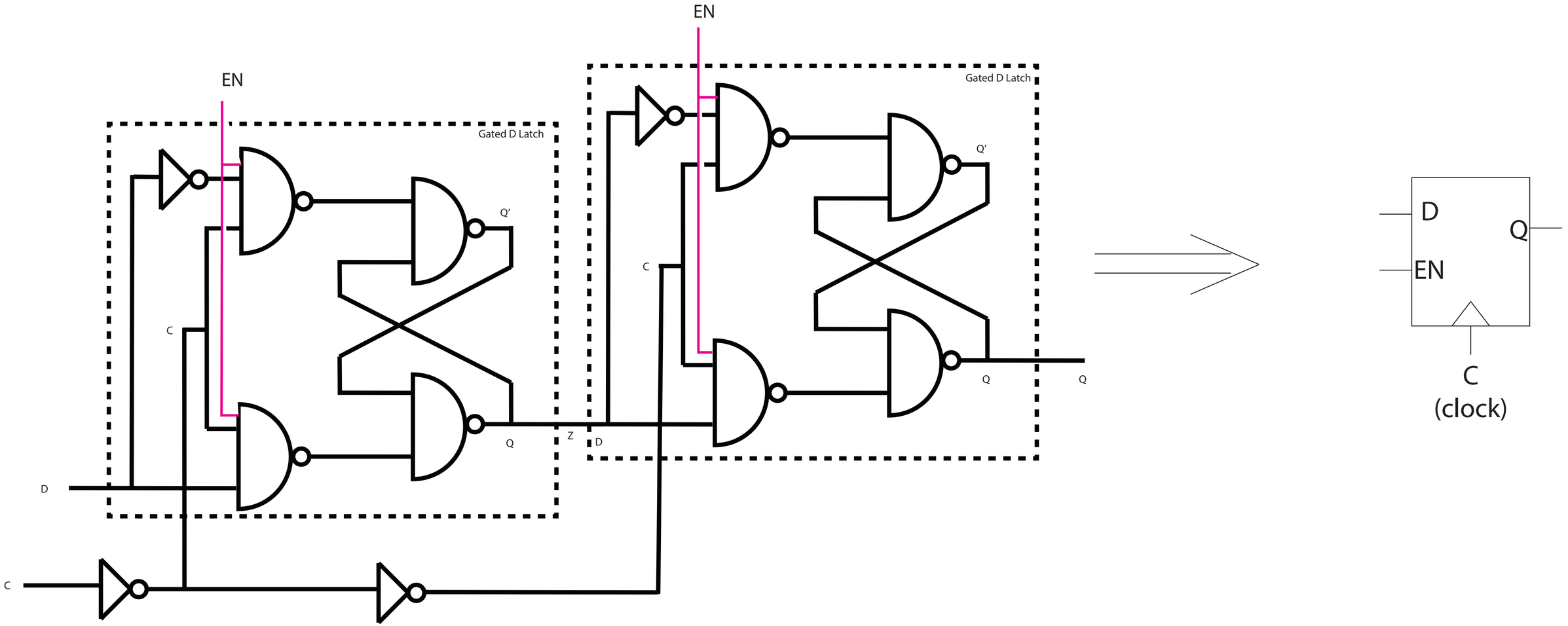
Flip-flop with enable input

- Enable signal==0:
  - D input is disabled
  - Flip-flop preserves value
- Enable signal==1:
  - Allow D input to change Q



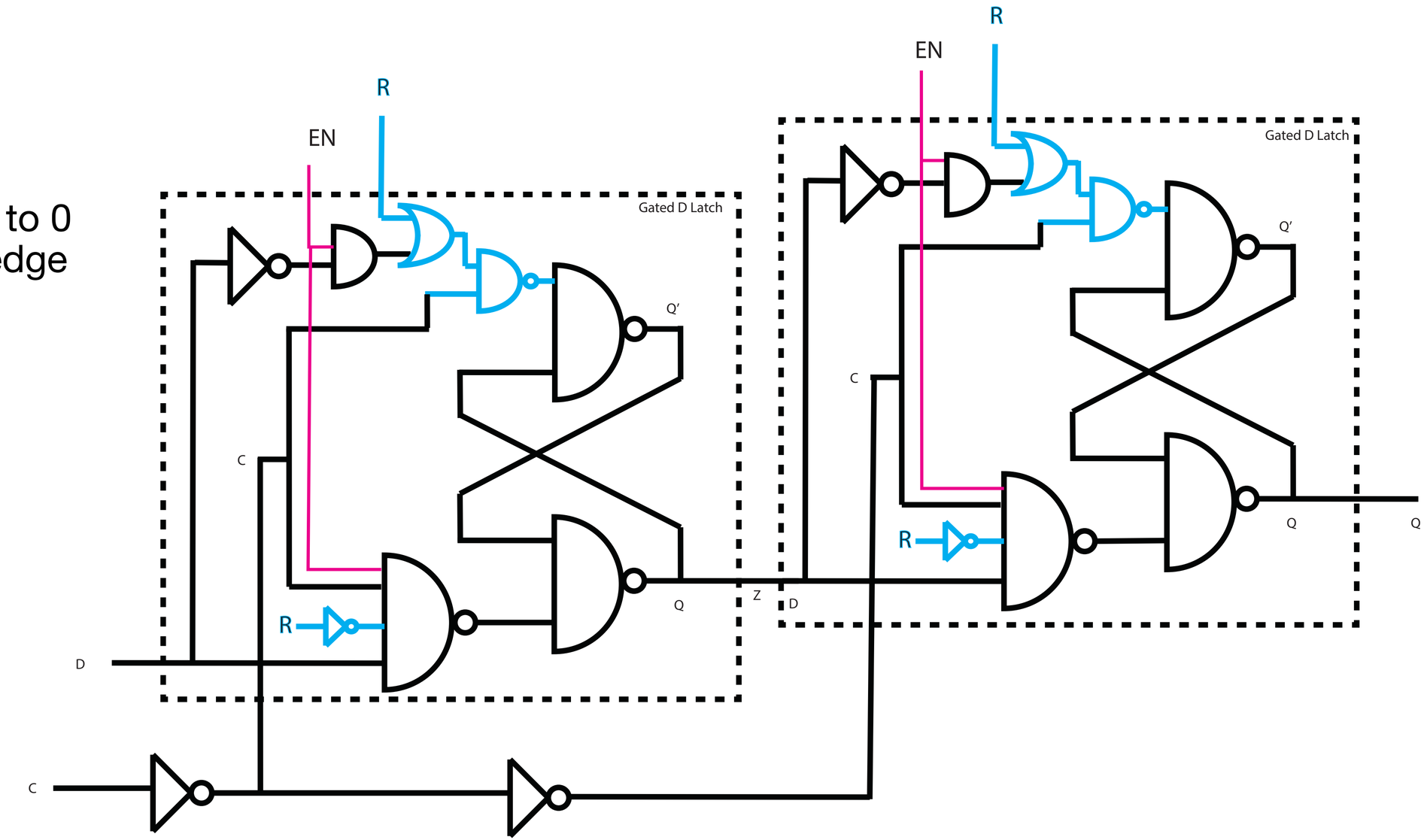
Flip-flop with enable input

Standard (Simplified) Representation



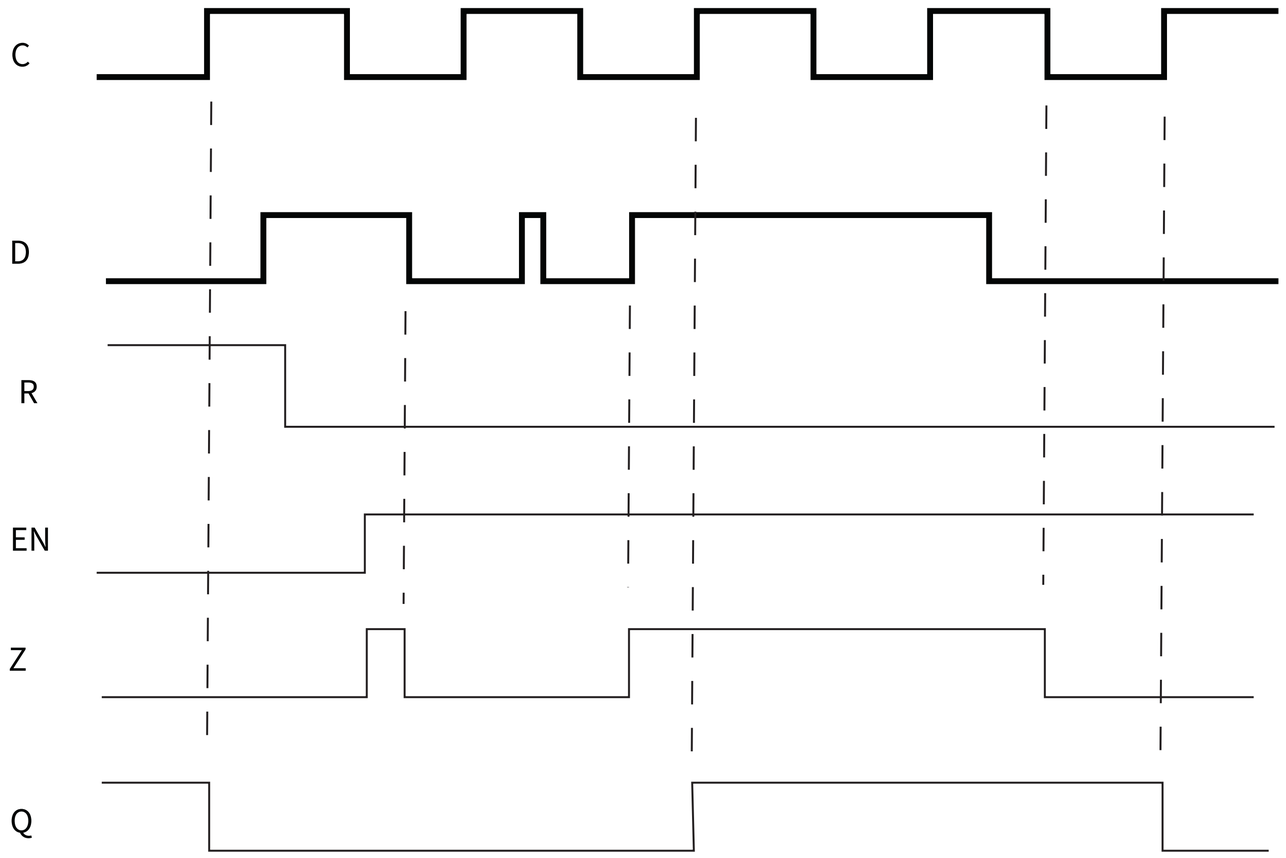
Flip-flop with reset input

- R==0: No impact
- R==1: clear output to 0 at clock (C) rising edge



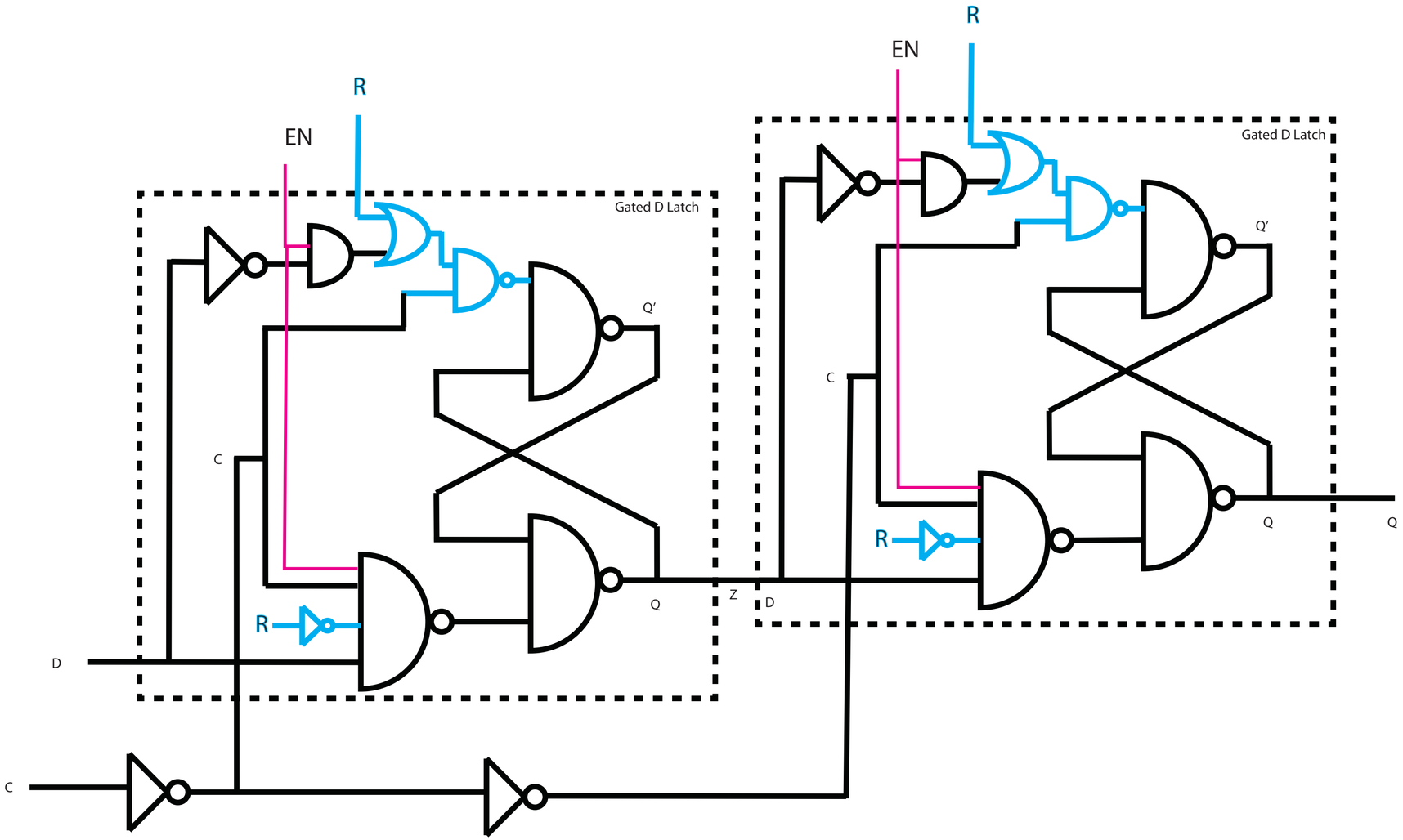
Flip-flop with reset input

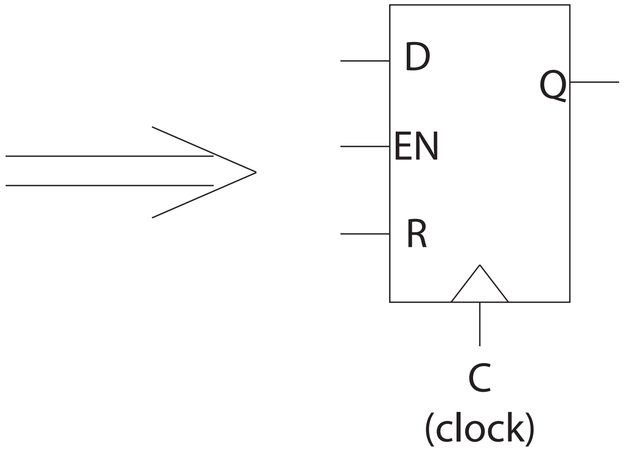
Timing Diagram



Flip-flop with reset input

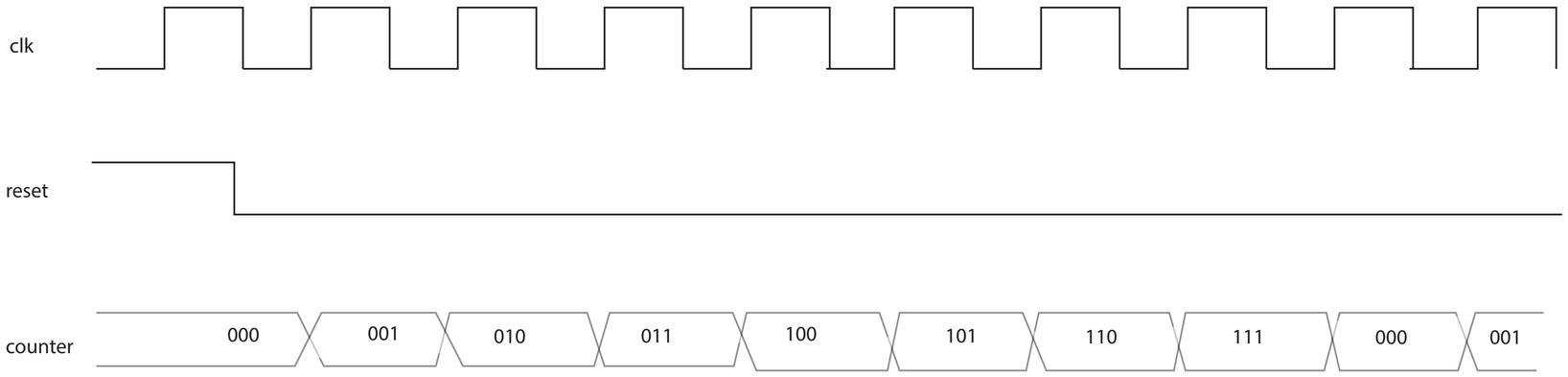
Standard (Simplified) Representation





Counter

- Use flip-flops to increment value every clock cycle
  - Add 1 to current value of counter
  - Store updated value back in counter
- 3-bit counter values (range 0-7):
  - 000
  - 001
  - 010
  - 011
  - 100
  - 101
  - 110
  - 111
  - 000 (Repeat from 0)
  - …



Counter

Truth Table

- Inputs:
  - Counter Old
    - 3-bits
- Outputs:
  - Counter New
    - 3-bits

Qold Bit 2 (Q2)	Qold Bit 1 (Q1)	Qold Bit 0 (Q0)	Qnew Bit 2	Qnew Bit 1	Qnew Bit 0
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	0

Counter

Equations

- Qnew Bit 0 = Q0’
- Qnew Bit 1 = Q1Q0’ + Q1’Q0
- Qnew Bit 2 = Q2’Q1Q0 + Q2Q1’ + Q2Q0’

Counter

Alternate implementation using enable input of flip-flop

- Tie inverted value of flip-flop (Q) to the input (D) of the flip-flop
  - D input of counter bit 2 = inverted value of Q output of counter bit 2
  - D input of counter bit 1 = inverted value of Q output of counter bit 1
  - D input of counter bit 0 = inverted value of Q output of counter bit 0
- Create truth table to see when inversion is needed
  - Enable input of flip-flop controls when the flip-flop is updated

Counter with enable input register

Truth Table

Qold Bit 2 (Q2)	Qnew Bit 2	Invert Bit 2
0	0	0
0	0	0
0	0	0
0	1	1
1	1	0
1	1	0
1	1	0
1	0	1

Qold Bit 1 (Q1)	Qnew Bit 1	Invert Bit 1
0	0	0
0	1	1
1	1	0
1	0	1
0	0	0
0	1	1
1	1	0
1	0	1

Qold Bit 0 (Q0)	Qnew Bit 0	Invert Bit 0
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1

Counter with enable input register

Equation

- Enable input for counter bit 0: 1
  - Bit 0 changes every clock cycle
- Enable input for counter bit 1: Q0
  - Bit 1 changes every other clock cycle
    - Q0 == 1 (no update when Q0 is 0)
- Enable input for counter bit 2: Q1Q0
  - Bit 2 changes every 4th clock cycle
    - Q1Q0 == 11 (no update when Q1Q0 is 00, 01, 10)

Counter with enable input register