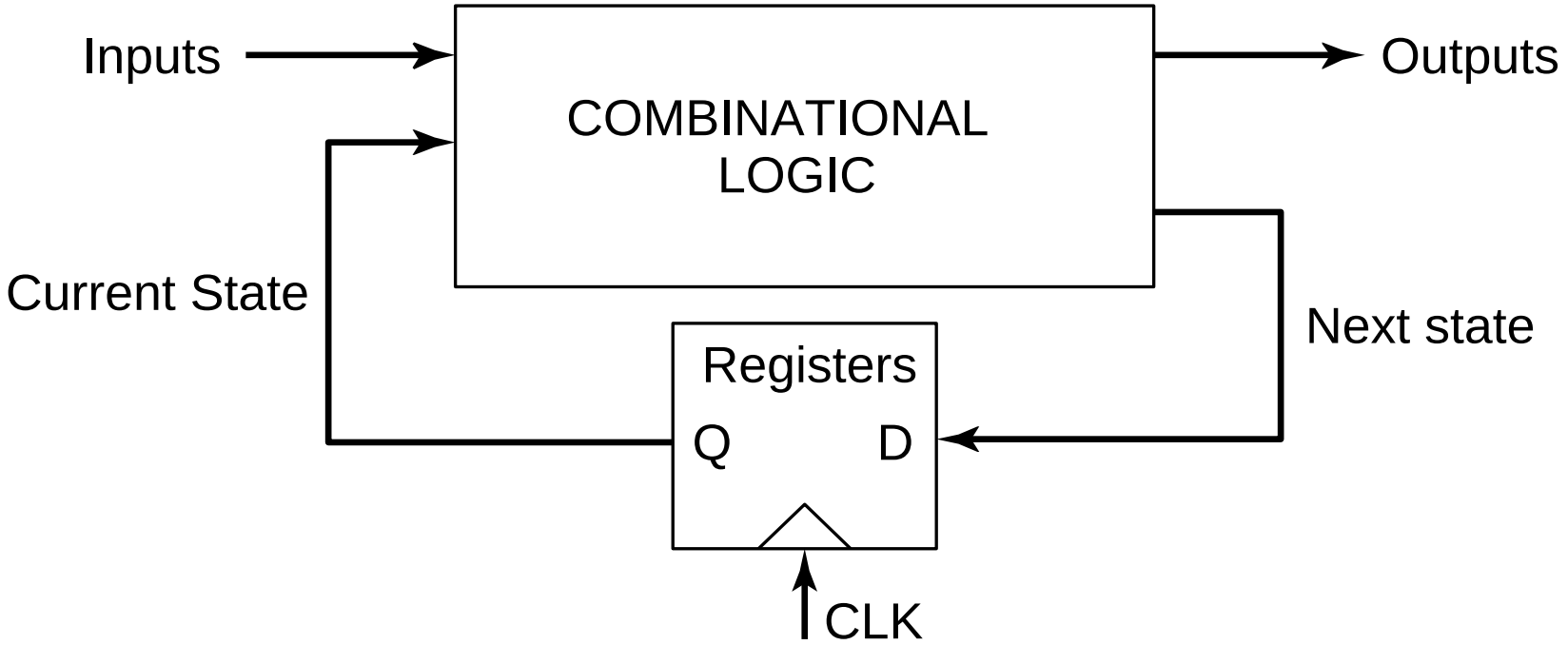
EE115C – Winter 2017
Digital Electronic Circuits

Lecture 17:
Latches and Flip-Flops

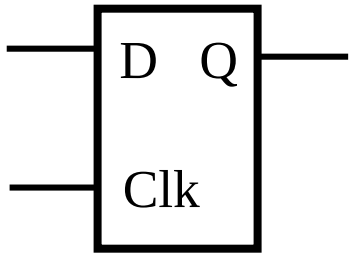
Sequential Logic

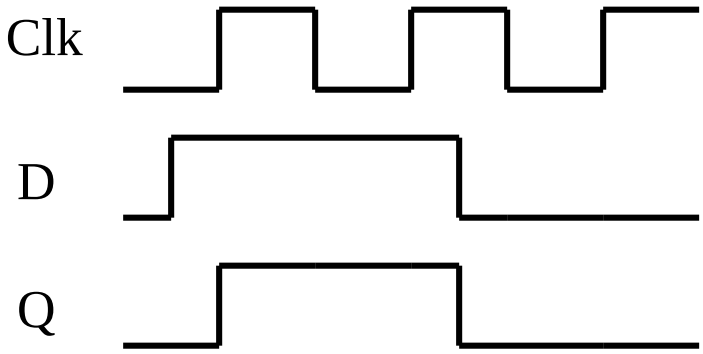


- **2 storage mechanisms**
  - Positive feedback
  - Charge based

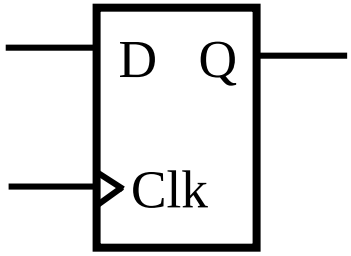
Latch versus Flip-Flop

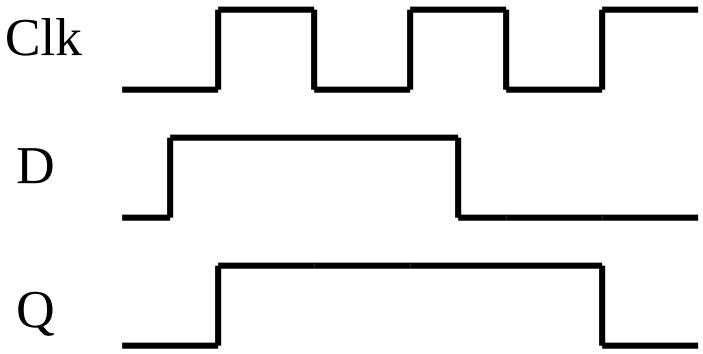
- **Latch: level-sensitive**
  clock is low – hold mode
  clock is high – transparent





- **Flip-flop: edge-triggered**
  stores data when
  clock rises





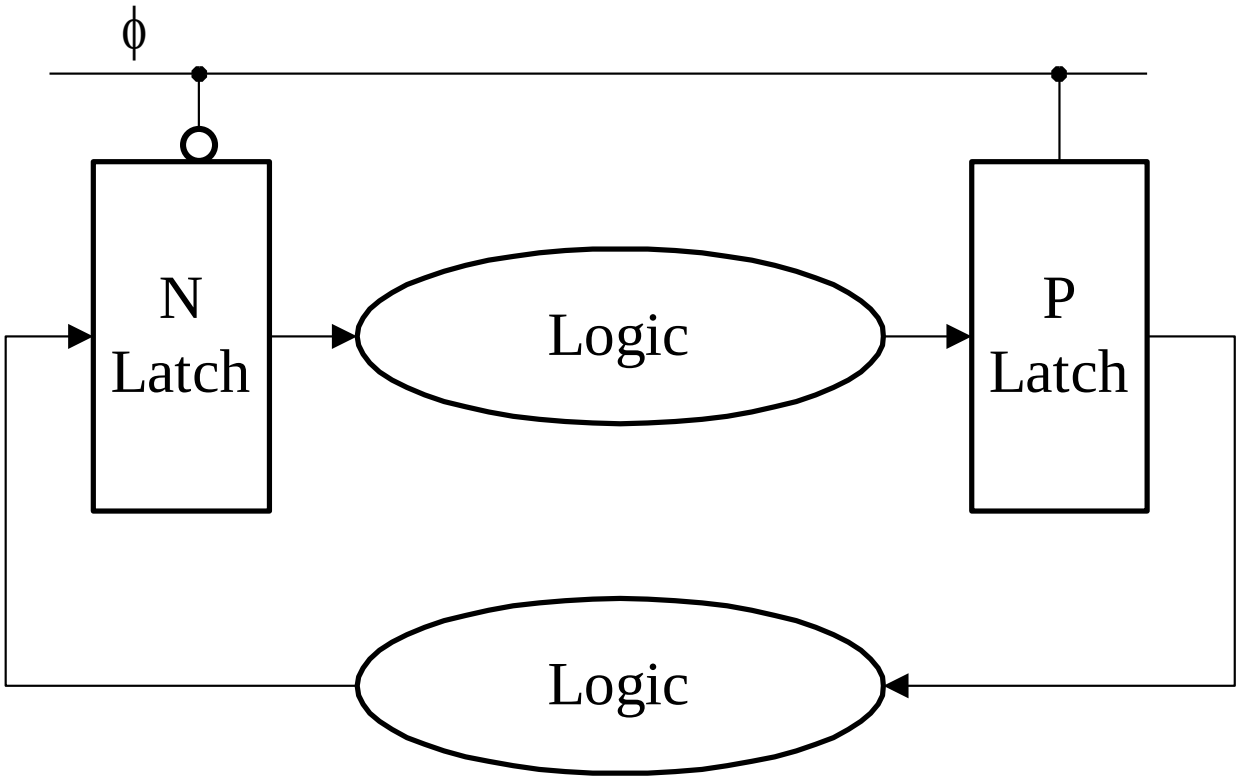
Naming Convention

- **In this class:**
  - Latch is level sensitive
  - Flip-flop is edge-triggered

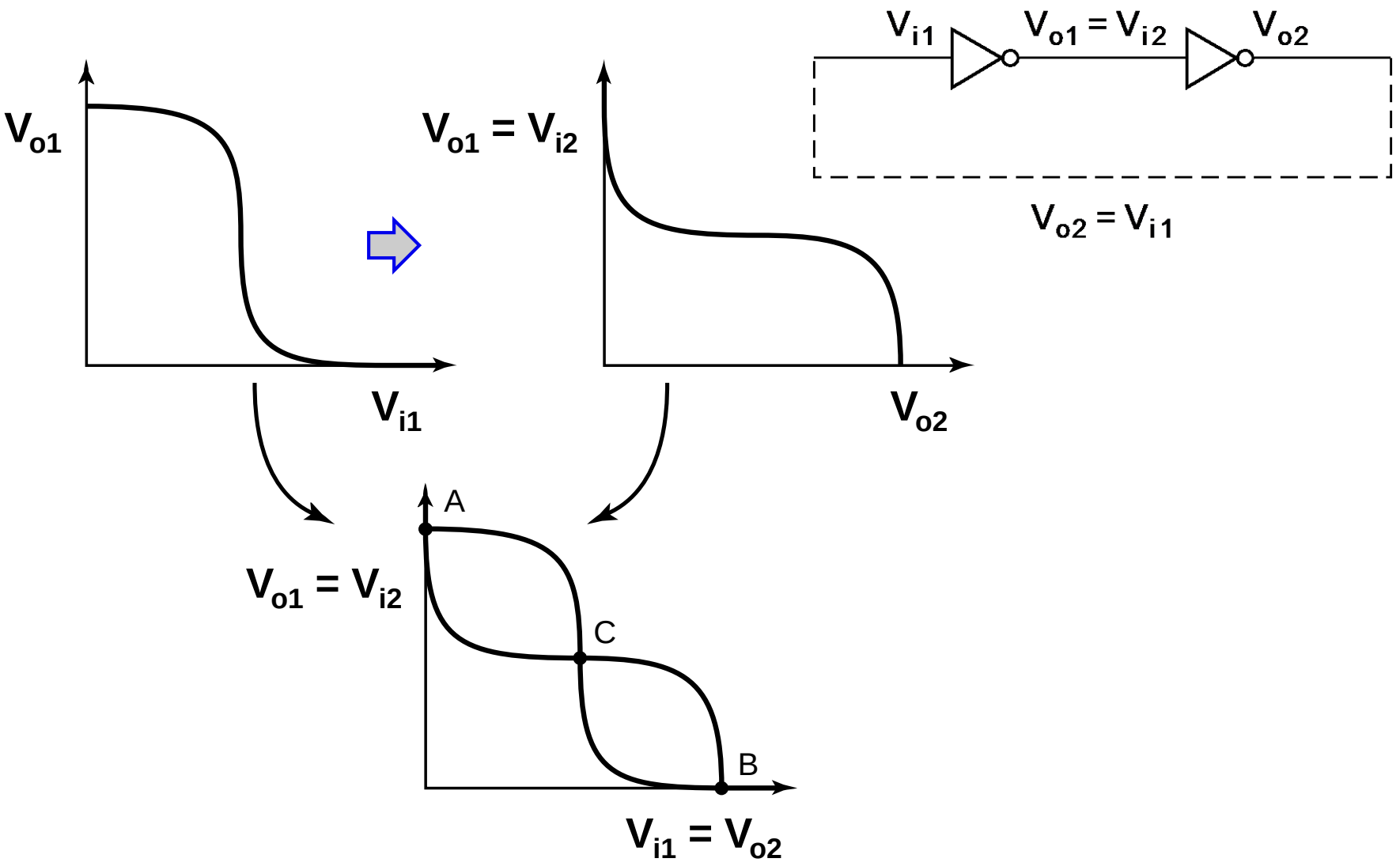
- **There are many different naming conventions**

Latch-Based Design

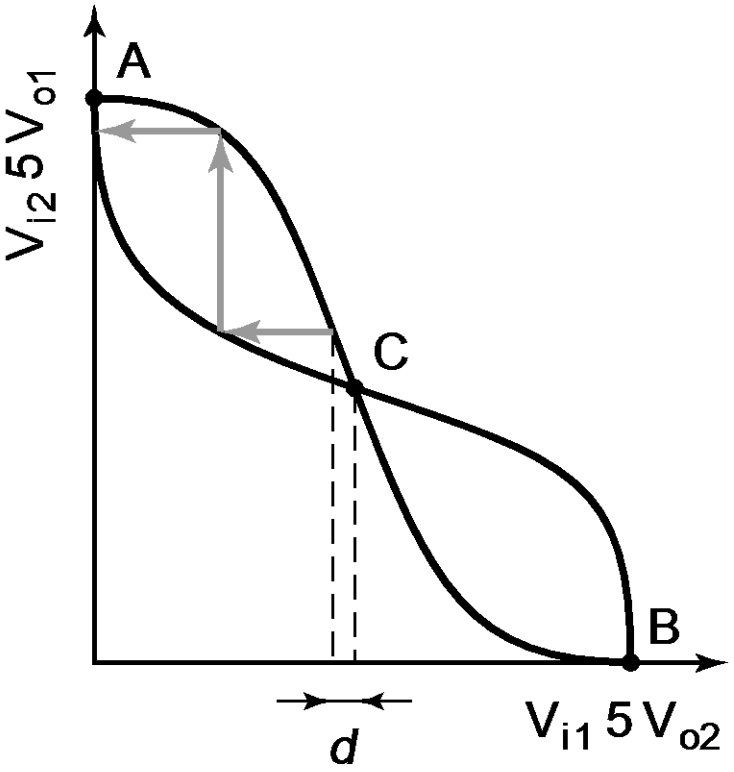
- N latch is transparent when Φ = 0
- P latch is transparent when Φ = 1

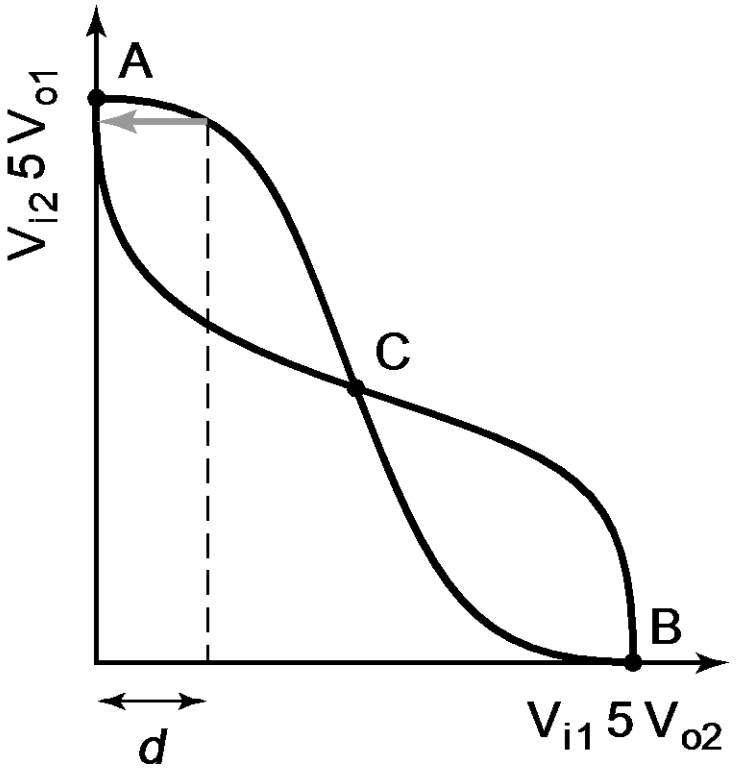


Positive Feedback: Bi-Stability



Meta-Stability

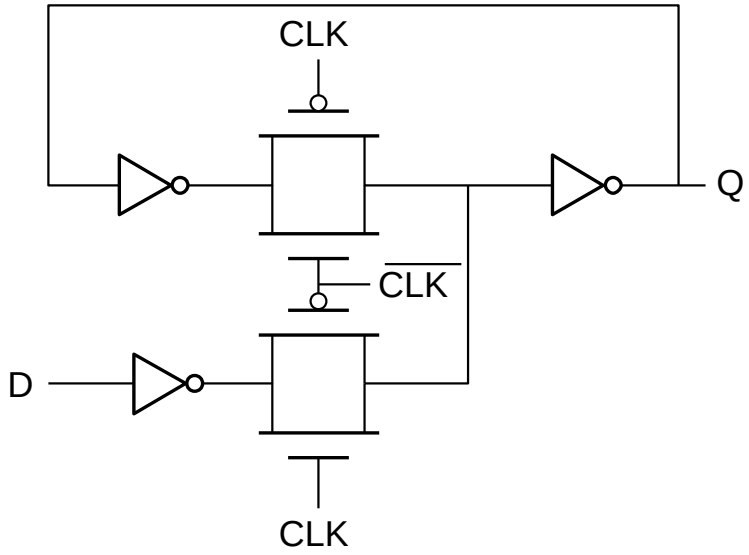


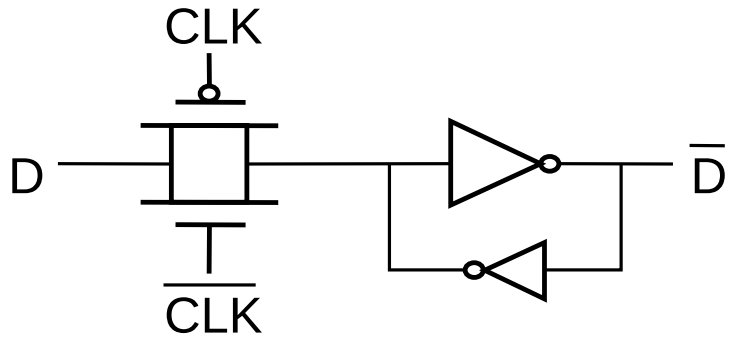


Gain should be larger than 1 in the transition region

Writing into a Static Latch

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



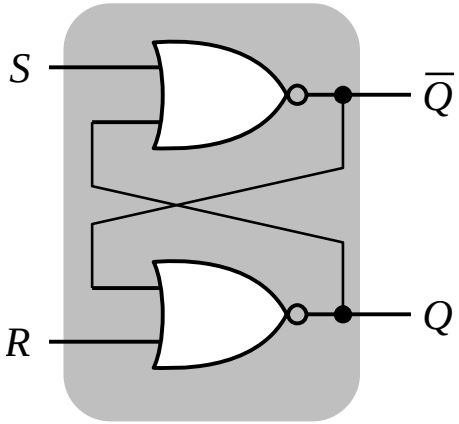


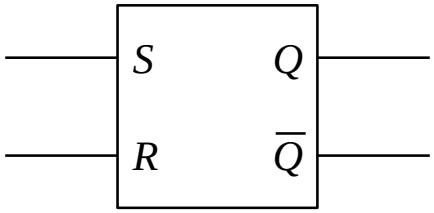
1 Converting into a MUX

2 Forcing the state
(can implement as NMOS-only)

Cross-Coupled Pairs

NOR-based set-reset





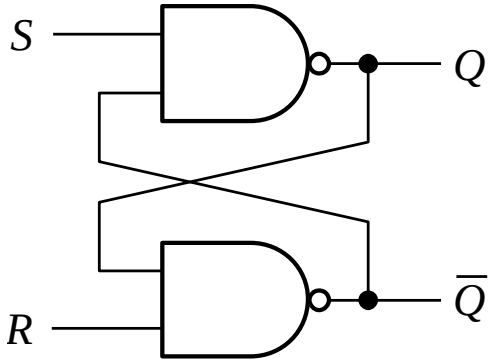
S	R	Q	$\overline{Q}$
0	0	Q	$\overline{Q}$
1	0	1	0
0	1	0	1
1	1	0	0

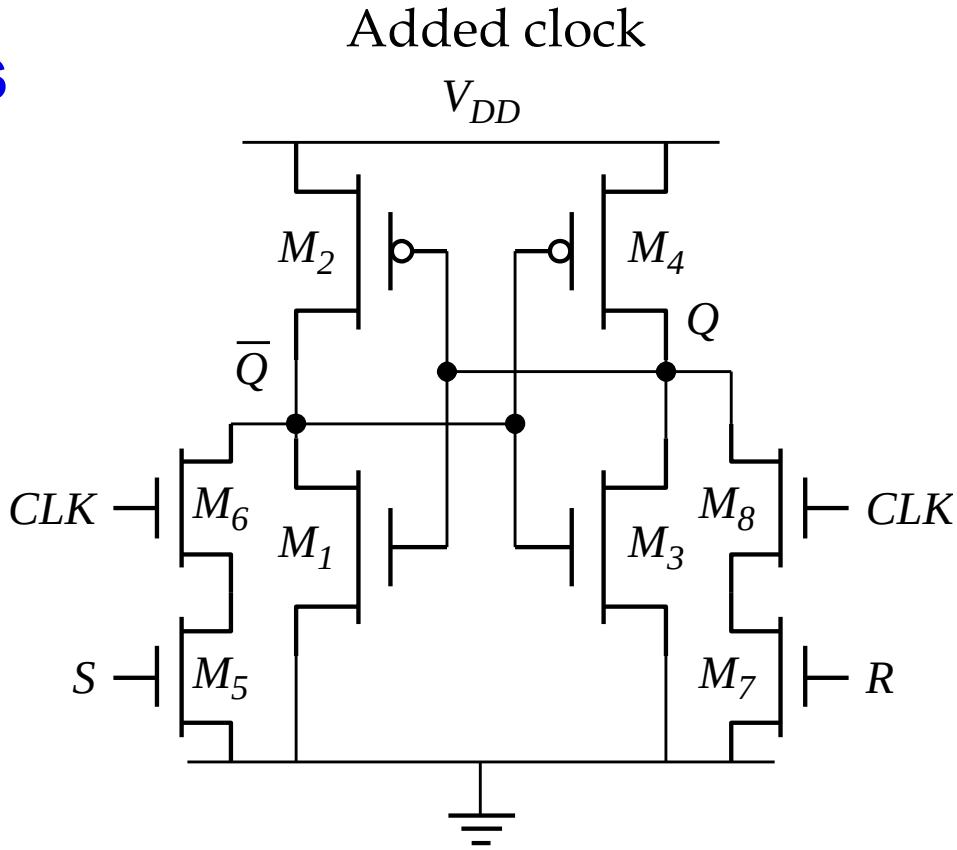
Forbidden State

The "Overpowering" Approach

Cross-Coupled NAND

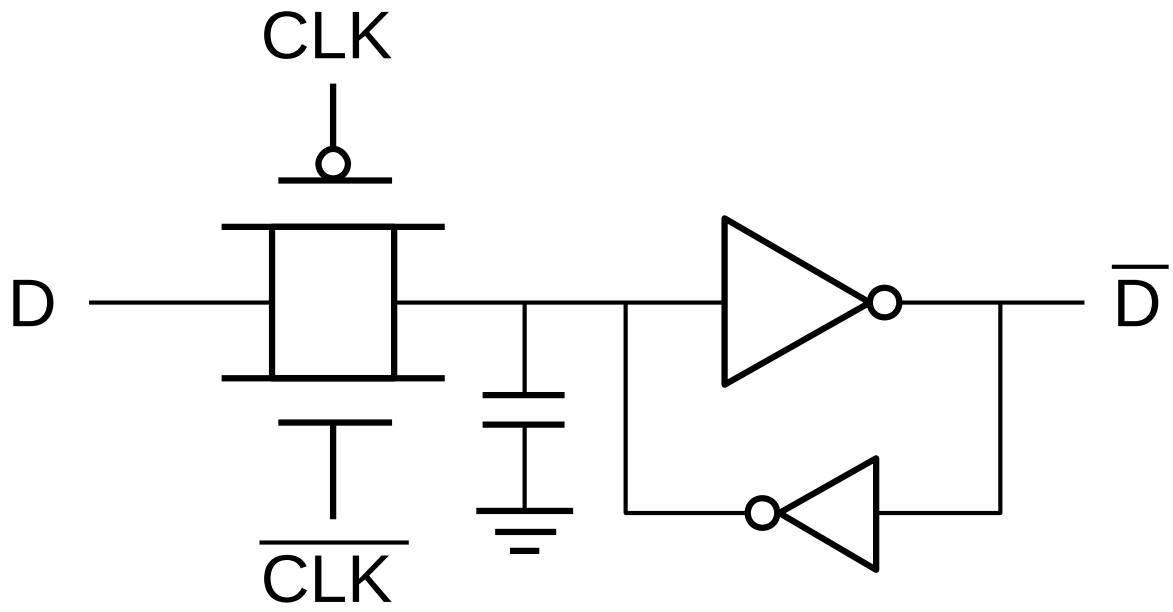
Cross-coupled NANDs





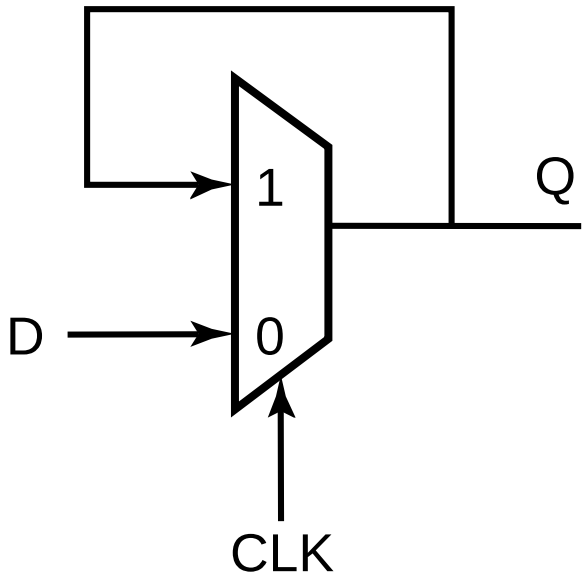
This is not used in datapaths any more,
but is a basic building memory cell

Pseudo-Static Latch



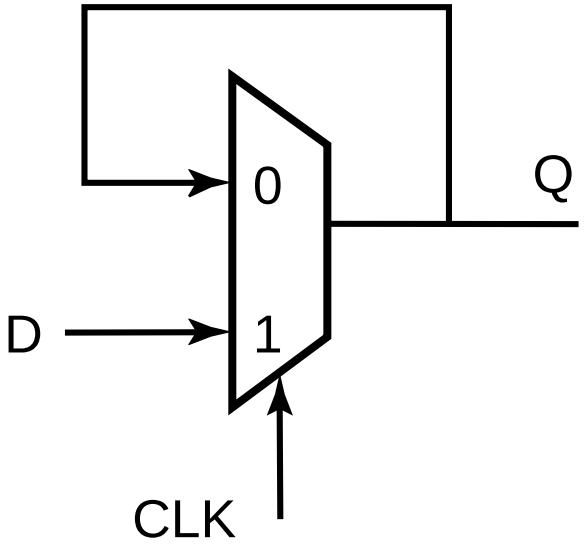
Mux-Based Latches

Negative latch (transparent when CLK= 0)



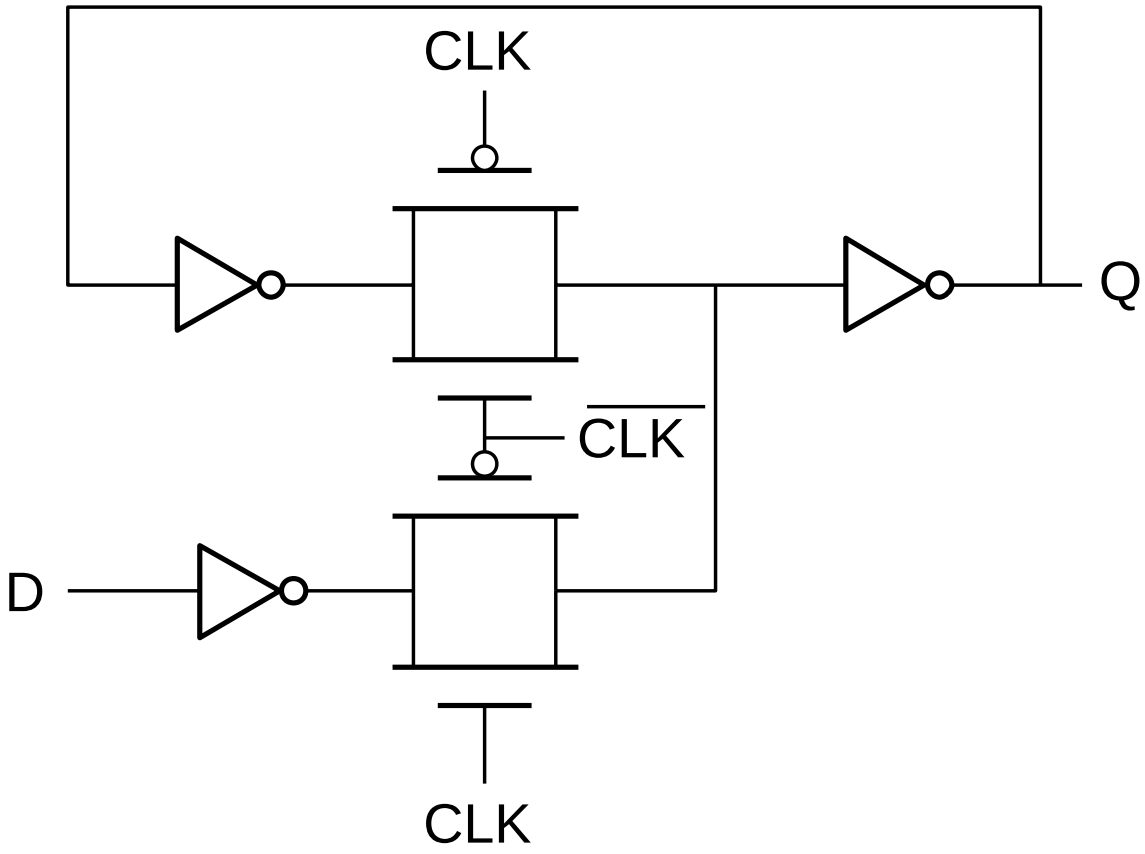
$$Q = Clk \cdot Q + \overline{Clk} \cdot In$$

Positive latch (transparent when CLK= 1)

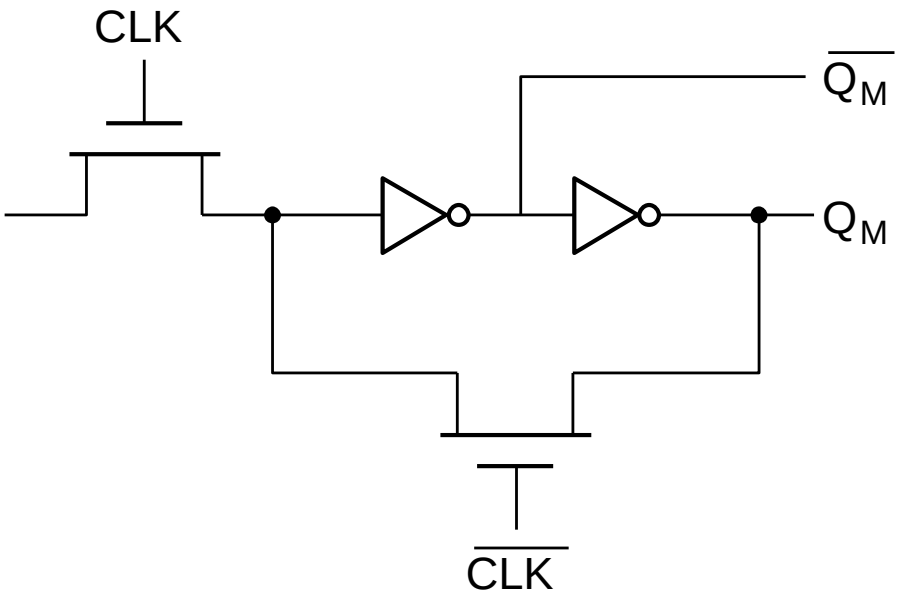


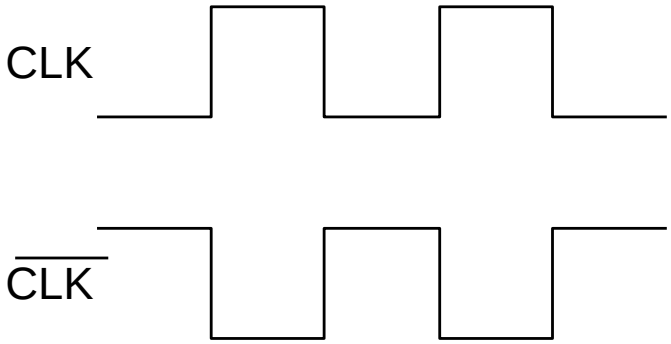
$$Q = \overline{Clk} \cdot Q + Clk \cdot In$$

Mux-Based Latch



Mux-Based Latch



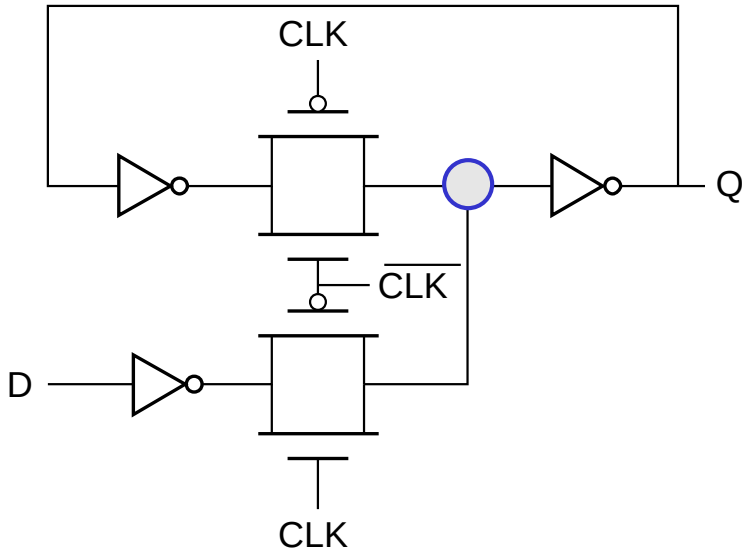


NMOS only

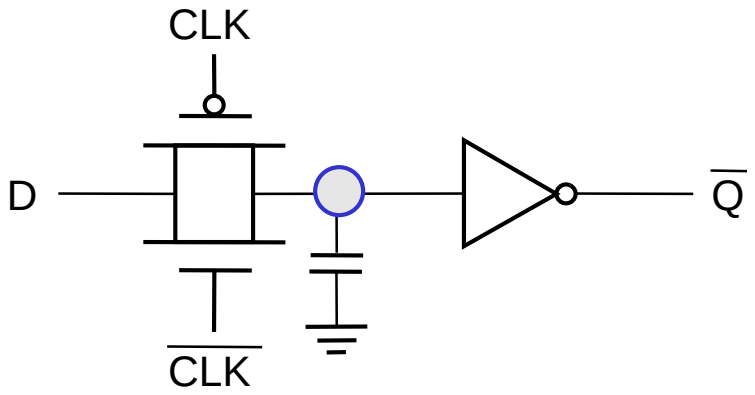
Non-overlapping clocks

Storage Mechanisms

Static

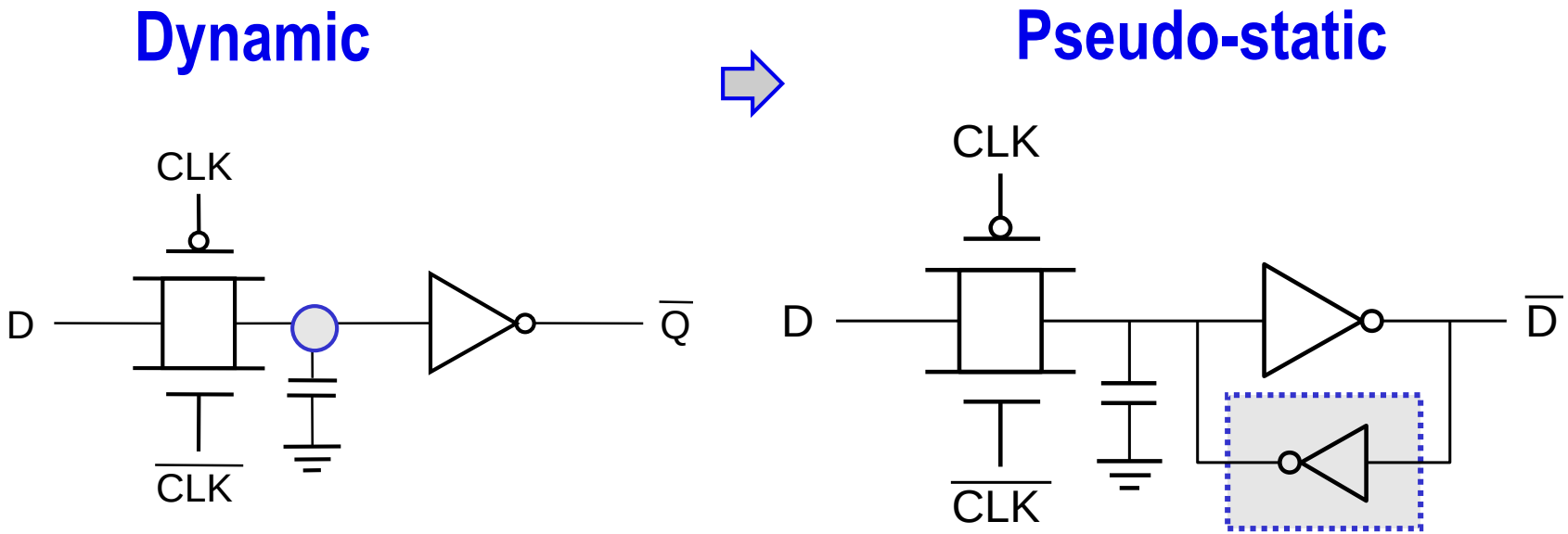


Dynamic



State node

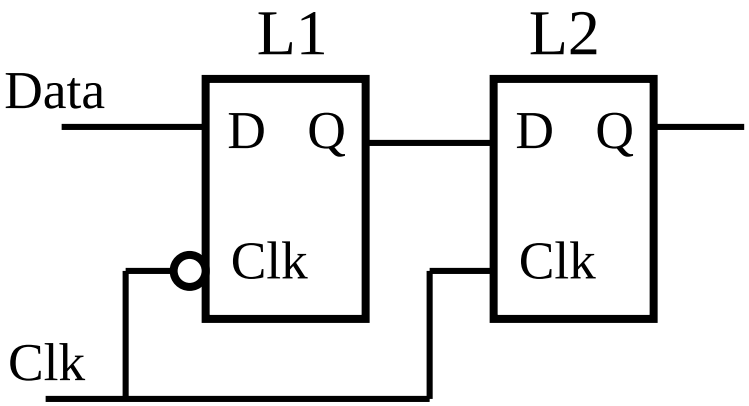
Pseudo-Static Latch



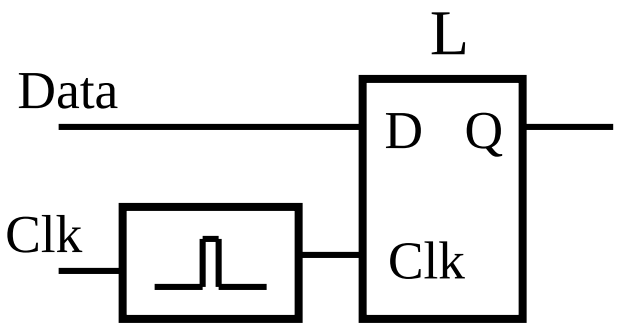
Principal Ways to Build a Flip-Flop

Ways to design an edge-triggered sequential cell:

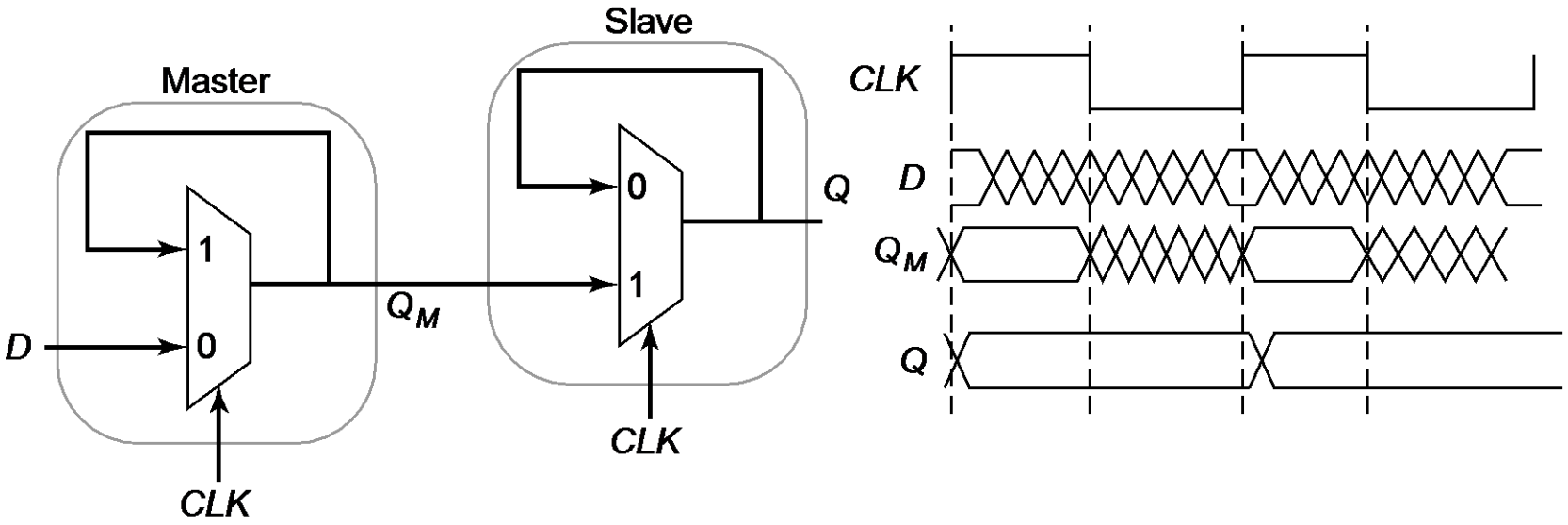
Master-Slave Latches



Pulse-Triggered Latch



Master-Slave (Edge-Triggered) Flip-Flop

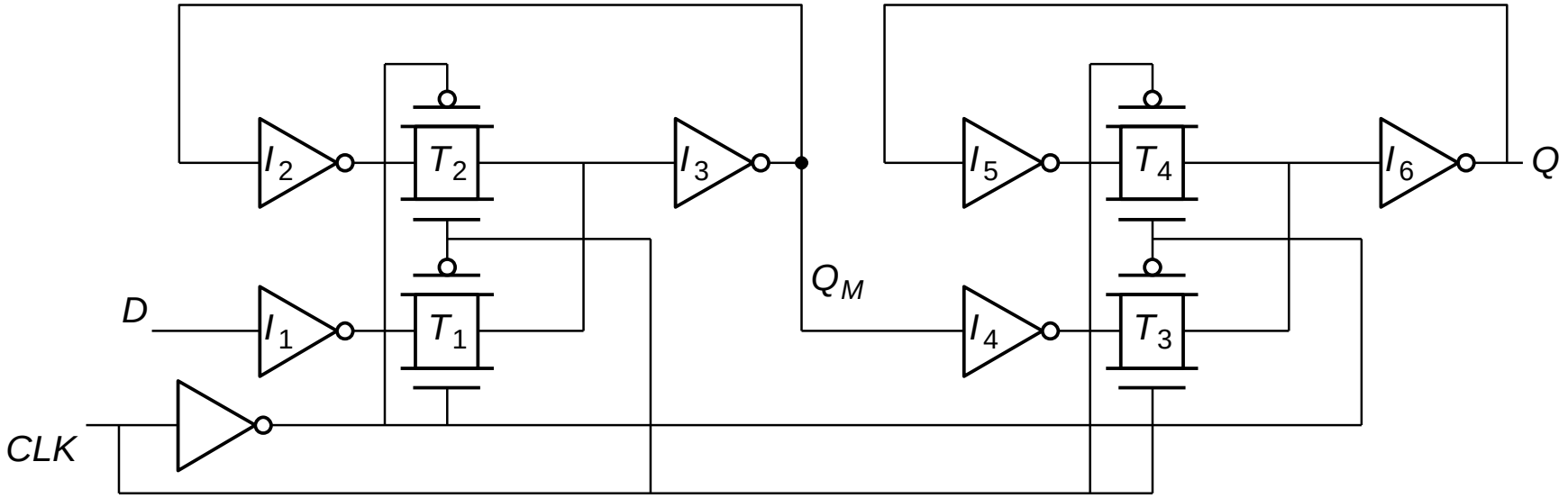


Two opposite latches trigger on edge

Also called master-slave latch pair

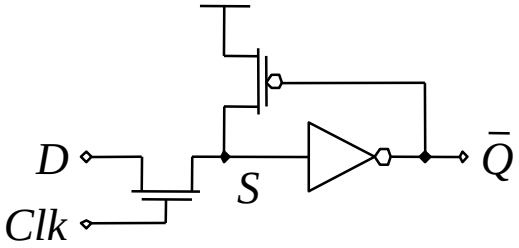
Master-Slave Flip-Flop: Example

Multiplexer-based latch pair



Transmission-Gate Latches

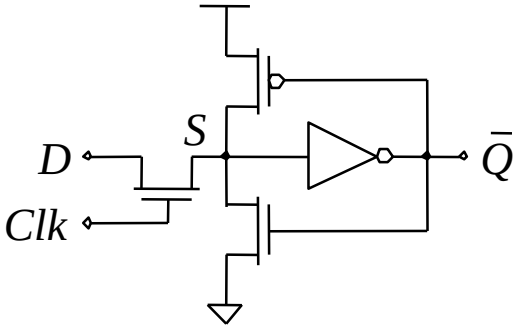
Simplest implementation



(a)

- *only 4 transistors*
- *Dynamic when S=1*
- *Susceptible to noise*

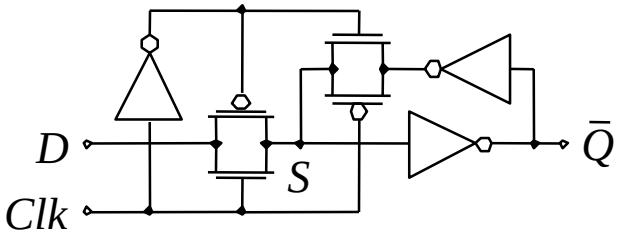
Basic static latch



(b)

- *pull-up/pull-down keeper*
- *Conflict at node S whenever new data is written*

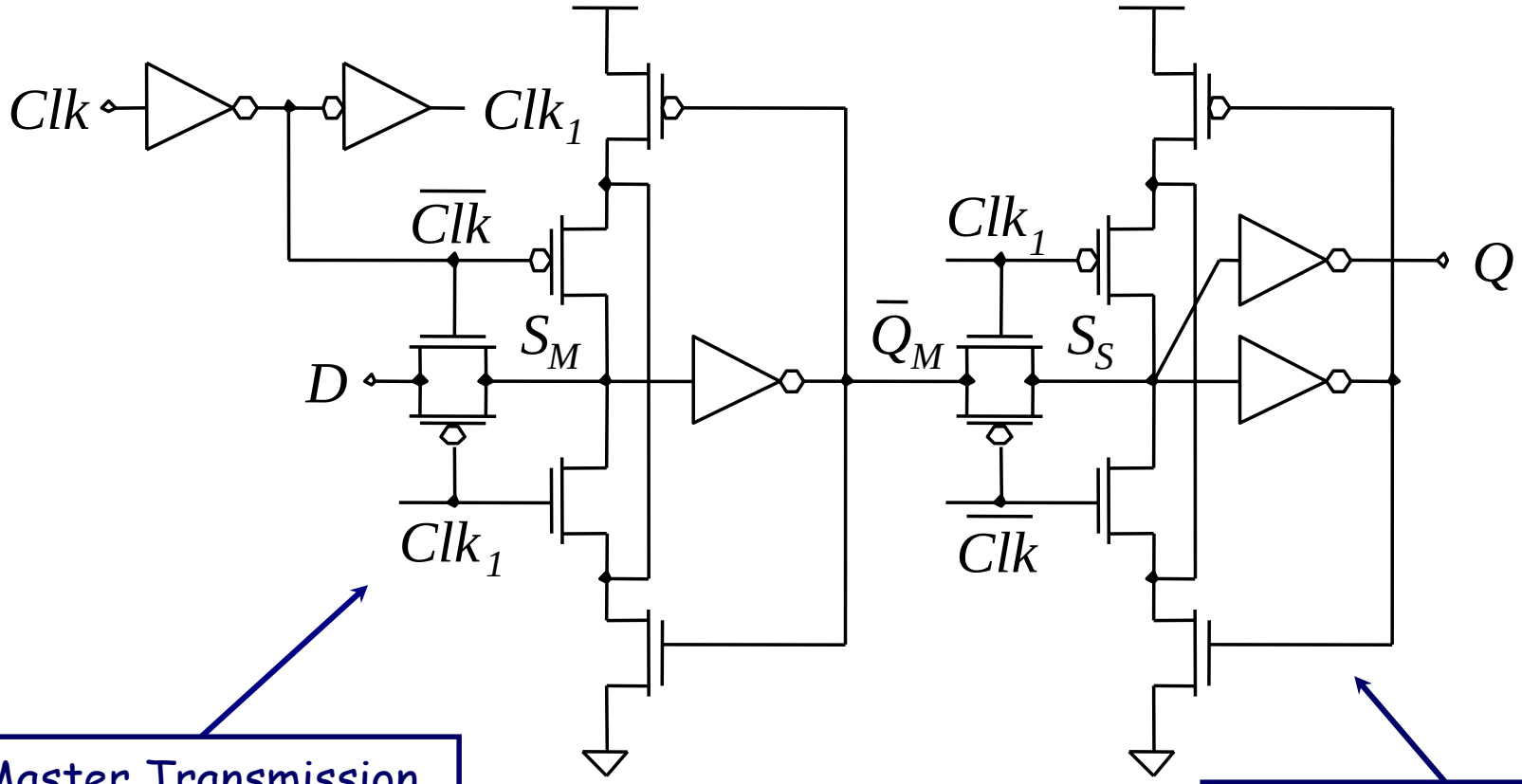
Complete implementation



(c)

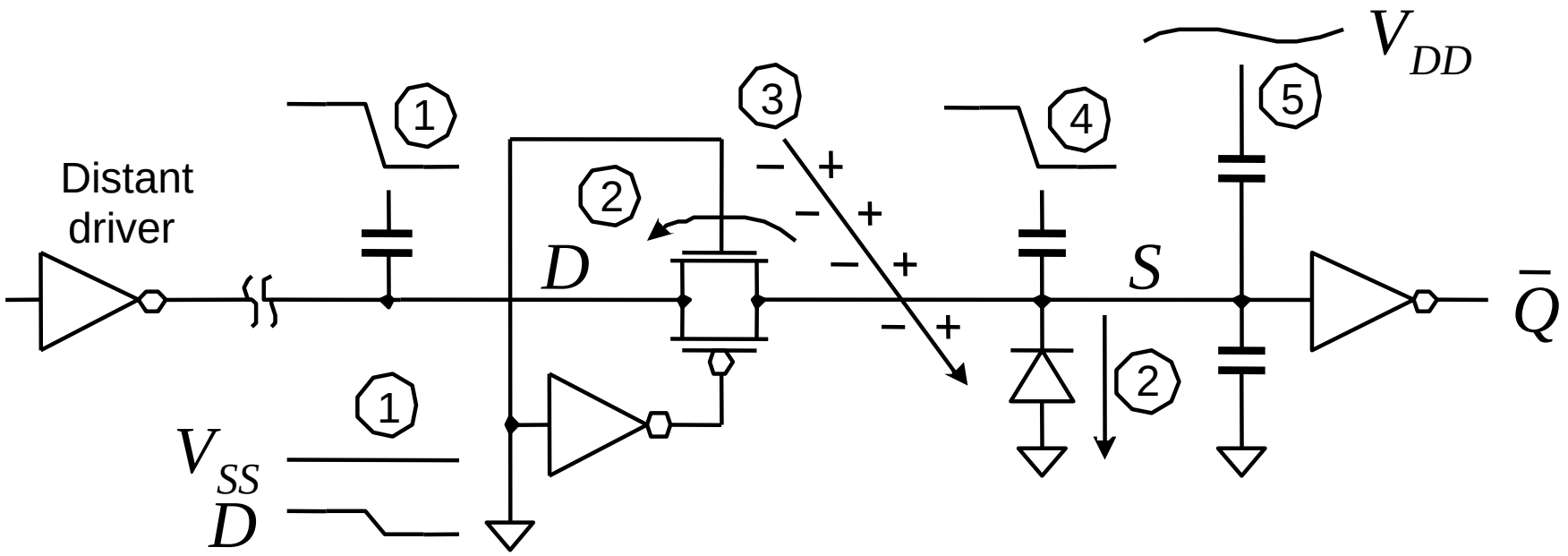
- *Feedback turned off when writing to the latch*
- *No conflict*
- *Larger clock load*

Transmission-Gate Flip-Flop



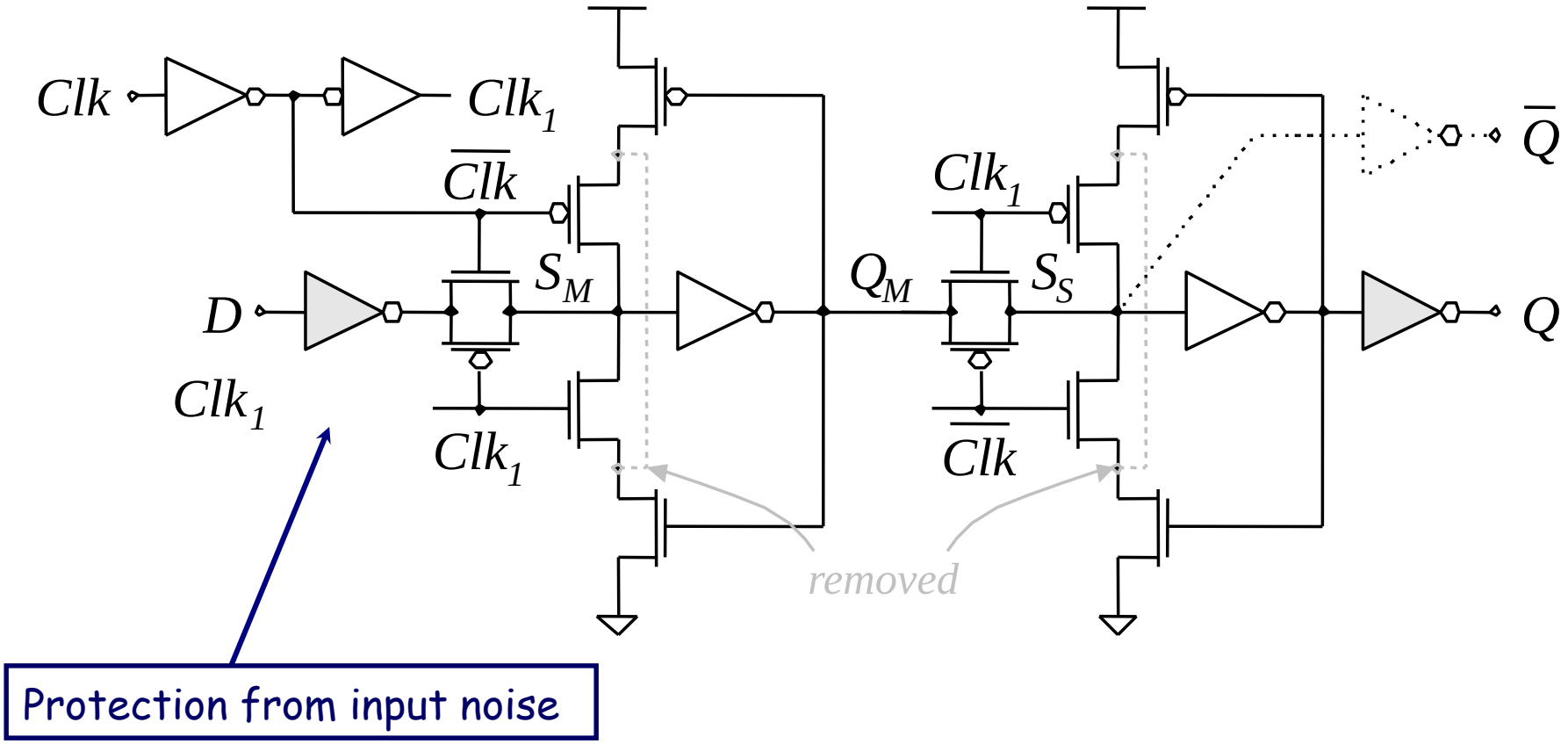
MSL with unprotected input (Gerosa et al. JSSC'94)

Noise Considerations

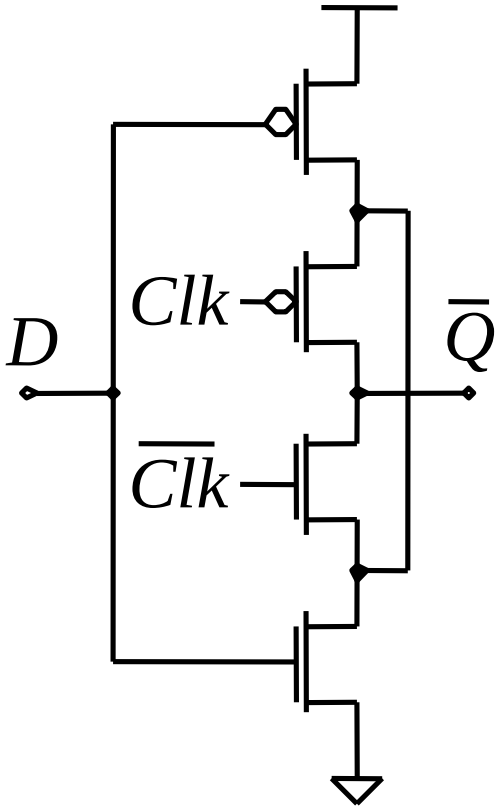


Sources of noise affecting the latch state node (Partovi in Chandrakasan et al. 2001)

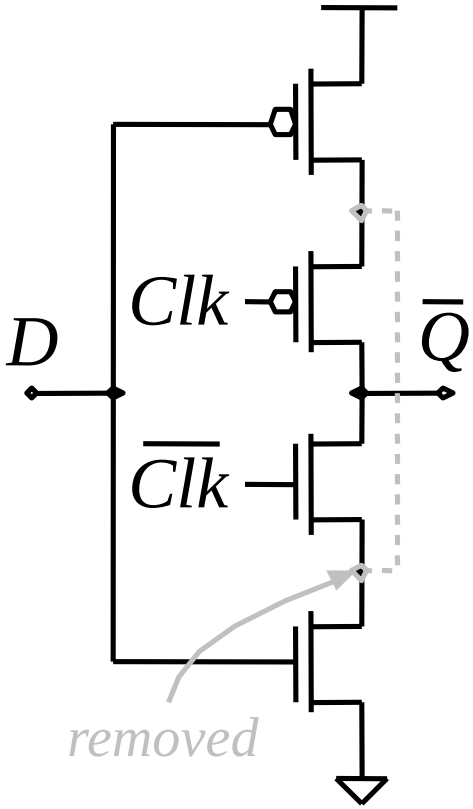
An Improved Version



Clocked CMOS (C^2MOS) Latch

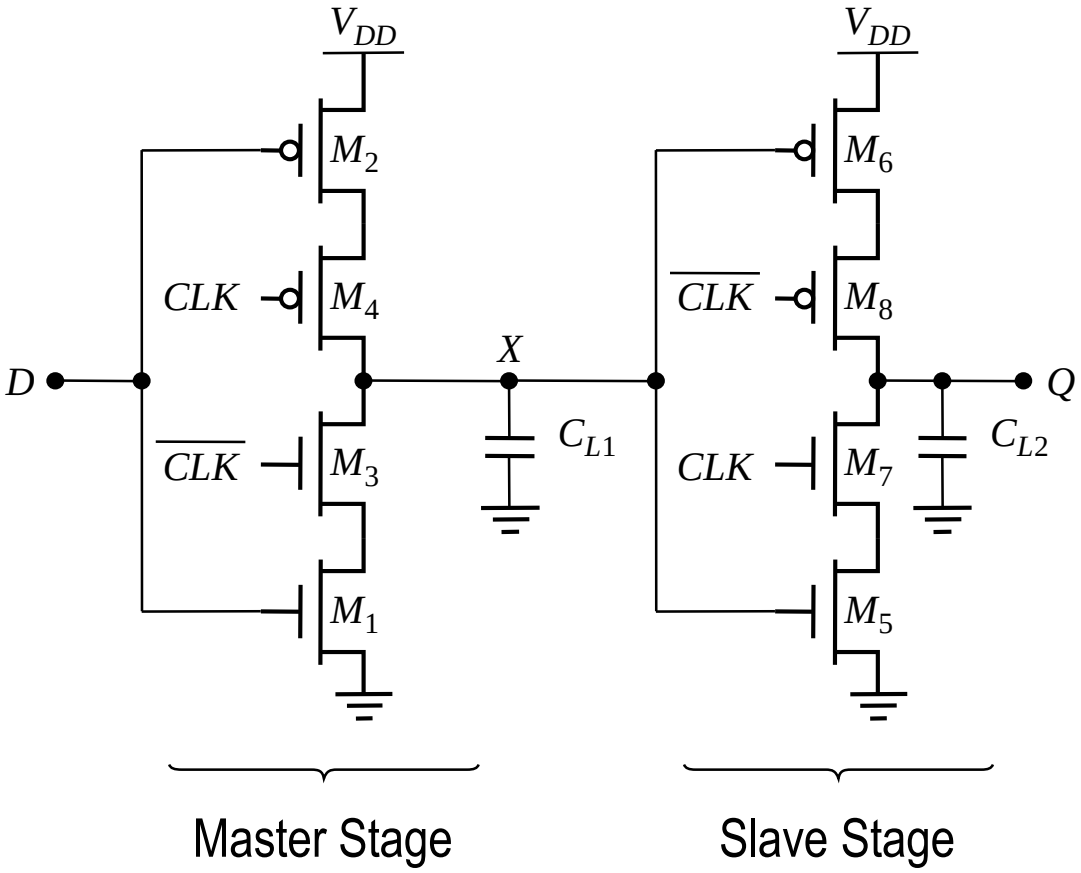


Transmission gate latch with gate isolation (dynamic)



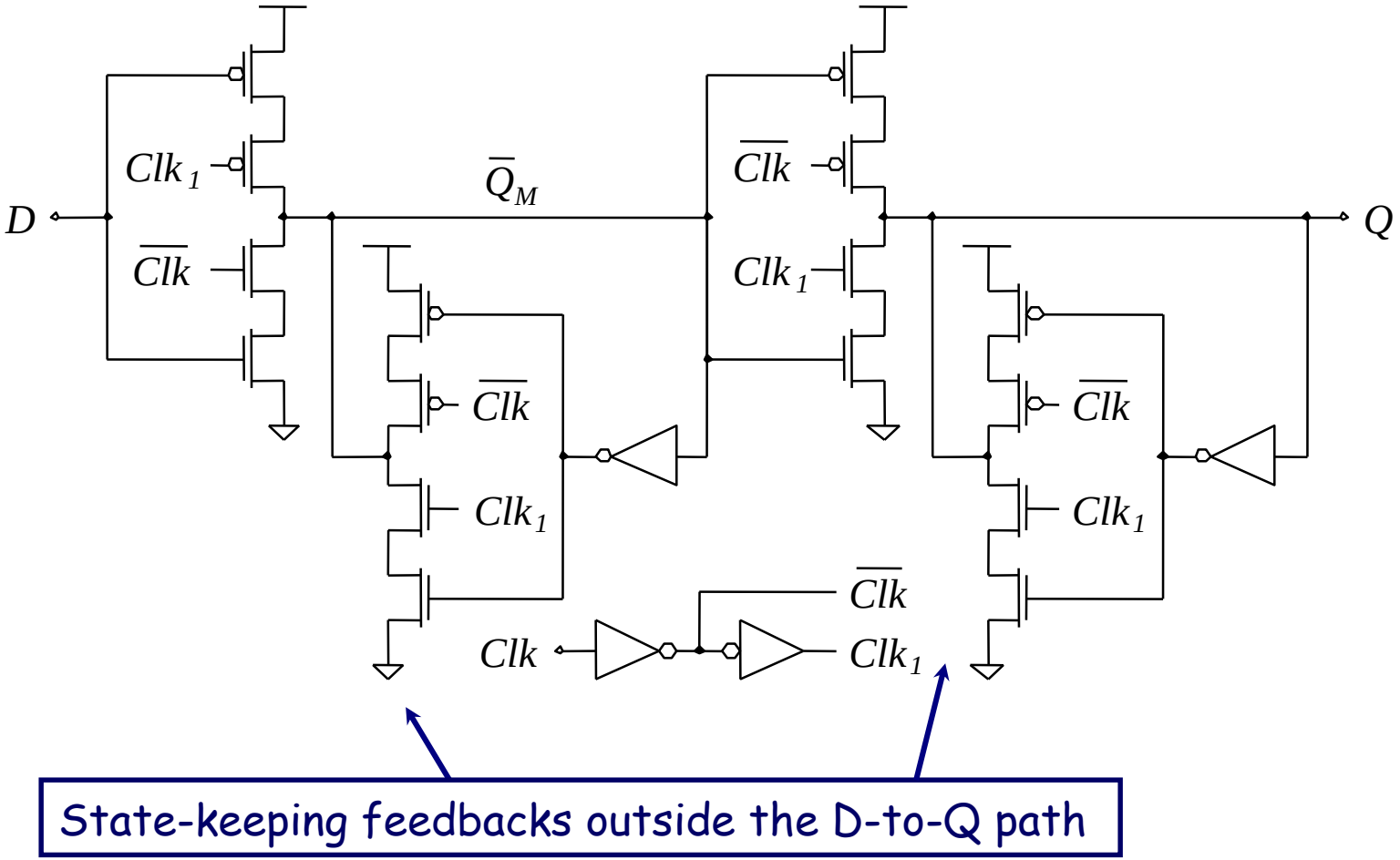
C^2MOS latch (dynamic)

C²MOS Flip-Flop



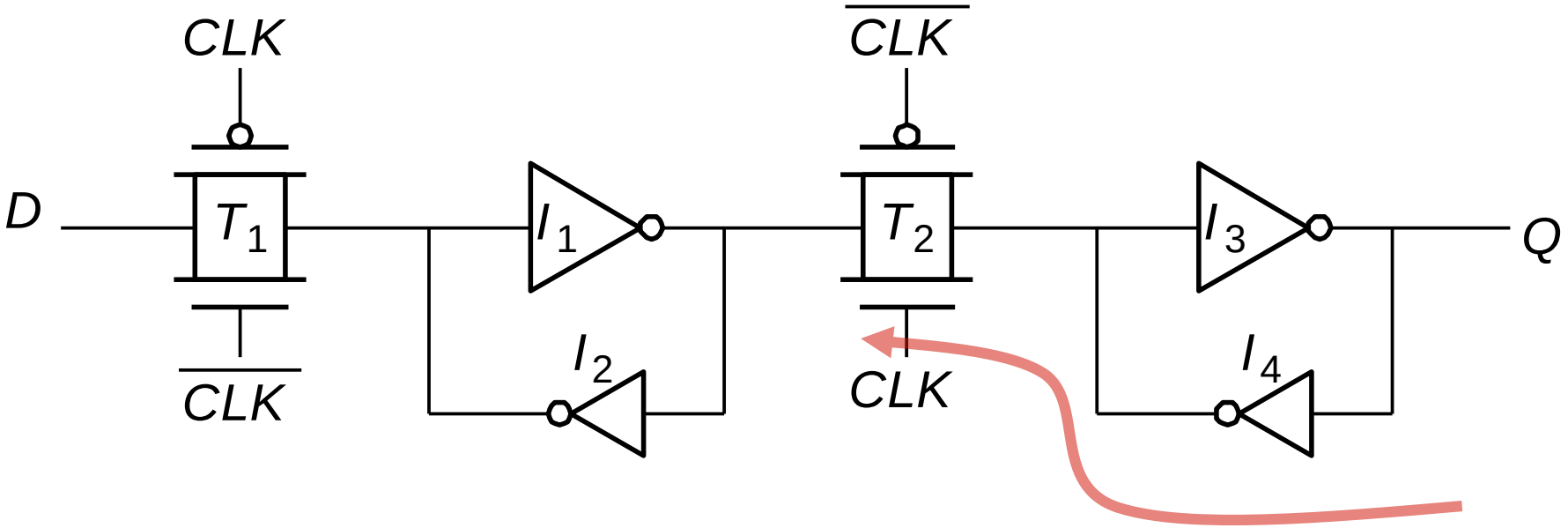
Keepers can be added to staticize

Static C²MOS Flip-Flop

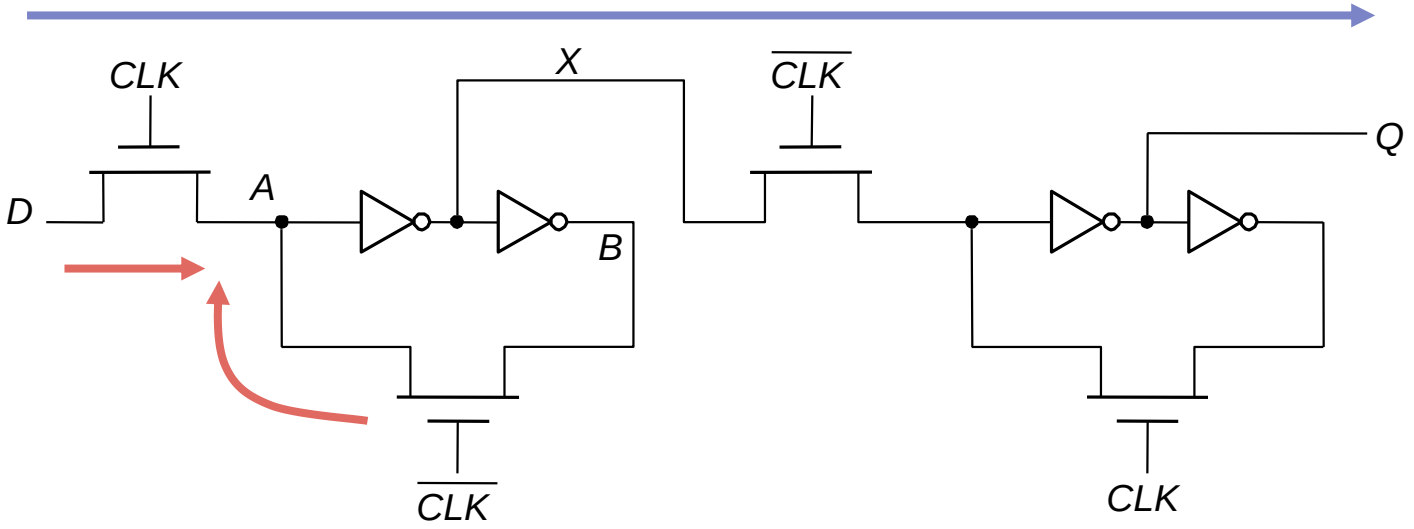


(Suzuki et al. 1973)

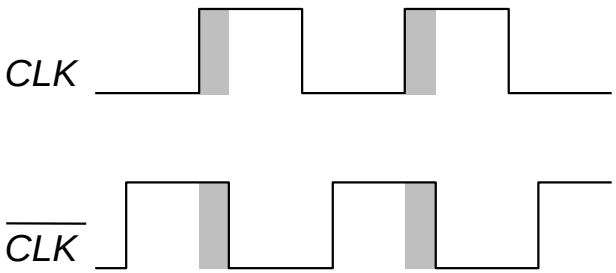
Reduced Clock Load Master-Slave Flip-Flop



Issue: Clock Overlap

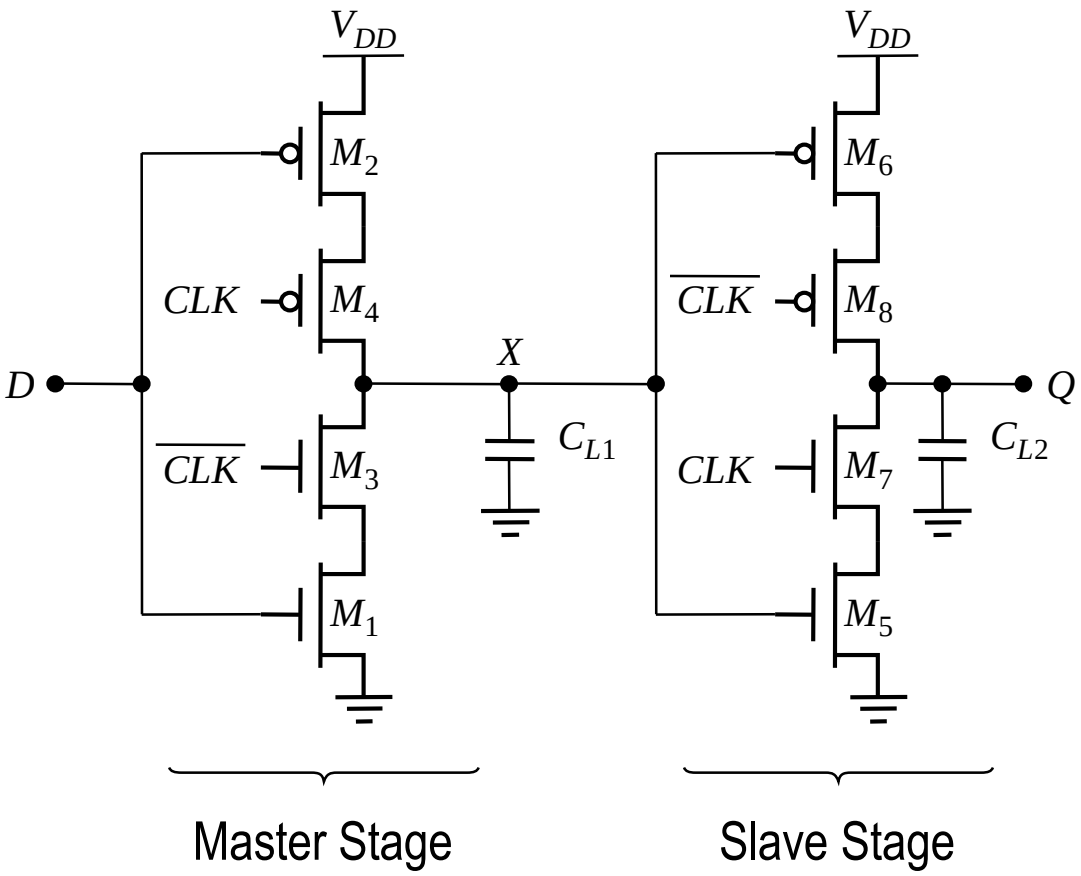


(a) Schematic diagram



(b) Overlapping clock pairs

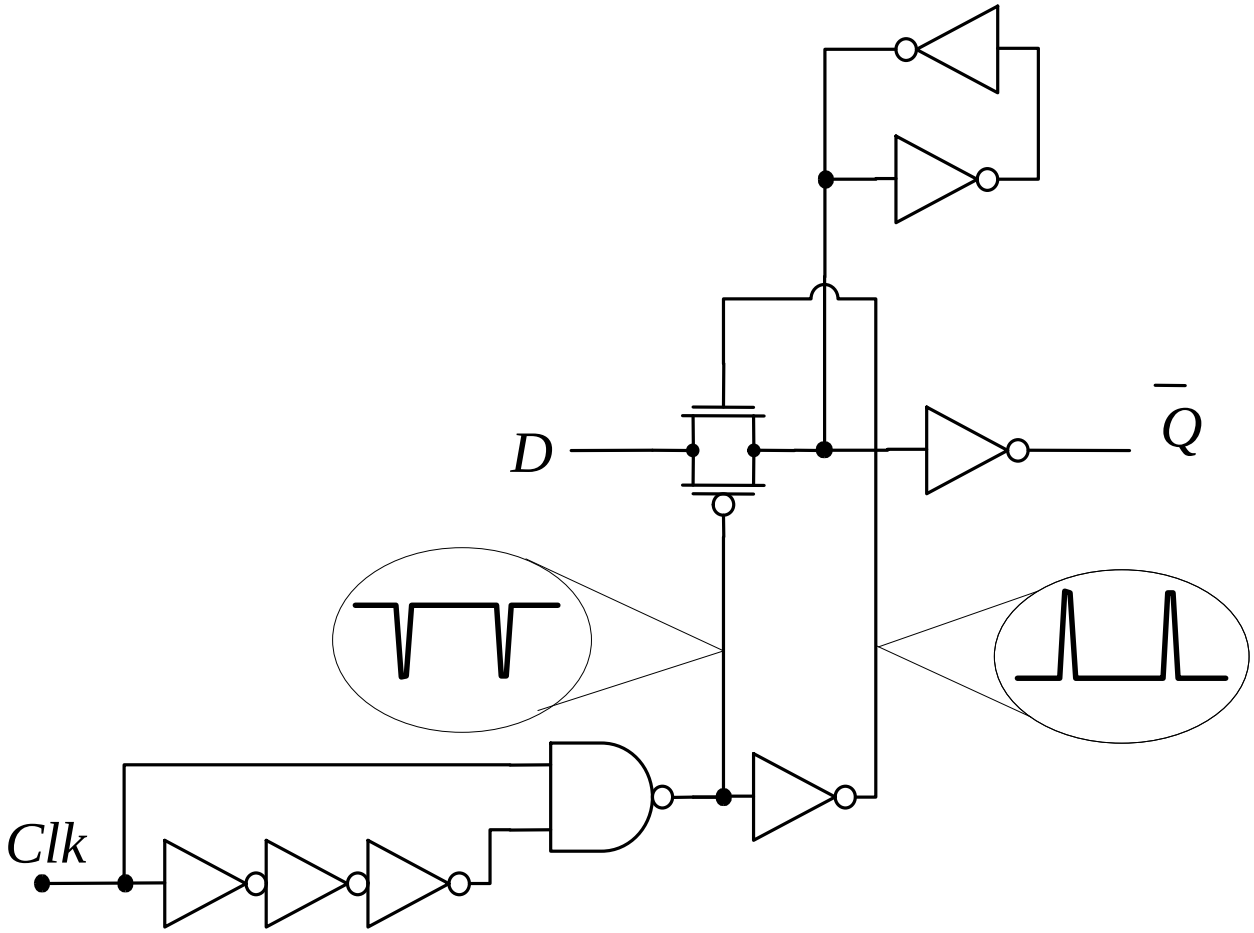
C²MOS Flip-Flop is Insensitive to Clock Overlap



Keepers can be added to staticize

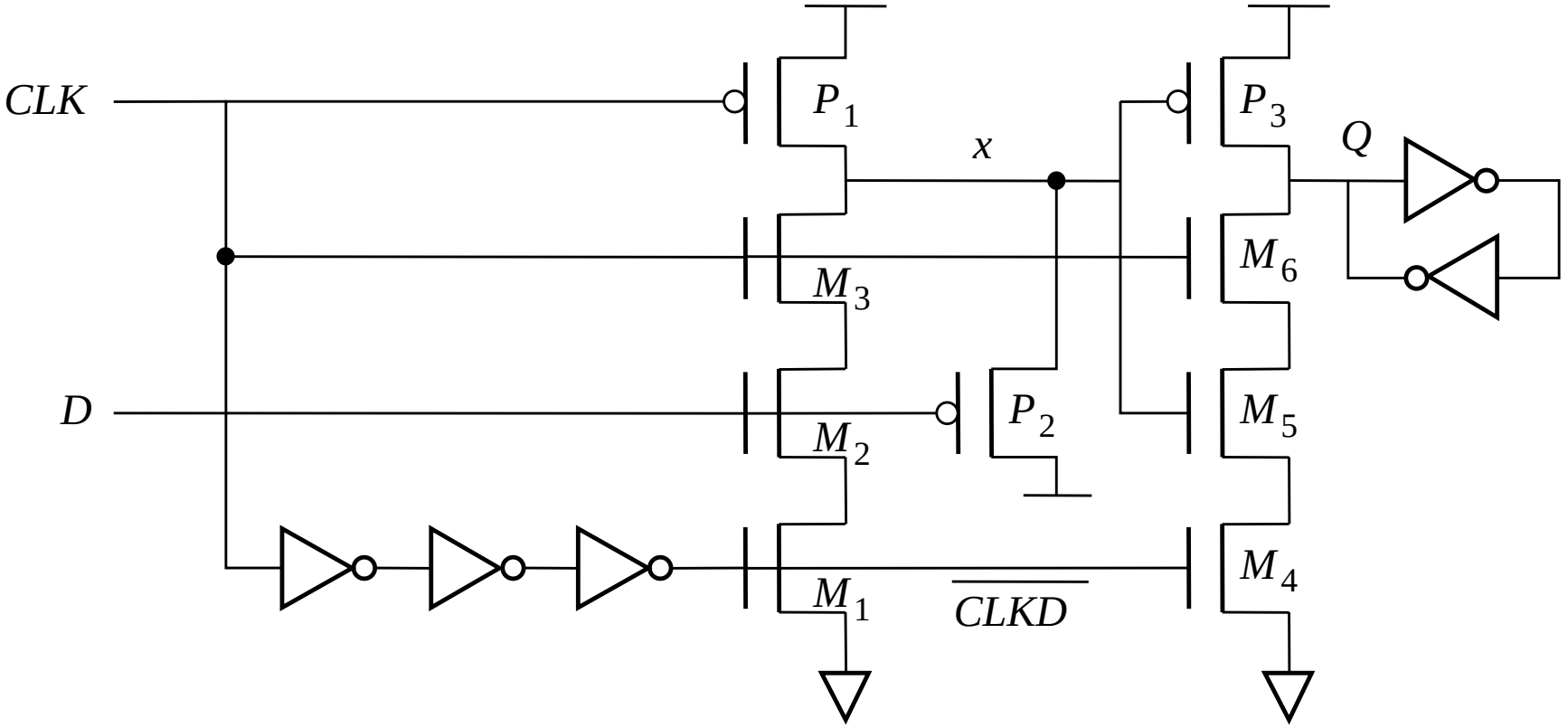
Pulsed latch: Intel's explicit pulsed latch

(Tschanz at al. 2001), Copyright © 2001 IEEE

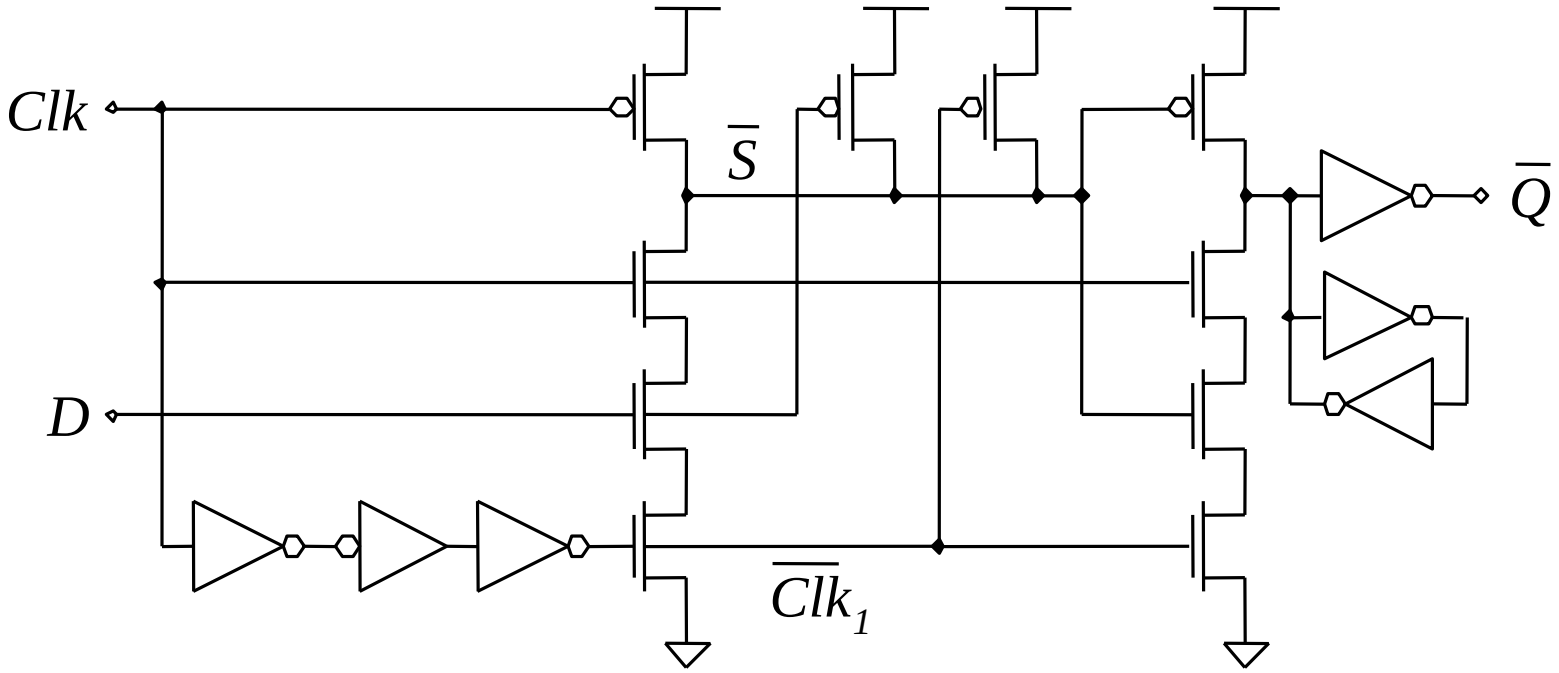


AMD's Pulsed Latches

Hybrid Latch – Flip-flop (HLFF), AMD K-6 and K-7 :



AMD's Hybrid Latch Flip-Flop (HLFF)



- Transparent to D only when Clk and $\overline{\text{Clk1}}$ are both high
- Limited clock uncertainty absorption
- Small D→Q delay
- Small clock load

(Partovi et al. 1996), Copyright © 1996 IEEE