ENEE 359a *Digital VLSI Design*

Sequential Logic

Prof. Bruce Jacob

blj@eng.umd.edu

Credit where credit is due:
Slides contain original artwork (© Jacob 2004) as well as material taken liberally from Irwin & Vijay's CSE477 slides (PSU), Schmit & Strojwas's 18-322 slides (CMU), Dally's EE273 slides (Stanford), Wolf's slides for *Modern VLSI Design*, and/or Rabaey's slides (UCB).

Overview

- Transmission gates
- Basic storage-cell concepts
- Metastability
- Static latches & registers, brief primer on dynamic logic, dynamic latches & registers, trading off complexity in individual memory elements vs. complexity in clock-delivery network
- Pipelining

Bruce Jacob

University of Maryland
ECE Dept.

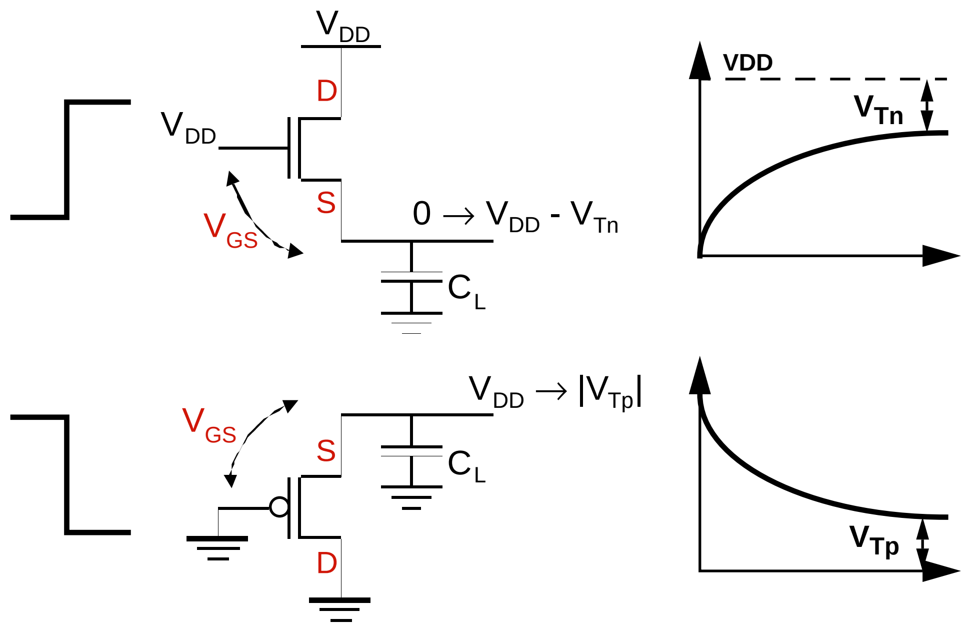
The Transmission Gate

Basic idea: reduce number of transistors in design by allowing inputs to drive not only *gate* terminals but also *source/drain* terminals.

(similar in this regard to *pass-transistor* logic)

Recall:

- **NMOS will not pass a “1”**
- **PMOS will not pass a “0”**



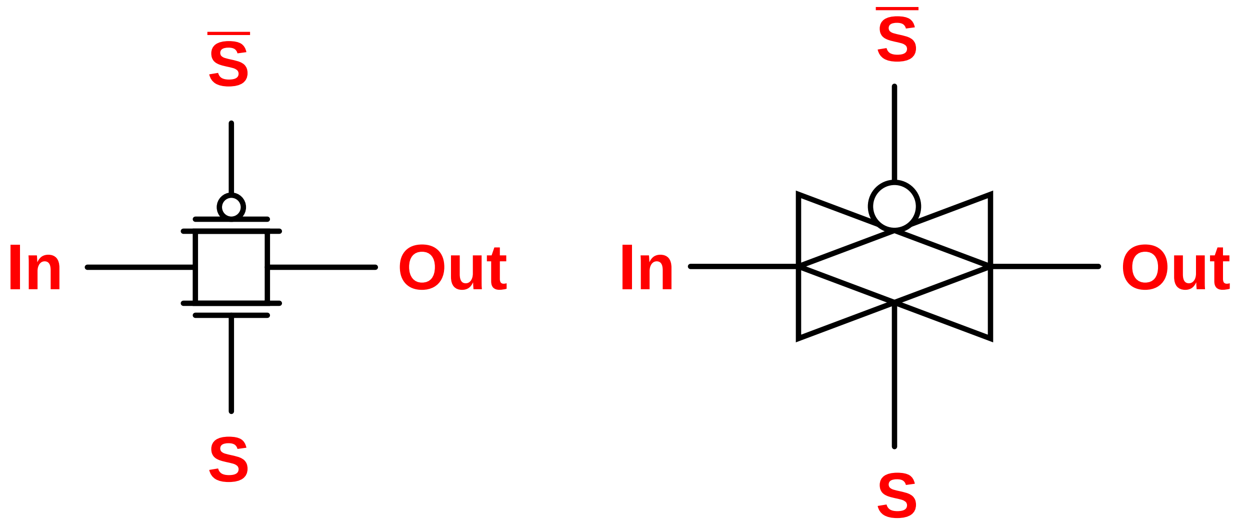
- **Body effect ($V_{BS} \neq 0$) causes $V_{Tn/p}$ to increase**

Bruce Jacob

University of Maryland
ECE Dept.

The Transmission Gate

Solution: use both



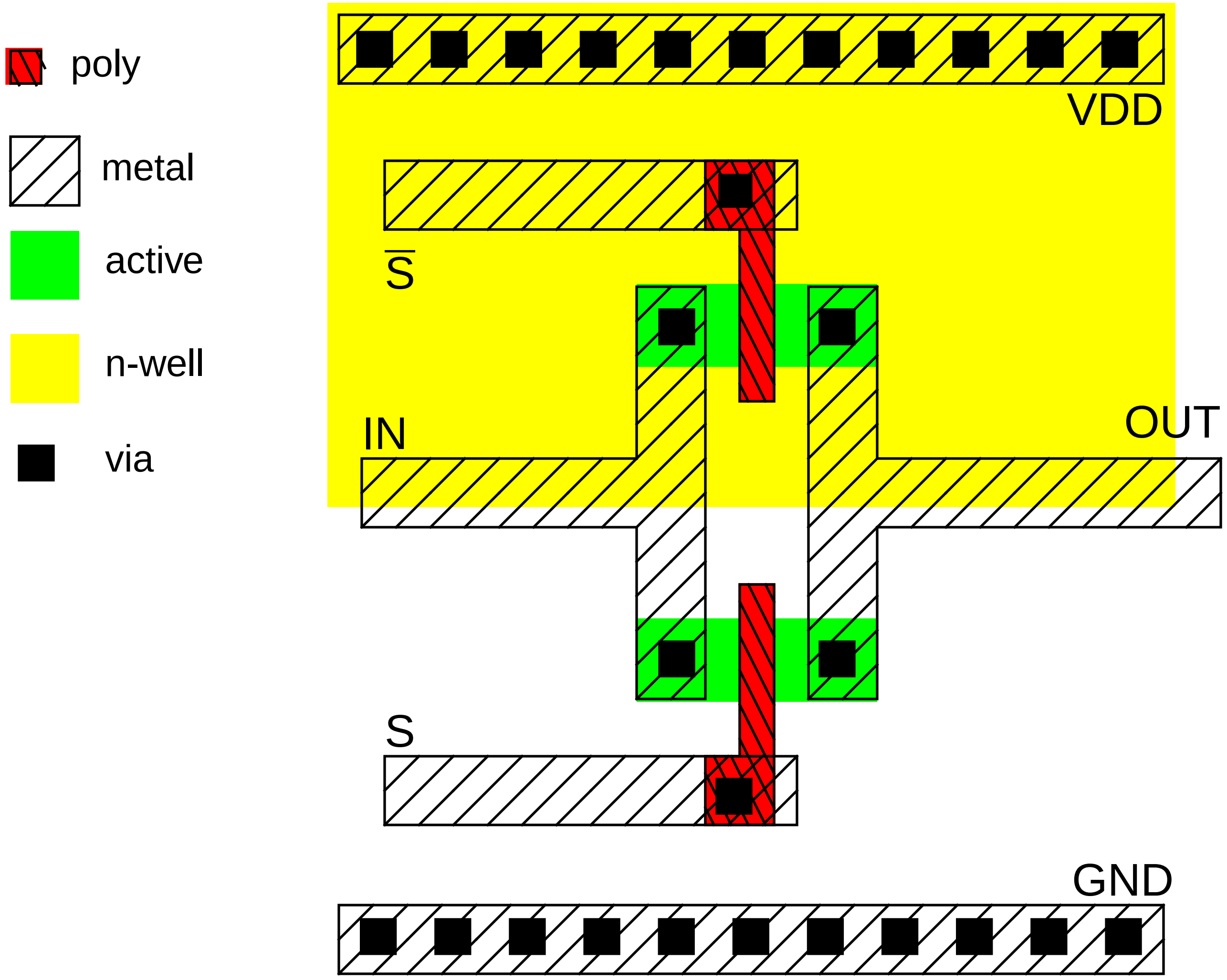
Out = (S) ? In : Z;

Potential problem: unlike other static logic gates, this can allow output to be *floating* (not connected — thus, easily perturbed by nearby signals such as metal wires above)

Bruce Jacob

University of
Maryland
ECE Dept.

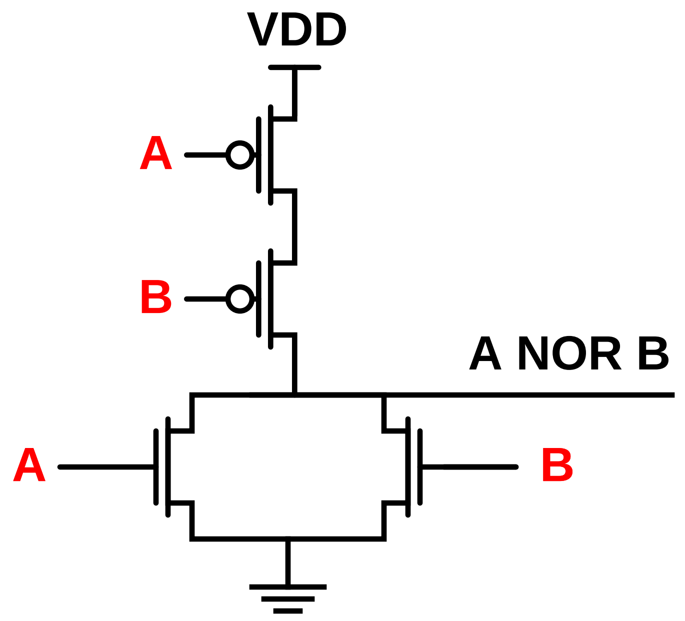
The Transmission Gate



Sometimes standard cell includes S inverter

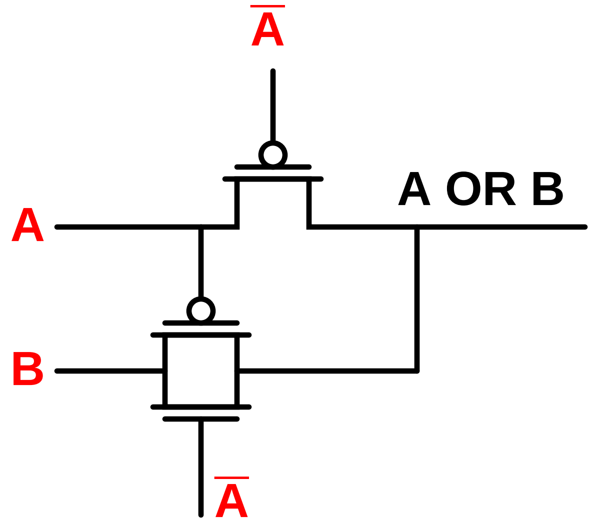
UNIVERSITY OF MARYLAND

The Transmission Gate

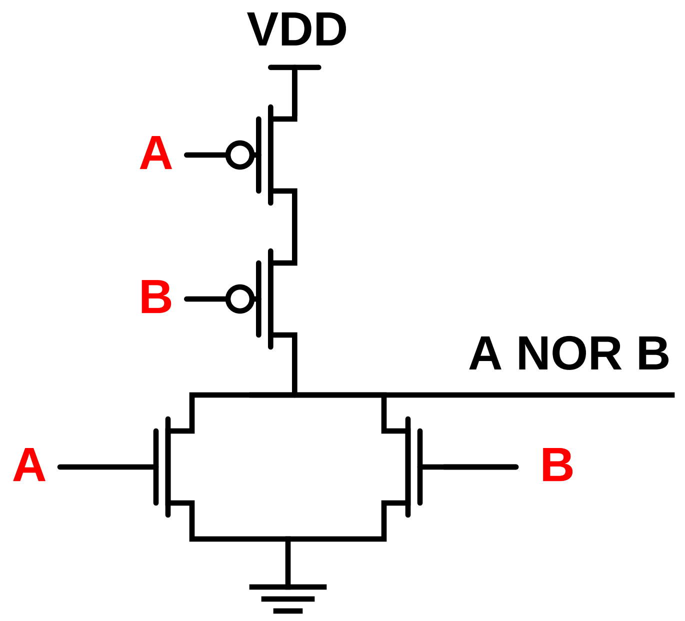


Complementary Implementation

TG Implementation

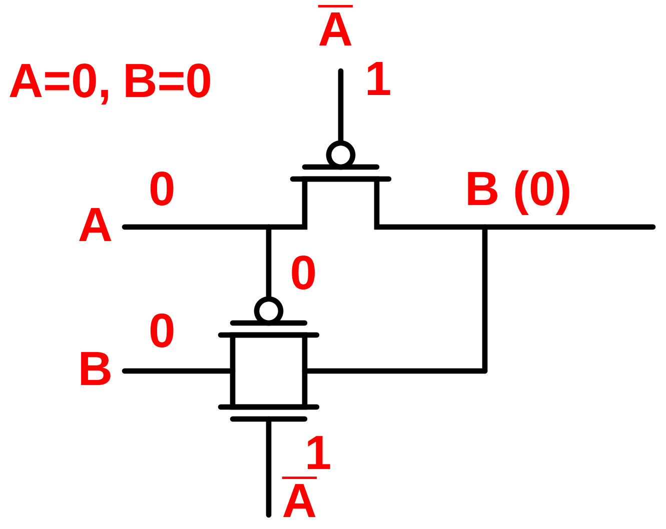


The Transmission Gate

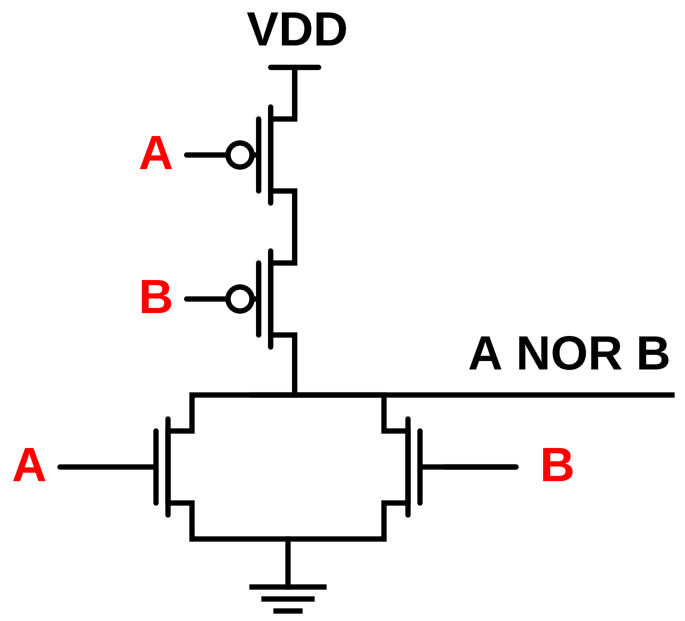


Complementary
Implementation

TG Implementation

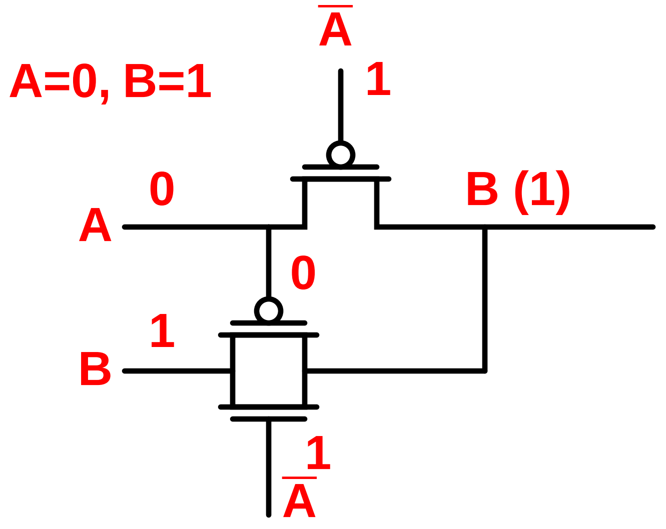


The Transmission Gate



Complementary
Implementation

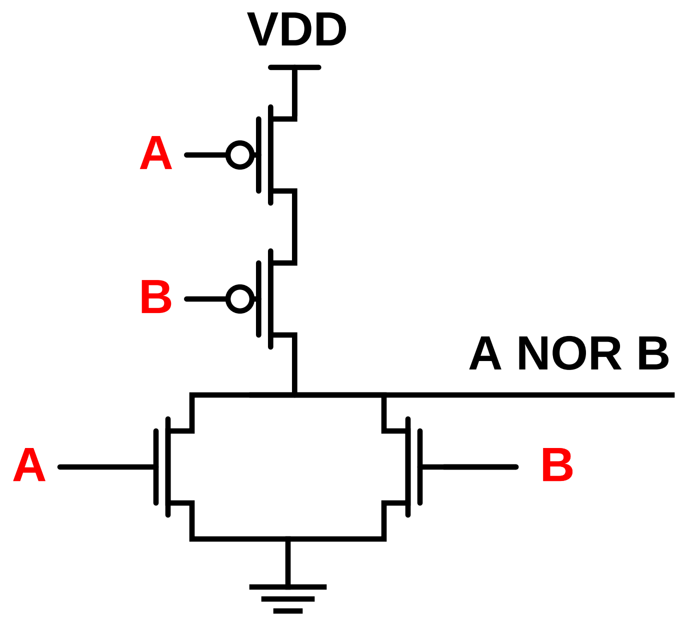
TG Implementation



Bruce Jacob

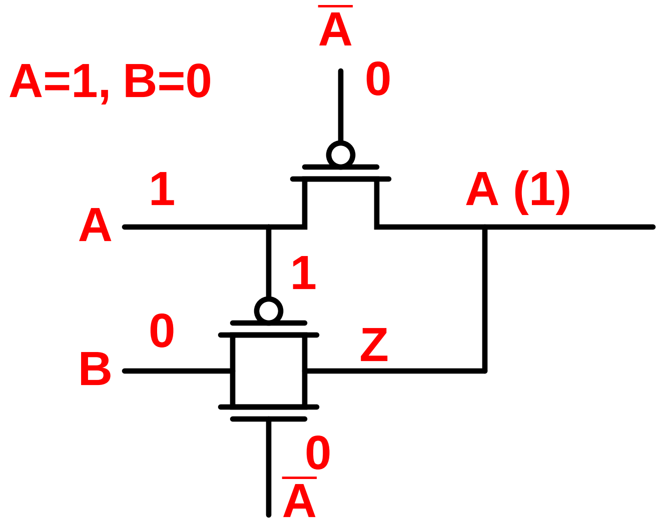
University of Maryland
ECE Dept.

The Transmission Gate

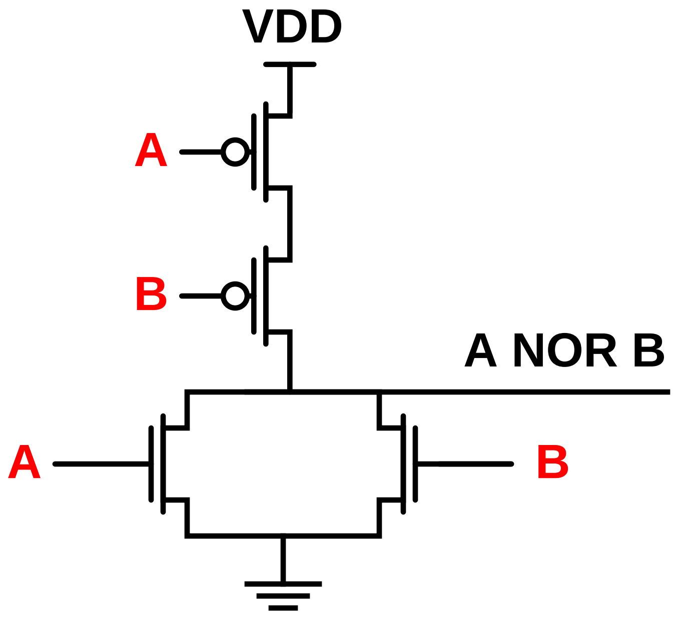


Complementary Implementation

TG Implementation

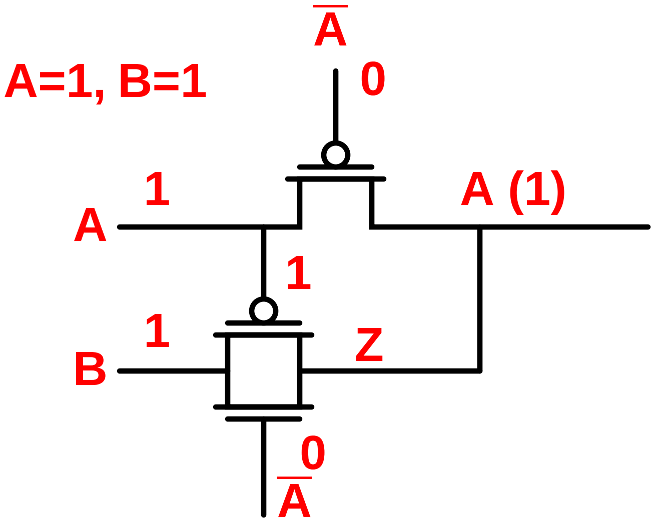


The Transmission Gate



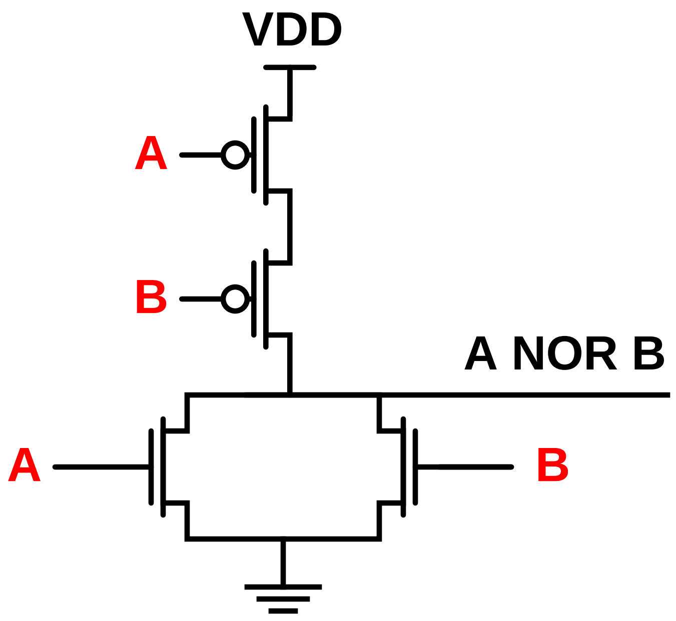
Complementary Implementation

TG Implementation



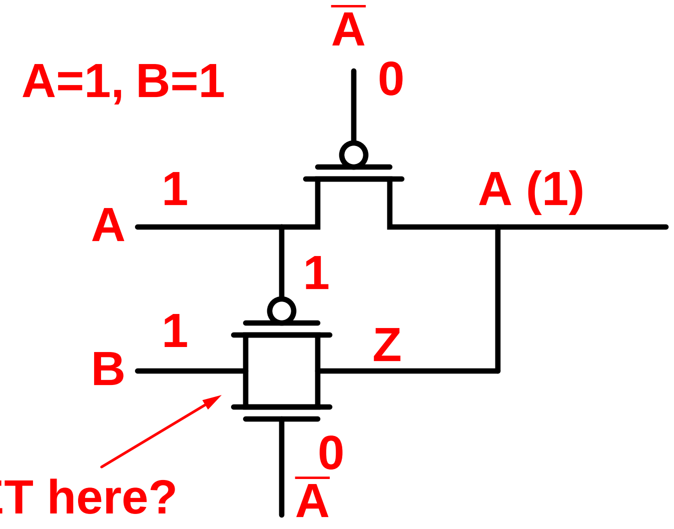
UNIVERSITY OF MARYLAND

The Transmission Gate



Complementary Implementation

TG Implementation

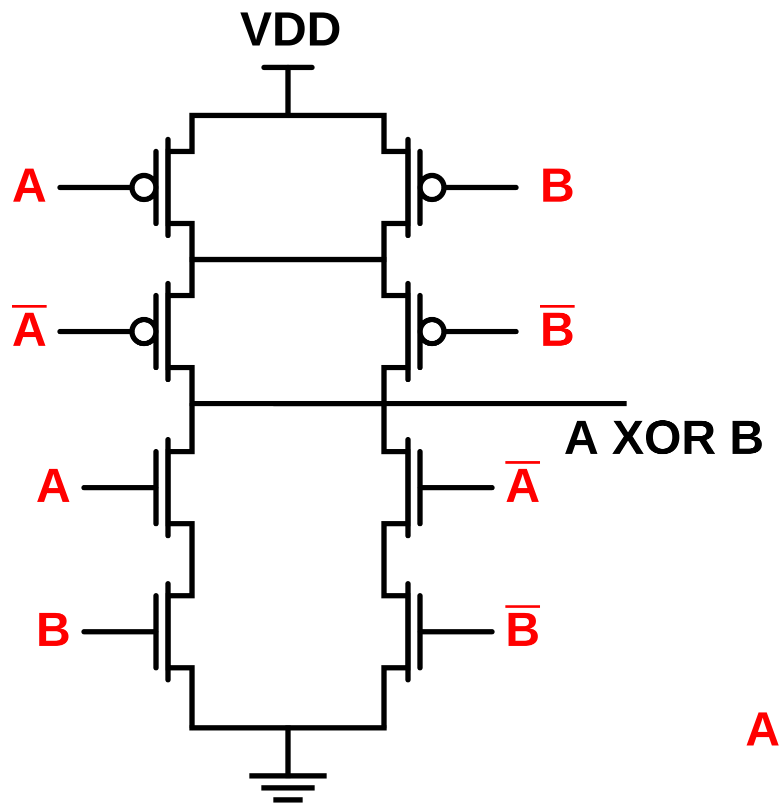


Q: why can't we just use nFET here?

Bruce Jacob

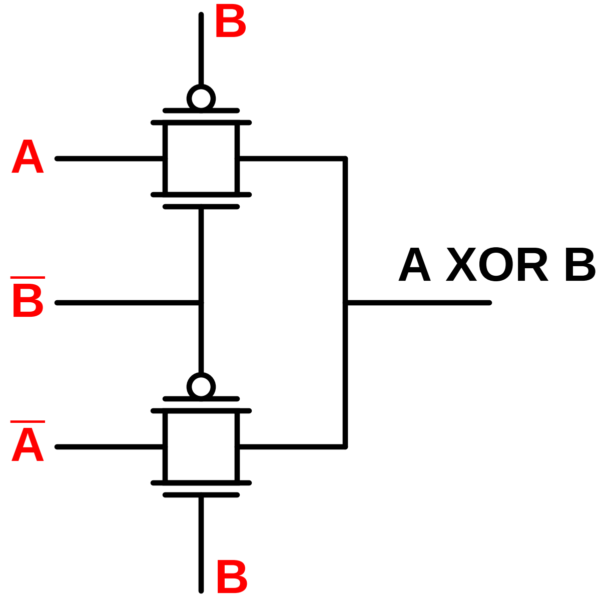
University of
Maryland
ECE Dept.

The Transmission Gate



Complementary
Implementation

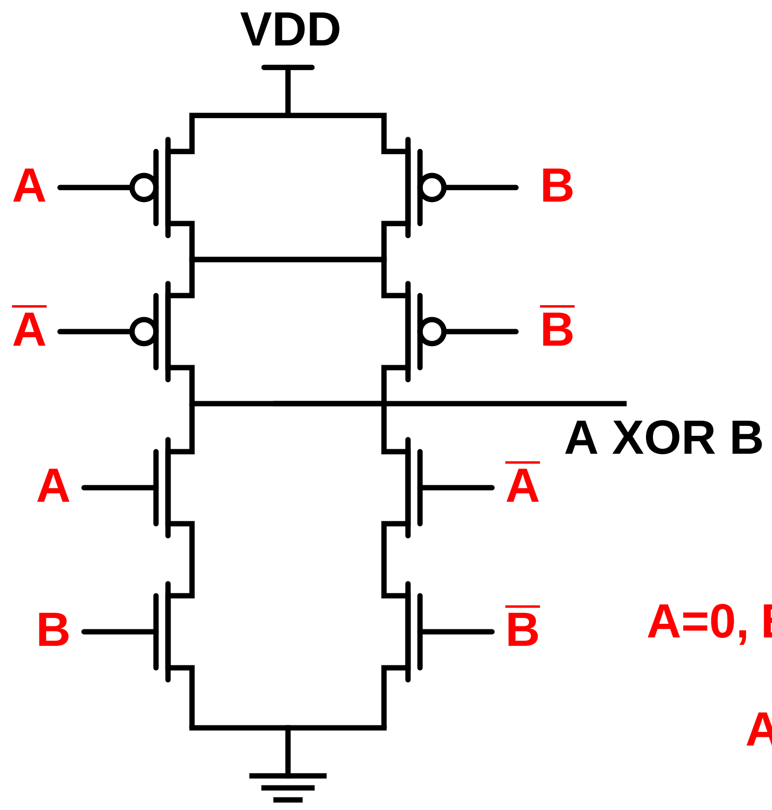
TG Implementation



Bruce Jacob

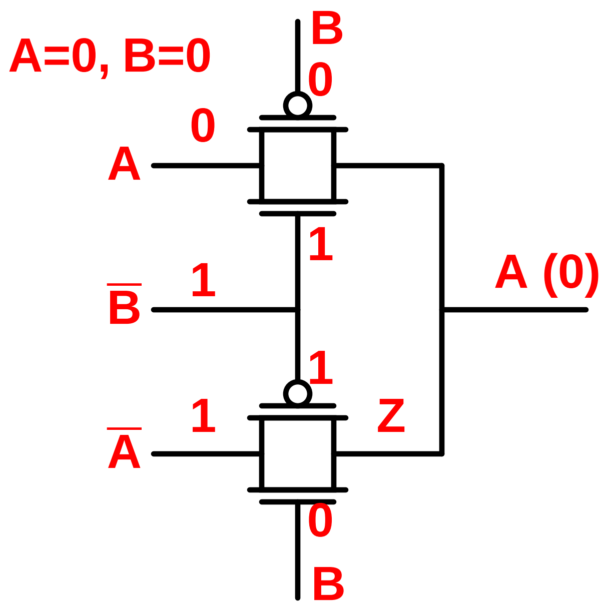
University of
Maryland
ECE Dept.

The Transmission Gate



Complementary
Implementation

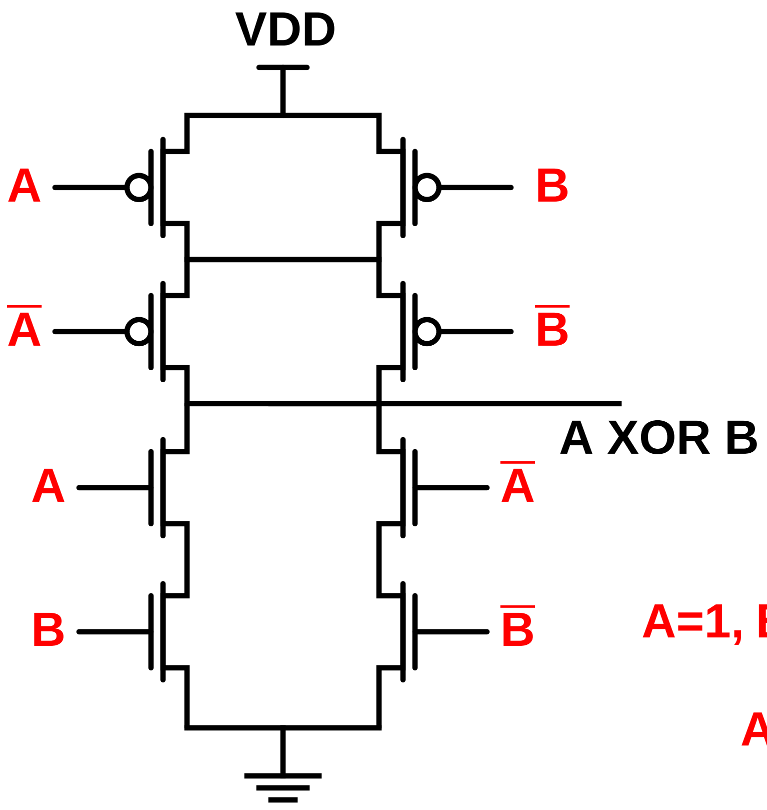
TG Implementation



Bruce Jacob

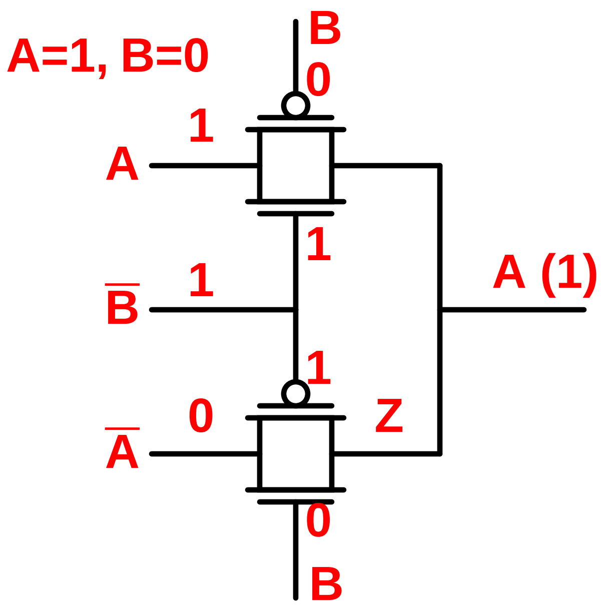
University of
Maryland
ECE Dept.

The Transmission Gate



Complementary
Implementation

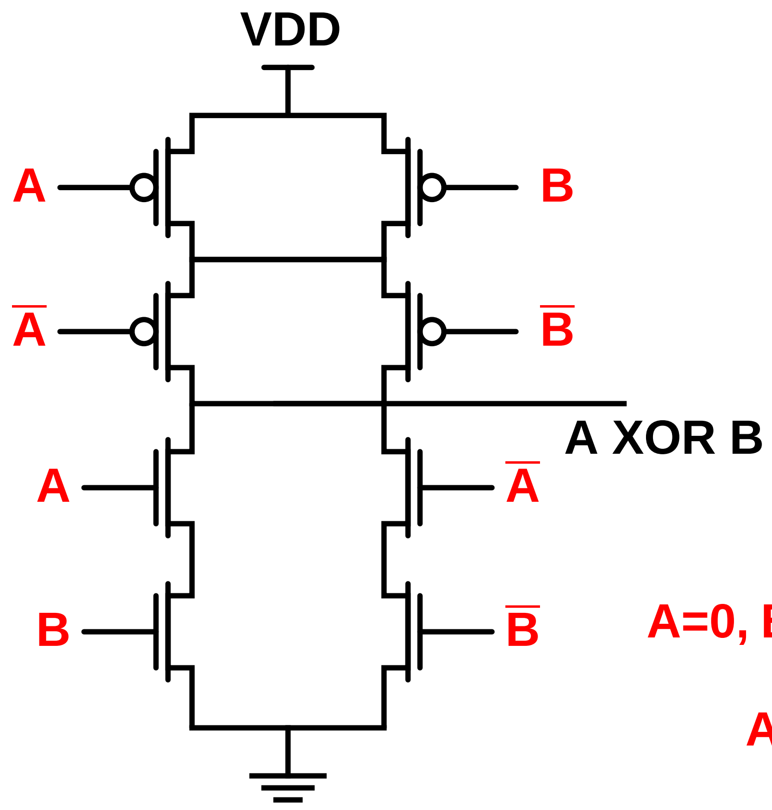
TG Implementation



Bruce Jacob

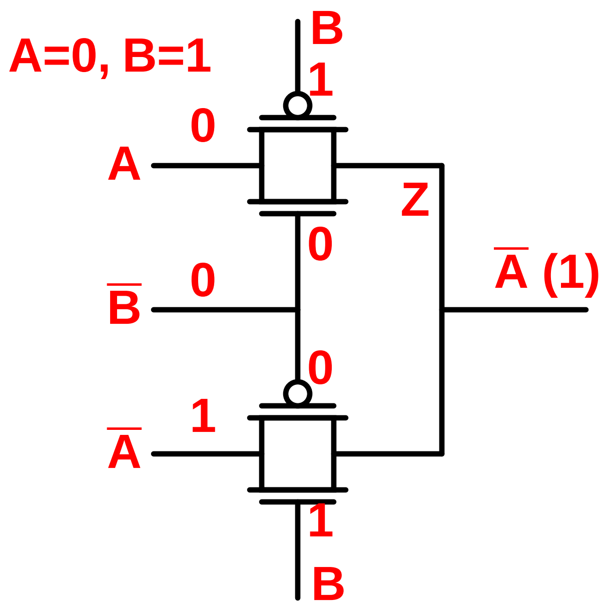
University of
Maryland
ECE Dept.

The Transmission Gate



Complementary
Implementation

TG Implementation

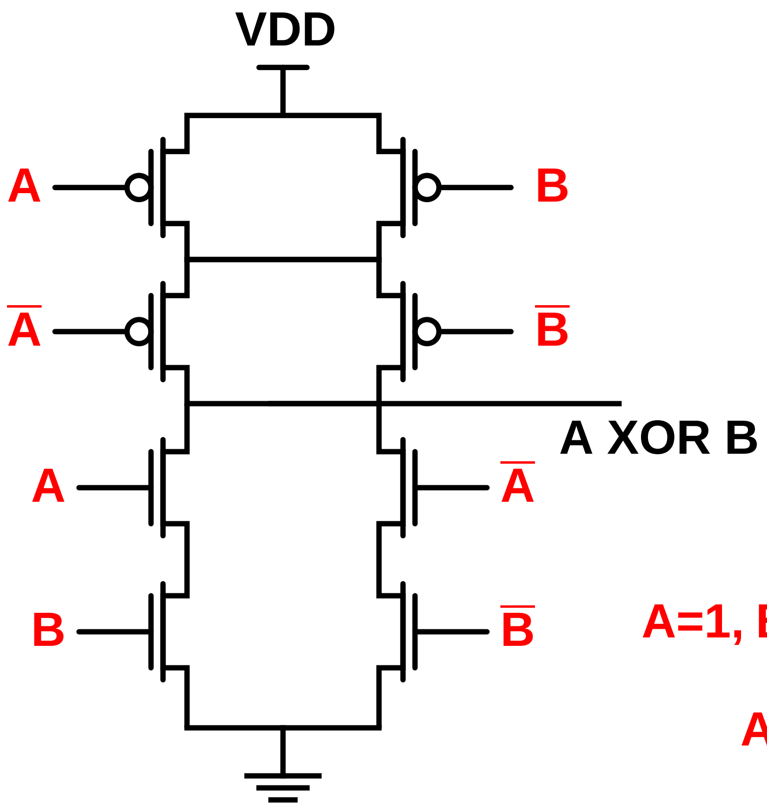


Bruce Jacob

University of
Maryland
ECE Dept.

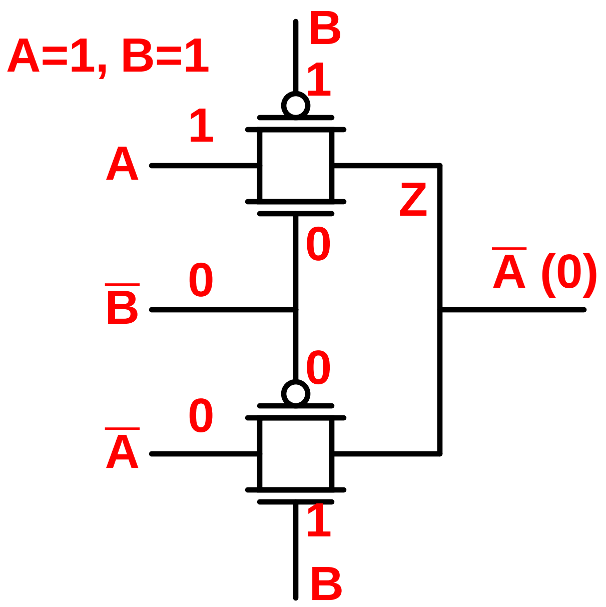
UNIVERSITY OF MARYLAND

The Transmission Gate



Complementary
Implementation

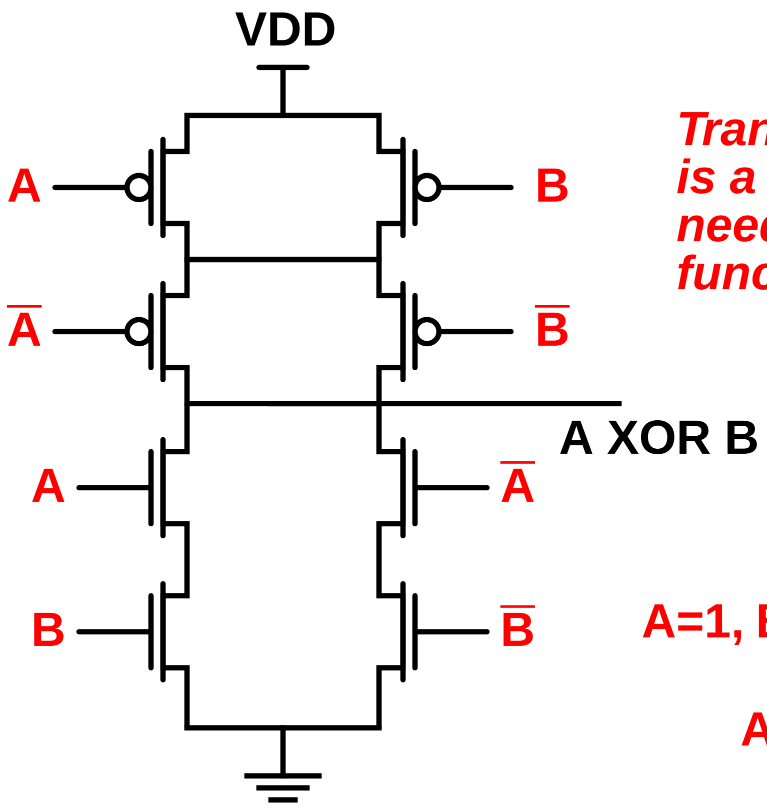
TG Implementation



Bruce Jacob

University of
Maryland
ECE Dept.

The Transmission Gate

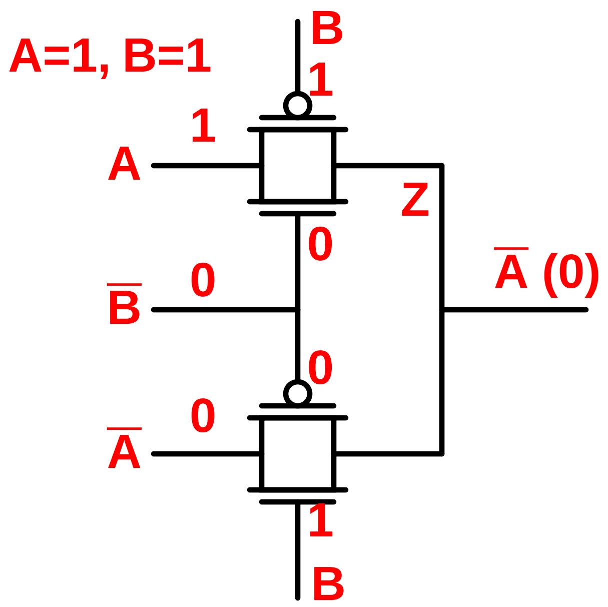


Complementary Implementation

Transmission-gate logic is a huge win for circuits that need XOR/XNOR and similar functions, e.g. full adders

XOR = A$\overline{B}$ + $\overline{A}$B

TG Implementation



Basic storage-cell concepts

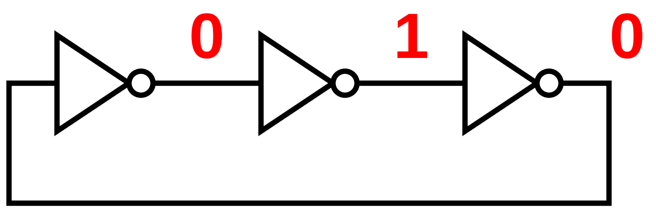
DEFINITIONS (from the book):

1. **Flip-flop: bistable component built by cross-coupling logic gates**
2. **Latch: *level-sensistive* storage element**
3. **Register: *edge-triggered* storage element**

**(the literature on the topic is not consistent in its use of these terms,
so beware and try to figure out from context what someone means …)**

Basic storage-cell concepts

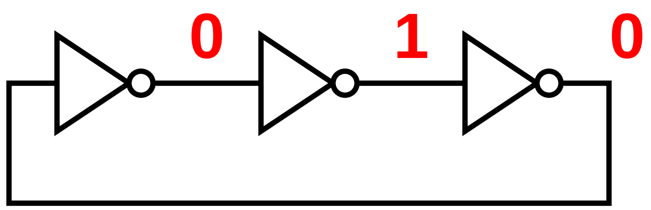
Bistable? Only two stable points on VTC



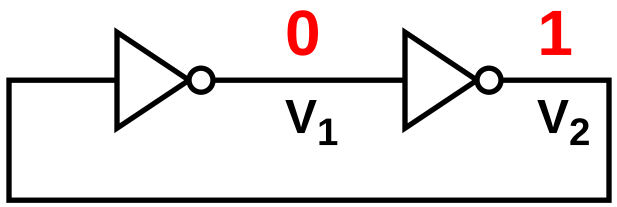
… what happens next?

Basic storage-cell concepts

Bistable? Only two stable points on VTC



… what happens next?



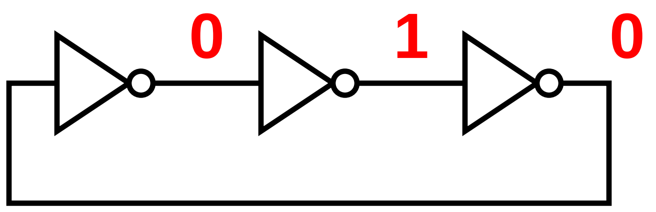
… what happens next?

Bruce Jacob

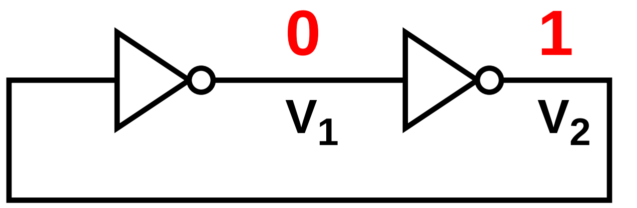
University of
Maryland
ECE Dept.

Basic storage-cell concepts

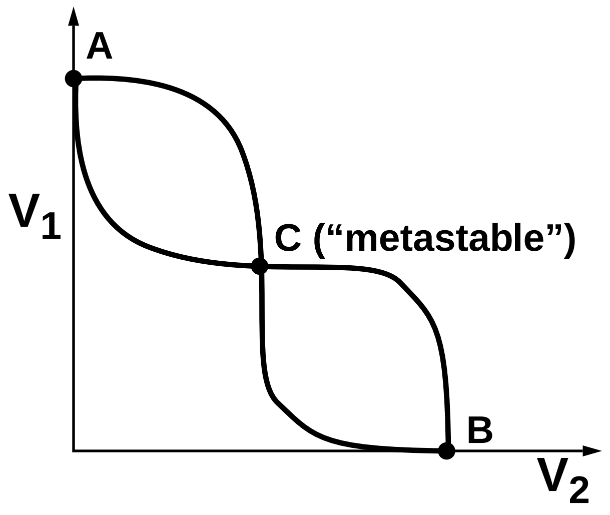
Bistable? Only two stable points on VTC



… what happens next?



this is a stable circuit
(and, in particular, bistable)



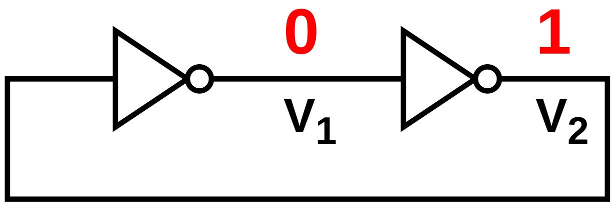
Bruce Jacob

University of Maryland
ECE Dept.

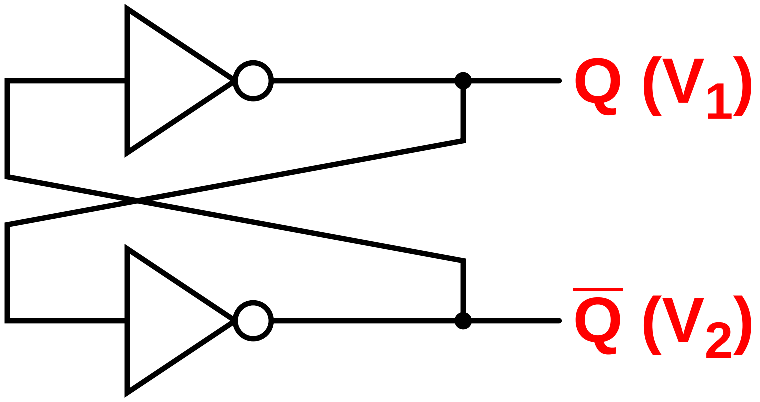
Basic storage-cell concepts

Cross-coupling? Memory as feedback

This:



Equals this:



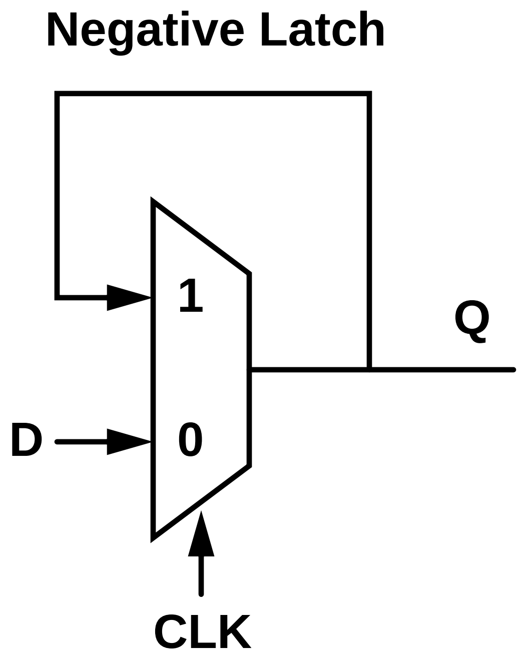
Question: how do we write a new value?

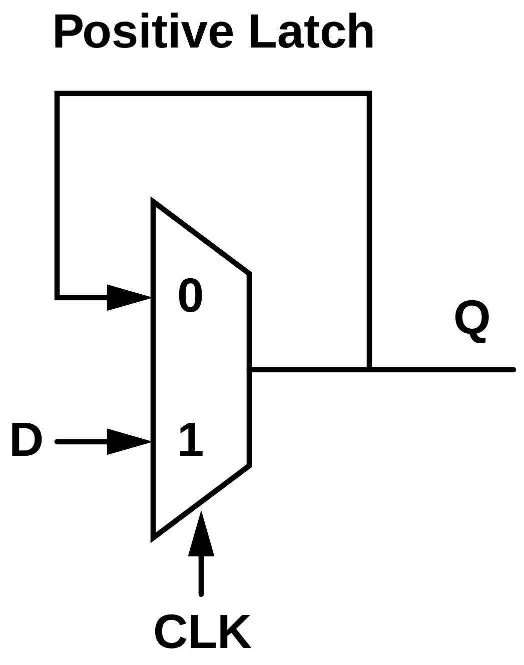
- Cutting feedback loop (multiplexer-based)
- Overpowering feedback loop

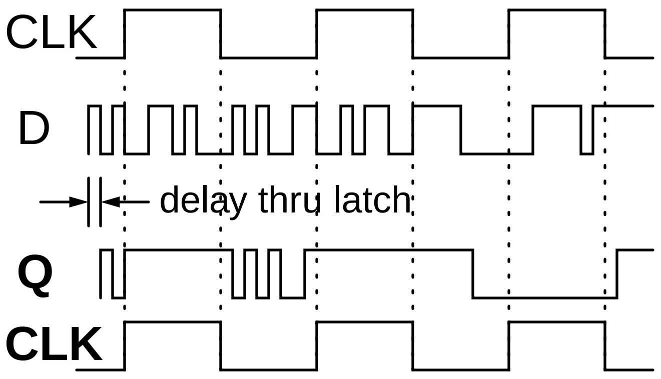
Bruce Jacob

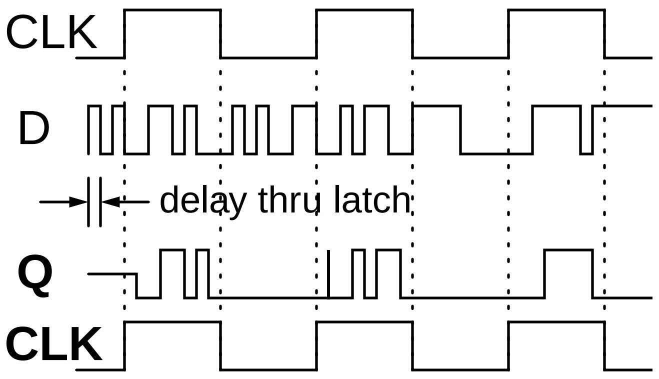
University of
Maryland
ECE Dept.

Multiplexer-Based Latches





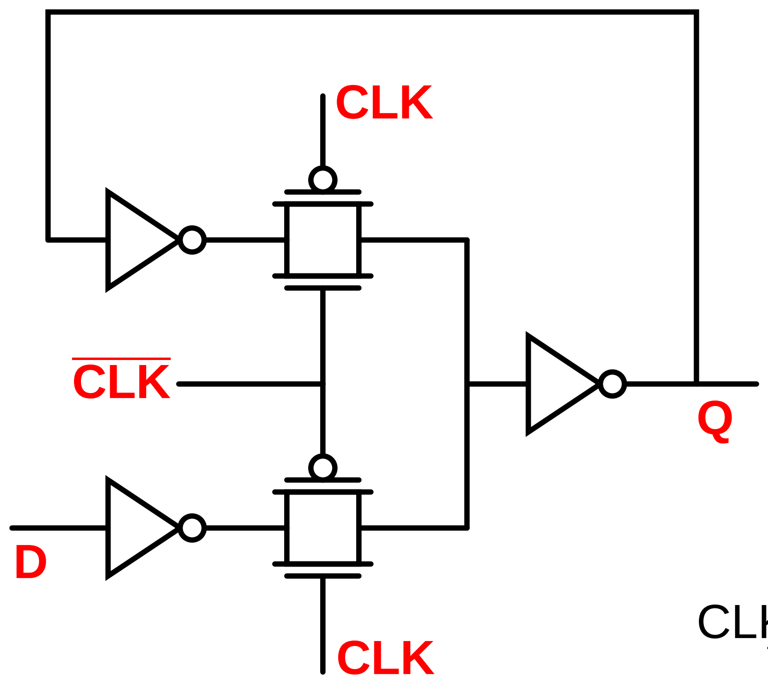




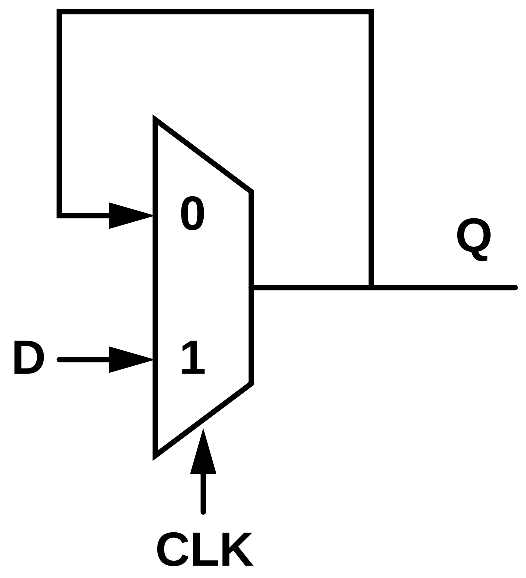
Bruce Jacob

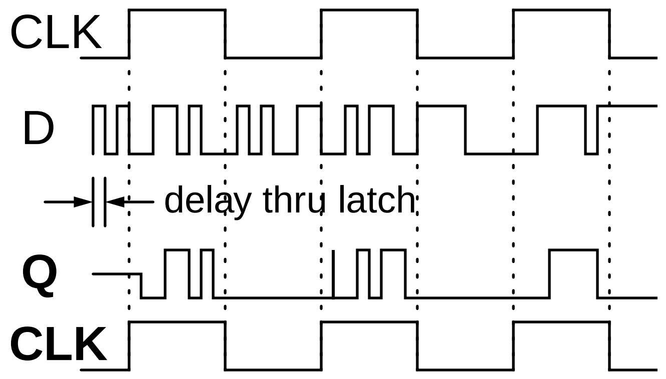
University of
Maryland
ECE Dept.

Multiplexer-Based Latches



Positive Latch





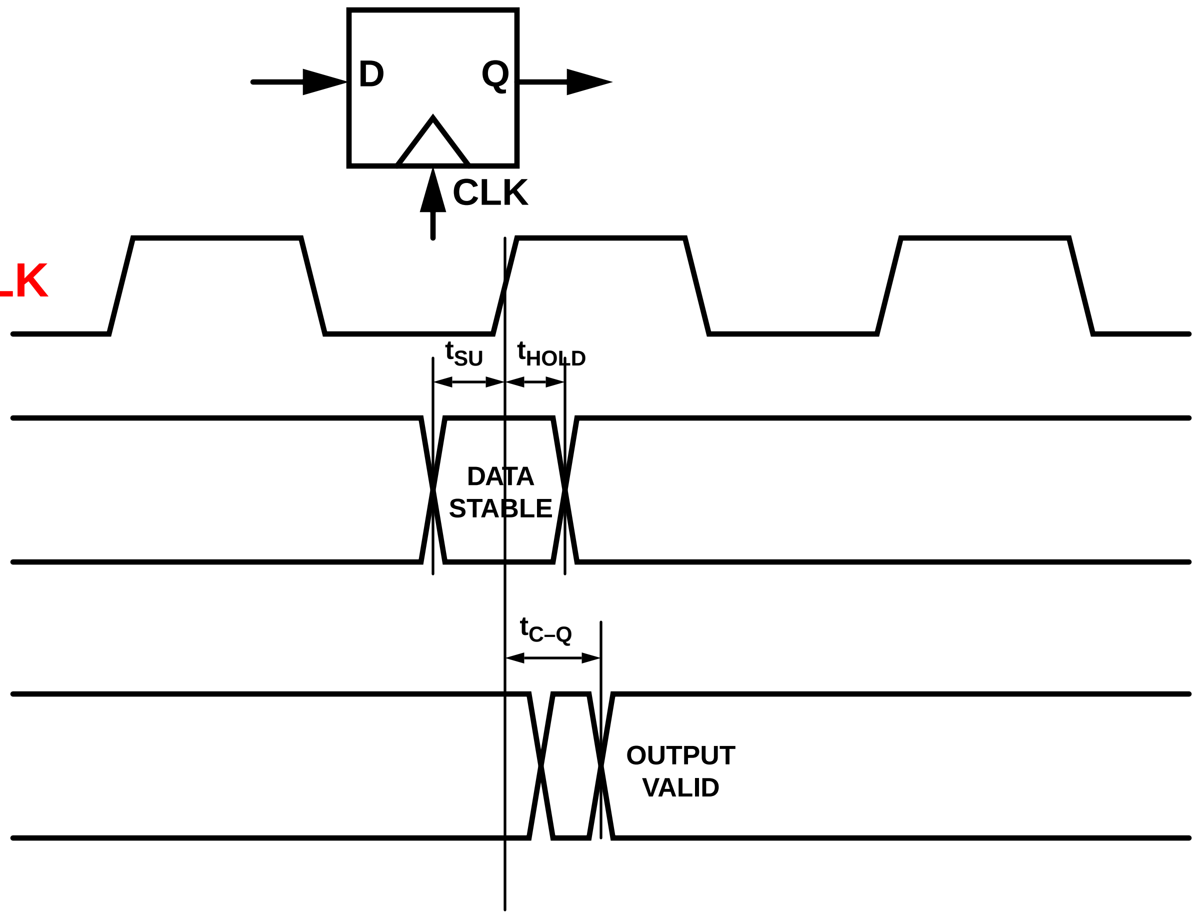
UNIVERSITY OF MARYLAND

Bruce Jacob

University of
Maryland
ECE Dept.

Metastability

SET-UP and HOLD times



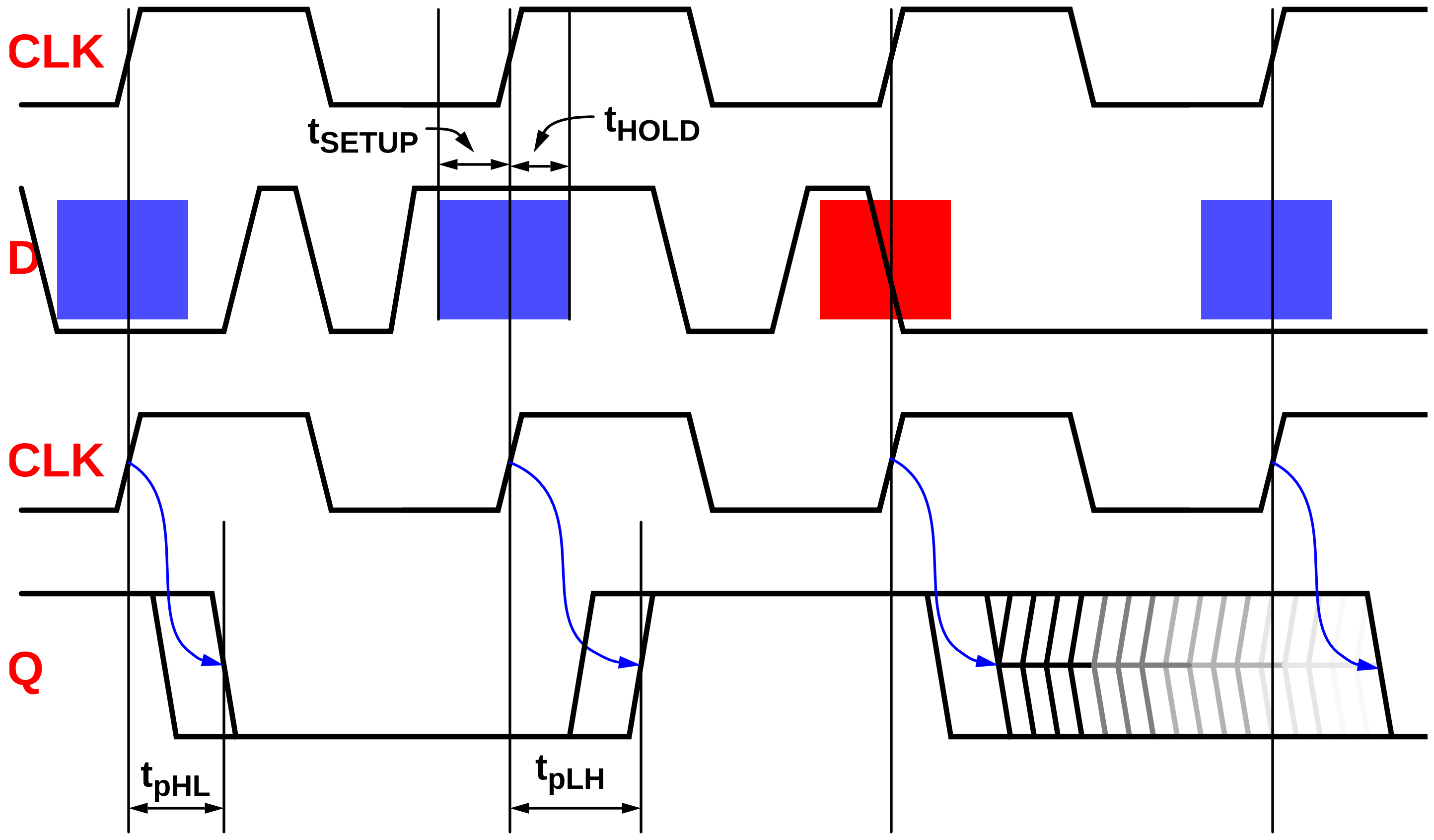
Bruce Jacob

University of
Maryland
ECE Dept.

UNIVERSITY OF MARYLAND

Metastability

SET-UP and HOLD time violations

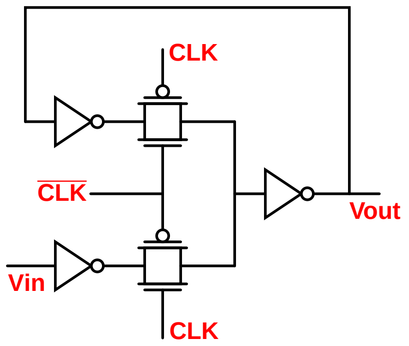


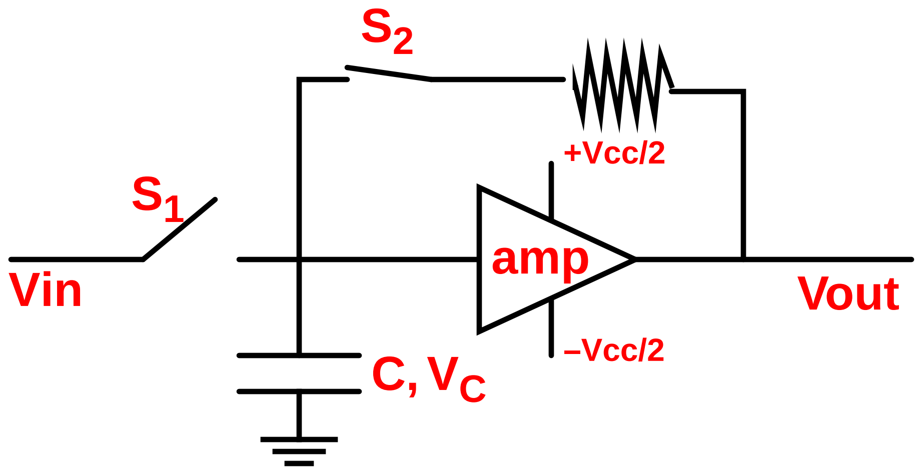
Bruce Jacob

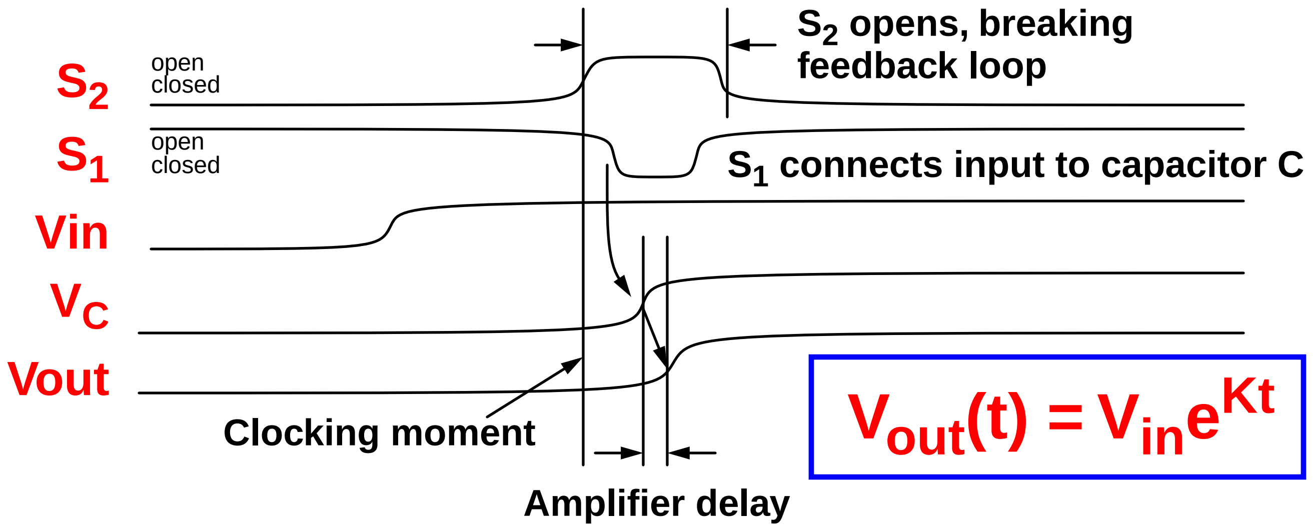
University of
Maryland
ECE Dept.

Metastability

Intuition behind the behavior







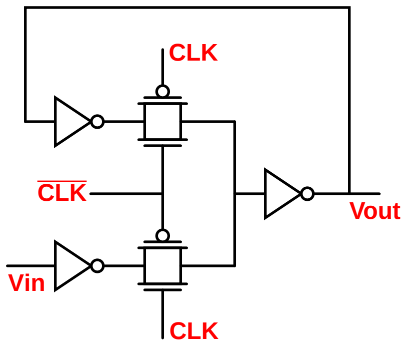
$$\boxed{V_{out}(t) = V_{in}e^{Kt}}$$

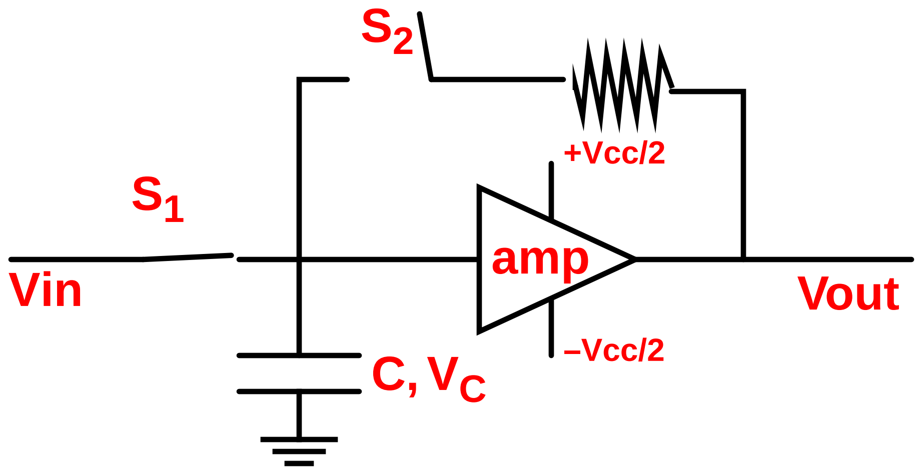
Bruce Jacob

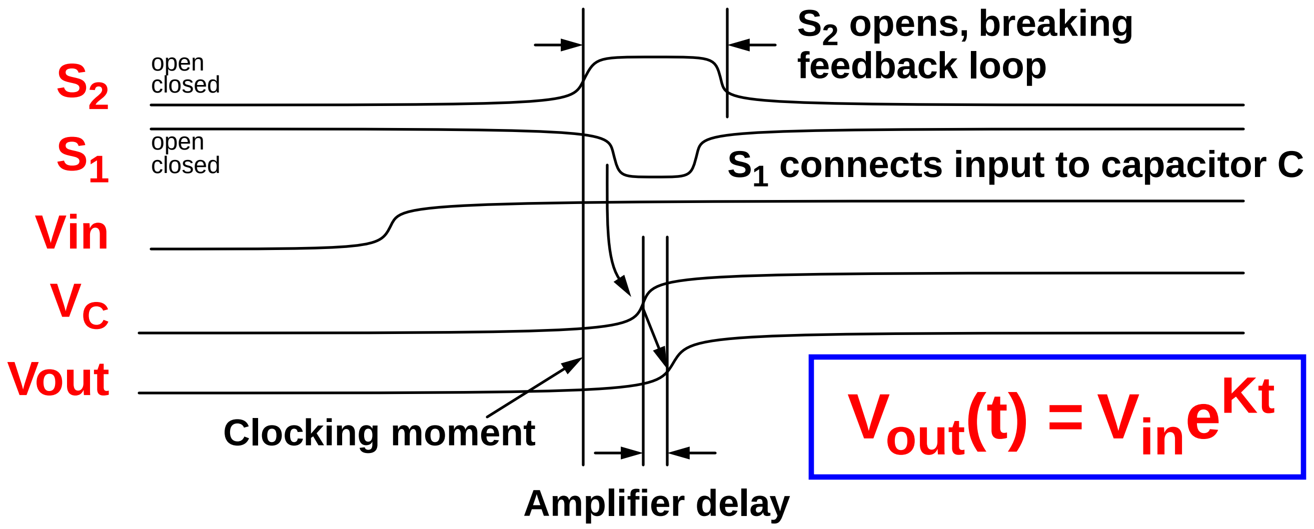
University of
Maryland
ECE Dept.

Metastability

Intuition behind the behavior



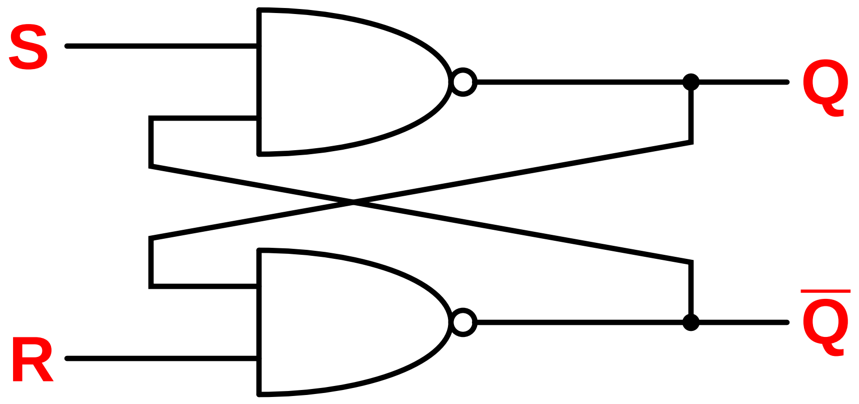




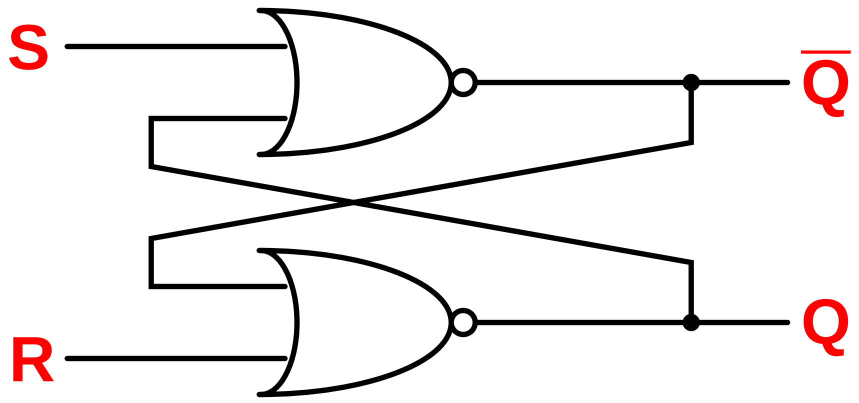
$$\boxed{V_{out}(t) = V_{in}e^{Kt}}$$

Static latches & registers

Set-Reset Flip-Flop (two types)



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



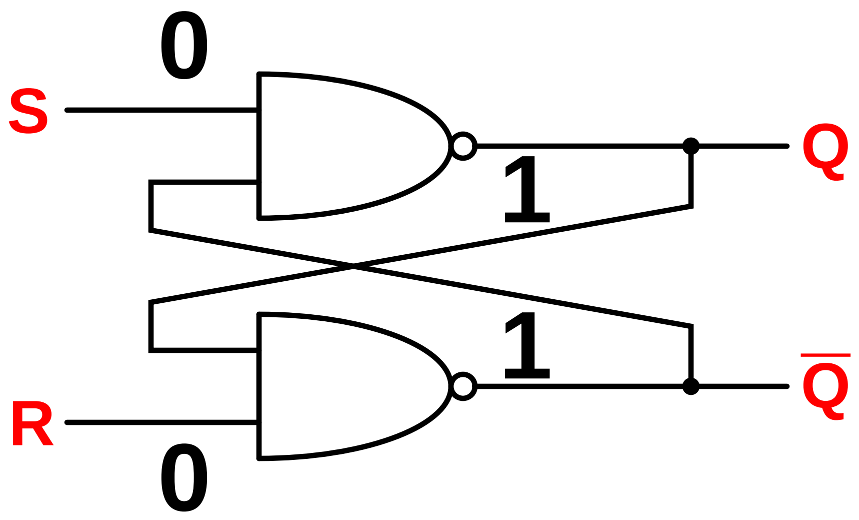
S	R	Q	$\overline{Q}$
0	0	Q	$\overline{Q}$
1	0	1	0
0	1	0	1
1	1	0	0

Bruce Jacob

University of
Maryland
ECE Dept.

Static latches & registers

NAND-based SR Flip-Flop

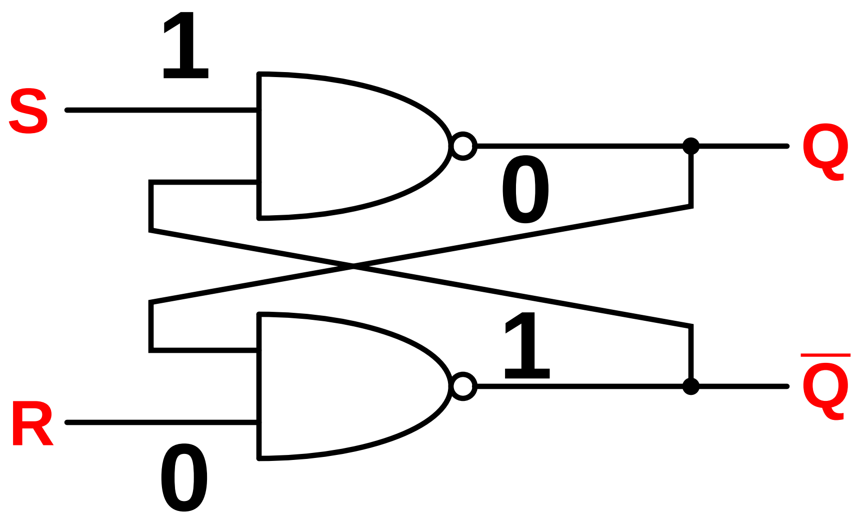


S	R	Q	$\overline{Q}$
0	0	1	1
1	0	0	1
0	1	1	0
1	1	Q	$\overline{Q}$

“forbidden” state, Q == $\overline{Q}$

Static latches & registers

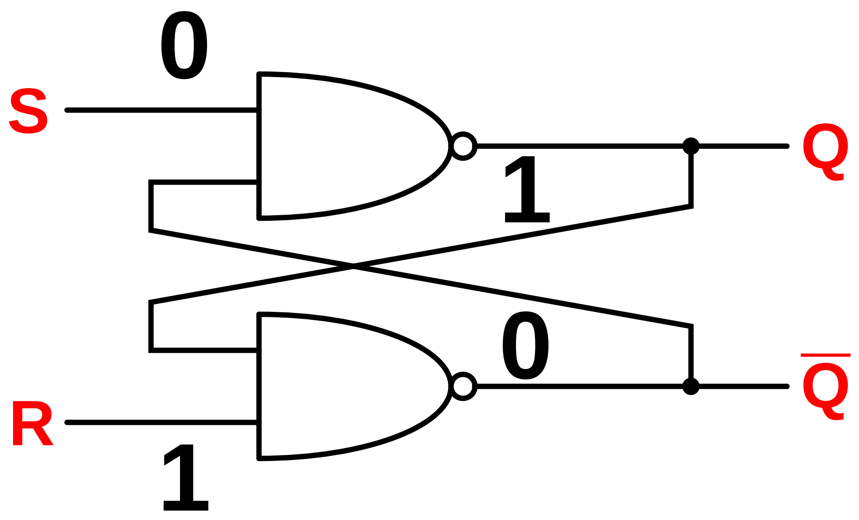
NAND-based SR Flip-Flop



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

Static latches & registers

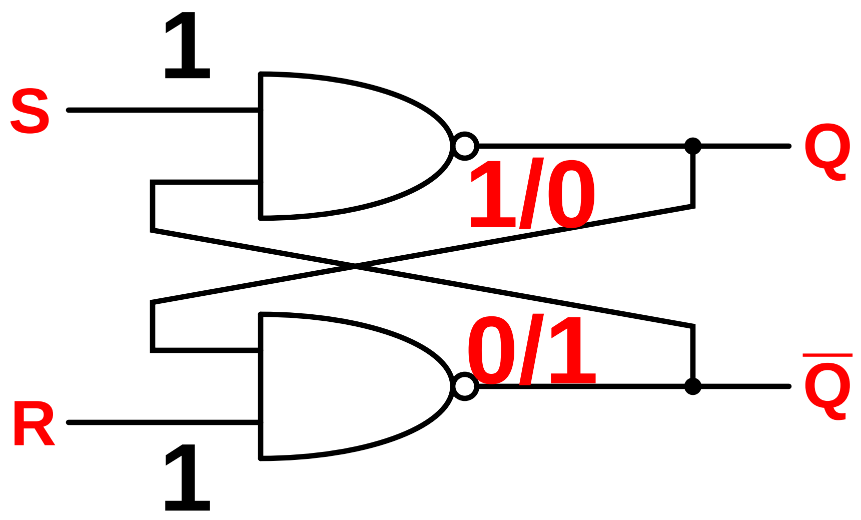
NAND-based SR Flip-Flop



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

Static latches & registers

NAND-based SR Flip-Flop

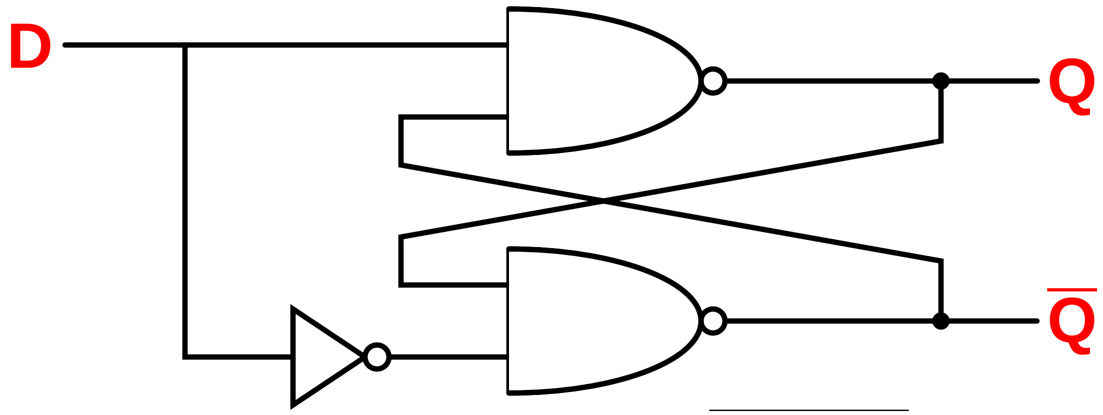


S	R	Q	$\overline{Q}$
0	0	1	1
1	0	0	1
0	1	1	0
1	1	Q	$\overline{Q}$

Q and $\overline{Q}$ keep their previous values if this state is reached from a non-forbidden state

Some Issues

Forbidden FF states

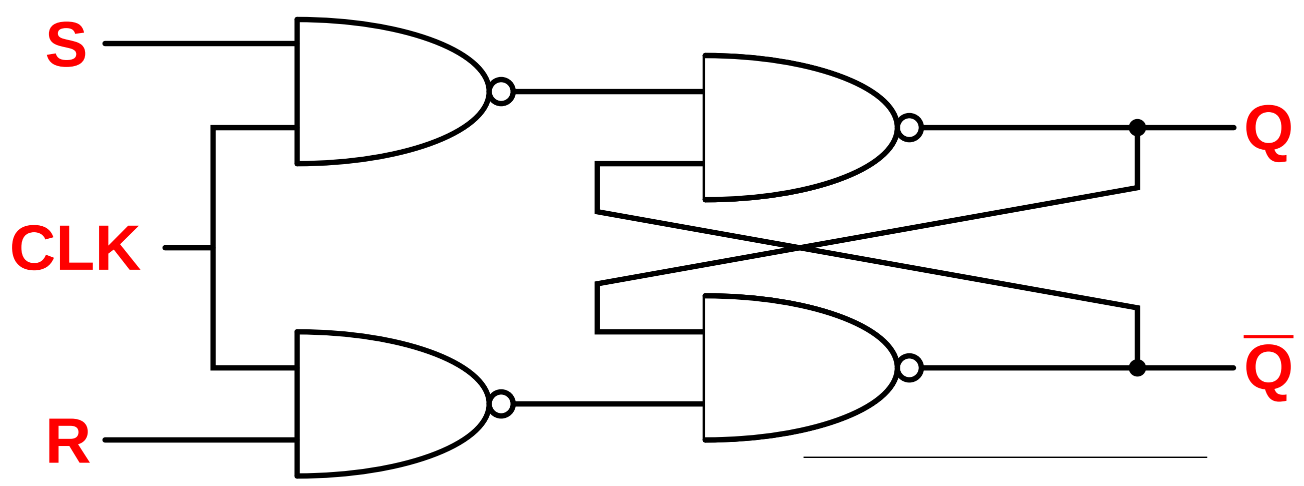


D	Q_{t+1}
1	0
0	1

D latch (transparent)

Some Issues

Forbidden FF states

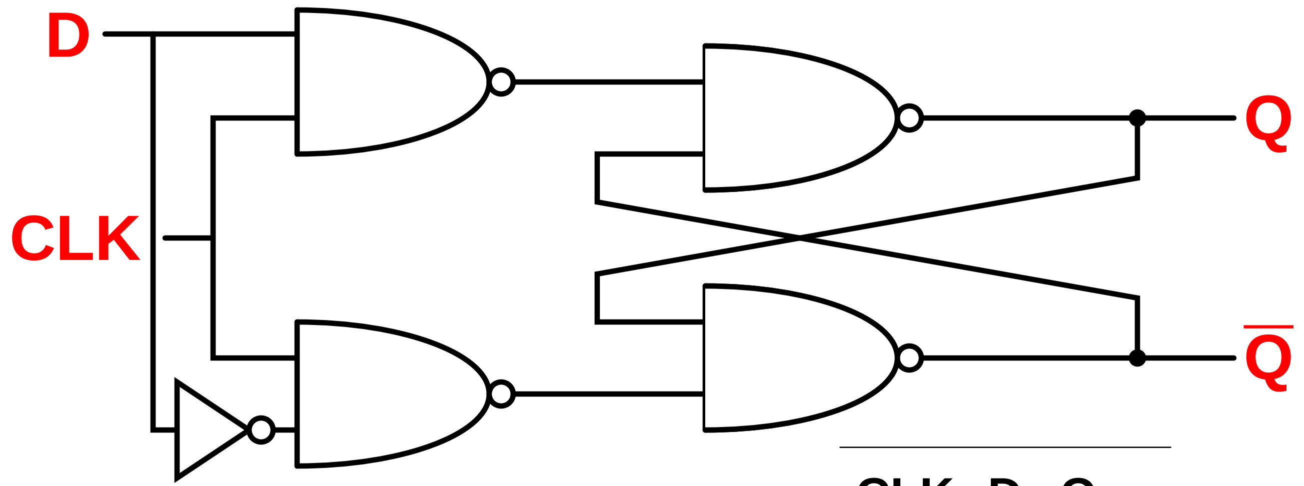


Gated SR latch (transparent)

CLK	S	R	Q_{t+1}
0	x	x	Q_t
1	0	0	Q_t
1	0	1	0
1	1	0	1
1	1	1	x

Some Issues

Forbidden FF states

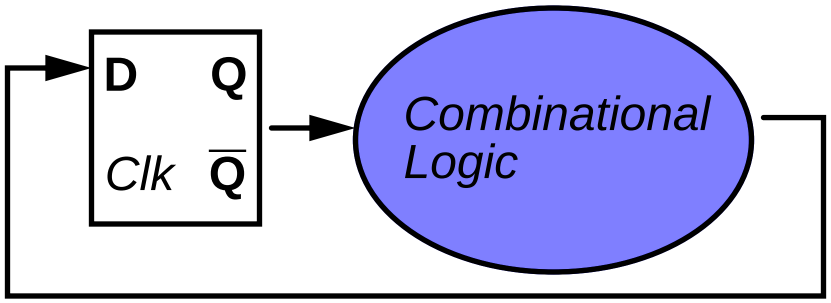


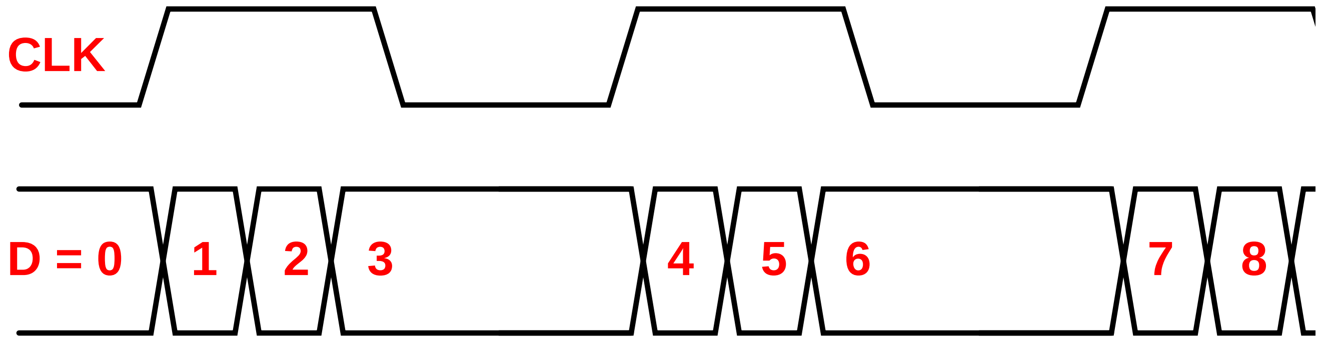
CLK	D	Q_{t+1}
0	x	Q_t
1	0	0
1	1	1

Gated D-latch (transparent)

Some Issues

Feed-through and race conditions

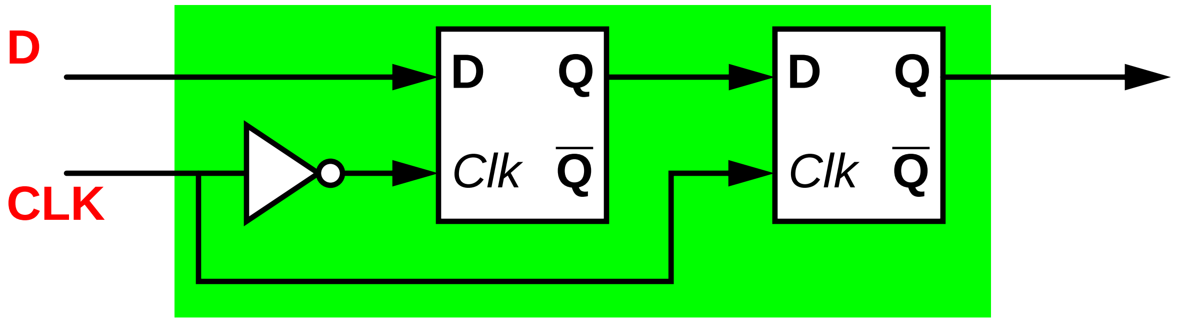




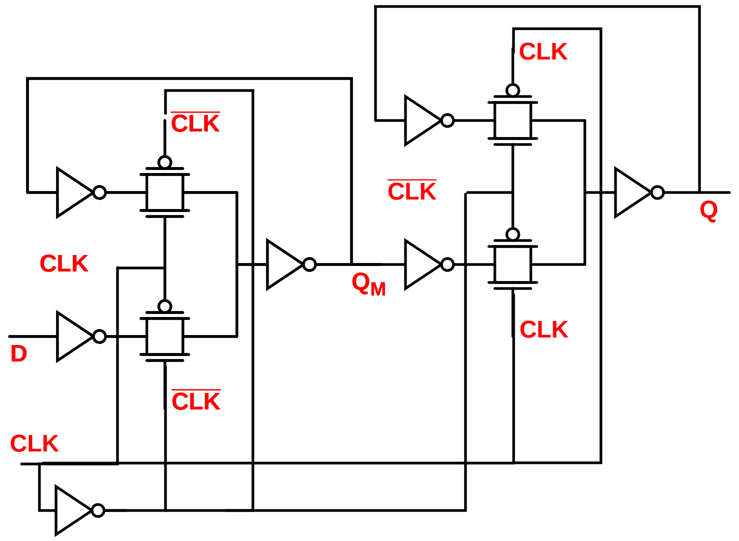
Transparent latches allow combinational logic *results* to be seen as its own *inputs*

Some Issues

Feed-through and race conditions



Master-slave D register (neg-edge triggered)

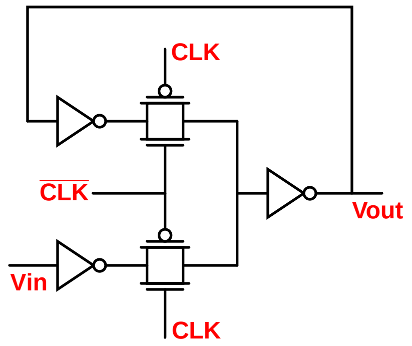


Bruce Jacob

University of
Maryland
ECE Dept.

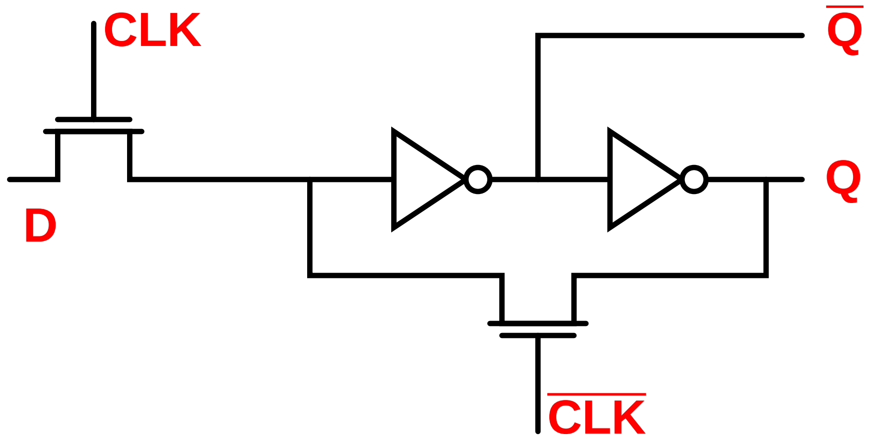
Some Issues

Cost of Clock Network (driving huge load)



Clock network drives 4 transistors per latch … power-expensive.

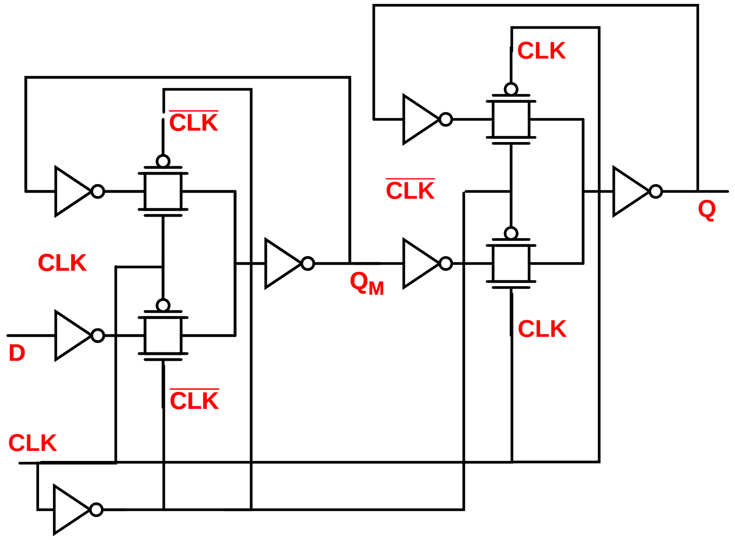
Alternative design:



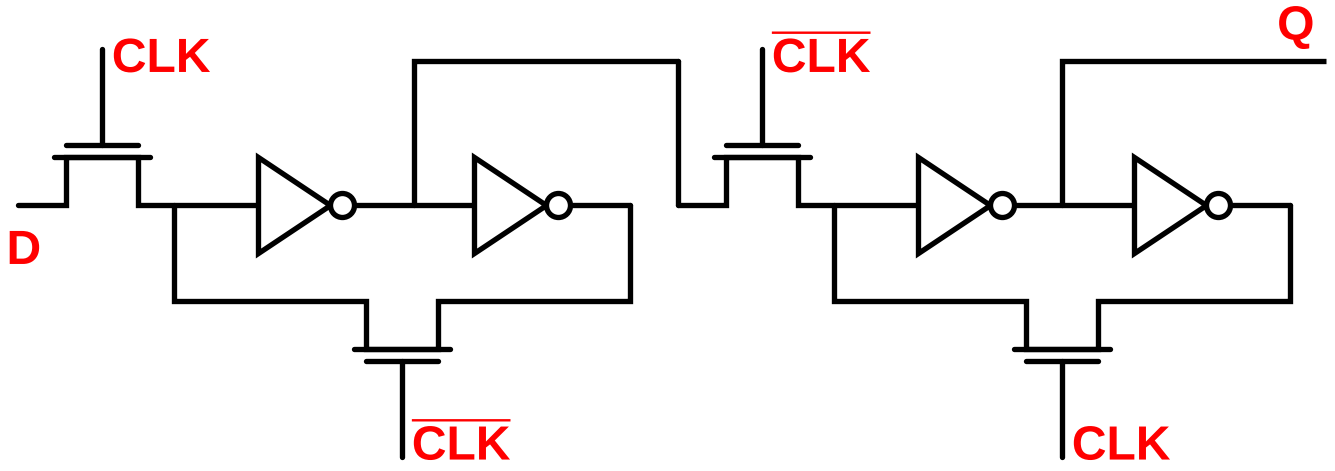
- Design w/ NMOS pass transistors presents smaller load to CLK; INV can recover low “1” but at a cost
- However: requires non-overlapping CLK/$\overline{\text{CLK}}$ … why?

Some Issues

Cost of Clock Network (driving huge load)



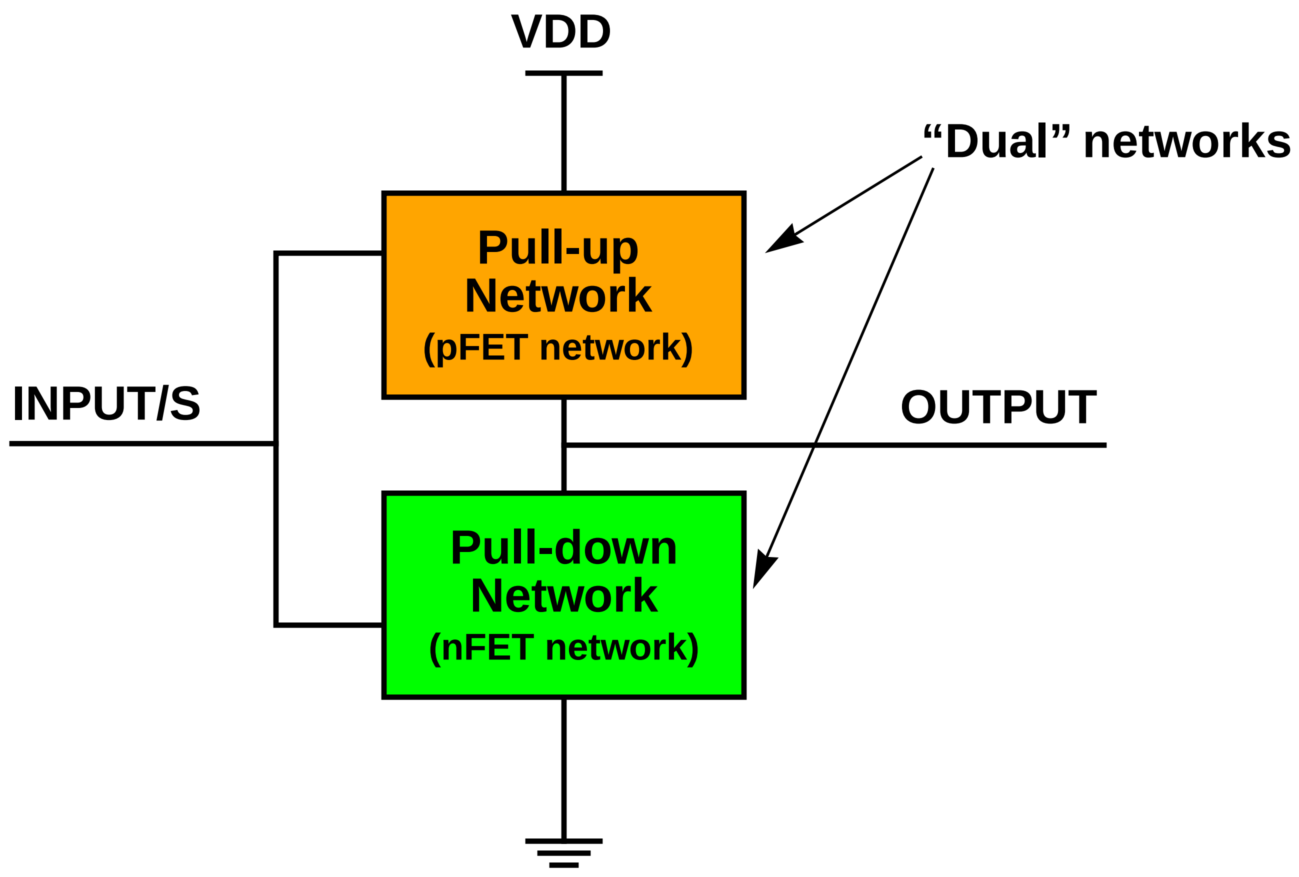
Similar example, with master-slave organzation



Some Issues

Number of Transistors: Dynamic Logic

(first: Primer … recall complementary logic)



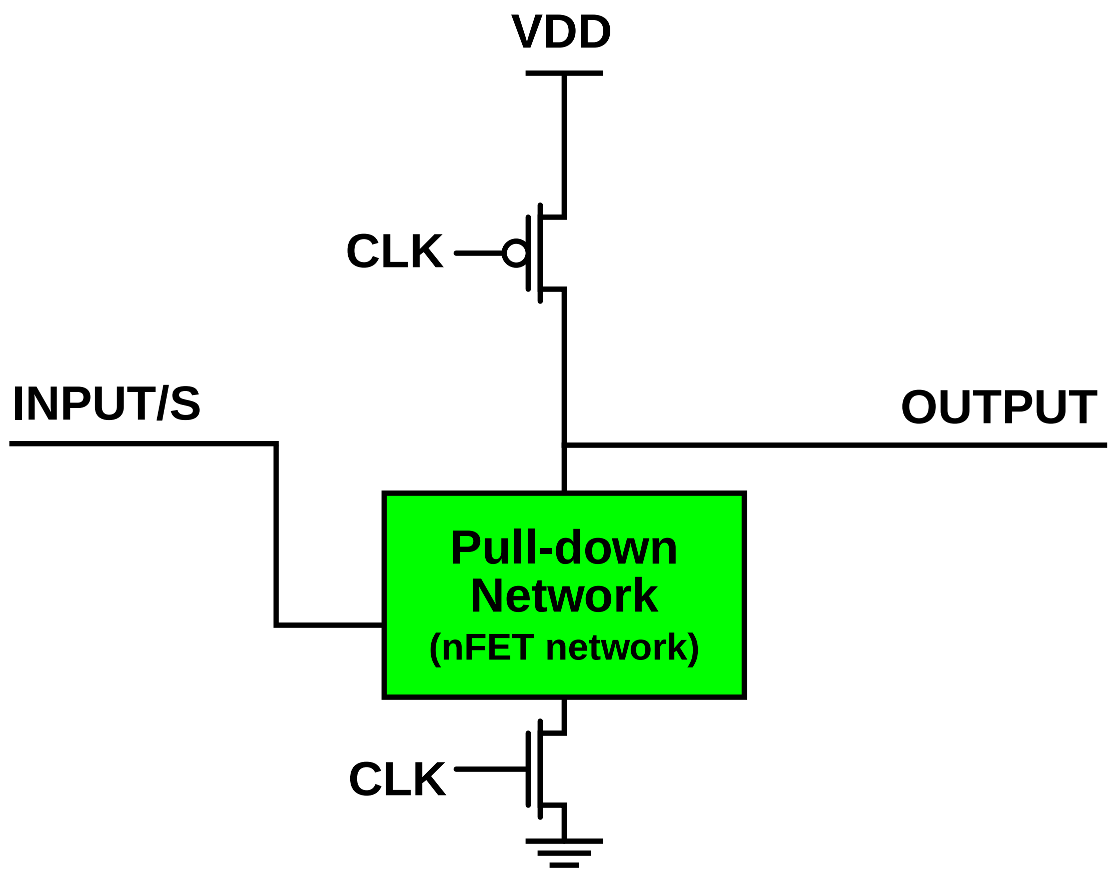
Bruce Jacob

University of
Maryland
ECE Dept.

Some Issues

Number of Transistors: Dynamic Logic

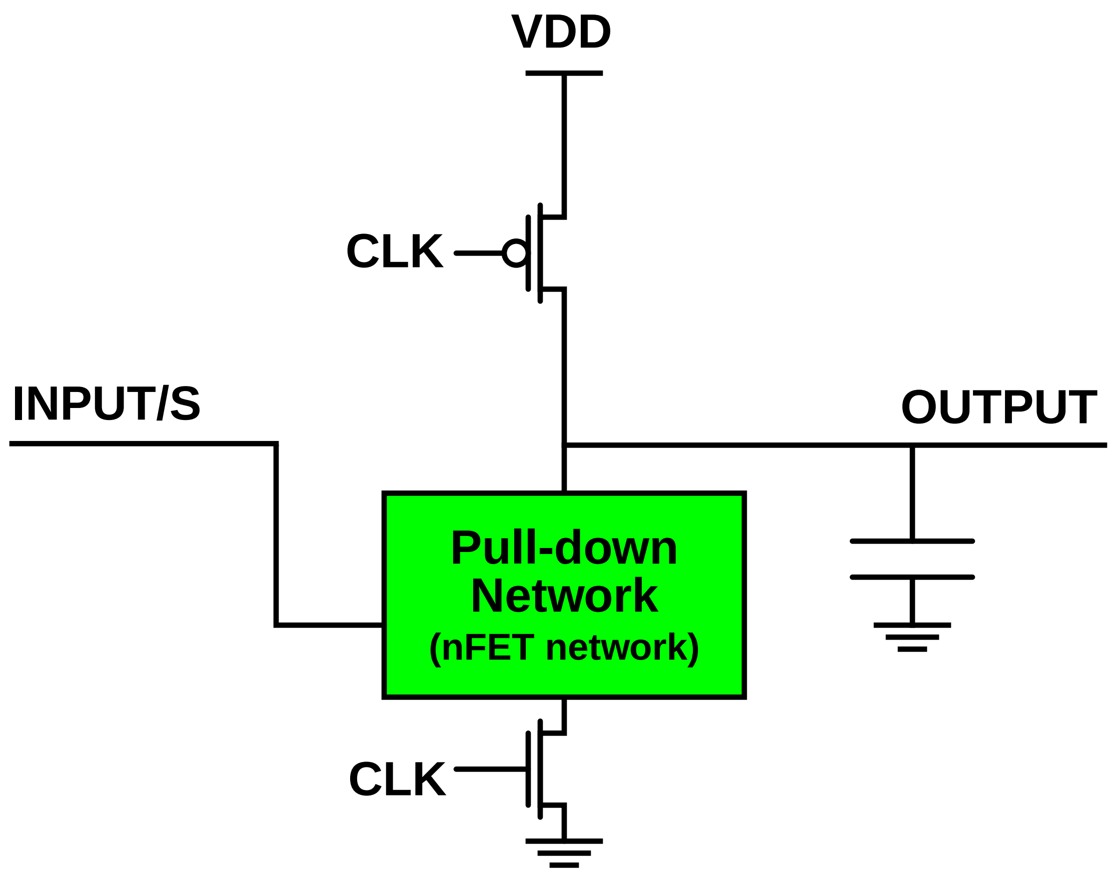
(… no longer complementary …)



Some Issues

Number of Transistors: Dynamic Logic

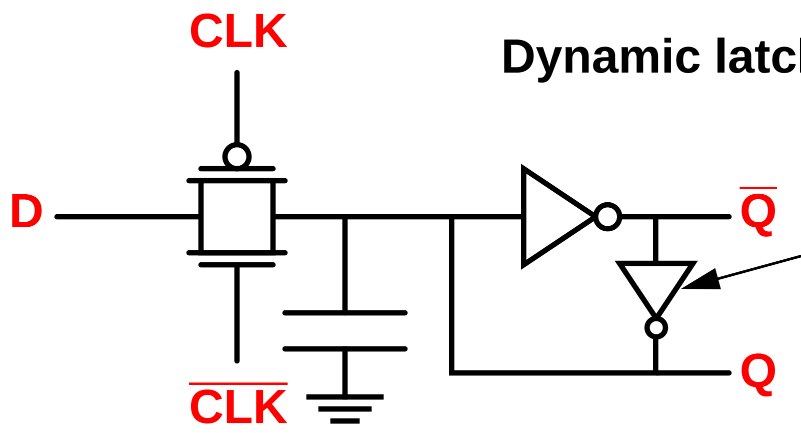
(… relies upon capacitance, burns power)



Some Issues

Number of Transistors: *Dynamic* Storage

Dynamic latch (transparent)

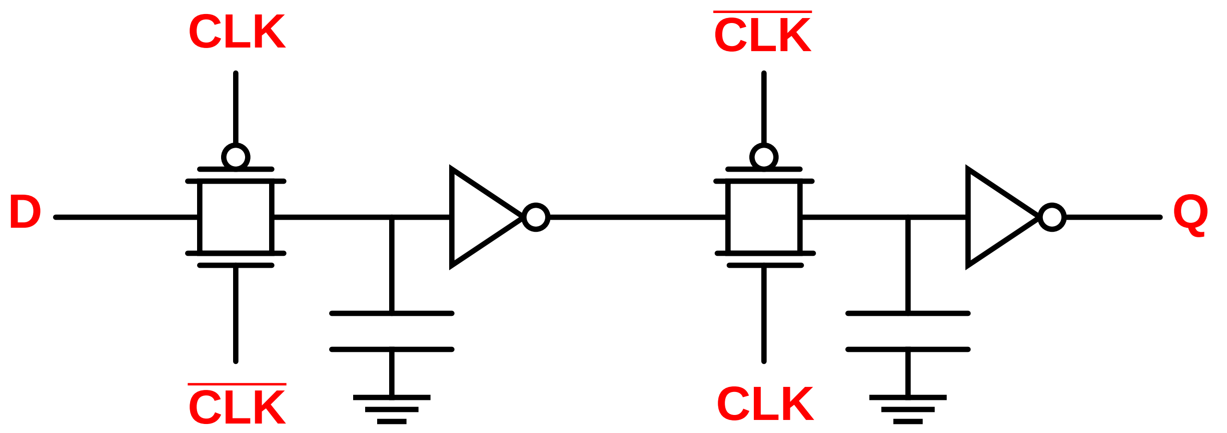


Create “pseudostatic” latch:

Make this inverter weak so that D input overpowers feedback loop.

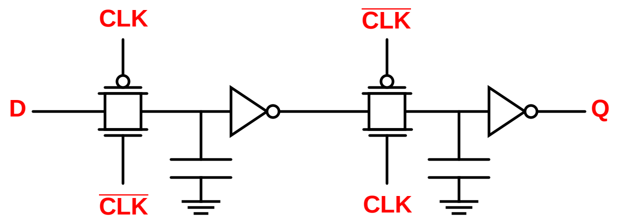
How to make weak inverter:
W/L: make W small or L large

Dynamic edge-triggered register



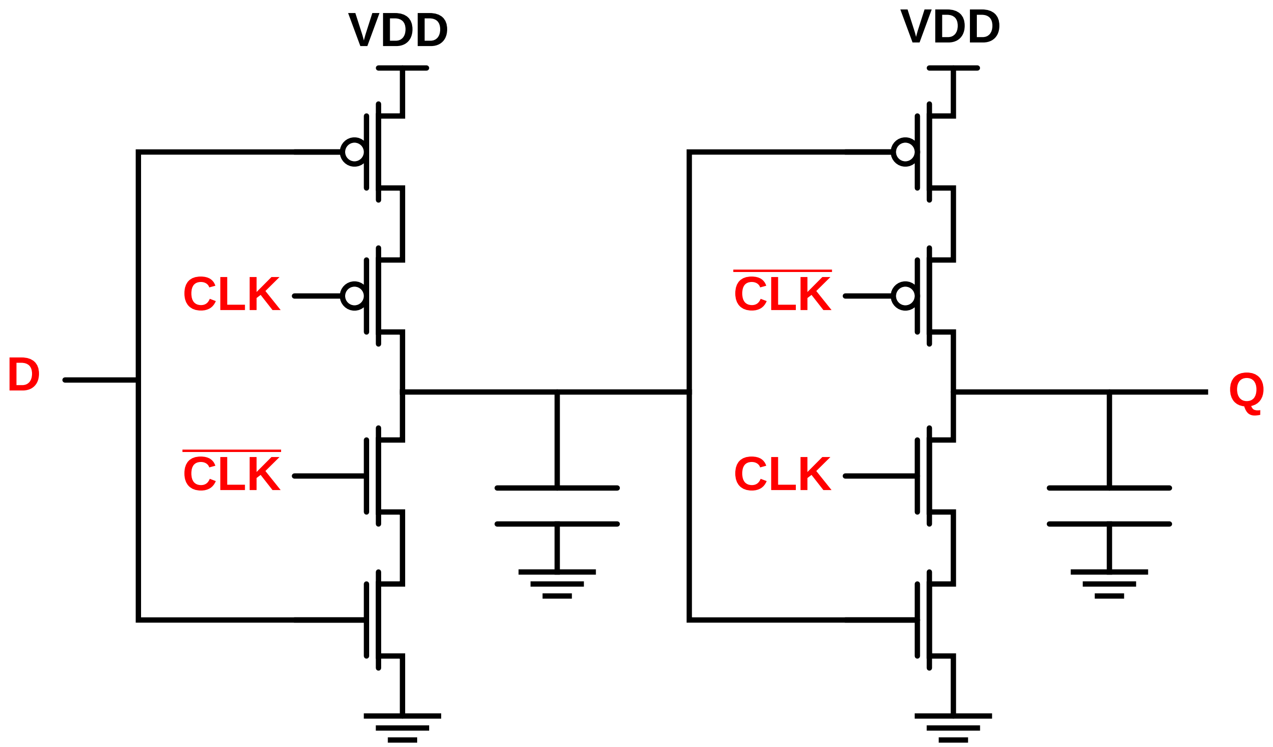
Some Issues

Non-overlapping clocks: Clocked CMOS



This allows feed-through when clocks overlap.

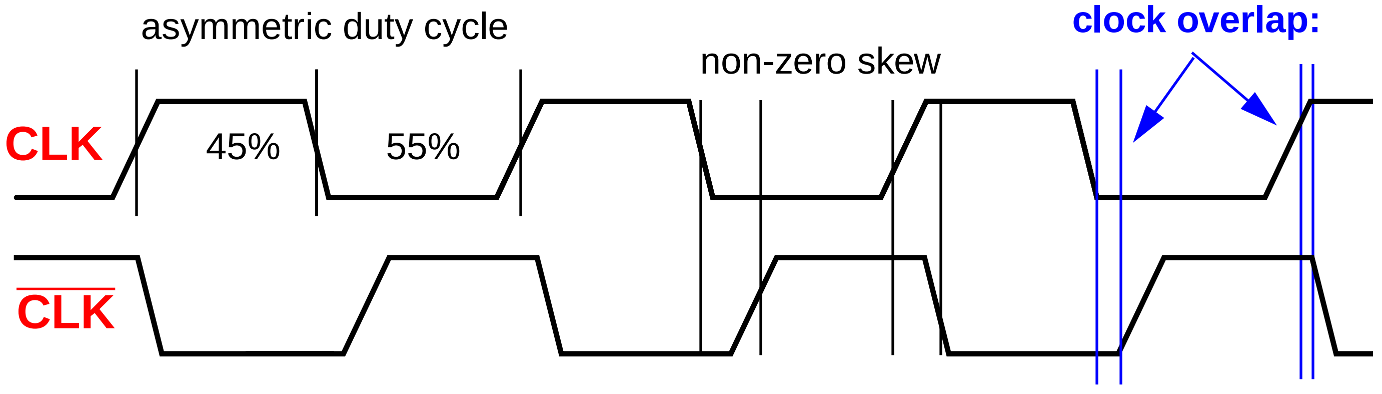
This design does not:

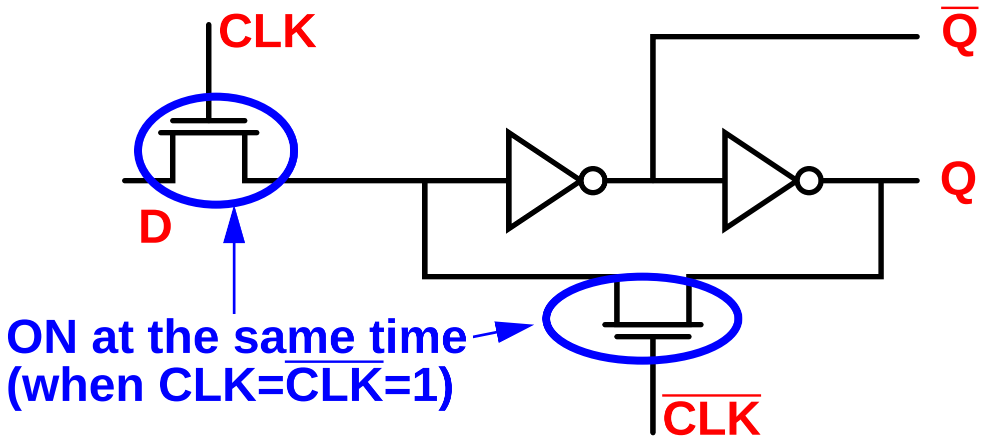


Some Issues

Non-overlapping clocks: Clocked CMOS

EXAMPLES:





Bruce Jacob

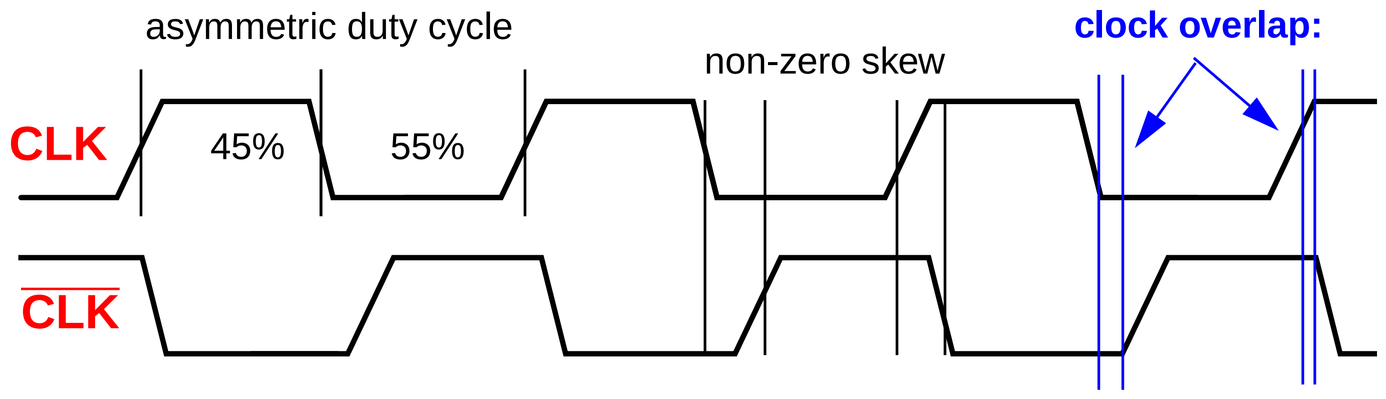
University of
Maryland
ECE Dept.

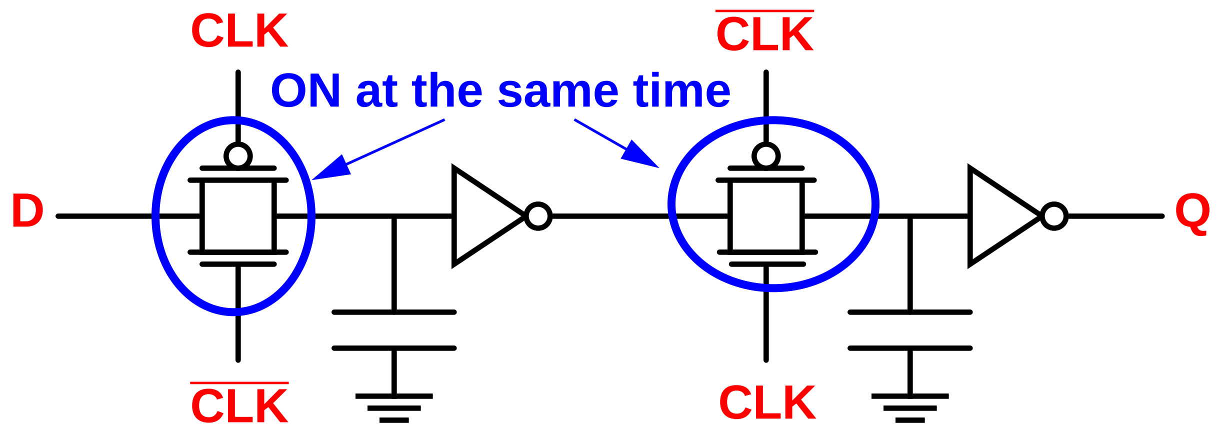
UNIVERSITY OF MARYLAND

Some Issues

Non-overlapping clocks: Clocked CMOS

EXAMPLES:

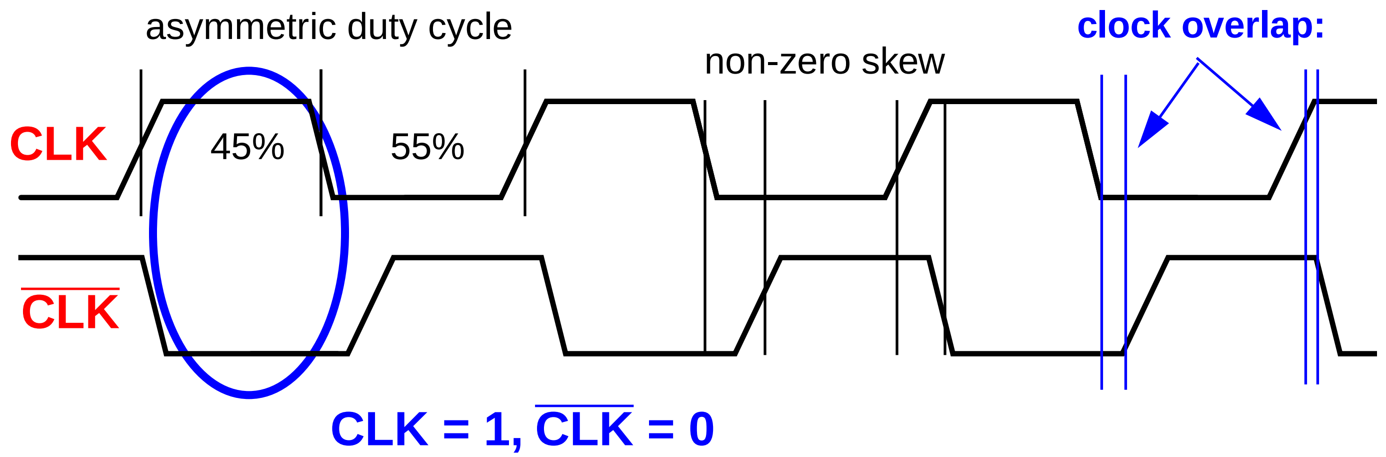


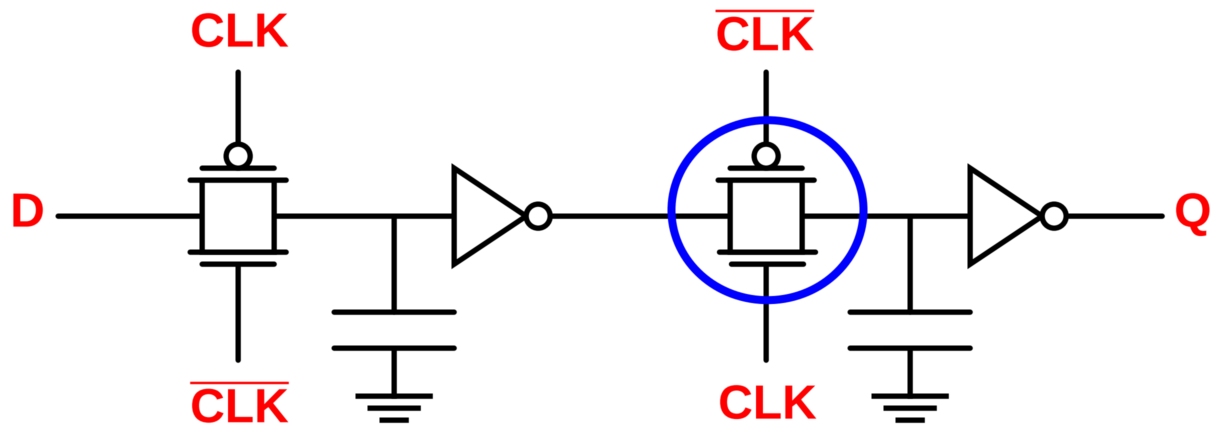


Some Issues

Non-overlapping clocks: Clocked CMOS

EXAMPLES:



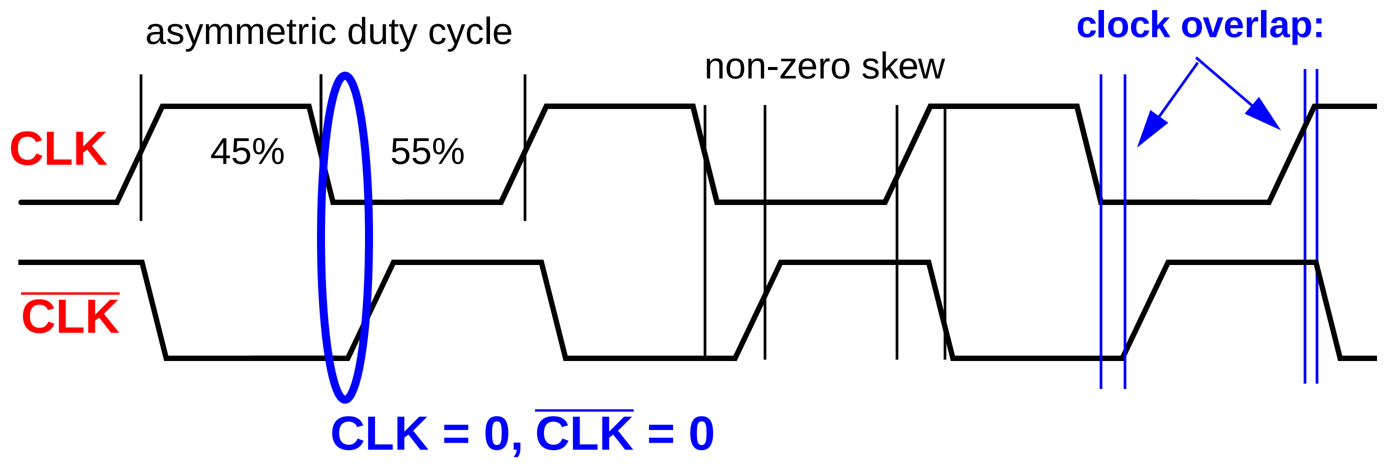


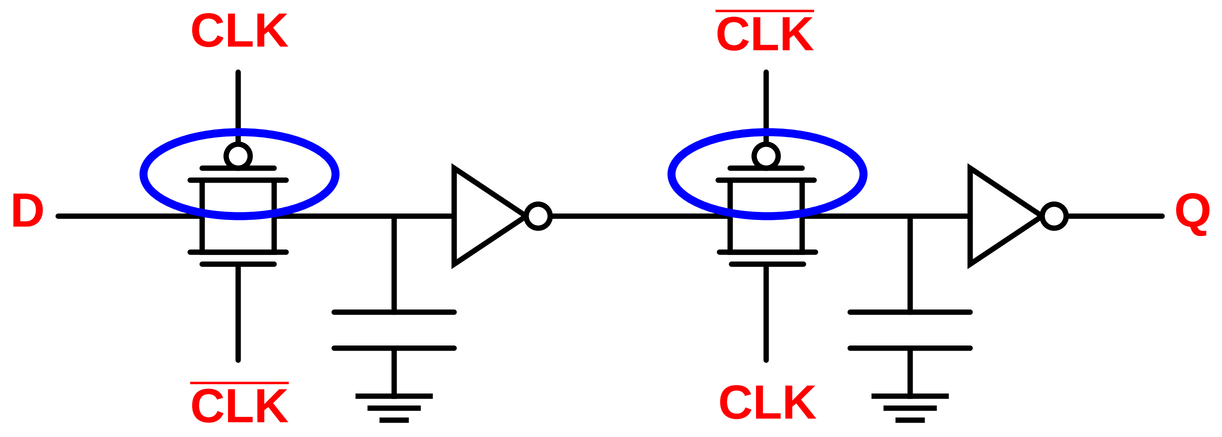
UNIVERSITY OF MARYLAND

Some Issues

Non-overlapping clocks: Clocked CMOS

EXAMPLES:

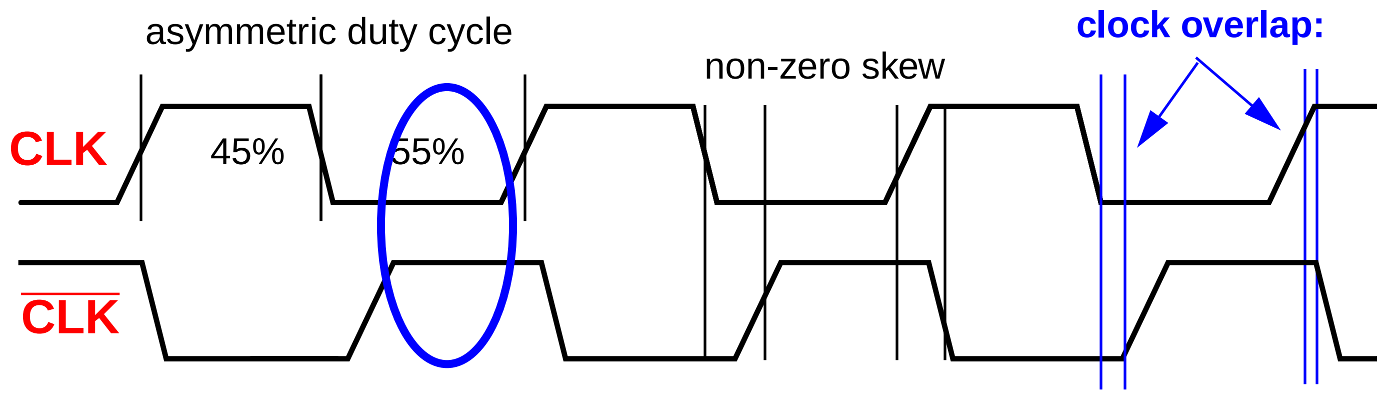




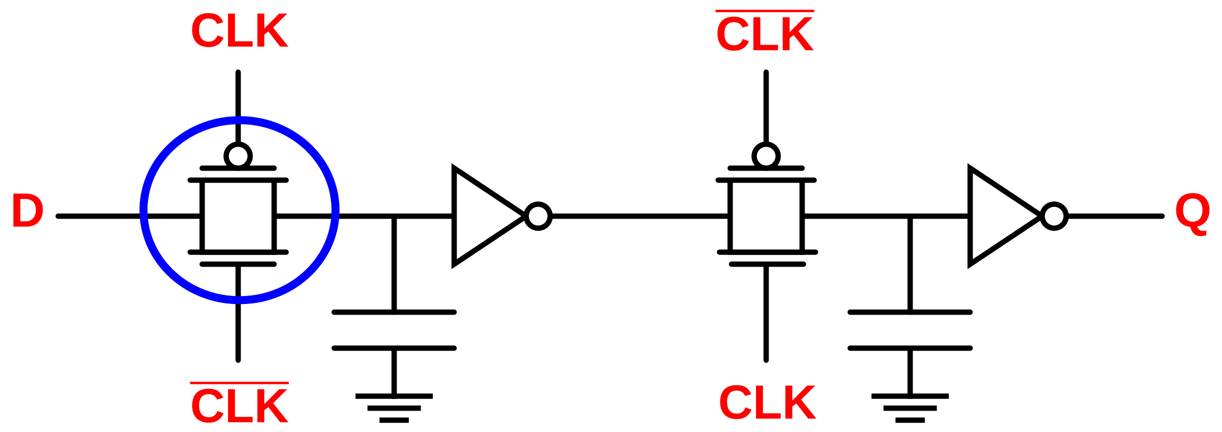
Some Issues

Non-overlapping clocks: Clocked CMOS

EXAMPLES:



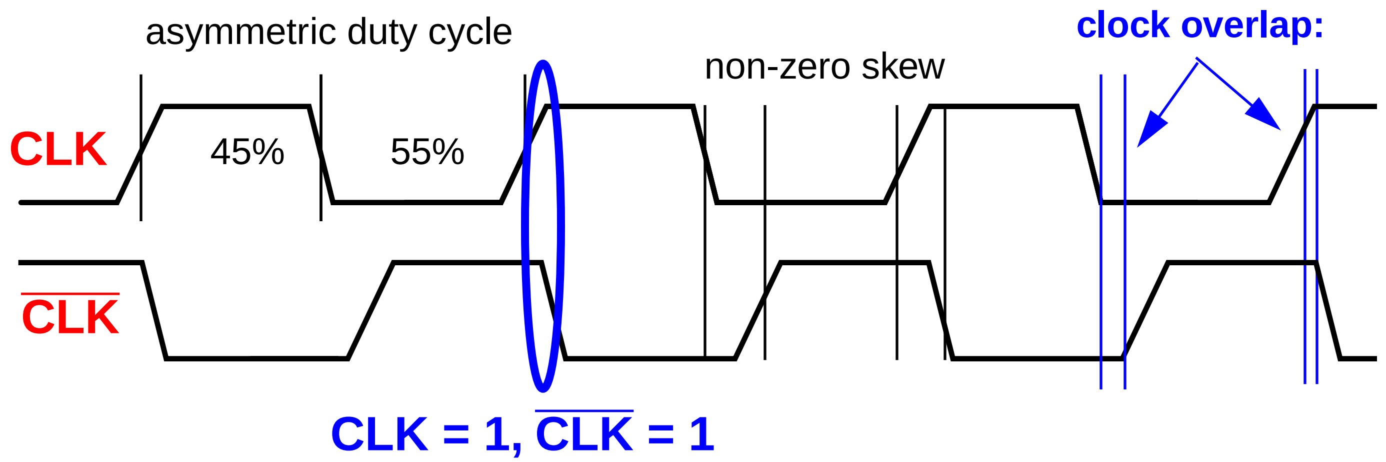
$\text{CLK} = 0, \overline{\text{CLK}} = 1$

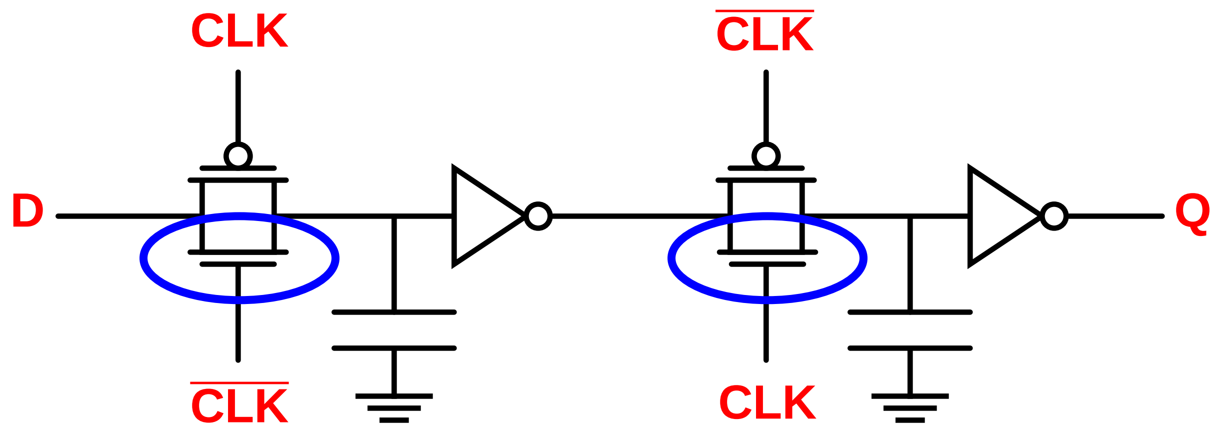


Some Issues

Non-overlapping clocks: Clocked CMOS

EXAMPLES:





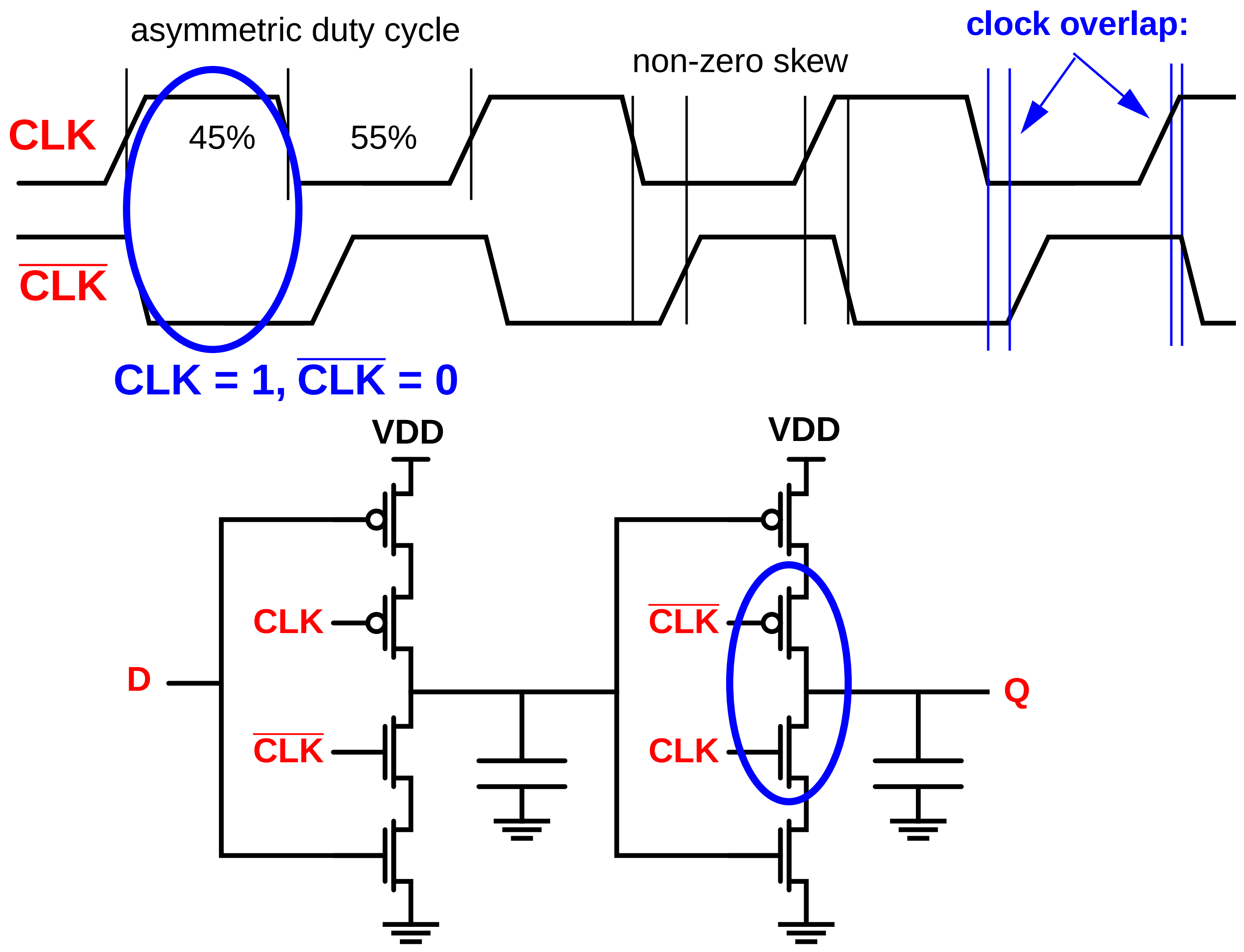
Bruce Jacob

University of Maryland
ECE Dept.

UNIVERSITY OF MARYLAND

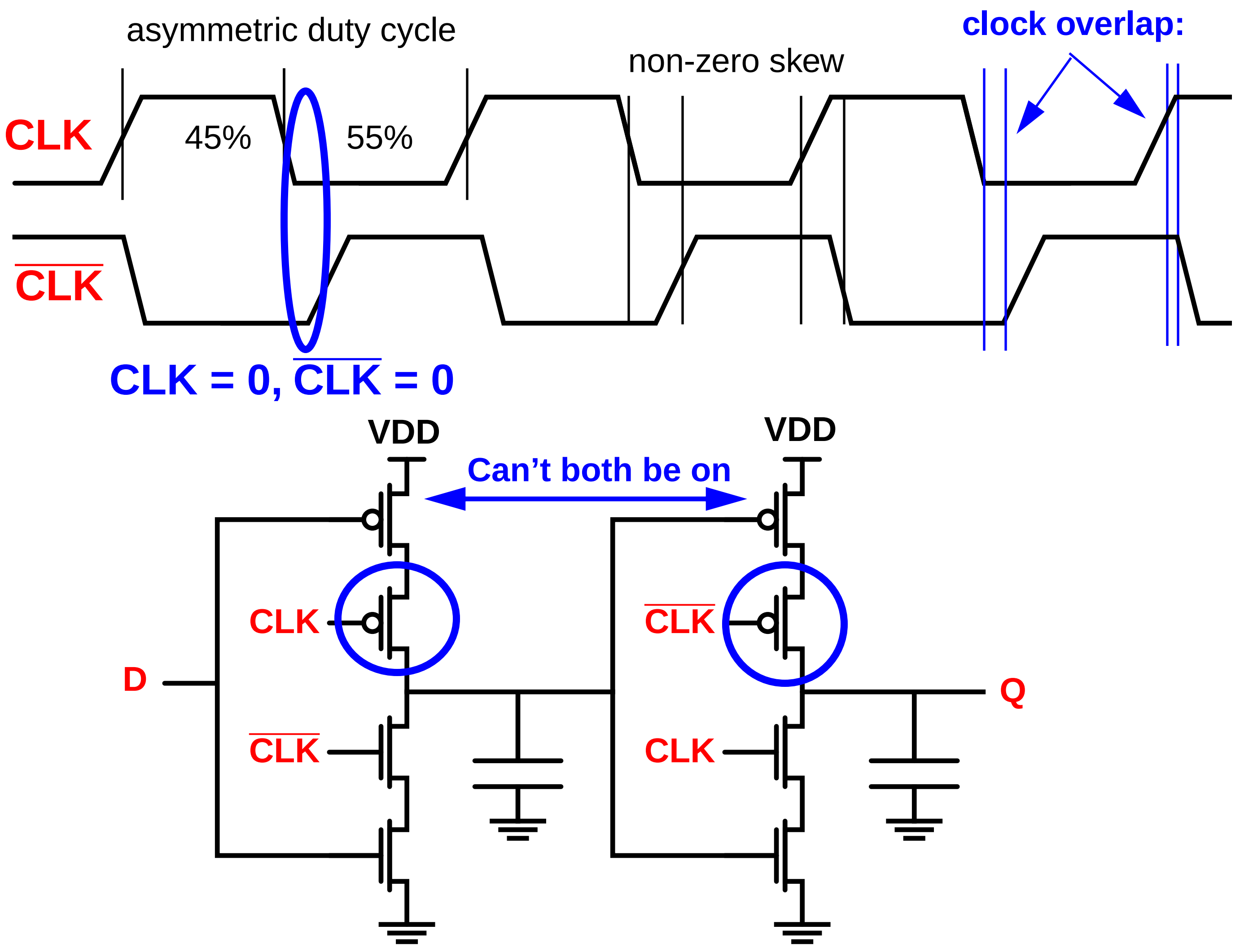
Some Issues

Non-overlapping clocks: Clocked CMOS



Some Issues

Non-overlapping clocks: Clocked CMOS



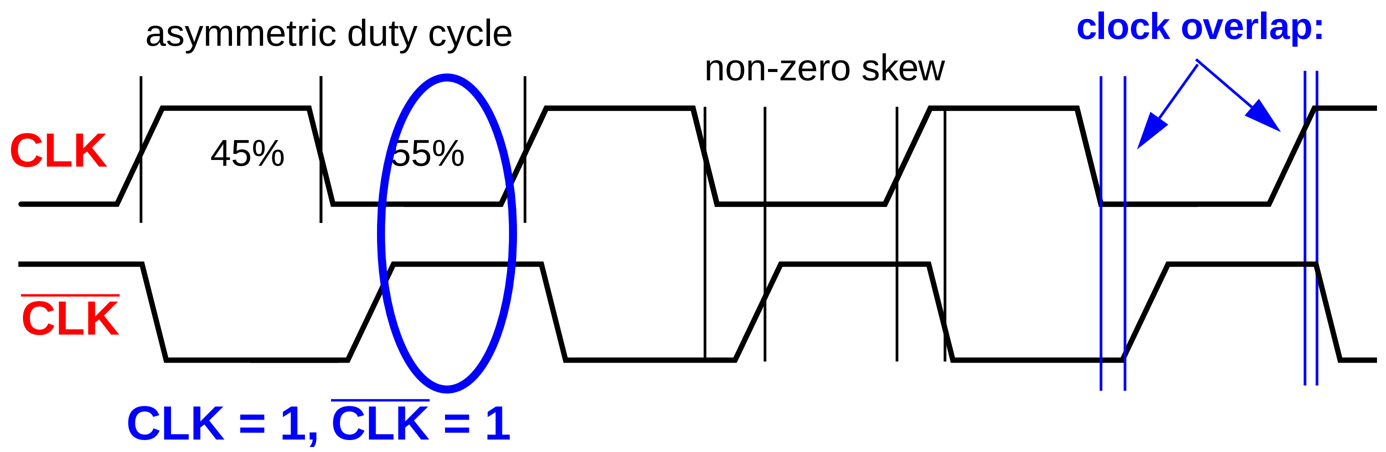
Bruce Jacob

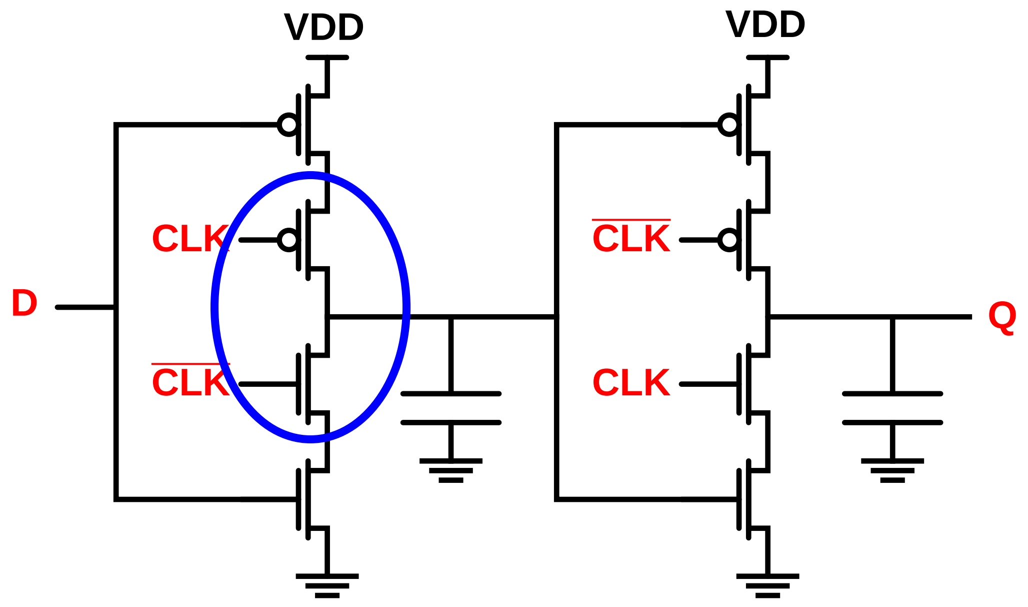
University of
Maryland
ECE Dept.

UNIVERSITY OF MARYLAND

Some Issues

Non-overlapping clocks: Clocked CMOS





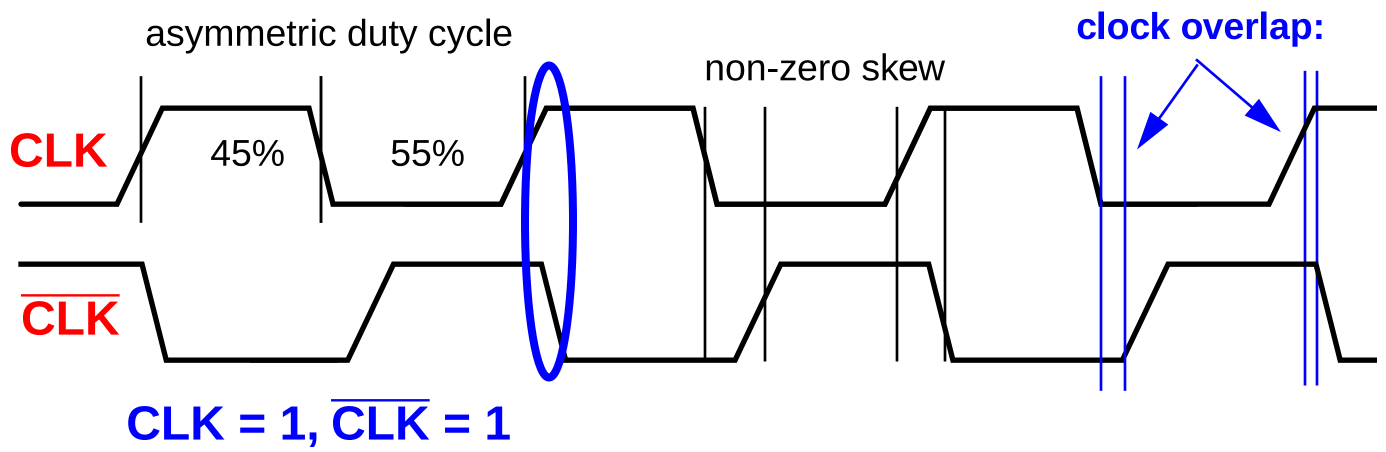
Bruce Jacob

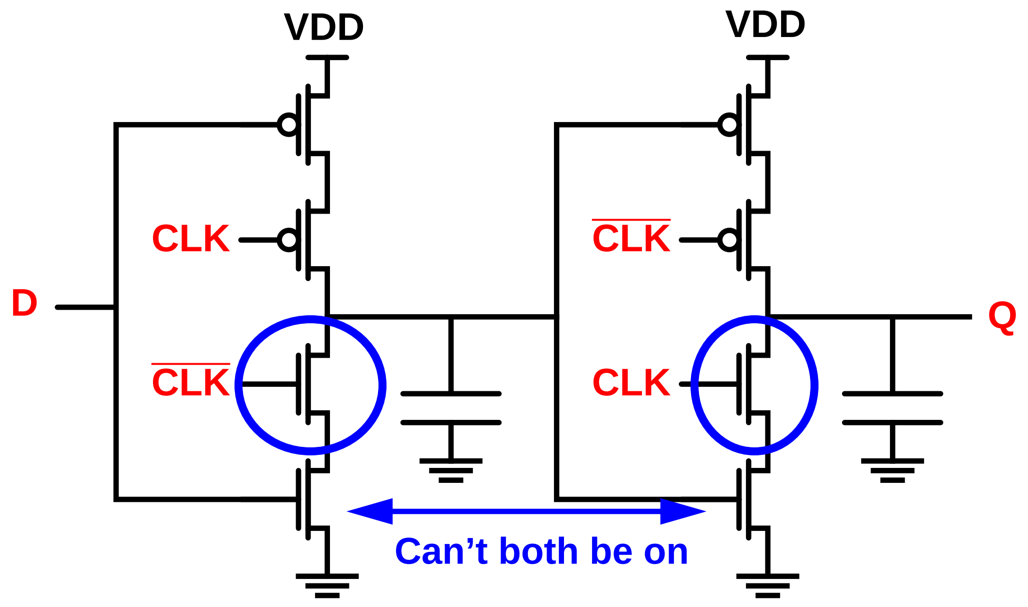
University of
Maryland
ECE Dept.

UNIVERSITY OF MARYLAND

Some Issues

Non-overlapping clocks: Clocked CMOS





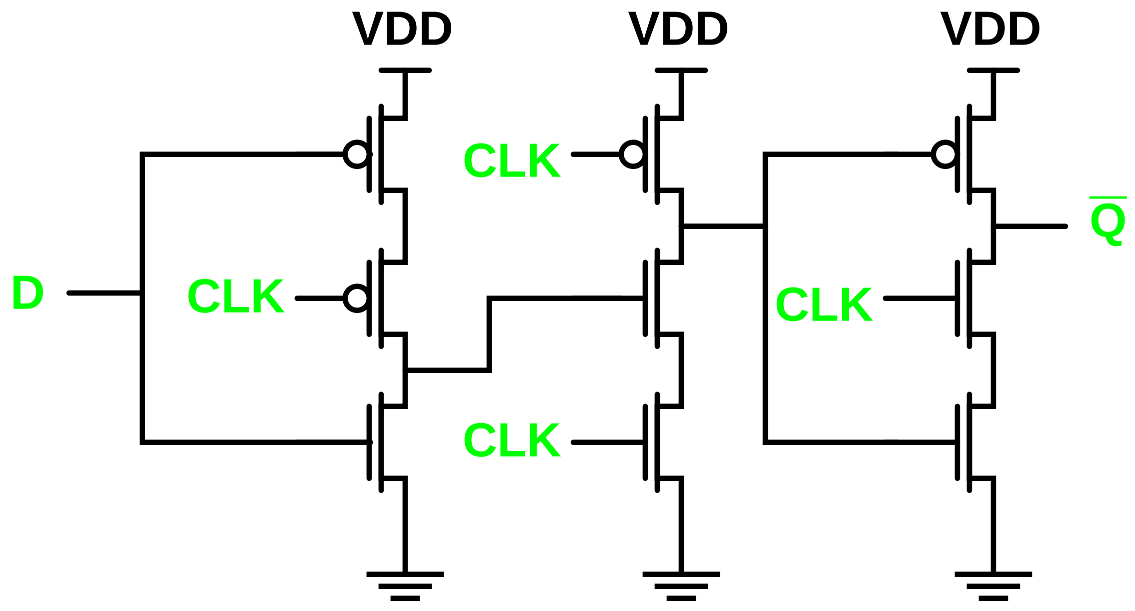
Bruce Jacob

University of
Maryland
ECE Dept.

Some Issues

Non-overlapping clocks:

“True” Single-Phase Clocked Register

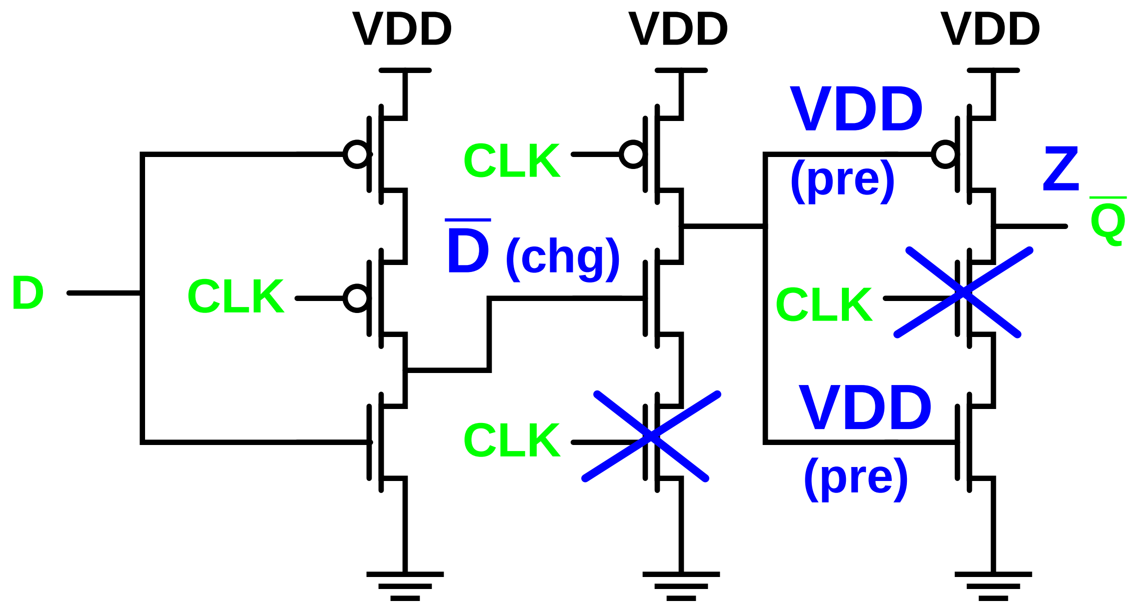


Positive edge-driven register

Some Issues

Non-overlapping clocks:

“True” Single-Phase Clocked Register



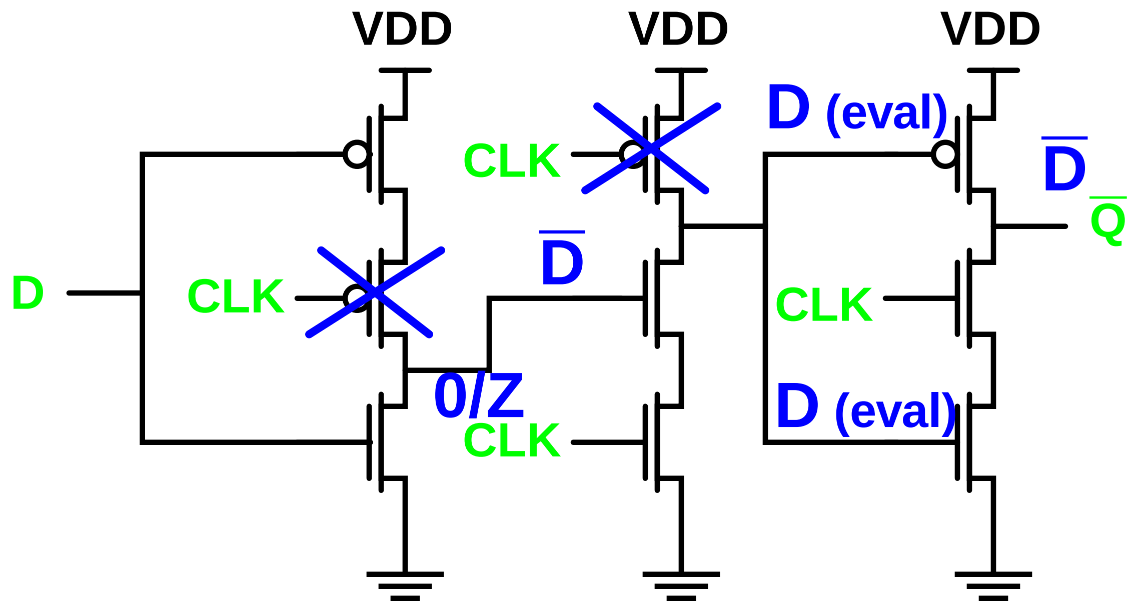
CLK = 0

Output = Z => output is *stable* (dynamic)

Some Issues

Non-overlapping clocks:

“True” Single-Phase Clocked Register

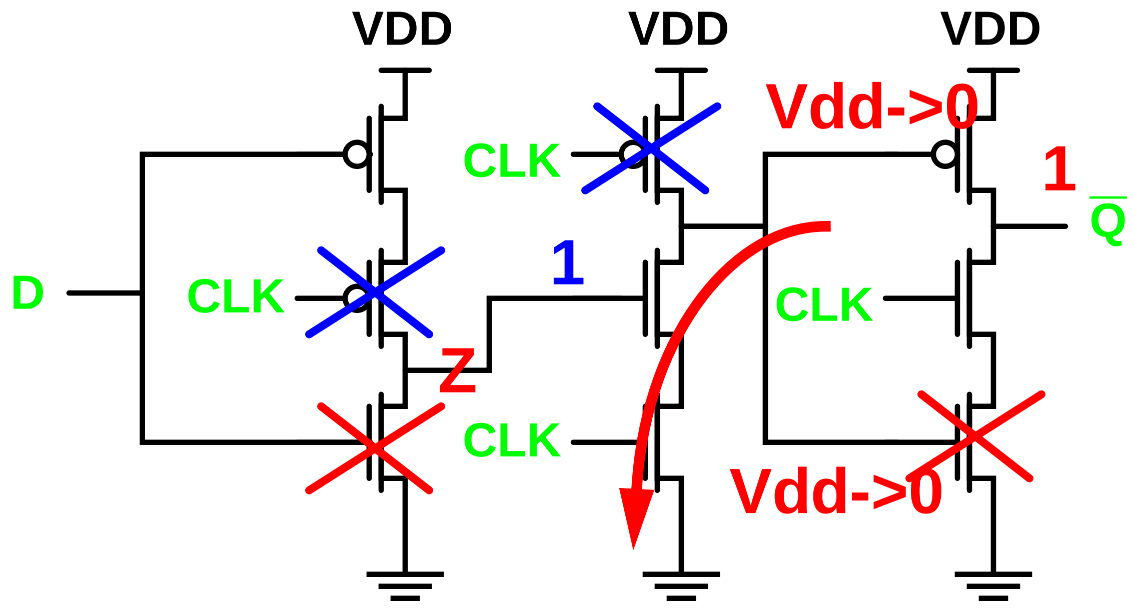


CLK = 1

Some Issues

Non-overlapping clocks:

“True” Single-Phase Clocked Register

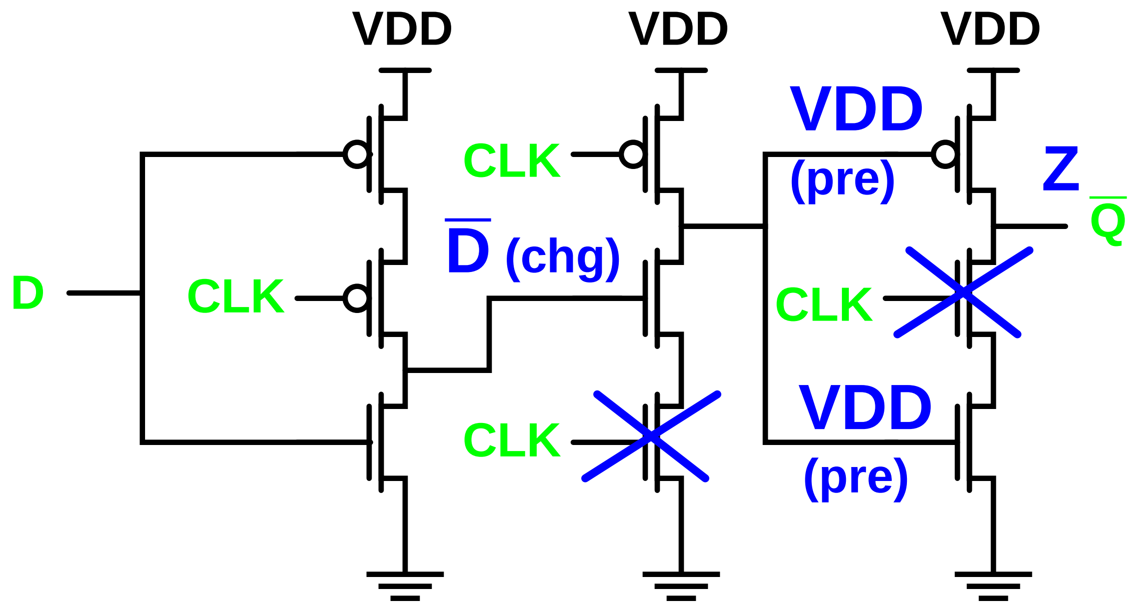


CLK = 1, D = 0

Some Issues

Non-overlapping clocks:

"True" Single-Phase Clocked Register

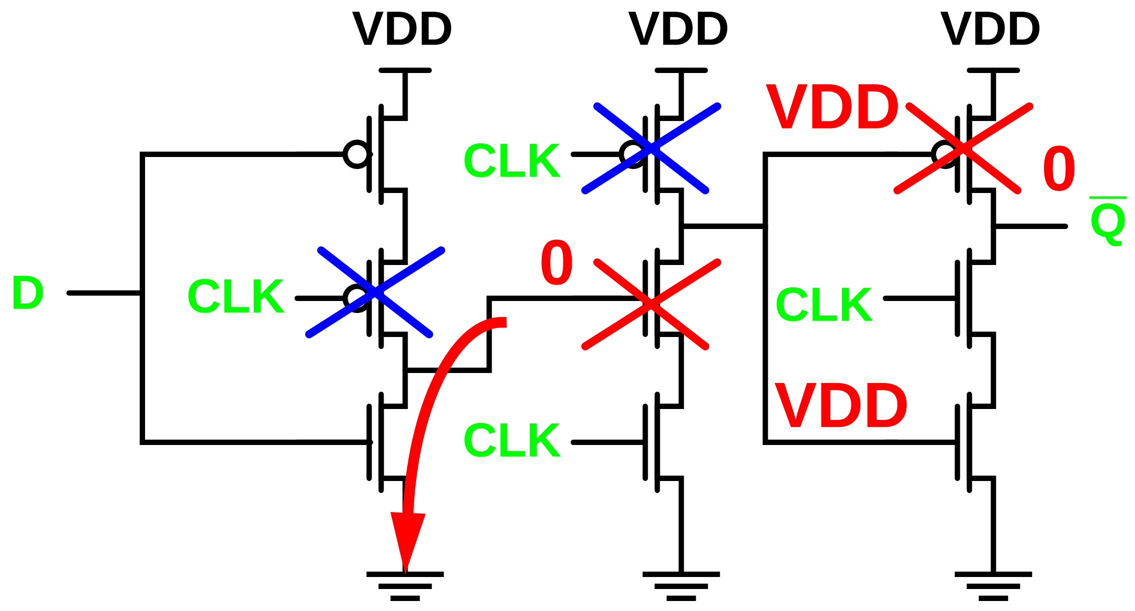


CLK = 0

Some Issues

Non-overlapping clocks:

“True” Single-Phase Clocked Register



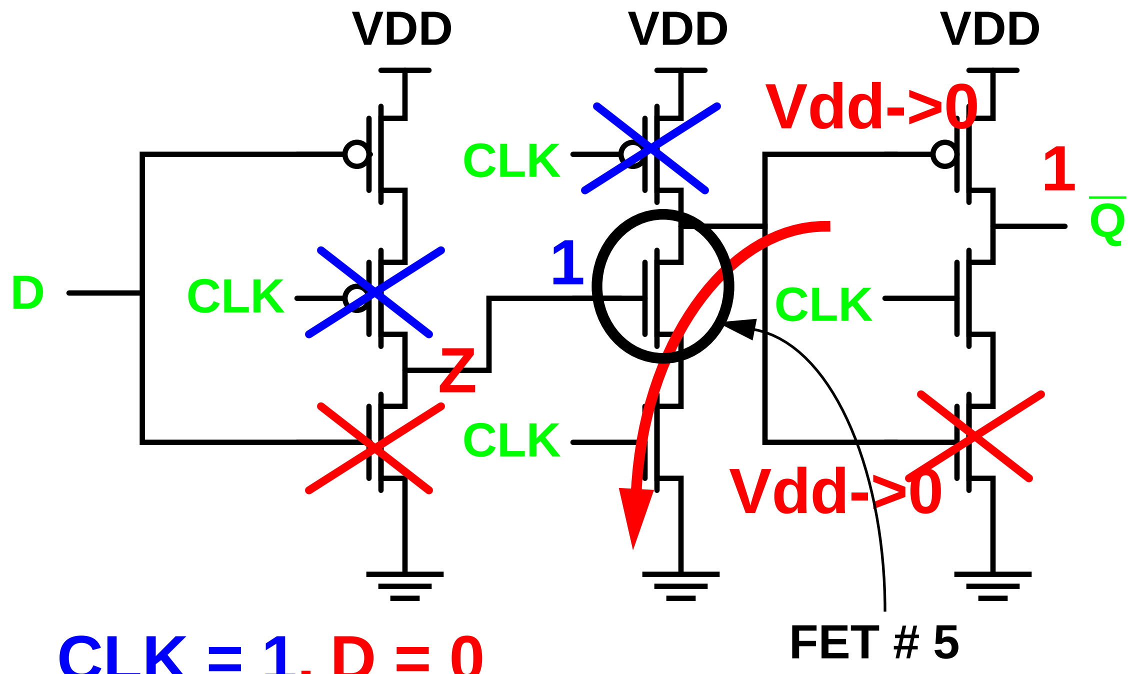
CLK = 1, D = 1

Bruce Jacob

University of
Maryland
ECE Dept.

Some Issues

Non-overlapping clocks:
"True" Single-Phase Clocked Register



CLK = 1, D = 0

Note: if D is allowed to transition from 0 to 1 too soon after CLK transitions 0->1, it is possible to close FET #5 before the final capacitance discharges (Vdd -> 0) … which would obviously pose a problem.

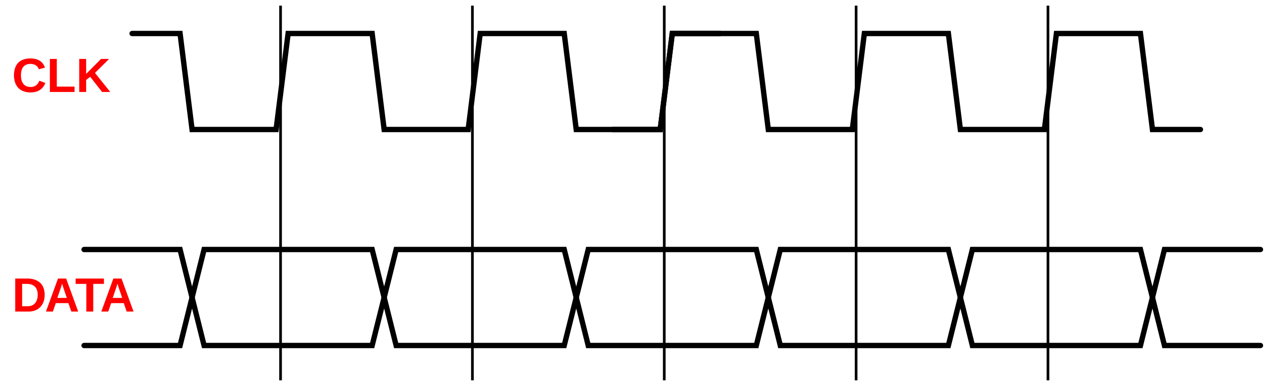
This represents the *hold time* of this register.

Bruce Jacob

University of
Maryland
ECE Dept.

Some Issues

Clock power: Dual-Edge-Triggered Reg.



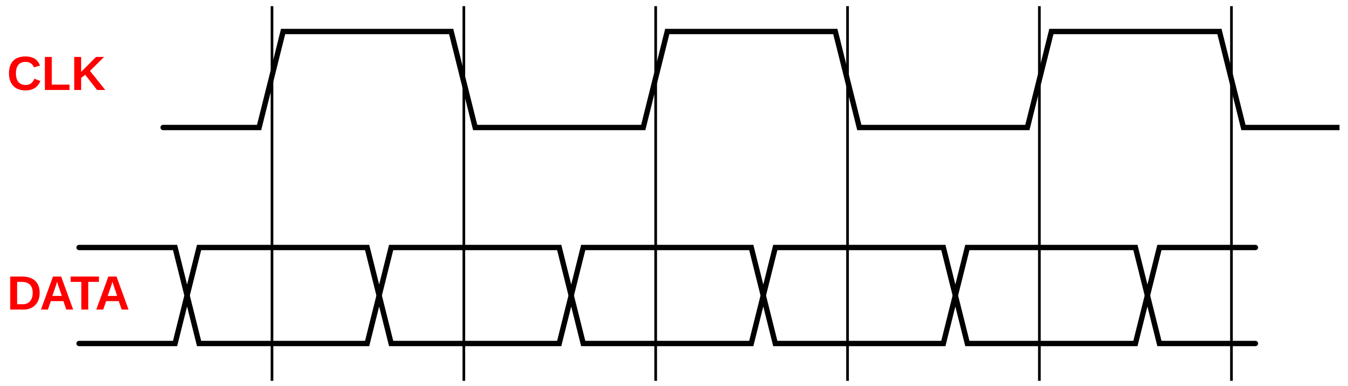
Note that clock transitions twice as often as data does (actually, even more, unless the data pattern happens to be 01010101010 …)

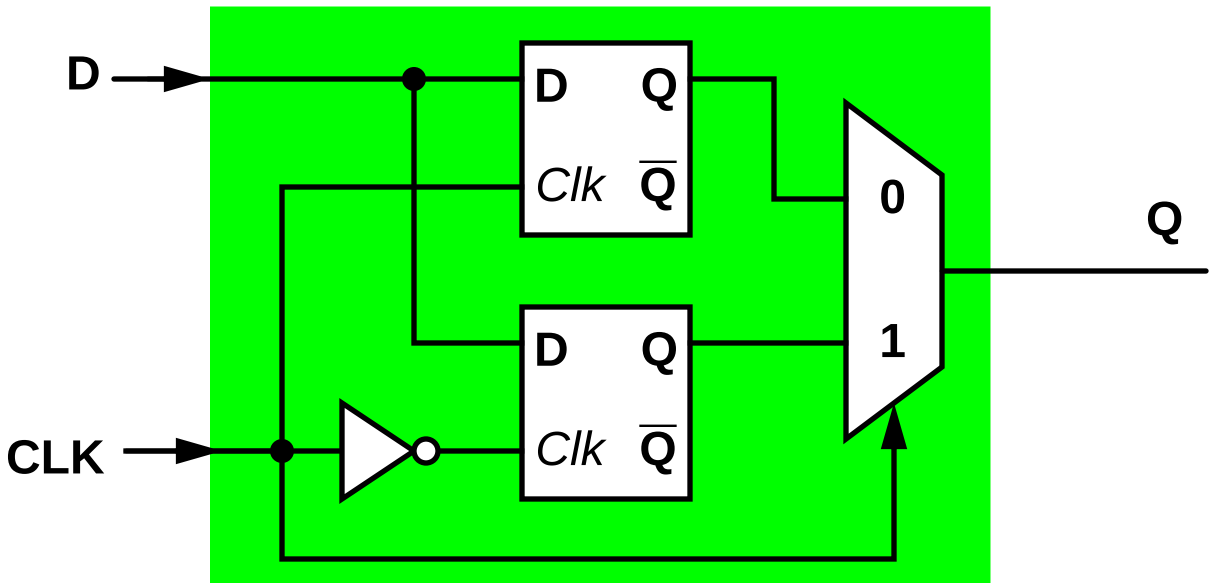
Max data rate = 1Gbps; clock rate = 2GHz

At high frequencies, this is a problem

Some Issues

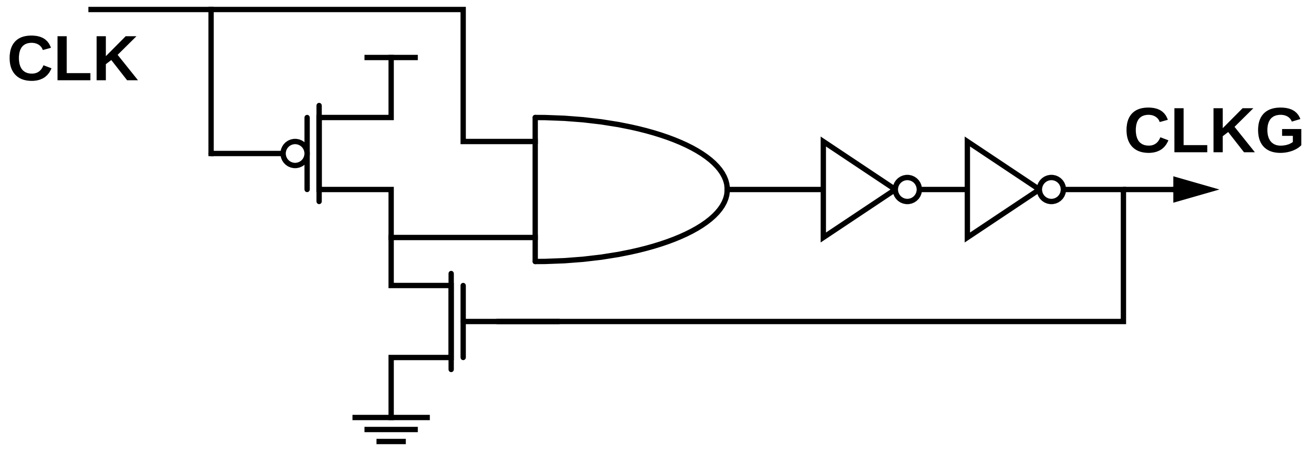
Clock power: Dual-Edge-Triggered Reg.

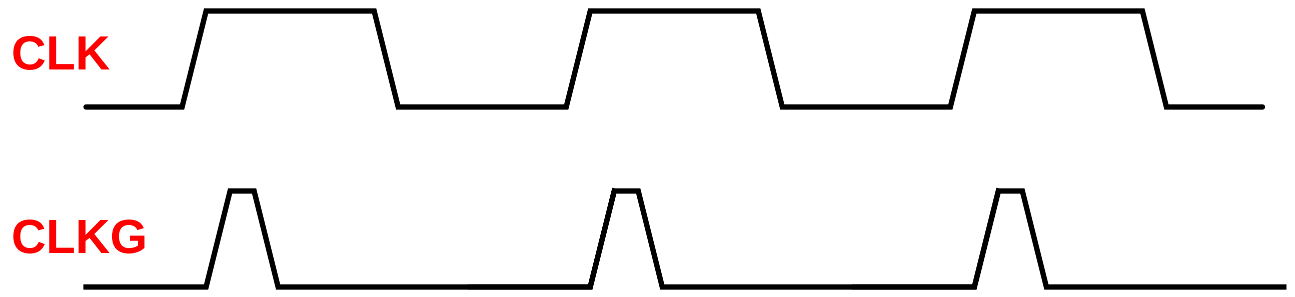




Some Issues

Simplicity, speed of design: Pulse registers





For use with transparent latches: creates *de facto* registers, provided you do thorough timing analysis to guarantee inputs to latch stable during transparent window.

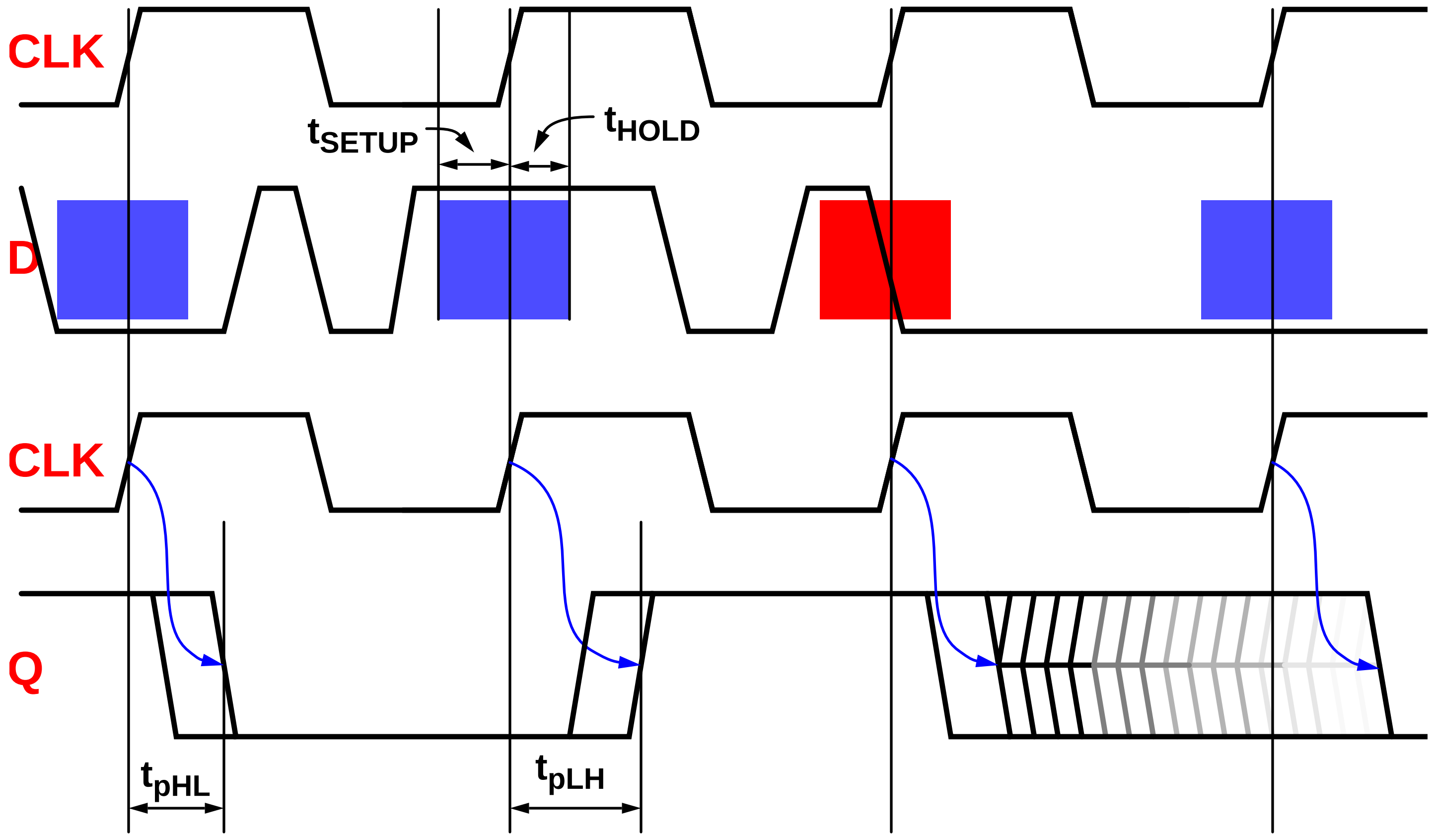
Benefits: latches much faster than registers, use fewer transistors, present lighter load to clock network (leads to lower power consumption).

Bruce Jacob

University of Maryland
ECE Dept.

Some Issues

SET-UP and HOLD time, metastability



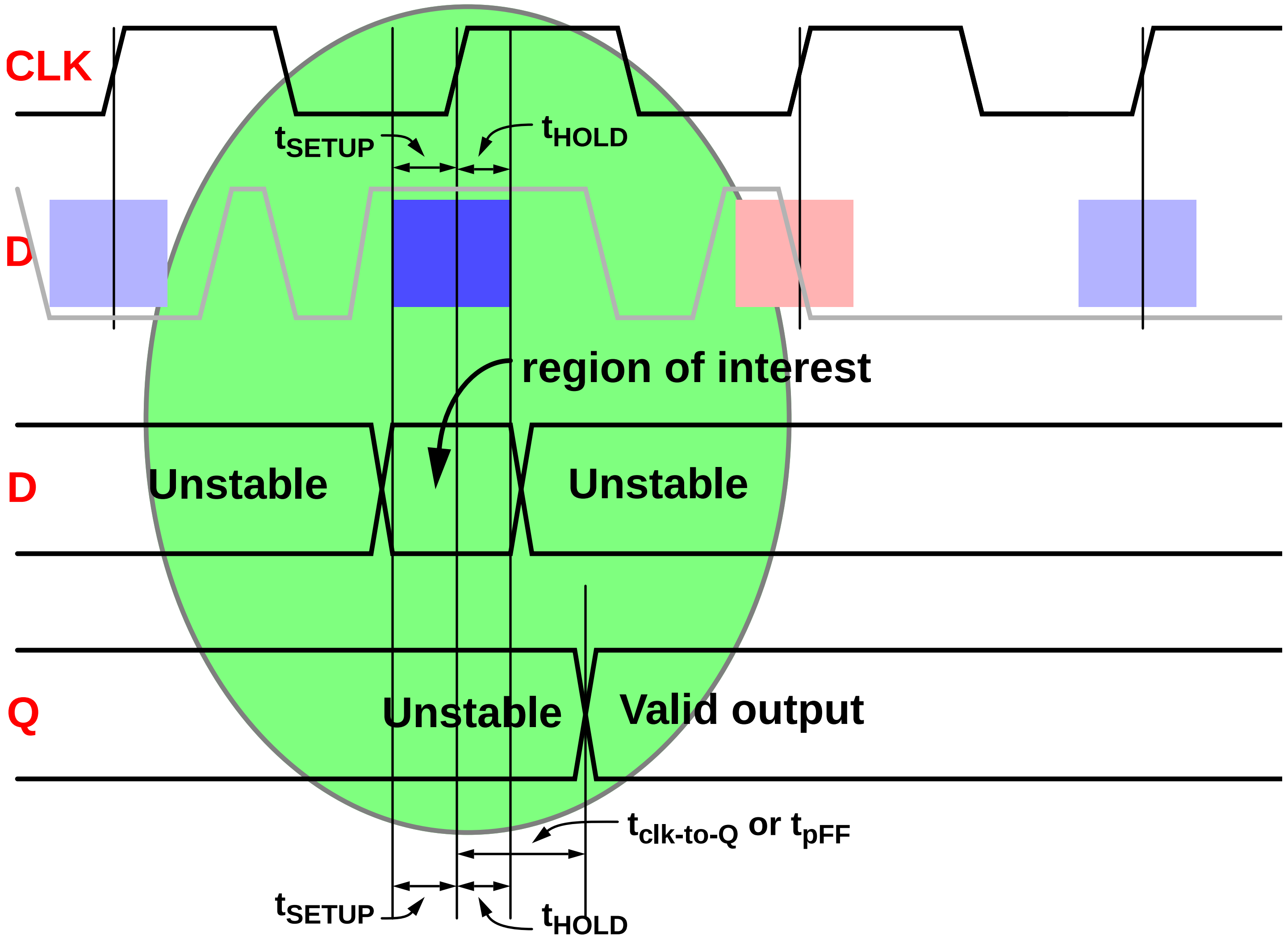
Bruce Jacob

University of
Maryland
ECE Dept.

UNIVERSITY OF MARYLAND

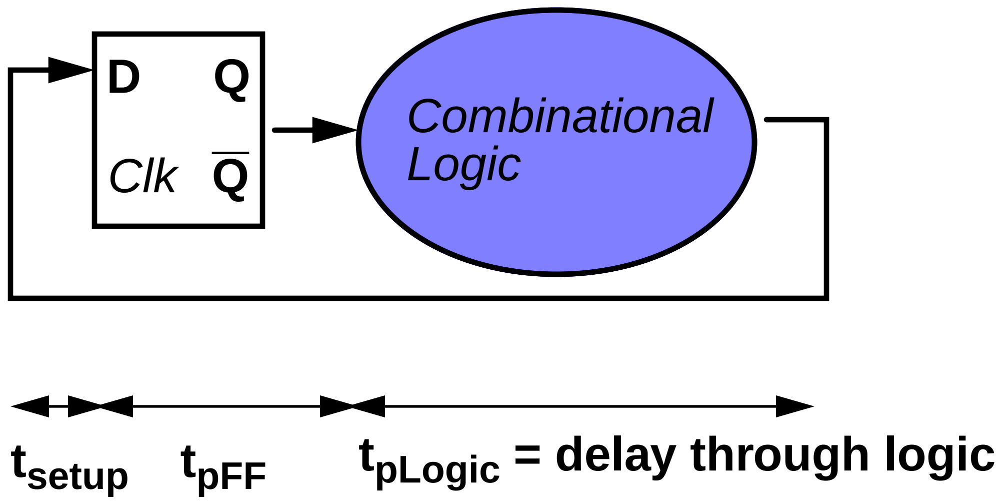
Some Issues

SET-UP and HOLD time, max. clock rate



Some Issues

SET-UP and HOLD time, max. clock rate



Max. clock frequency:

$$T_{min} \geq t_{pFF} + t_{pLogic} + t_{setup}$$

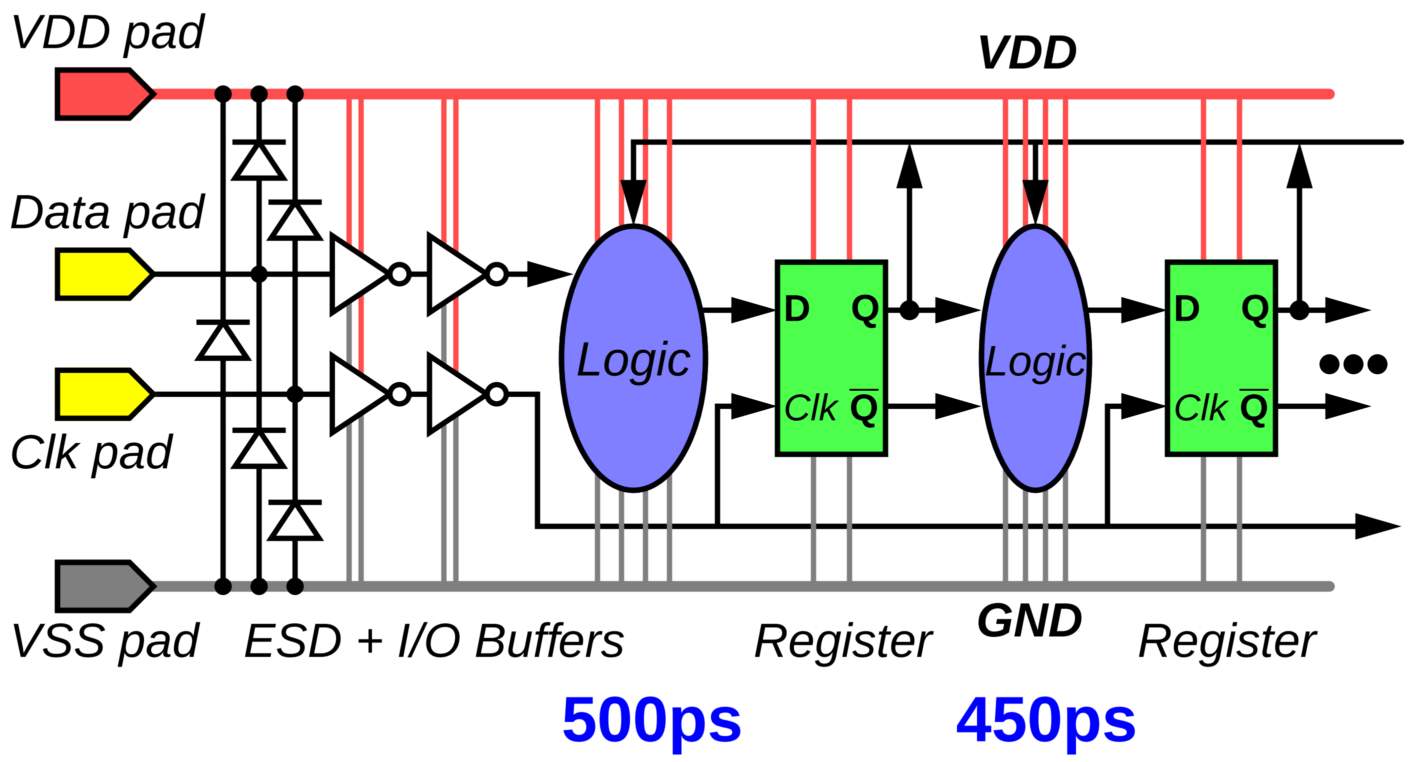
$$F_{max} \leq 1 / (t_{pFF} + t_{pLogic} + t_{setup})$$

Bruce Jacob

University of
Maryland
ECE Dept.

Pipelining

Goal: Increase max. clock rate



Worst-case logic delay: 950ps vs. 500ps

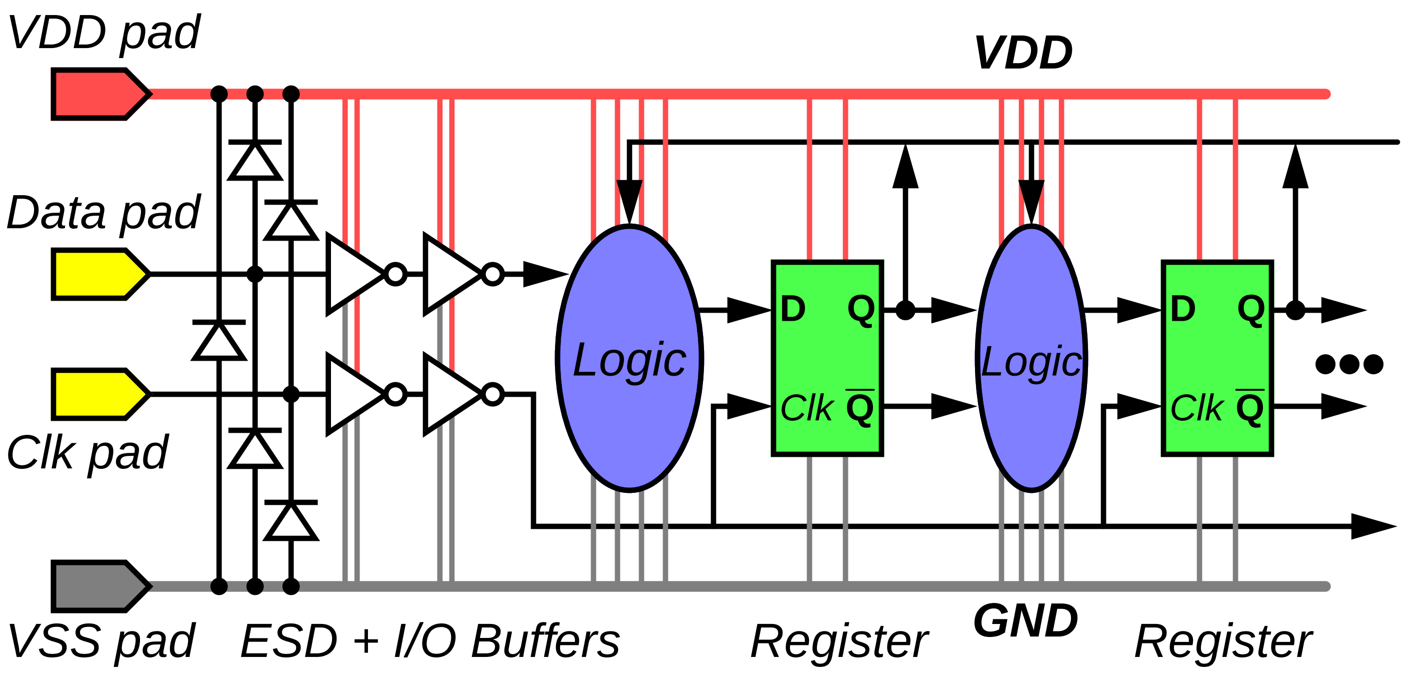
Bruce Jacob

University of
Maryland
ECE Dept.

UNIVERSITY OF MARYLAND

Pipelining

Goal: Decrease power dissipation



500ps, 100pF **450ps, 80pF**

Dynamic Power ~ CV^2*f*

- **Can reduce V_{DD} & f_{MAX} and maintain throughput**