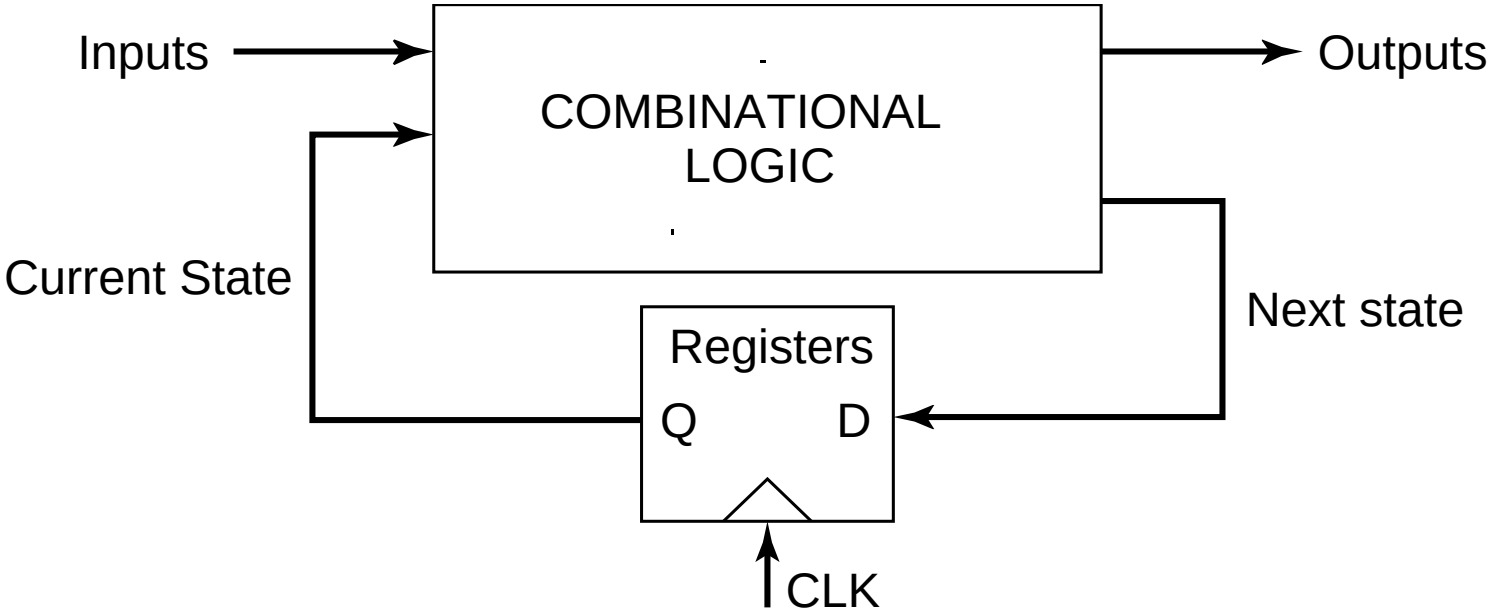
EE 466/586
VLSI Design

Partha Pande
School of EECS
Washington State University
pande@eecs.wsu.edu

Lecture 16

Sequential Logic

Sequential Logic



2 storage mechanisms

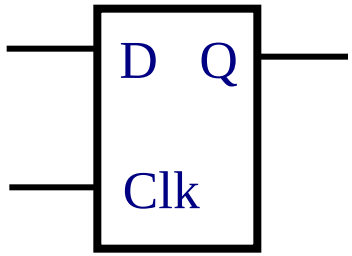
- **positive feedback**
- **charge-based**

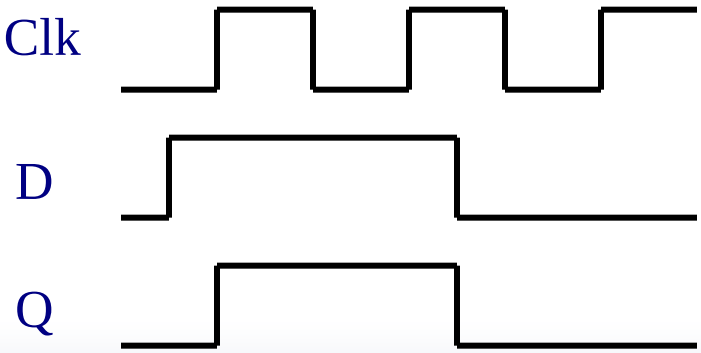
A latch is level sensitive

A register is edge-triggered

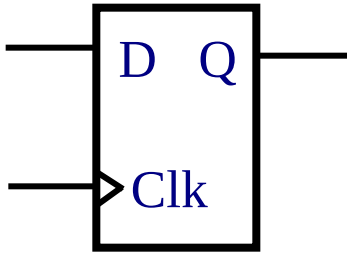
Latch versus Register

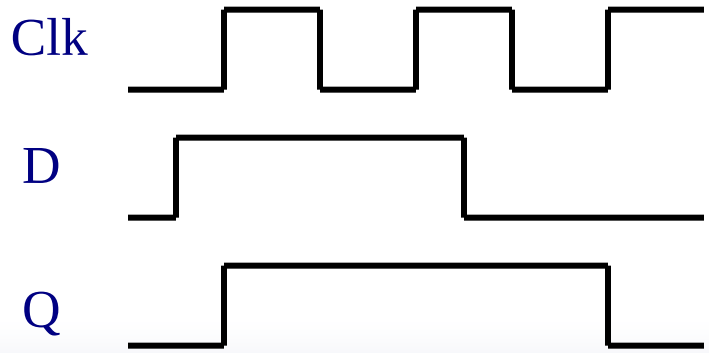
- Latch Stores data depending on the level of the clock



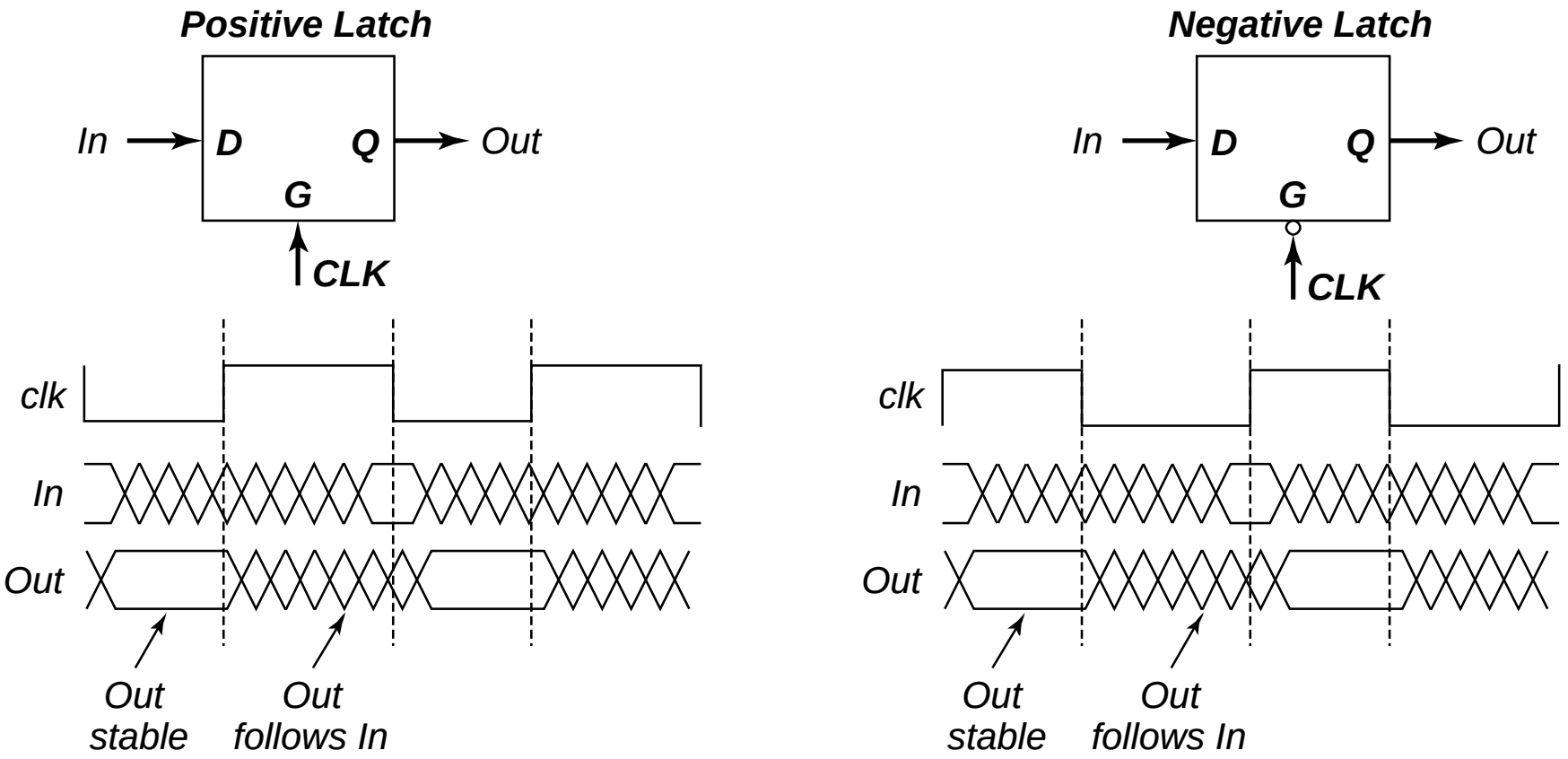


- Register stores data when clock rises





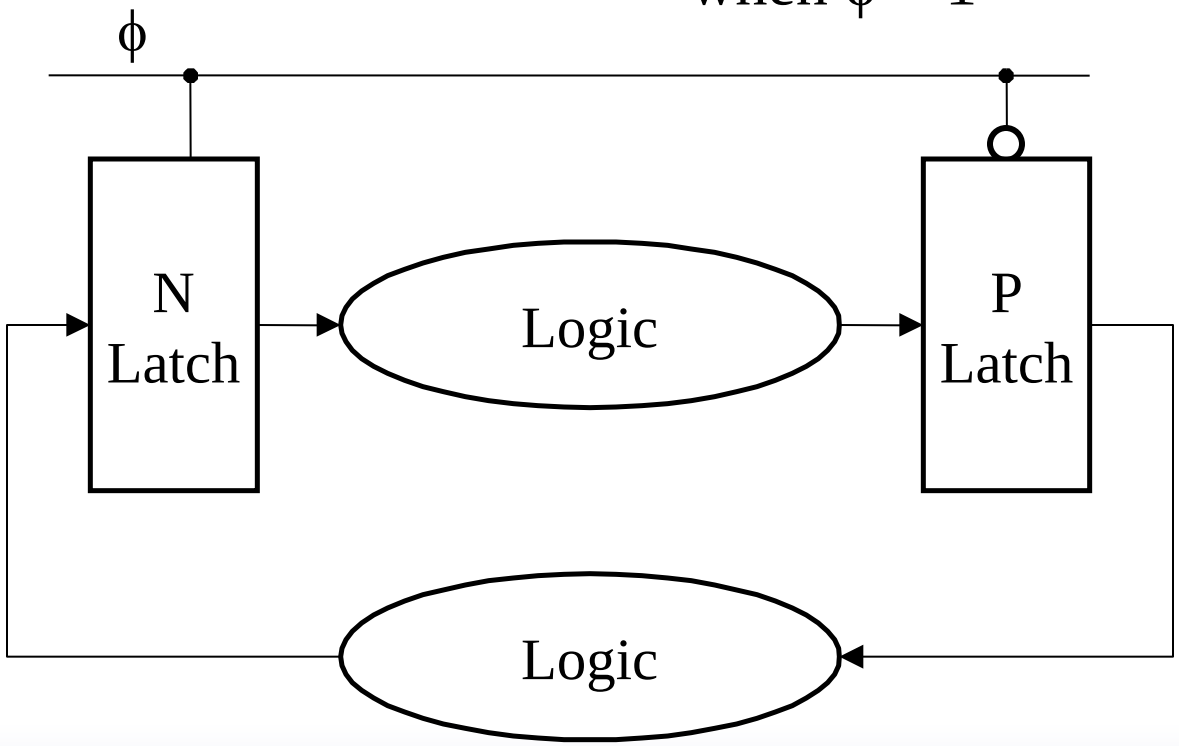
Latches


Positive Latch
In
D
Q
Out
G
CLK
clk
In
Out
Out stable
Out follows In
Negative Latch
In
D
Q
Out
G
CLK
clk
In
Out
Out stable
Out follows In

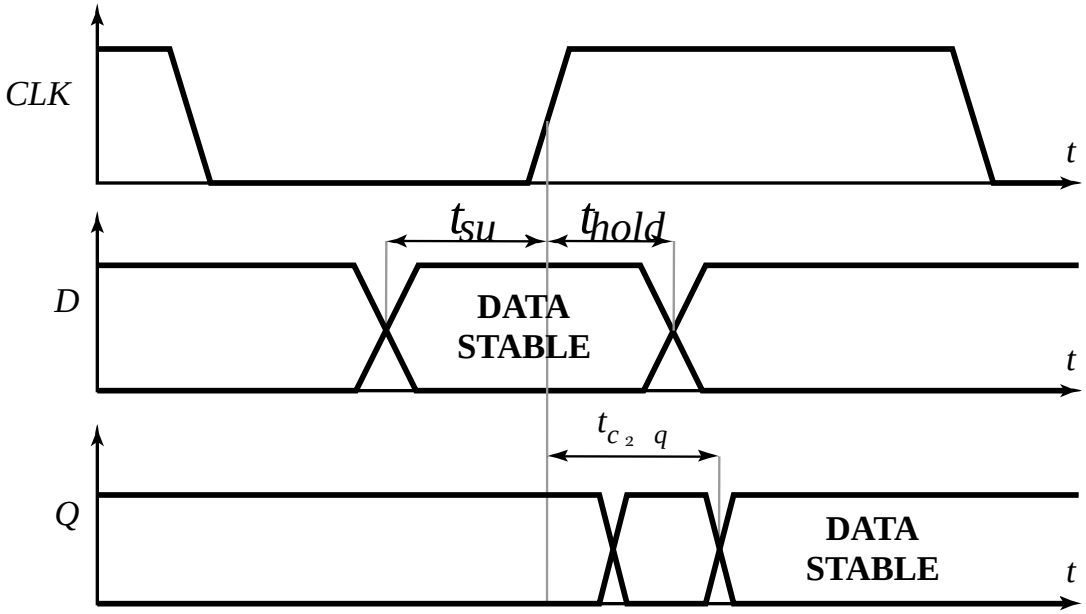
Latch-Based Design

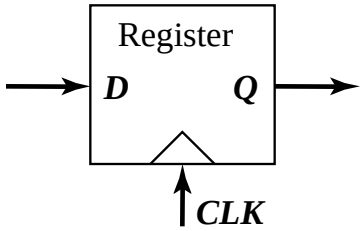
• N latch is transparent when $\phi = 0$

• P latch is transparent when $\phi = 1$



Timing Definitions

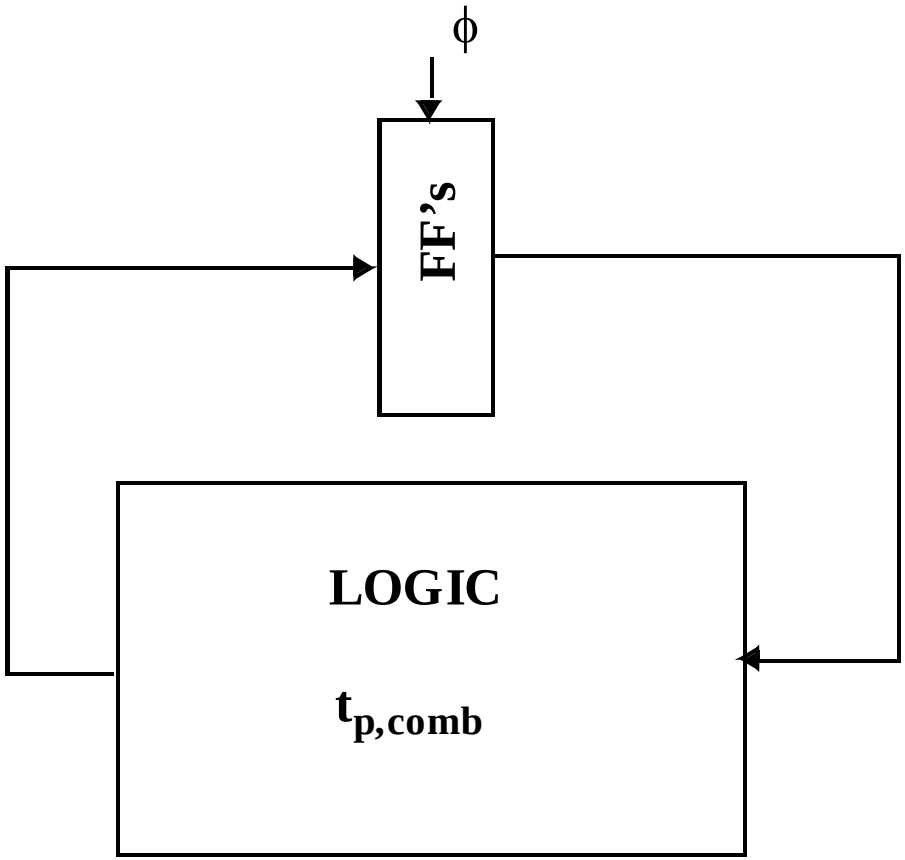




Timing Definitions

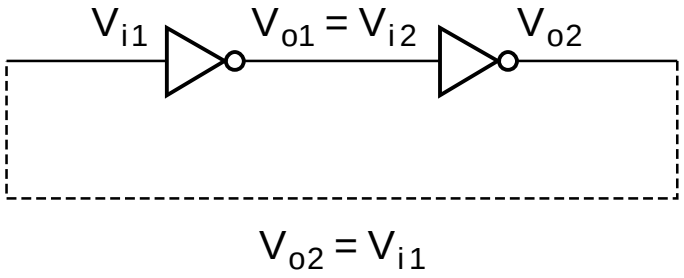
- T_{su} (Setup time)
  - Incoming data must be stable before the clock arrives
- T_{hold} (Hold time)
  - The length of time the data remains stable after the clock arrives for proper operation
- If the data is stable before the setup time and continues to be stable after the hold time, the register will properly capture the data
- $T_{clk\text{-}Q}$ (clk to Q delay)
  - This is the delay from the time the clock arrives to the point at which the Q output stabilizes

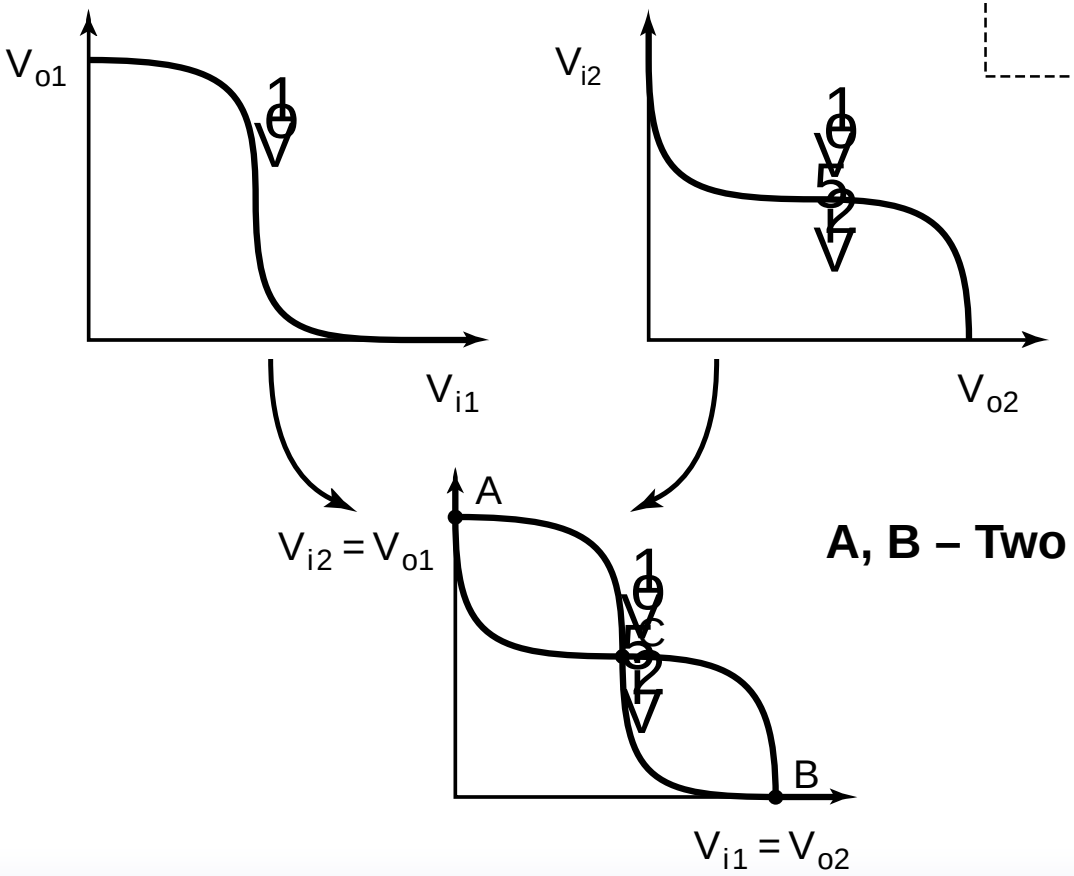
Maximum Clock Frequency



$$t_{clk\text{-}Q} + t_{p,comb} + t_{setup} = T$$

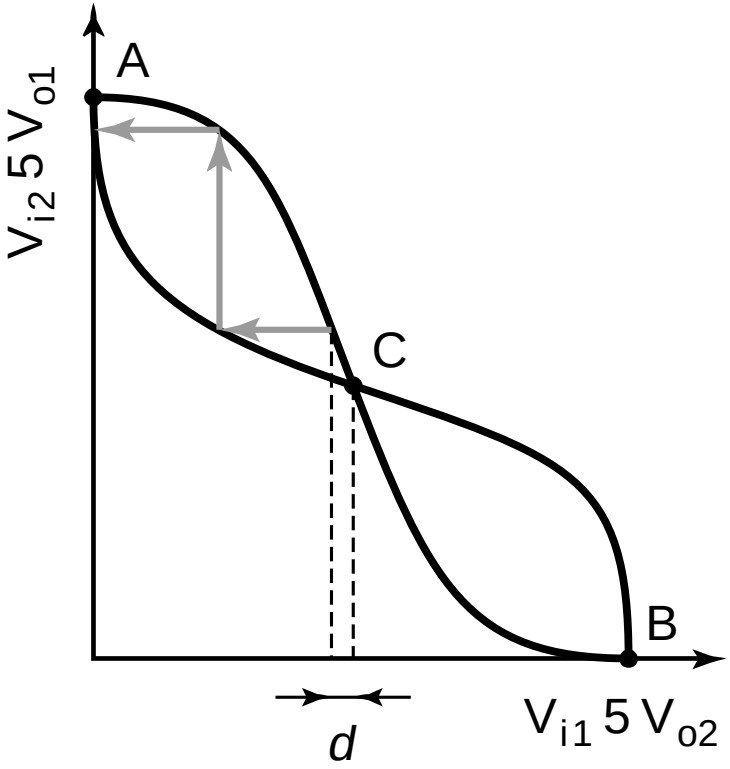
Positive Feedback: Bi-Stability

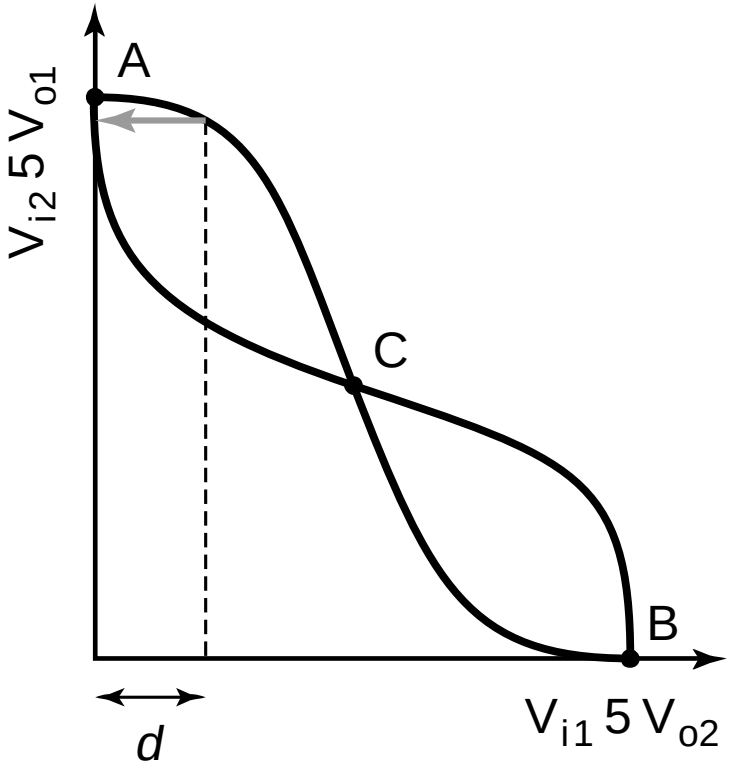




A, B – Two stable operating points

Meta-Stability

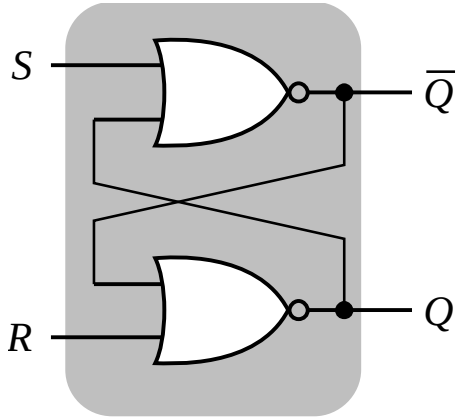


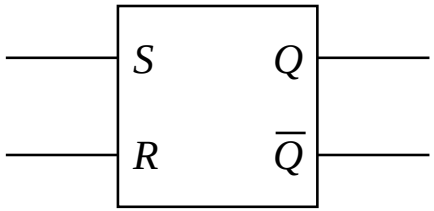


Gain should be larger than 1 in the transition region

Cross-Coupled Pairs

NOR-based set-reset



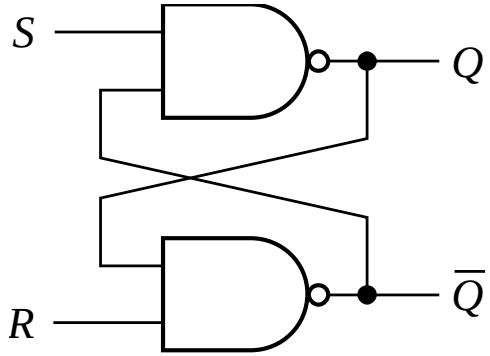


S	R	Q	$\overline{Q}$
0	0	Q	$\overline{Q}$
1	0	1	0
0	1	0	1
1	1	0	0

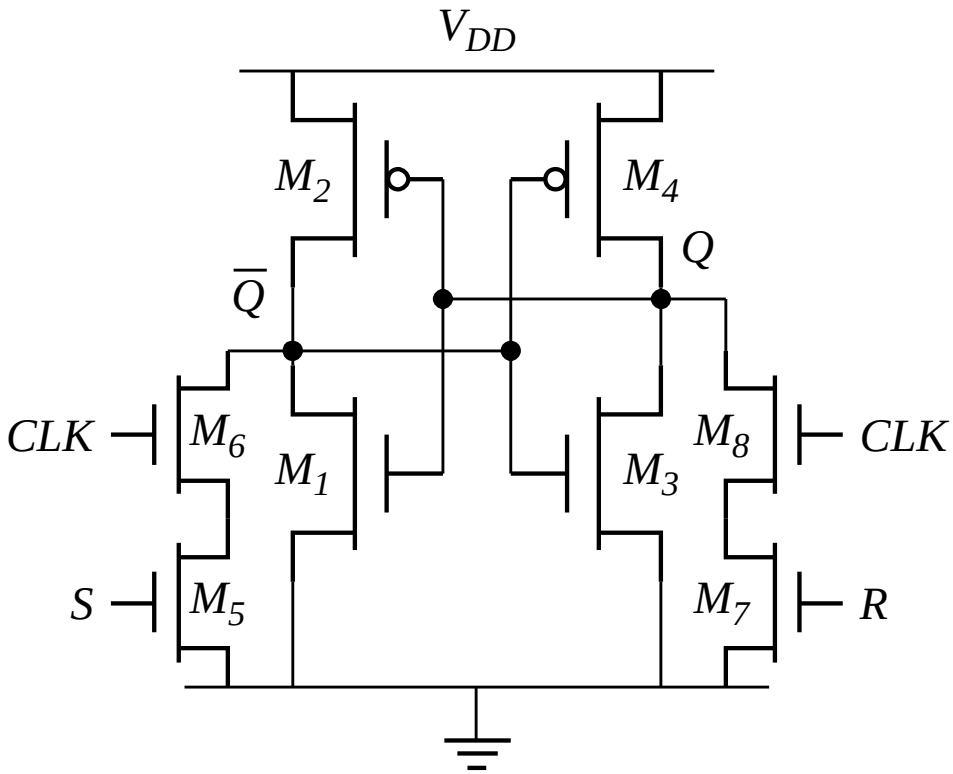
Forbidden State

Cross-Coupled NAND

Cross-coupled NANDs



Added clock



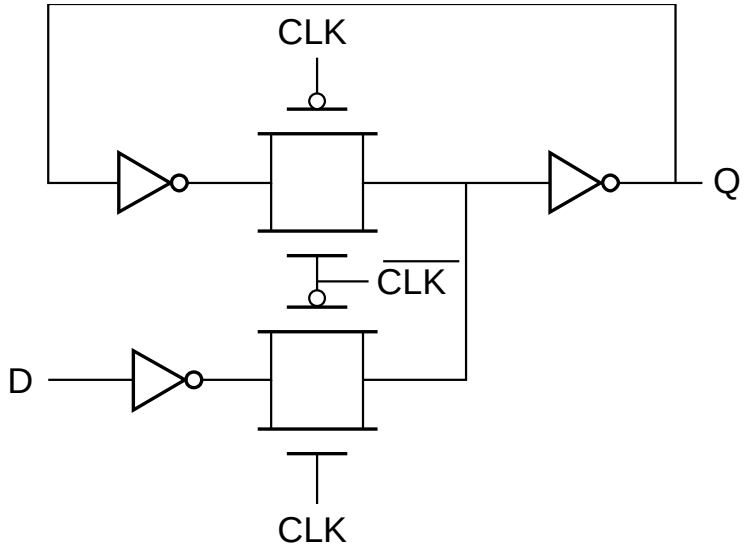
This is not used in datapaths any more, but is a basic building memory cell

Design of D-Latch

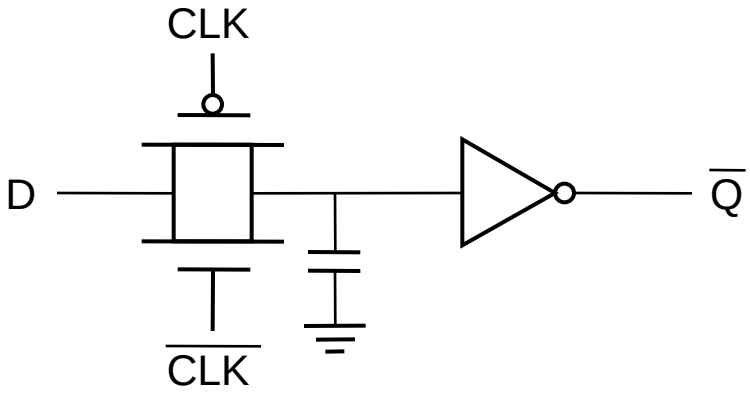
- Follow board notes

Storage Mechanisms

Static

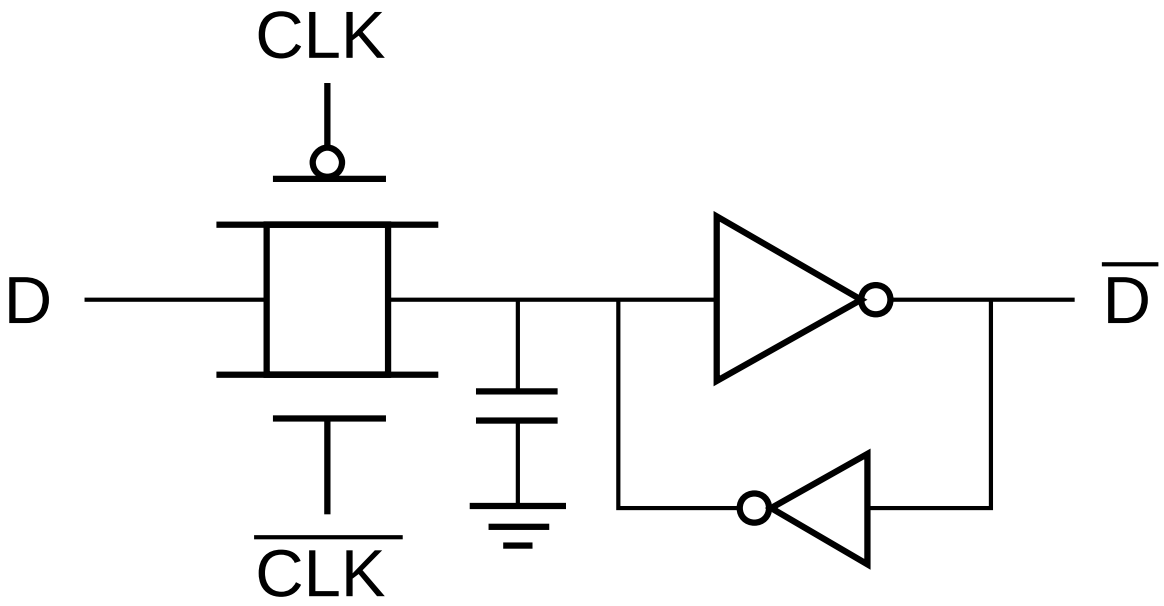


Dynamic (charge-based)



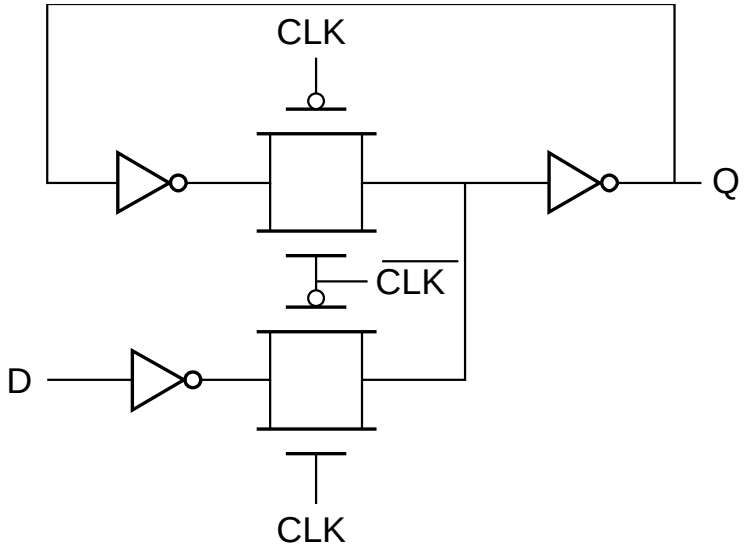
Need a feedback loop to hold the data

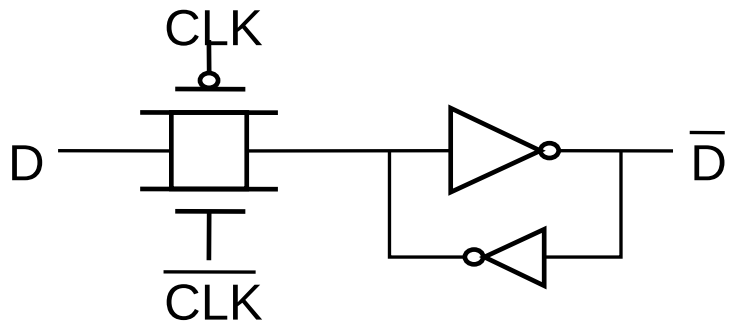
Making a Dynamic Latch Pseudo-Static



Writing into a Static Latch

Use the clock as a decoupling signal,
that distinguishes between the transparent and opaque states

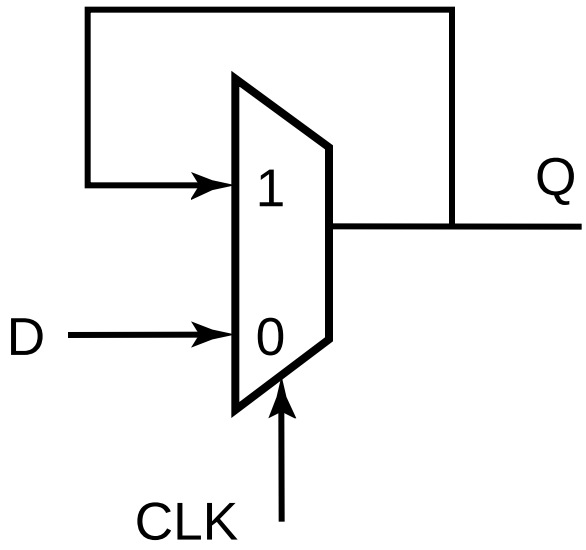




Converting into a MUX

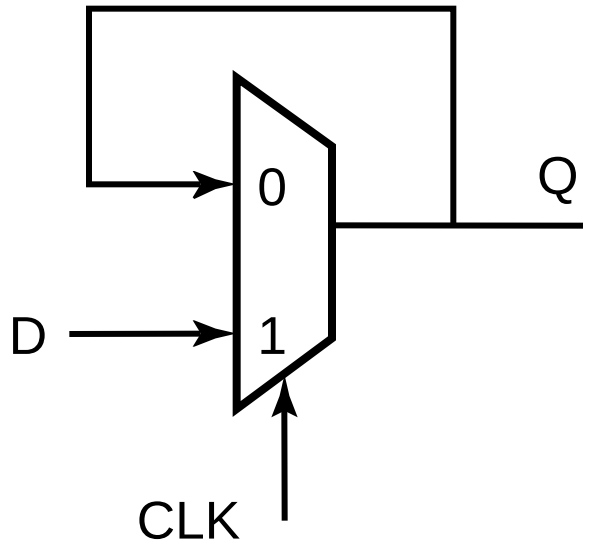
Mux-Based Latches

Negative latch
(transparent when CLK= 0)



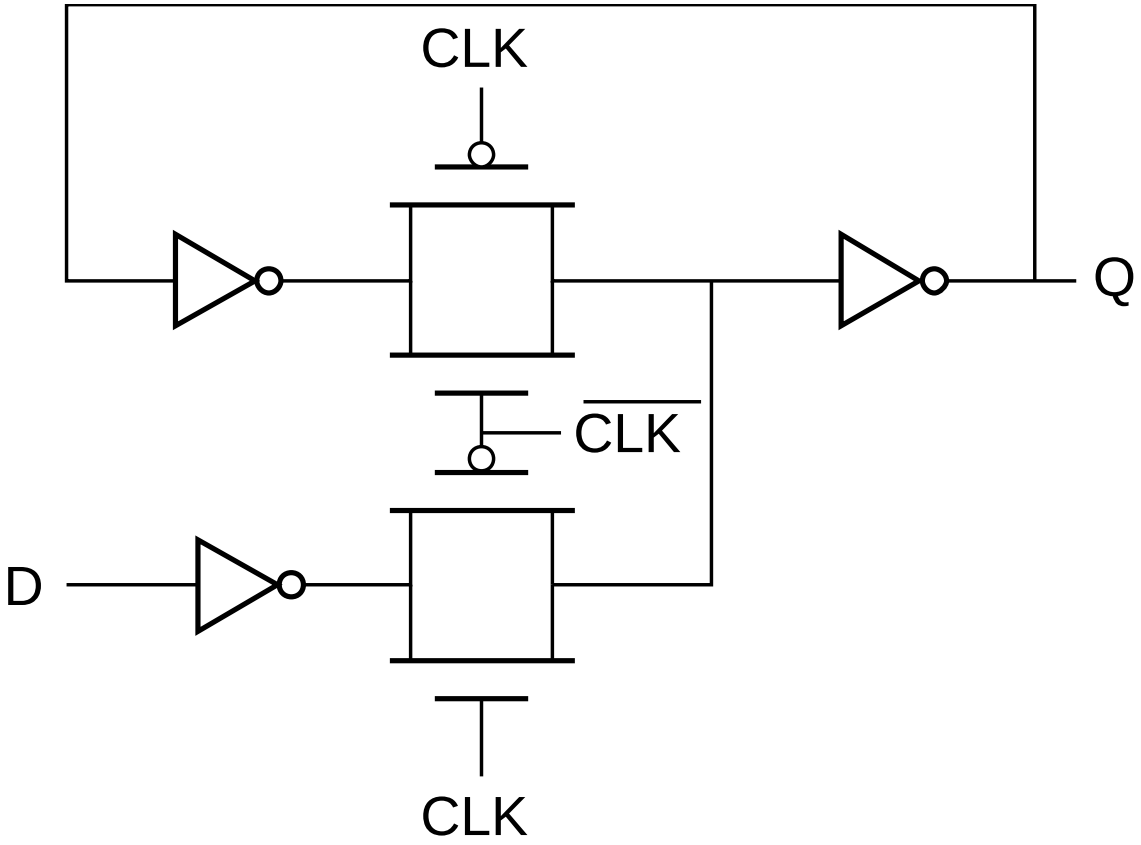
$$Q = \overline{Clk} \cdot Q + Clk \cdot In$$

Positive latch
(transparent when CLK= 1)

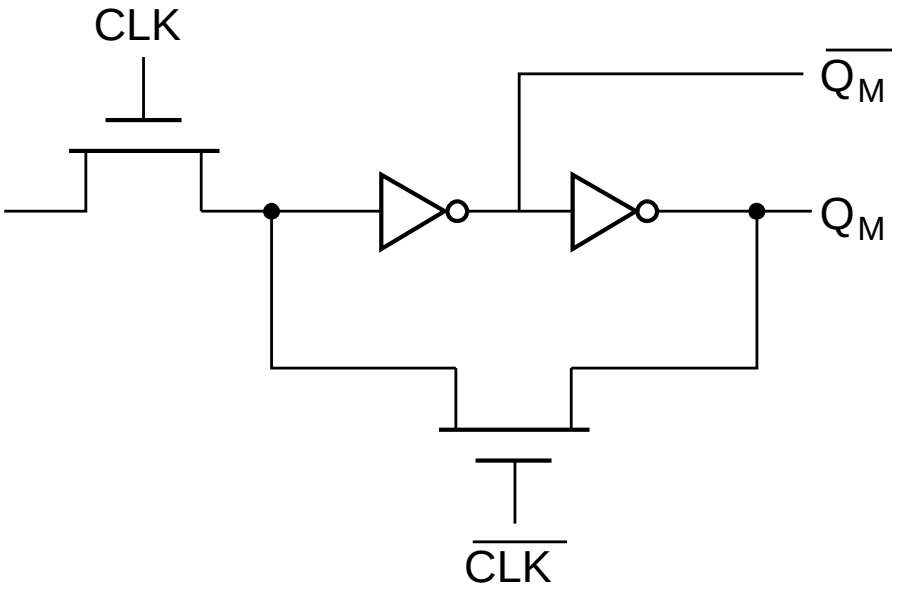


$$Q = Clk \cdot Q + \overline{Clk} \cdot In$$

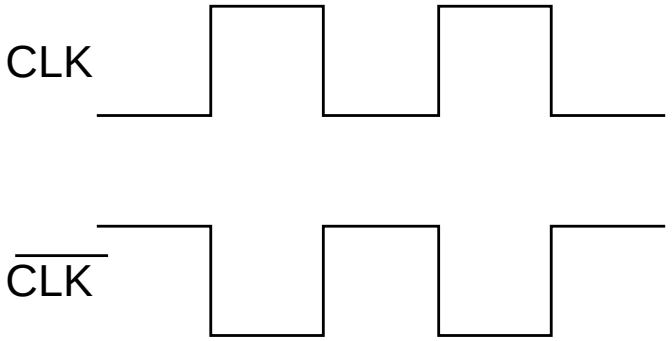
Mux-Based Latch



Mux-Based Latch

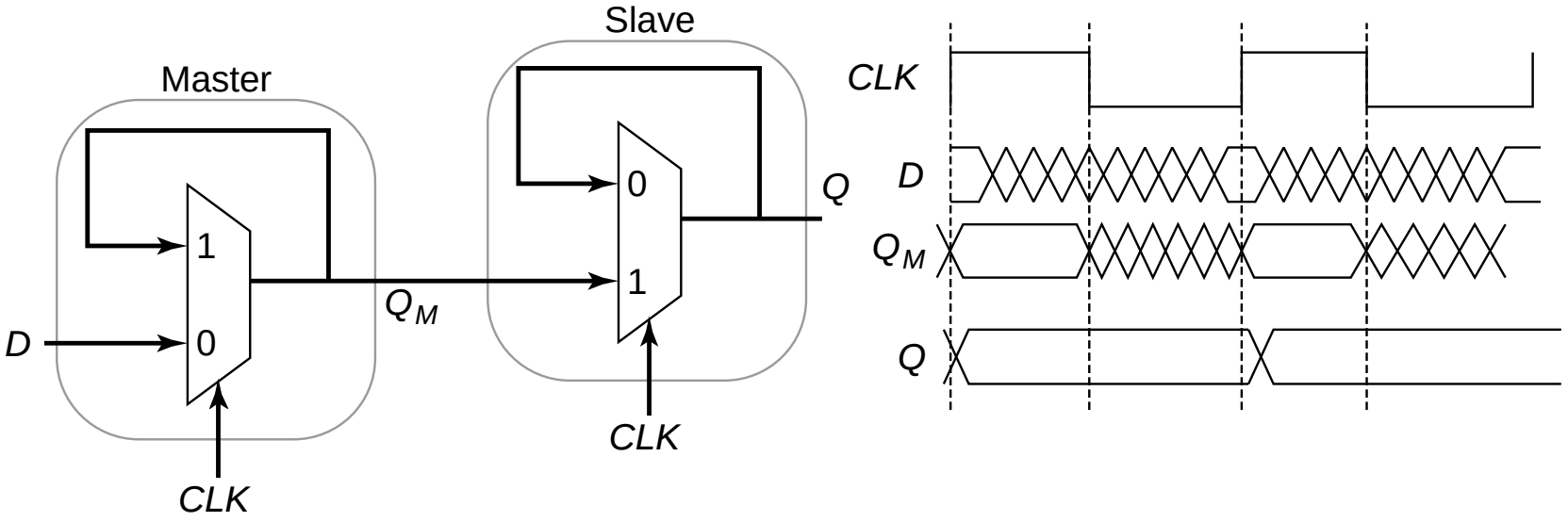


NMOS only



Non-overlapping clocks

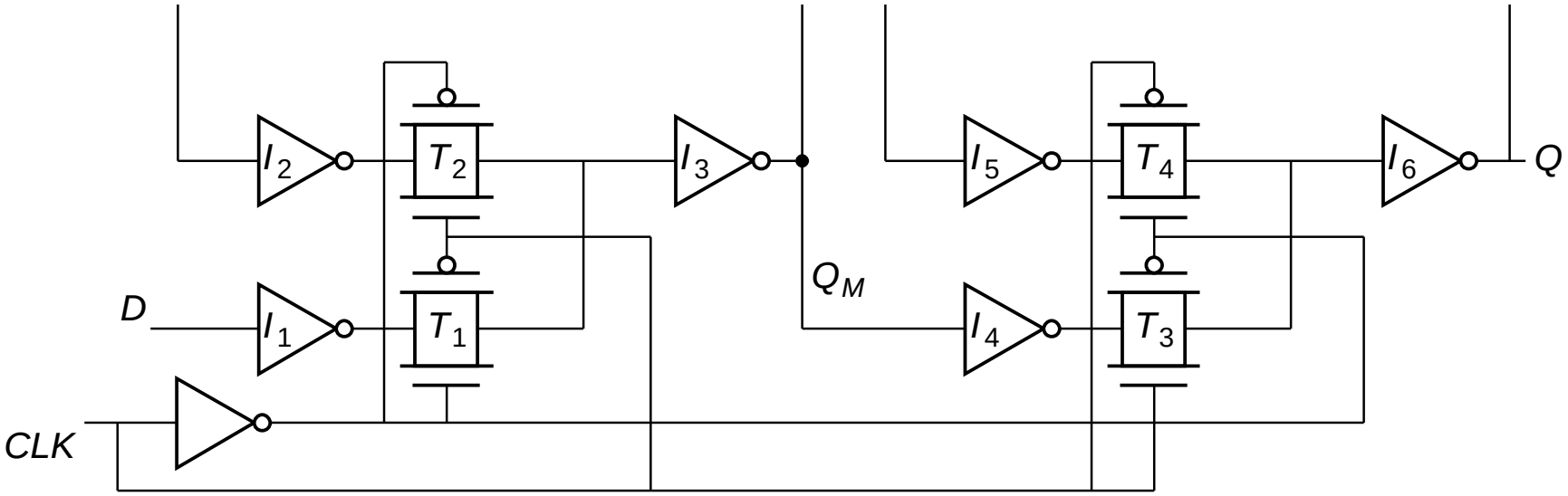
Master-Slave (Edge-Triggered) Register



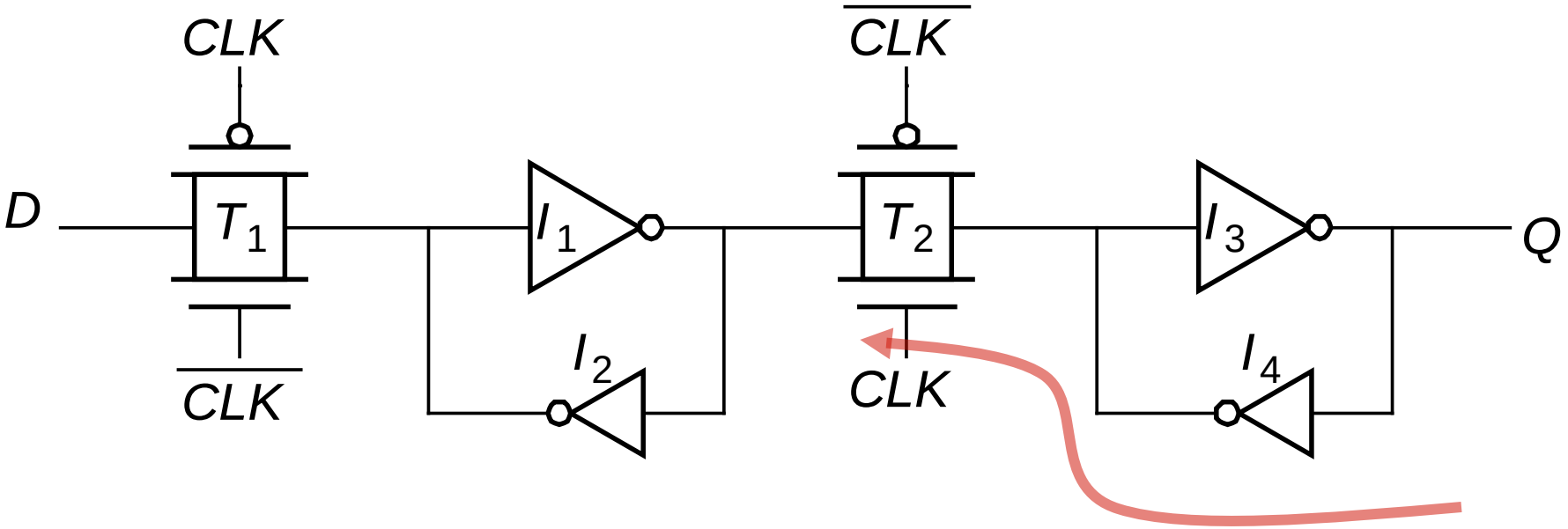
Two opposite latches trigger on edge
Also called master-slave latch pair

Master-Slave Register

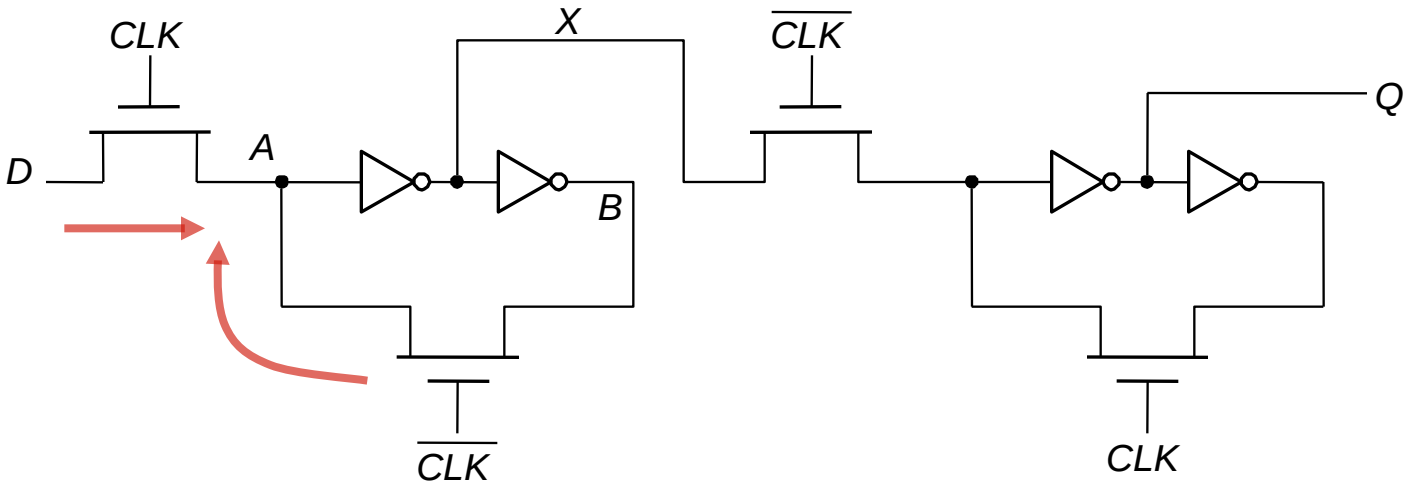
Multiplexer-based latch pair



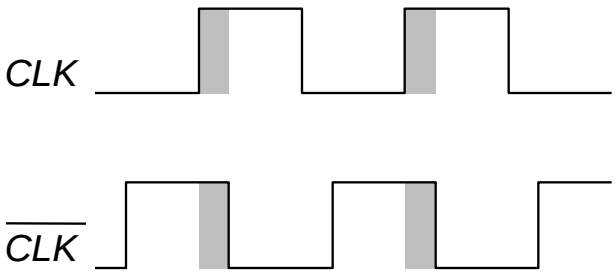
Reduced Clock Load Master-Slave Register



Avoiding Clock Overlap



(a) Schematic diagram



(b) Overlapping clock pairs