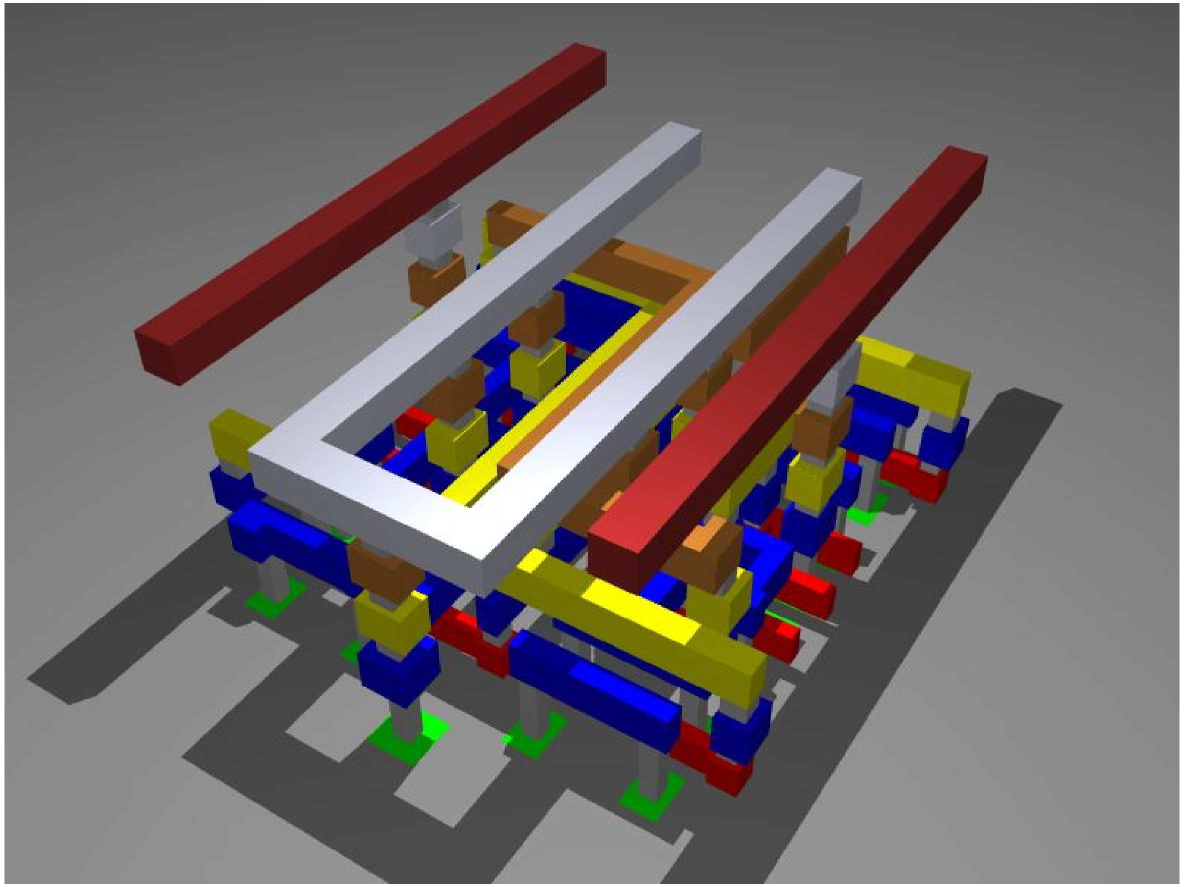
MODULE 6



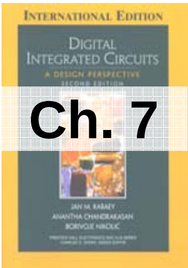
SEQUENTIAL ELEMENTS

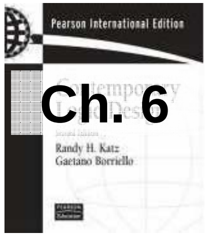
§ 7.5 Will be discussed with module 08 timing design

Sequential Elements - Outline

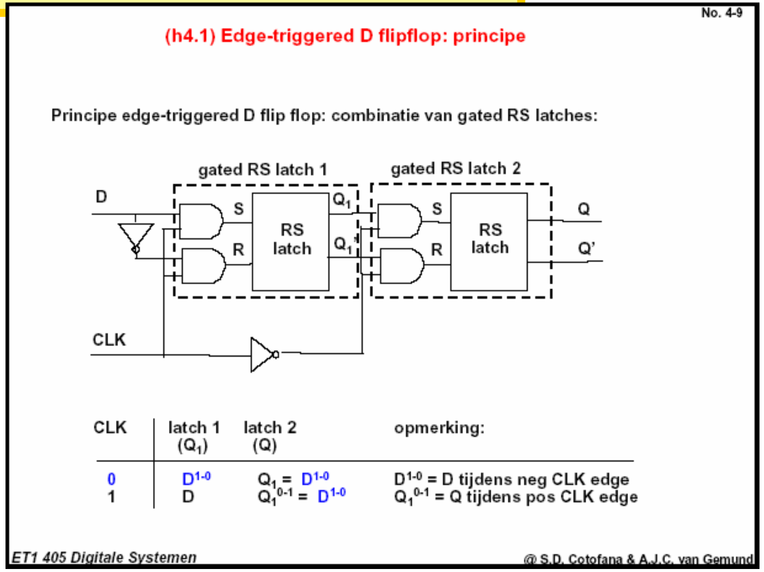
- **Background**
  - **Timing, terminology, classification**
- **Static Flipflops**
  - **Latches**
  - **Registers**
- **Dynamic Flipflops**
  - **Latches**
  - **Registers**
- **Non-bistable elements**
  - **Schmitt Trigger**

- **Much of this material has already been covered in CS1 (v Genderen), Lecture 4(?), Katz Section 6.1**
- **Here we will add transistor-level implementation, dynamic flipflops**

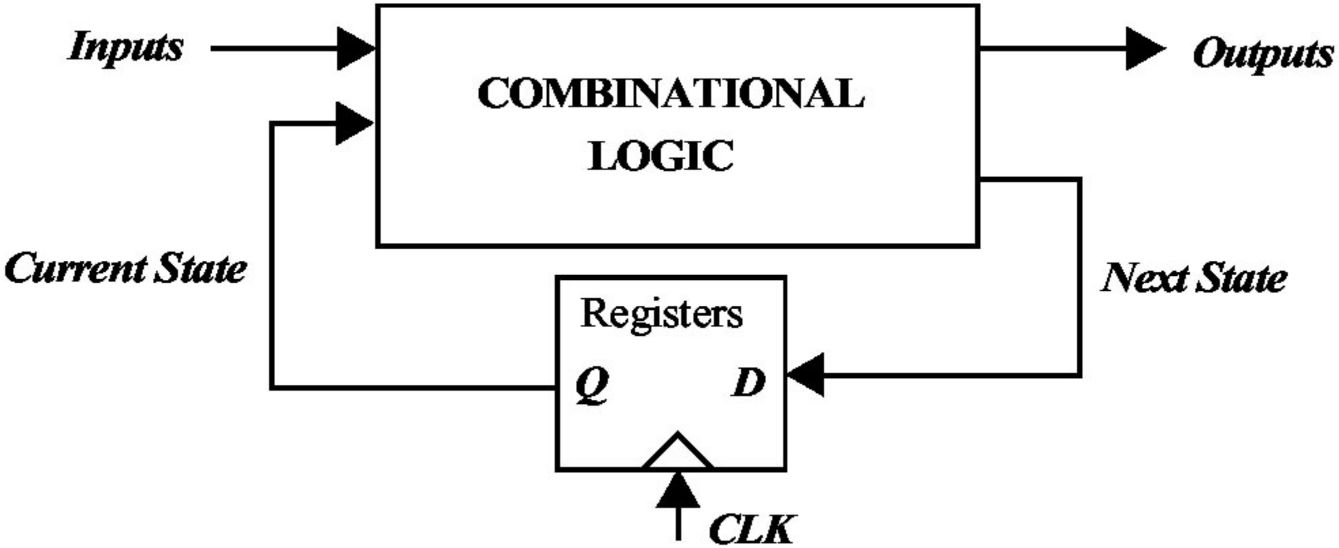


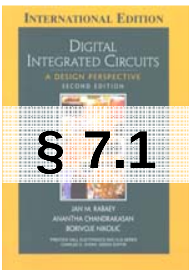


[Sorin Cotofana]



FSM with Positive Edge Triggered Registers





- Flip-flops provide **memory/state**
- VLSI uses predominantly **D-type flip-flops**

Memory elements

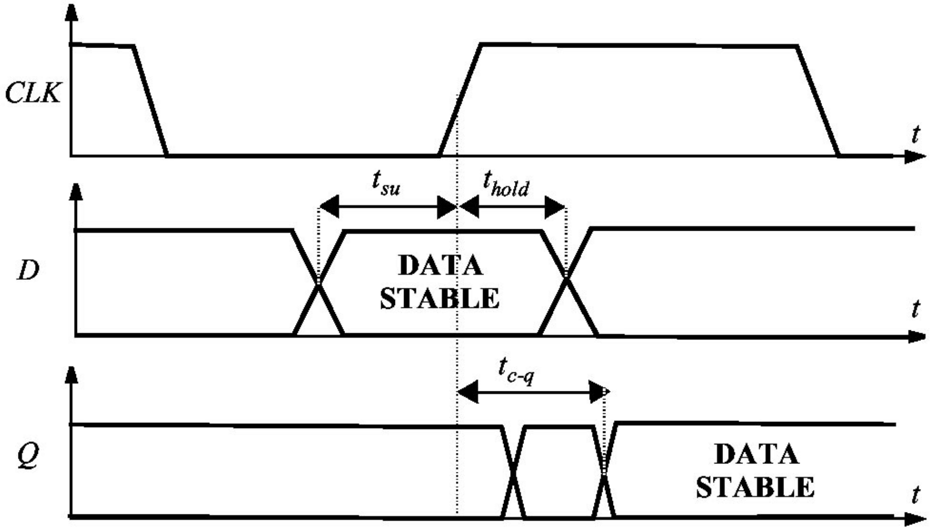
- **Store a temporary value, remember a state**
- **Typically controlled by clock.**
- **May have load signal, etc.**
- **In CMOS, memory is created by:**
  - **capacitance (dynamic);**
  - **feedback (static).**

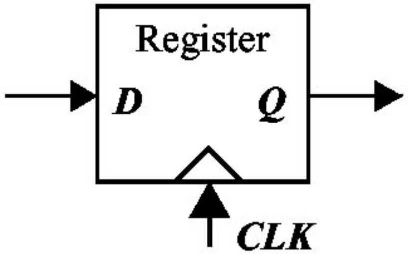
- **Also see http://en.wikipedia.org/wiki/Flip-flop_(electronics)**

Variations in memory elements

- **Form of required clock signal.**
- **How behavior of data input around clock affects the stored value.**
- **When the stored value is presented to the output.**
- **Whether there is ever a combinational path from input to output.**

Timing Metrics Reminder





t_{c-q} : delay from clock (edge) to Q

t_{su} : setup time

t_{hold} : hold time

t_{plogic} : worst case propagation delay of logic

t_{cd} : best case propagation delay (contamination delay)

T : clock period

$$T \geq t_{c-q} + t_{plogic} + t_{su}$$

$$t_{cdregister} + t_{cdlogic} \geq t_{hold}$$

Nomenclature

Beware for confusion

	Katz (CS1)	Rabaey
Latch	Level sensitive storage element	Level sensitive storage element
Register	Group of storage elements	Edge triggered storage element
Flip Flop	Edge triggered storage element	Bistable element using feedback

$CLK = CK = \phi$

Latches vs. Registers

Latch

Level-sensitive

Transparent when clock is active

Clock active high: positive latch

Clock active low: negative latch

Faster, smaller

Register

Edge-triggered

Input and output isolated

Sampling on 0 $\rightarrow$ 1 clock: positive edge triggered

Sampling on 1 $\rightarrow$ 0 clock: negative edge triggered

Safer

Static vs. Dynamic Memory Elements

Static

Operate through **positive feedback**

Preserve state as long as power is on

Can work when clock is **off**

More **robust**

Dynamic

Store charge on (parasitic) capacitor

Charge **leaks** away (in milliseconds)

Clock must be kept **running** (for periodic refresh)

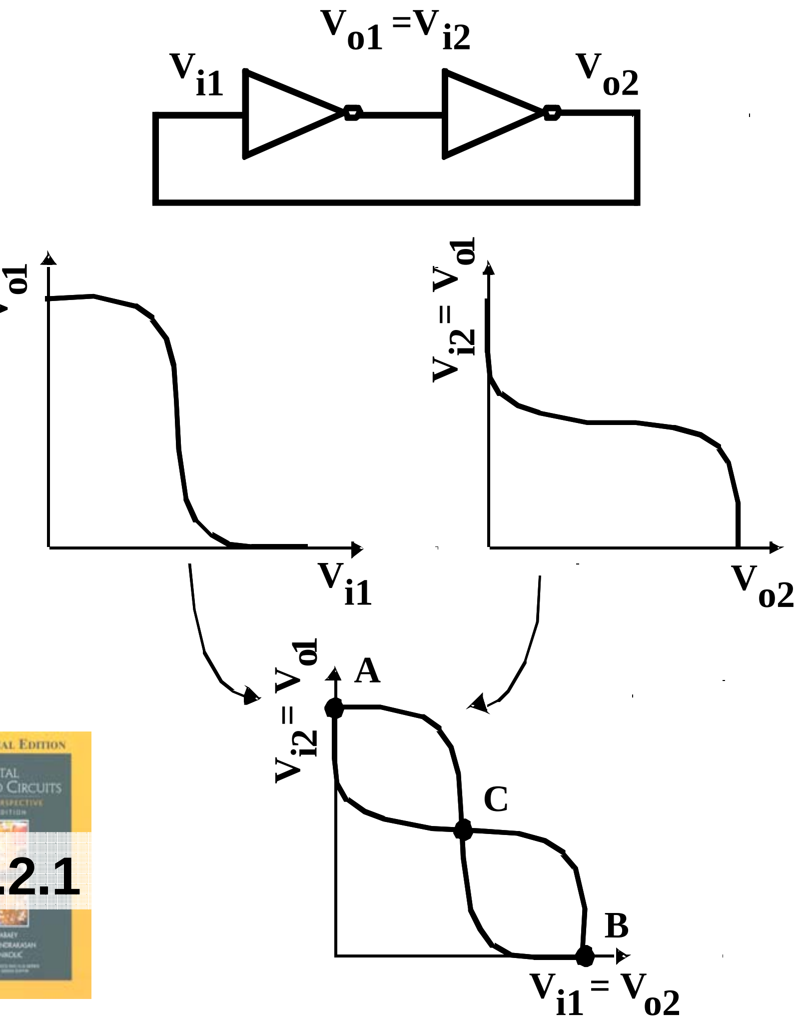
Faster, smaller

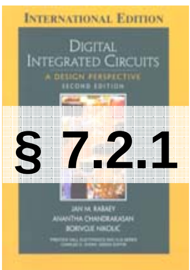
Rabaey: bistable elements are called Flip Flops

Static Latches and Registers

- **Latches → can be gated or not**
- **Registers**

Positive Feedback: Bi-Stability

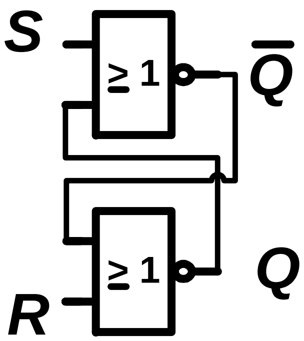


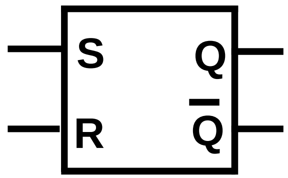


§ 7.2.1

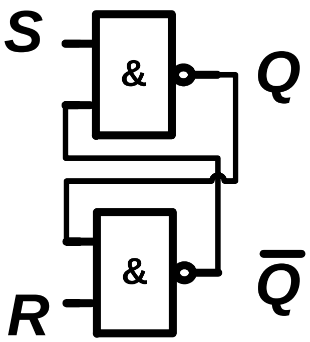
- **Loop-gain in A,B <<1**
- **A,B: stable points**
- **Loop-gain in C >> 1**
- **C: meta-stable point**

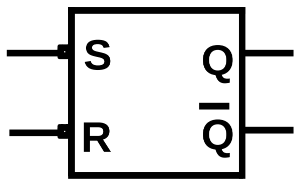
SR-Latch





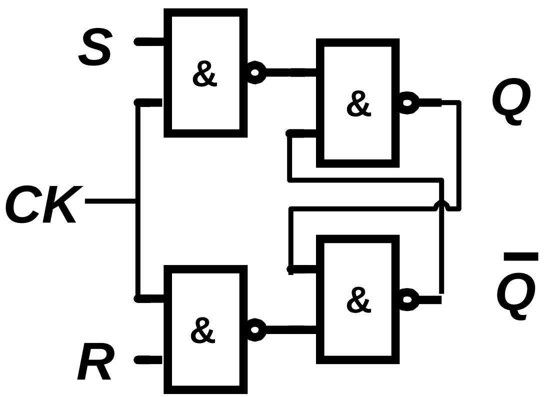
S	R	Q	$\overline{Q}$	
0	0	Q	$\overline{Q}$	
1	0	1	0	
0	1	0	1	
1	1	0	0	← **forbidden**



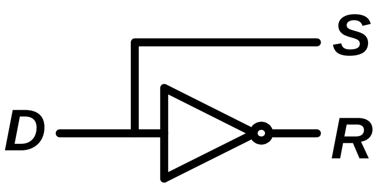


S	R	Q	$\overline{Q}$	
1	1	Q	$\overline{Q}$	
0	1	1	0	
1	0	0	1	
0	0	1	1	← **forbidden**

Clocked SR-Latch

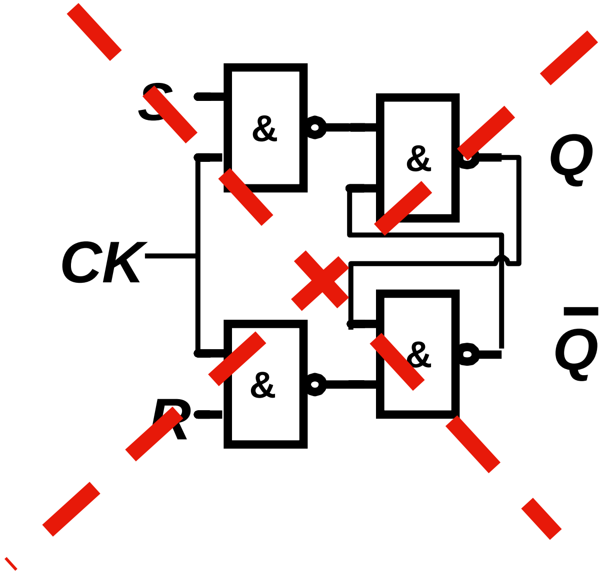


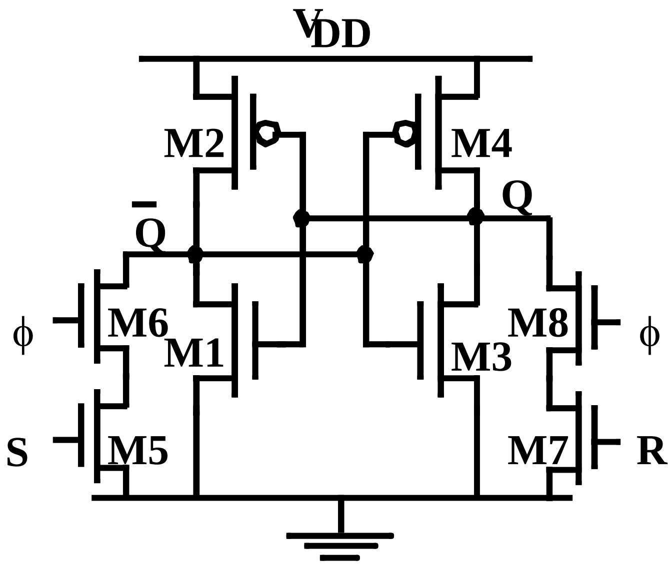
- Katz: gated latch
- Positive latch (active on CK high)
- Naïve implementation
- 16 transistors
- D latch requires 9xN, 9xP
- Larger area, cost, power



- Construction of D-latch
- D-latch, D-register most common in VLSI

CMOS Clocked SR-Latch



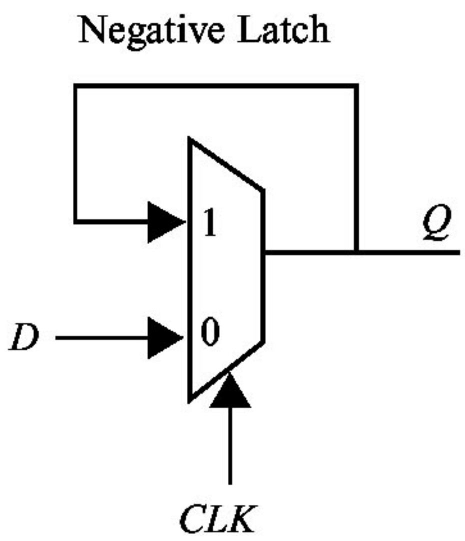


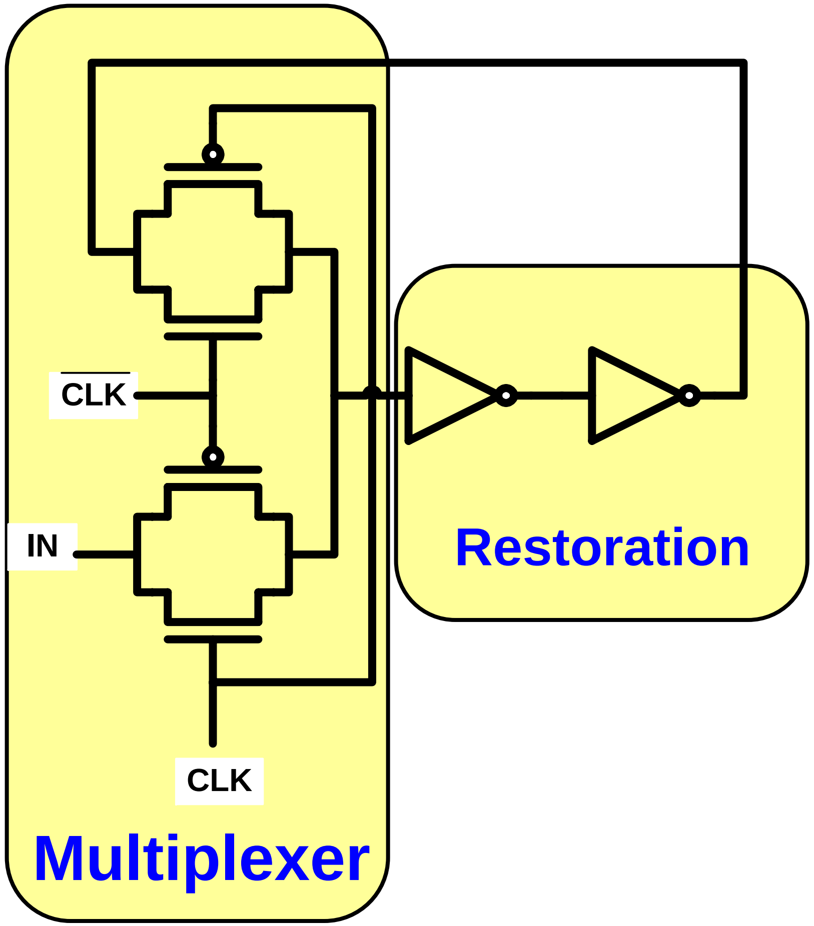
$\phi = CK$

- Save 6 (incl. 4 large) PMOS transistors and 2 NMOS
- D-latch requires 7 x N, 3 x P (instead of 9xN, 9xP)

Q: Is this a **ratioed** design or not?
Does it consume static power?

Multiplexer-Based Latches

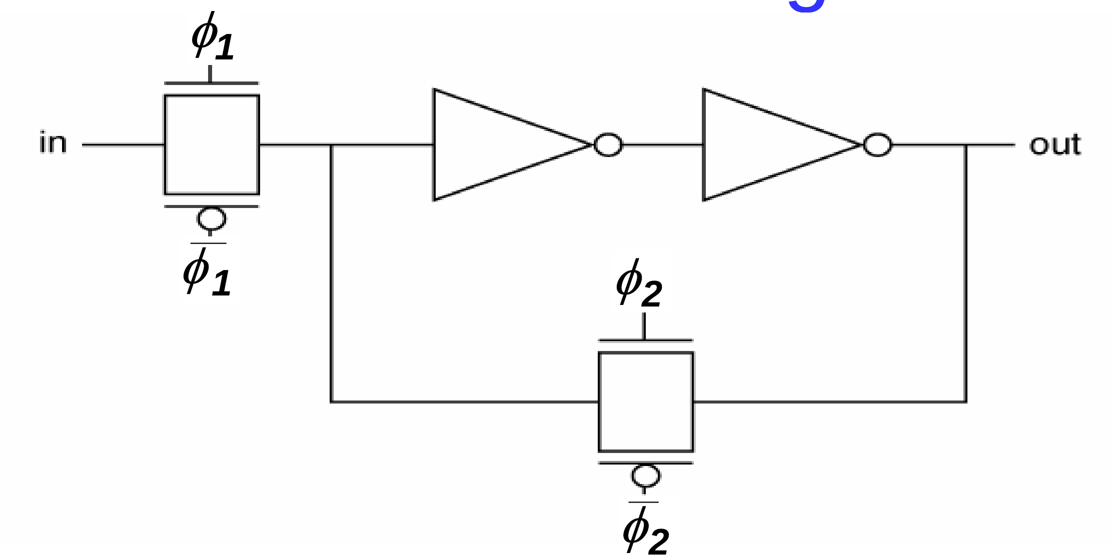




Recirculating latch

Mux-based latches much more common in modern dig. IC's

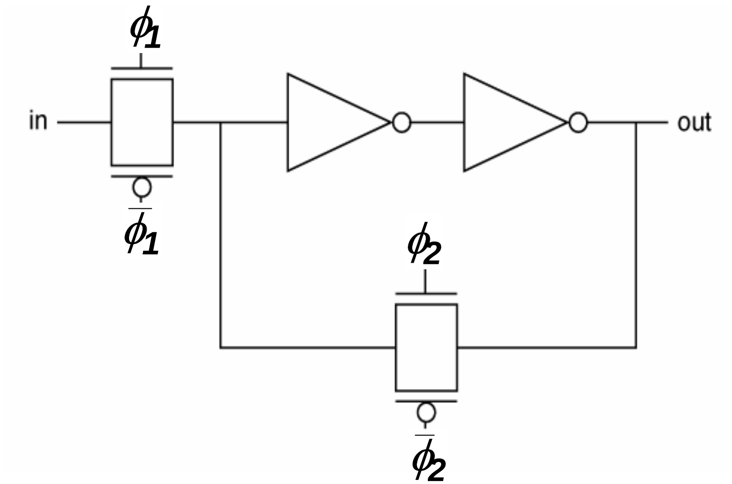
Recirculating latch

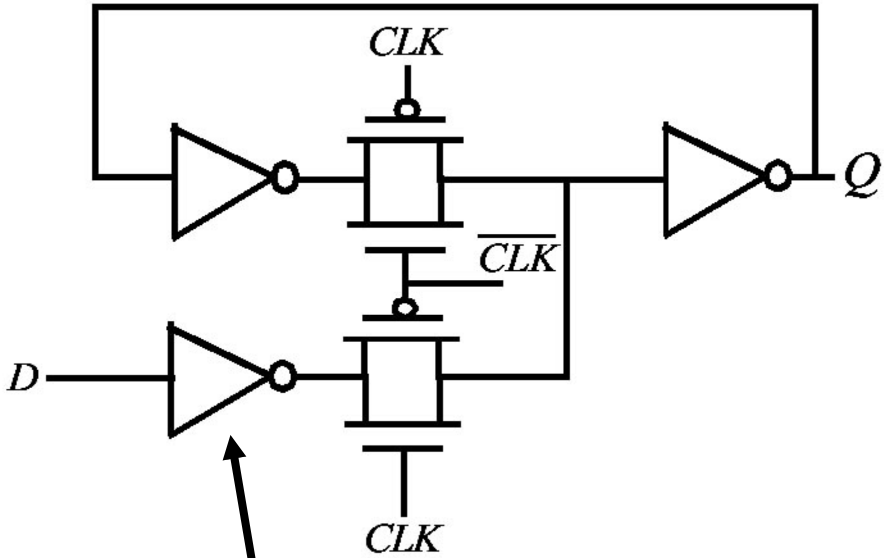


- **Quasi-static, static on one phase**
- **Feedback restores value**
- **Requires 4 x N, 4 x P, minimum size**

  (compare 7 x N, 3 x P, non-minimum size)
- **ϕ_1 and ϕ_2 inverse but should be non-overlapping**
- **Can suffer from charge sharing (when ϕ not non-overlapping)**
  C_{in} and C_{load} form communicating vessels when Output connected directly to input

Insensitive for Charge Sharing

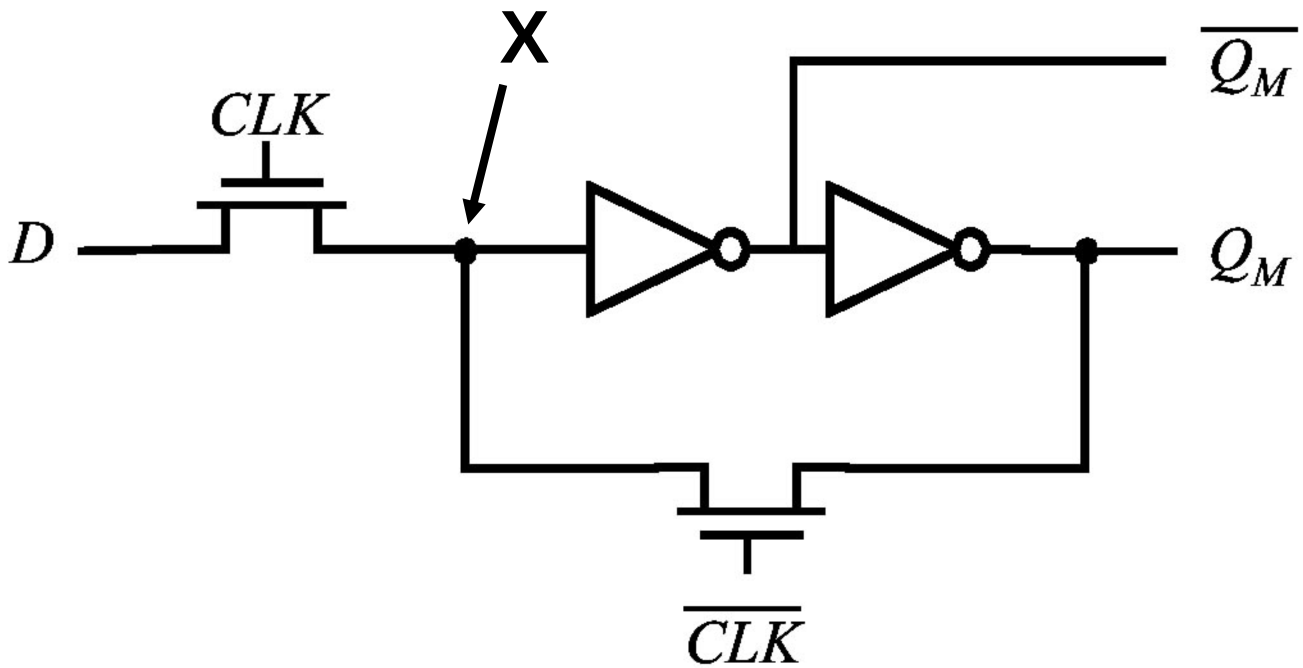




Uni-directionality of this inverter prevents coupling between Q and D

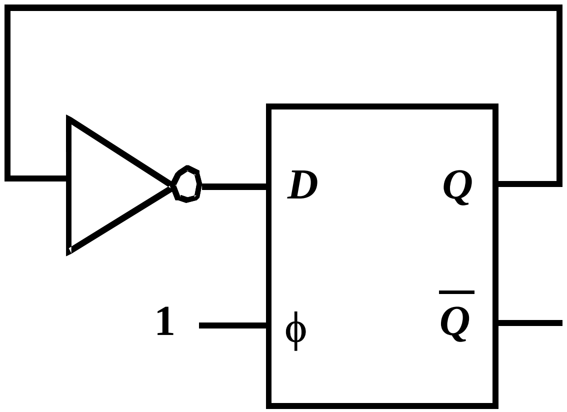
- **Non ratioed**
- **High load to CLK**

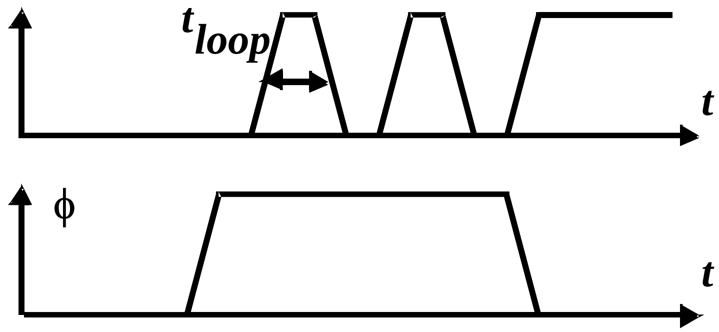
Recirculating NMOS Latch.



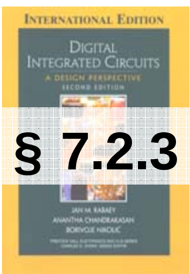
- **Degraded 1 at X**
- **Lower noise margin, higher delay, power**

Latch Designs can Suffer from Race Problems



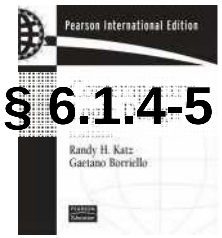


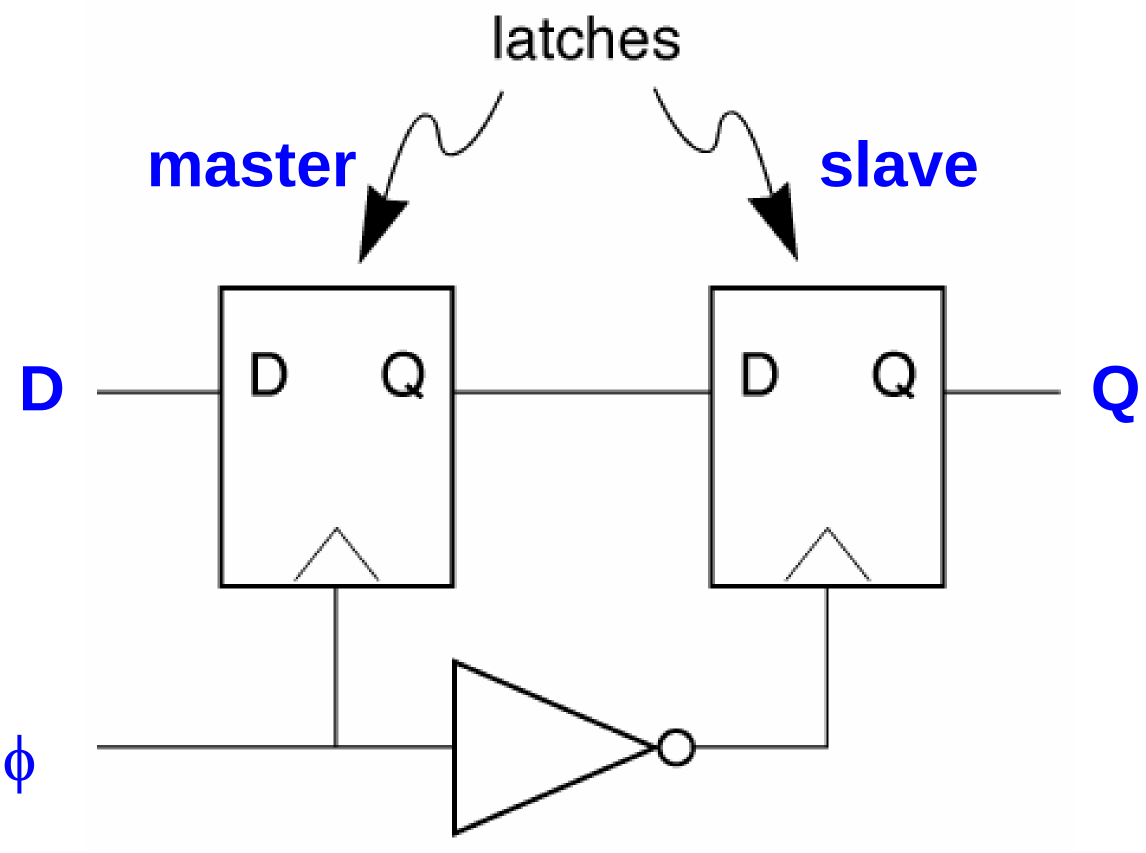
Signal can race around during $\phi = 1$



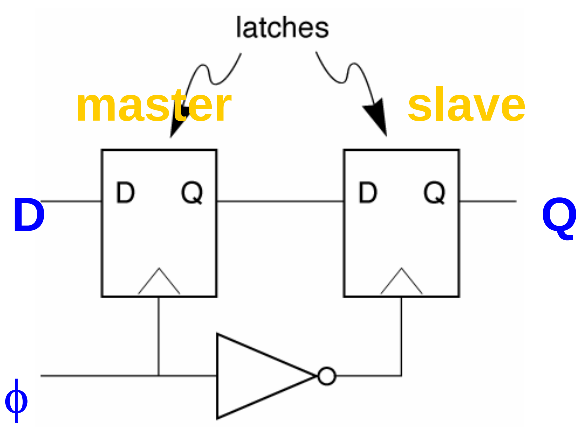
Registers

- **Not transparent—use multiple storage elements to isolate output from input.**
- **Master-slave, edge triggered principle**





Master-slave operation



ϕ = 0:

- **master latch is disabled;**
- **slave latch is enabled,**
- **but master latch output is stable,**
- **so output does not change.**

ϕ = 1:

- **master latch is enabled,**
- **loading value from input;**
- **slave latch is disabled,**
- **maintaining old output value.**

CMOS Flip-Flop Construction - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://www.play-hookey.com/digital/cmos_d_flip-flop.html

CMOS Flip-Flop Construction

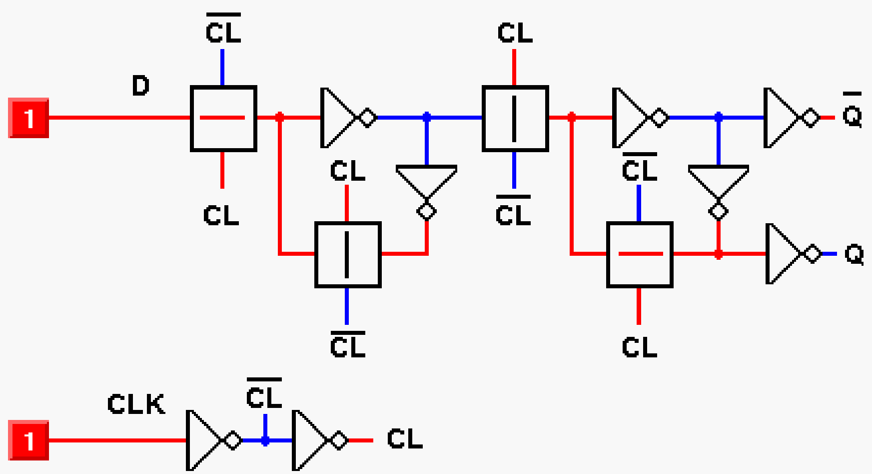
CMOS technology allows a very different approach to flip-flop design and construction. Instead of using logic gates to connect the clock signal to the master and slave sections of the flip-flop, a CMOS flip-flop uses *transmission gates* to control the data connections. (See the CMOS gate electronics page for a closer look at the transmission gate itself.)

The result is that a controllable flip-flop can be built with only inverters and transmission gates — a very small and simple structure for an IC.

The basic CMOS D flip-flop is shown below.

www.play-hookey.com

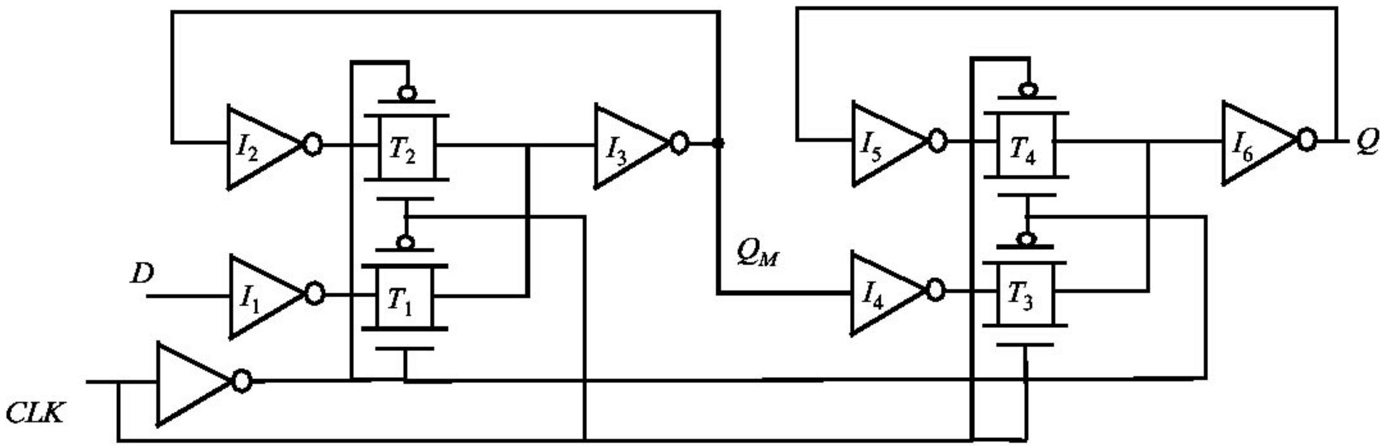
clickable



link

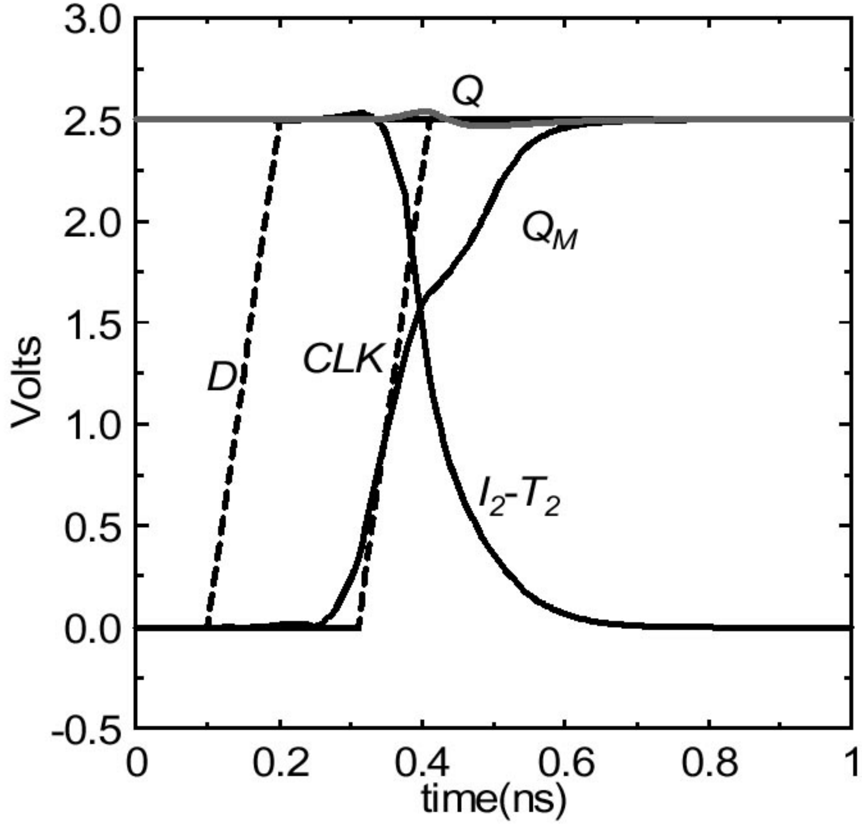
Internet

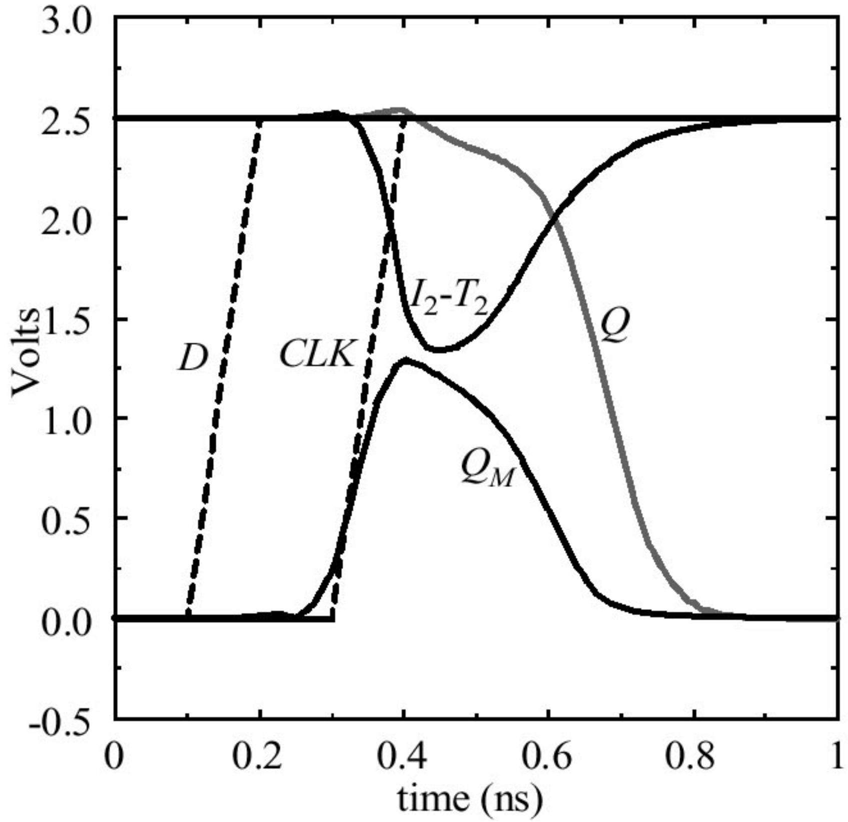
Transistor Level Master Slave Positive Edge Triggered Register



- **Robust Design**
- **Can eliminate I_1 and I_4, however, they make design more robust (avoid charge sharing, robust input)**
- **High Clock Load (8 x)**

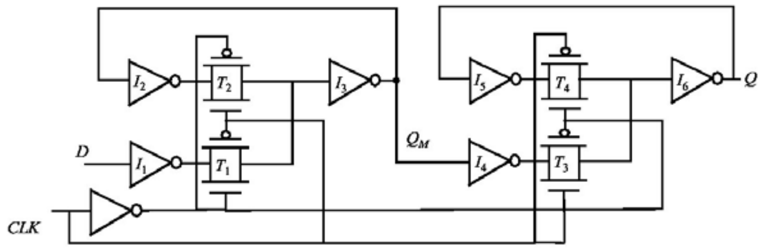
Set-up Time Simulation

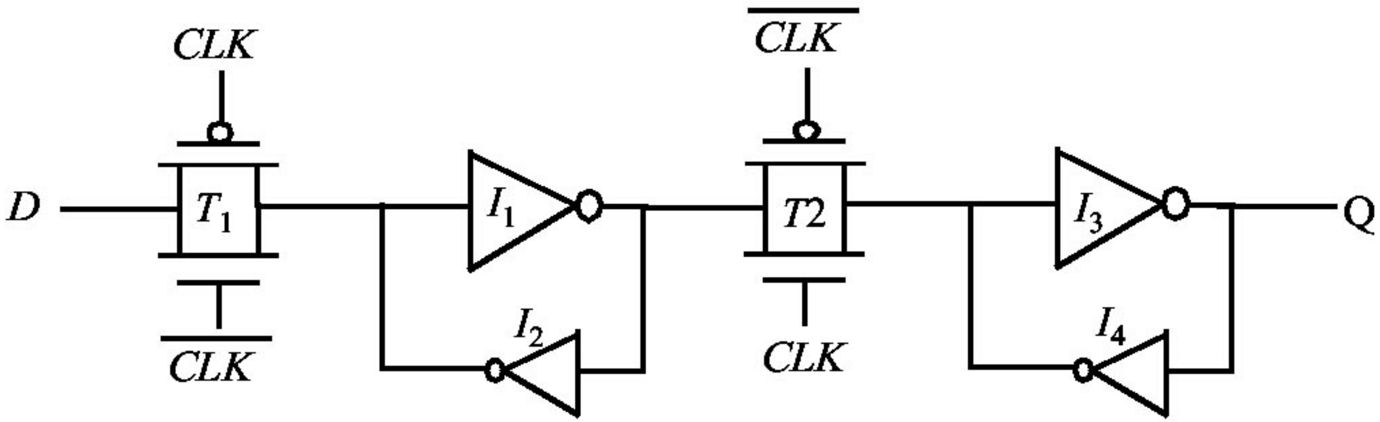




Slightly smaller delay between D and CLK

Ratioed Reduced Clock Load Register

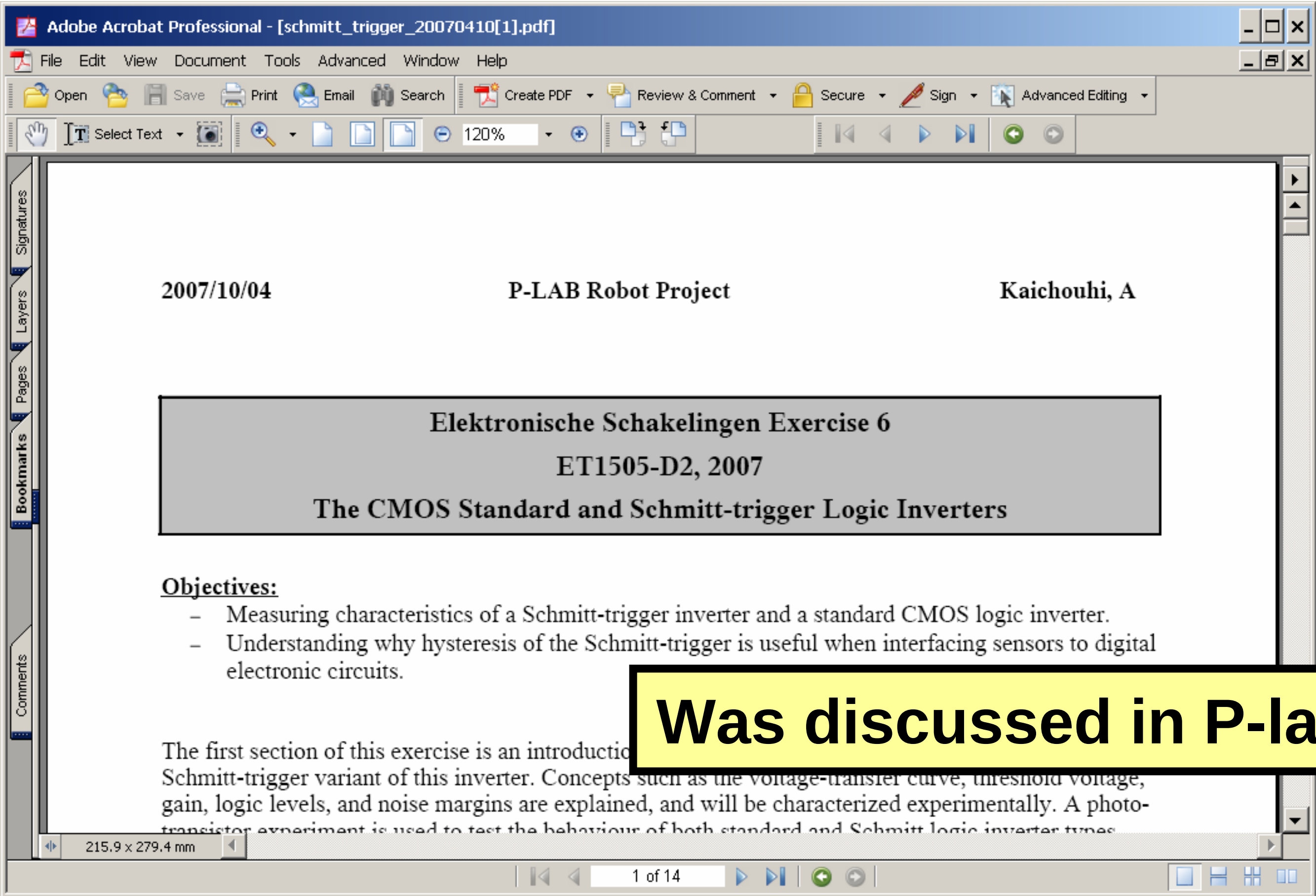




- I_2 and I_4 are small, even long
- Lower clock load
- Increased design complexity
- Reduced robustness (reverse conduction)

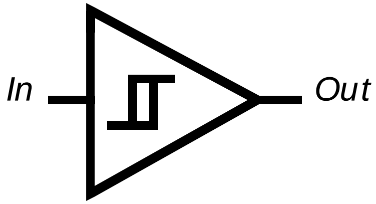
Non-bistable Elements

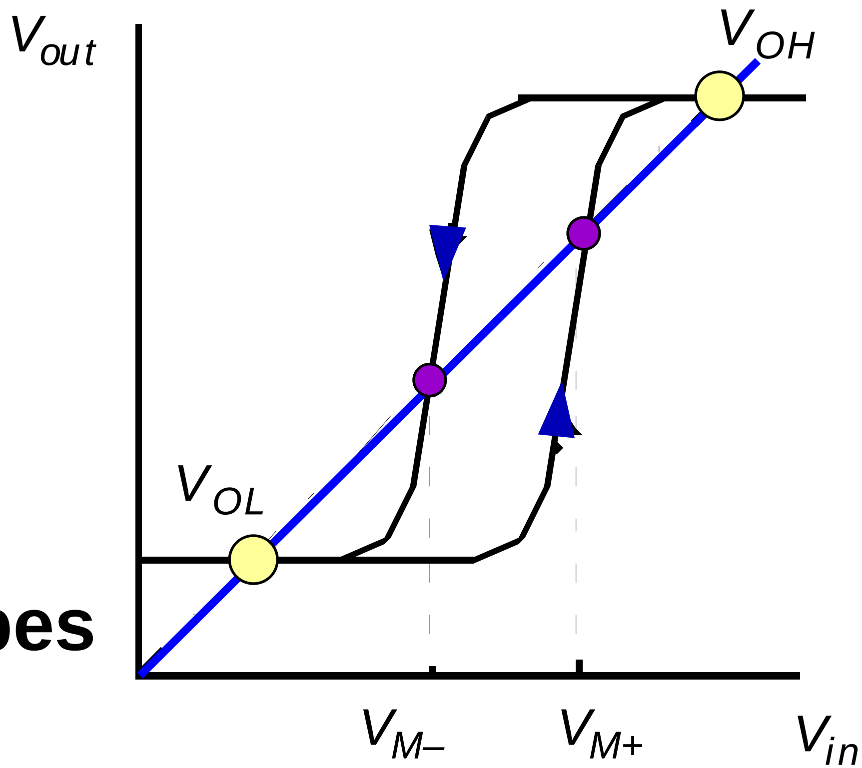
■ Schmitt Trigger



Was discussed in P-lab

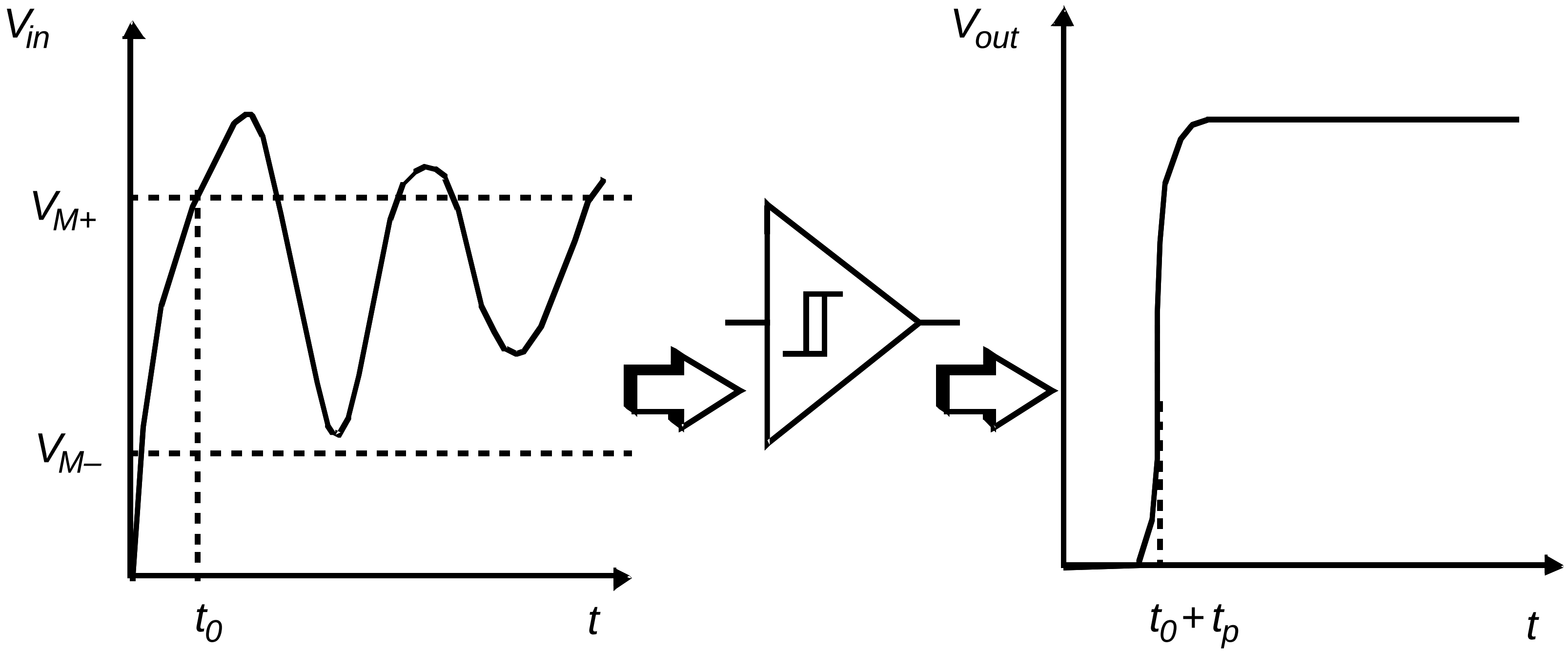
Schmitt Trigger



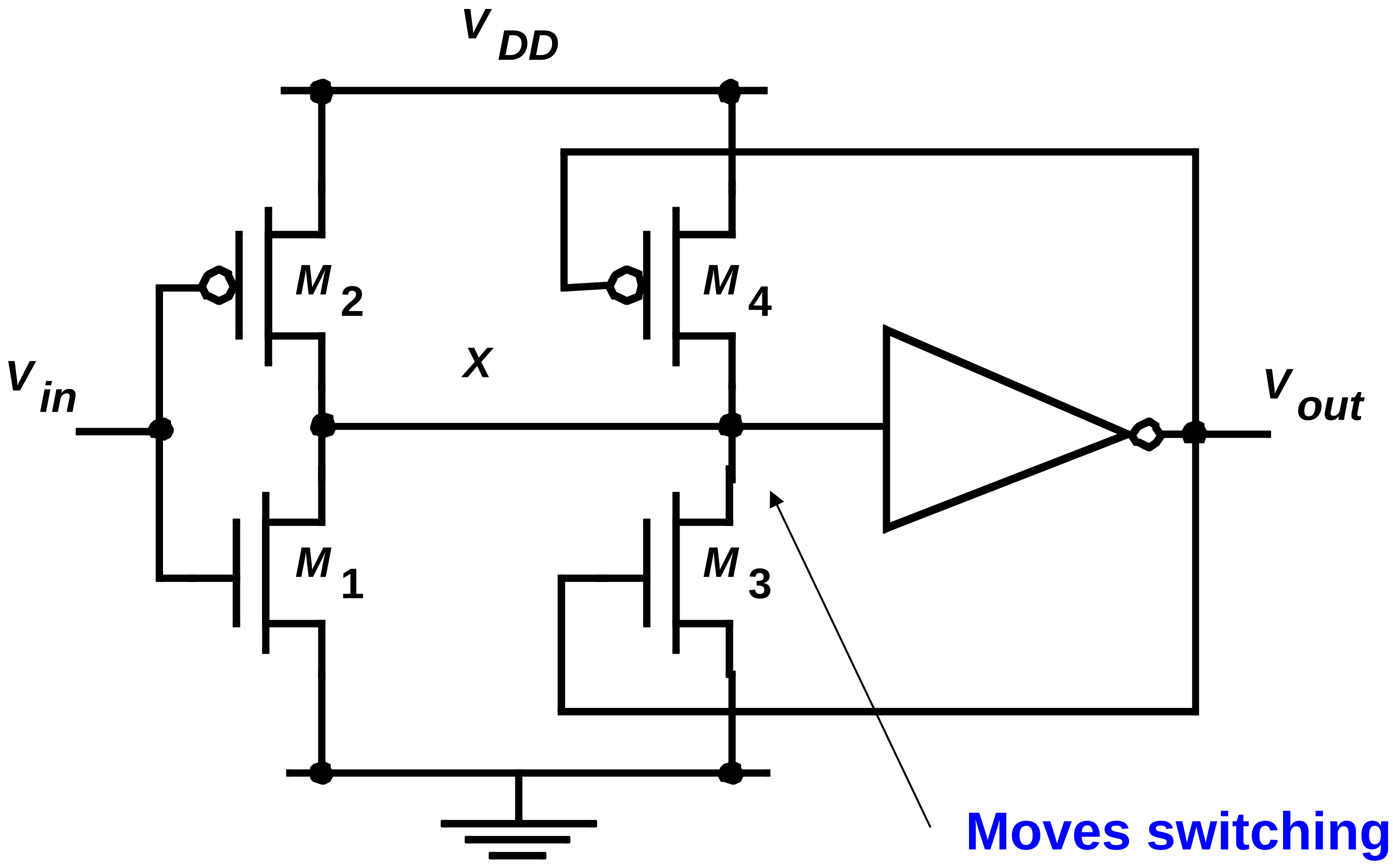


- **VTC with hysteresis**
- **Restores signal slopes**

Noise Suppression using Schmitt Trigger

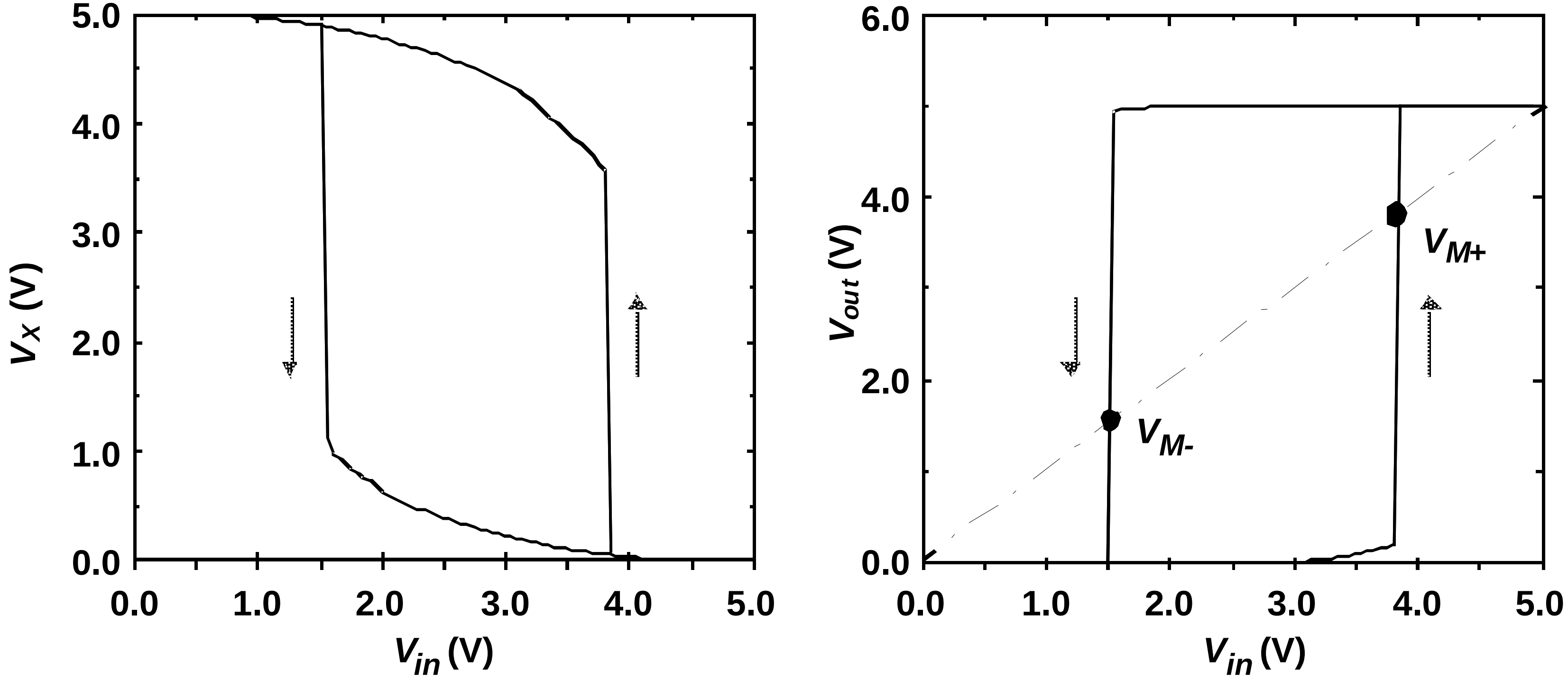


CMOS Schmitt Trigger

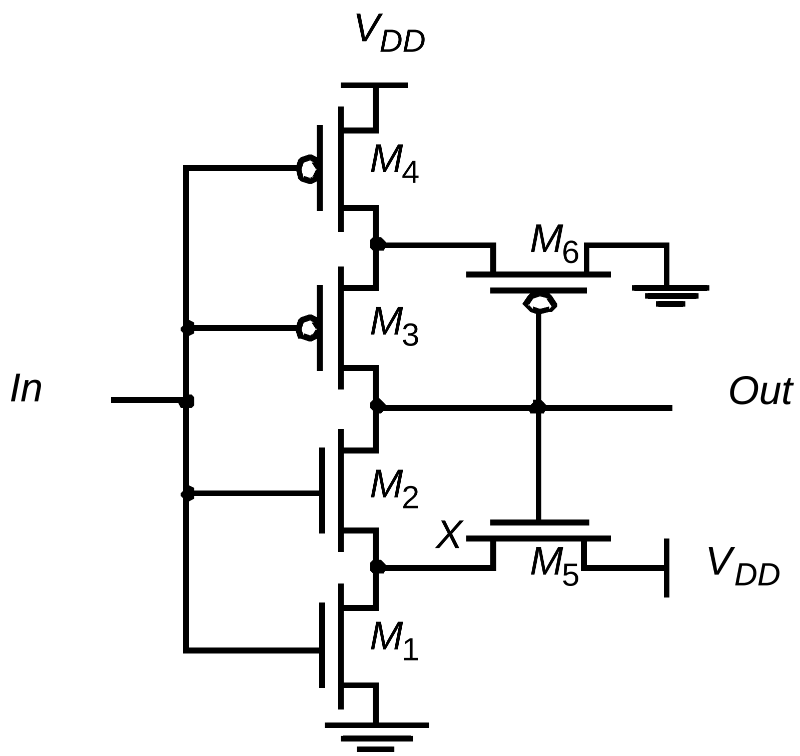


Moves switching threshold of first inverter

Schmitt Trigger Simulated VTC



CMOS Schmitt Trigger (2)



Summary

- **Background**
  - **Timing, terminology, classification**
- **Static Flipflops**
  - **Latches**
  - **Registers**
- **Dynamic Flipflops**
  - **Latches**
  - **Registers**
- **Non-bistable elements**
  - **Schmitt Trigger**