EN1600
Design and Implementation of VLSI Systems
Fall 2016

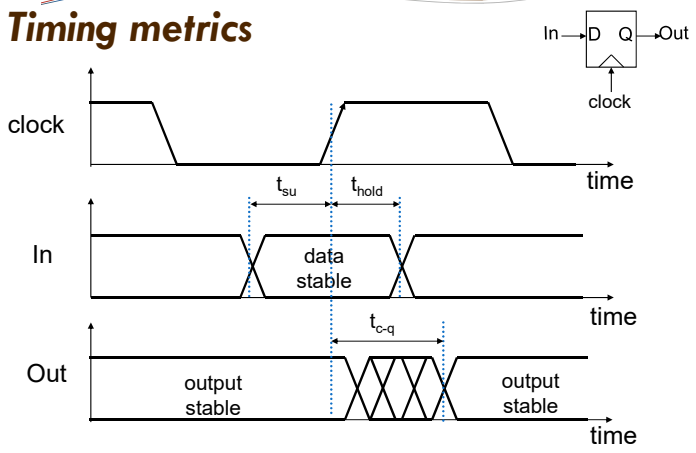
Lecture 13 & 14: static/dynamic latch design & power dissipation

Reading: Chap. 10, sections 10.1-10.3
Chap. 5, sec. 5.1-5.3
Weste & Harris

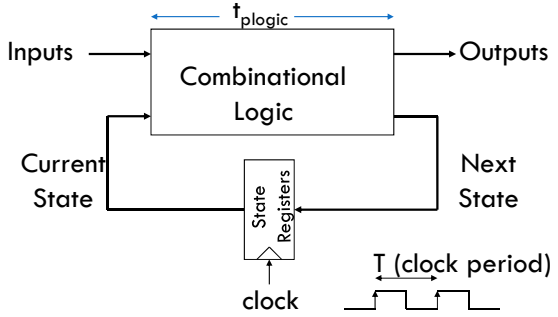
Oct. 31, Nov. 2, 2016
Prof. R. Iris Bahar

BROWN

Timing metrics



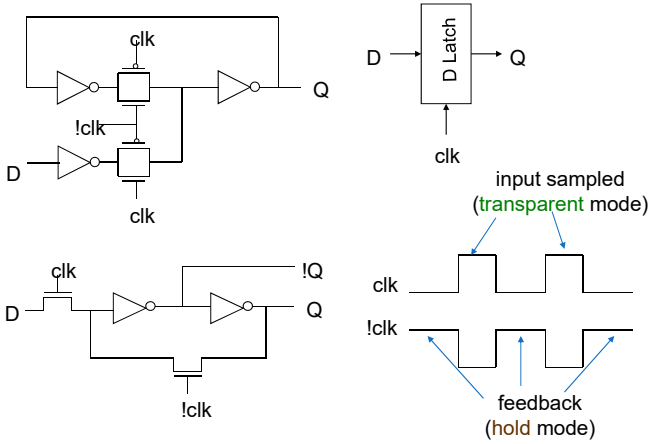
System timing constraints



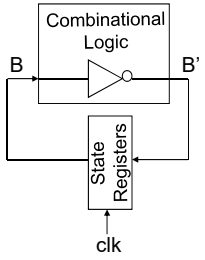
$T \geq t_{c\text{-}q} + t_{plogic} + t_{su}$

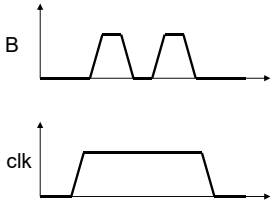
$t_{cdreg} + t_{cdlogic} \geq t_{hold}$

TG MUX based latch implementation



Latch race problem





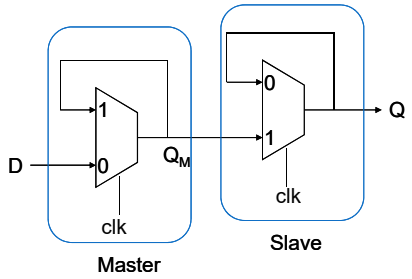
Which value of B is stored?

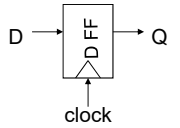
Two-sided clock constraint

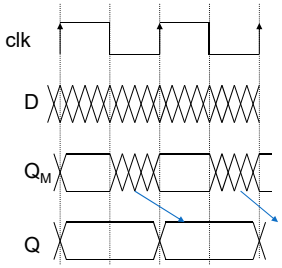
$$T \geq t_{c\text{-}q} + t_{plogic} + t_{su}$$

$$T_{high} < t_{c\text{-}q} + t_{cdlogic}$$

Master slave edge-triggered flip-flop



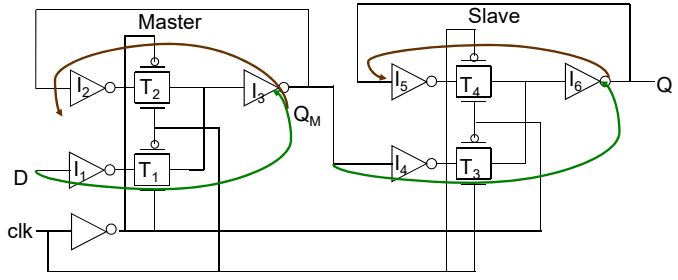


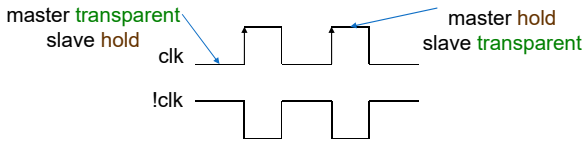


clk = 0 transparent hold

clk = 0→1 hold transparent

Master slave edge-triggered design



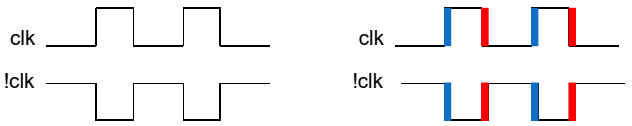


Master slave timing properties

- Assume propagation delays are t_{pd_inv} and t_{pd_tx}, that the contamination delay is 0, and that the inverter delay to derive !clk is 0
- Set-up time - time before rising edge of clk that D must be valid $3 * t_{pd\_inv} + t_{pd\_tx}$
- Propagation delay - time for Q_M to reach Q $t_{pd\_inv} + t_{pd\_tx}$
- Hold time - time D must be stable after rising edge of clk zero

BROWN

Non-Ideal Clocks



Ideal clocks

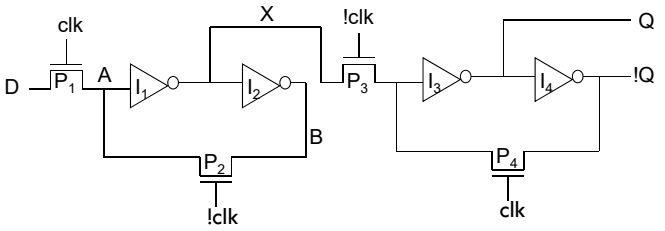
Non-ideal clocks
clock skew

1-1 overlap

0-0 overlap

BROWN

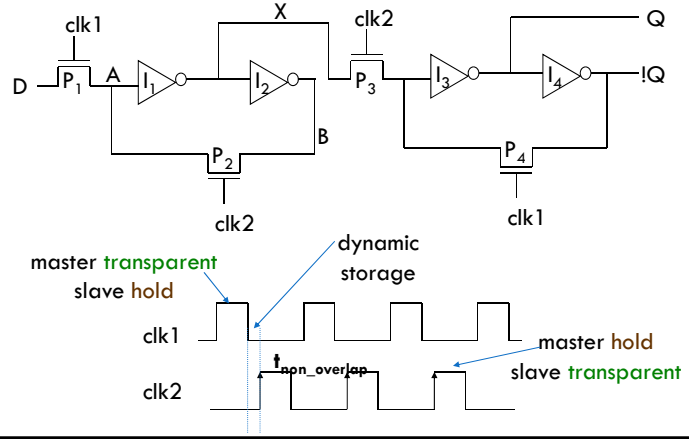
Example of clock skew problems



- Race condition: direct path from D to Q during the short time when both clk and !clk are high (1-1 overlap)
- Undefined state: both B and D are driving A when clk and !clk are both high
- Dynamic storage: when clk and !clk are both low (0-0 overlap)

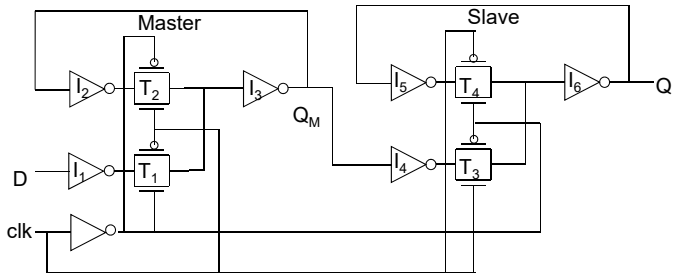
BROWN

Pseudo-static two-phase flip-flop



BROWN

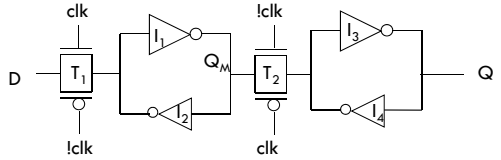
Master slave edge-triggered design



Nice design but requires a lot of transistors

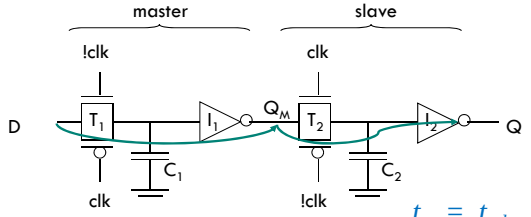
Reduced load flip-flop

- Clock load per register is important since it directly impacts the power dissipation of the clock network.
- Can reduce the clock load (at the cost of robustness) by making the circuit ratioed



- To switch the state of the master, T_1 must be sized to overpower I_2
- To avoid reverse conduction, I_4 must be weaker than I_1.

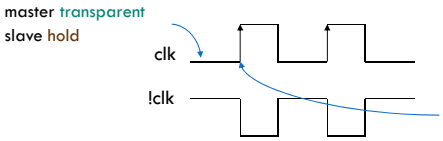
Dynamic edge triggered flip-flop



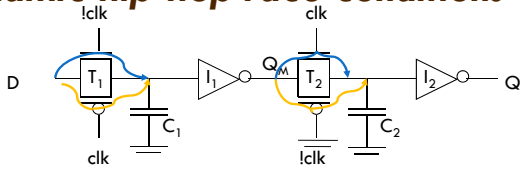
$t_{su} = t_{pd\_tx}$

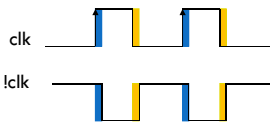
t_{hold} = zero

$t_{c\text{-}q} = 2t_{pd\_inv} + t_{pd\_tx}$



Dynamic flip-flop race conditions





0-0 overlap race condition

$t_{overlap0\text{-}0} < t_{T1} + t_{I1} + t_{T2}$

No change on output should be detected

1-1 overlap race condition

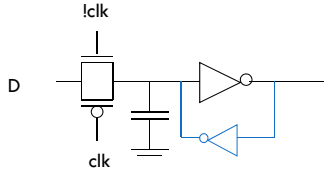
$t_{overlap1\text{-}1} < t_{hold}$

Change at output should reflect stable value

Note that you can have 1-1 or 0-0 overlap on either rising or falling edges

Pseudo-static dynamic latch

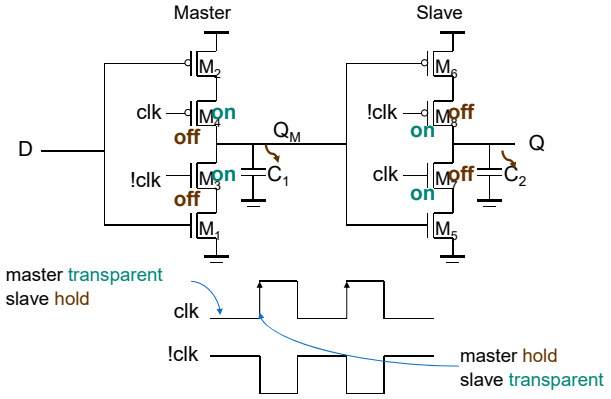
- Robustness considerations limit the use of dynamic FFs
  - coupling between signal nets and internal storage nodes can inject significant noise and destroy the FF state
  - leakage currents cause state to leak away with time
  - internal dynamic nodes don't track fluctuations in V_{DD} that reduces noise margins
- A simple fix is to make the circuit pseudostatic



- Above logic added to all dynamic latches

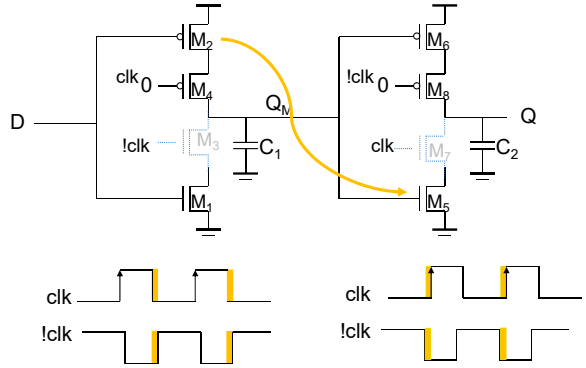
C^2MOS (clocked CMOS) flip-flop

- A clock skew insensitive FF

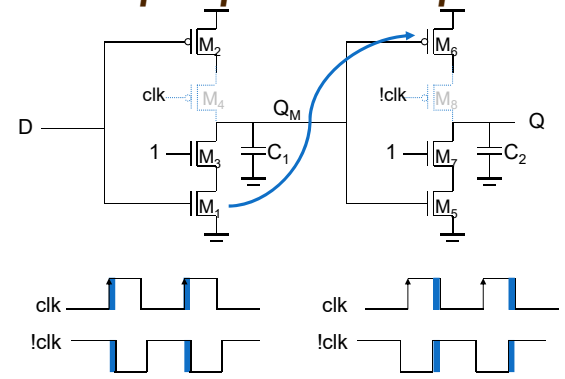


C^2MOS flip-flop 0-0 overlap case

- Clock-skew insensitive as long as the rise and fall times of the clock edges are sufficiently small



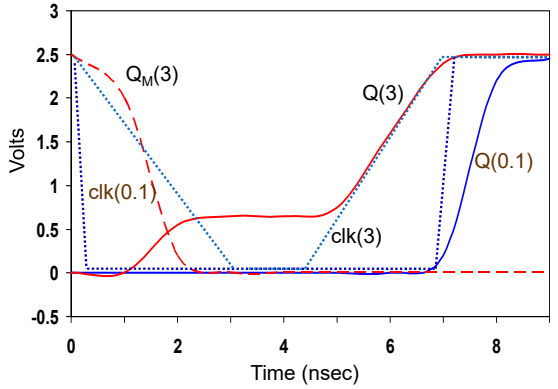
C^2MOS flip-flop 1-1 overlap case



1-1 overlap constraint

$t_{overlap1\text{-}1} < t_{hold}$

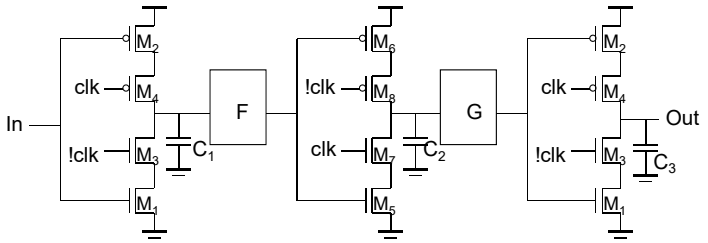
C^2MOS transient response



For a 0.1 ns clock rise time

For a 3 ns clock rise time (race condition exists)

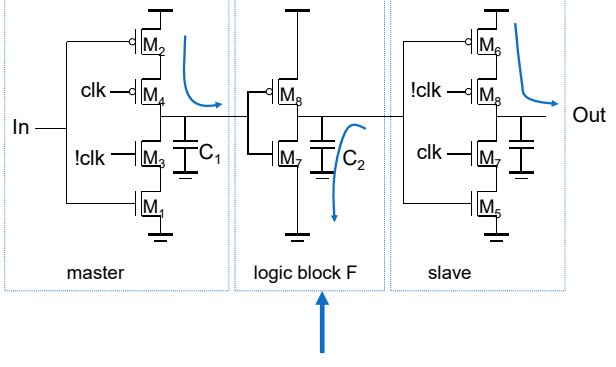
Pipelining using C²MOS



NORA Logic

What are the constraints on F and G to ensure that the FF is race free?

Pipelining using C²MOS



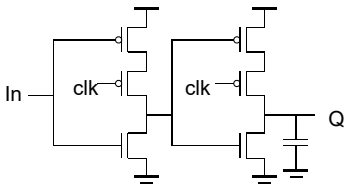
WRONG !!

HW and Lecture for next week

- Homework #4 will be posted later today
  - Due Friday, Nov. 11, byy 5pm
  - Covers latch design and multi-Vdd design
  - 2 big layout component → start early!
- Lecture next week
  - Given by Marco Donato
  - Finish lectures on sources of power dissipation
  - Start to cover memory SRAM and cache design

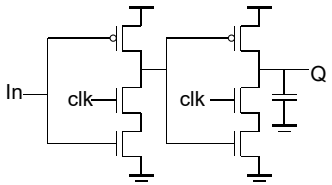
True single phase clocked (TSPC) latches

Negative Latch



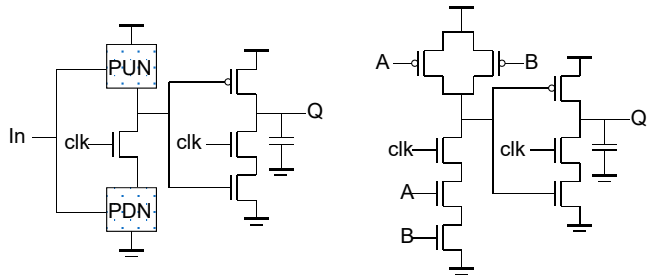
hold when clk = 1
transparent when clk = 0

Positive Latch

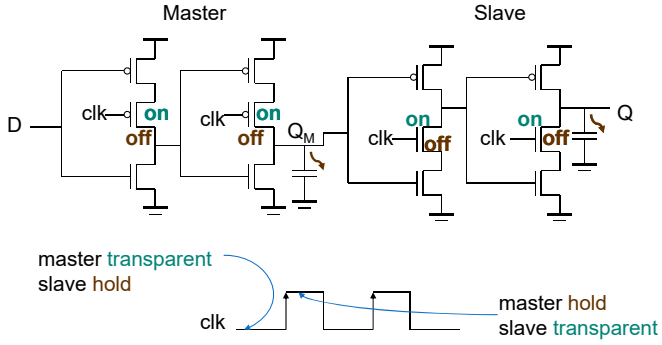


transparent when clk = 1
hold when clk = 0

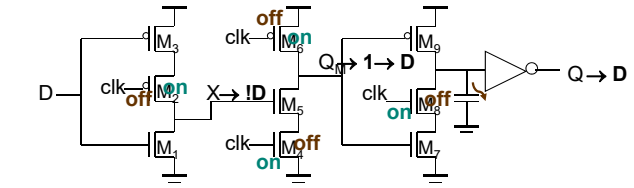
Embedding Logic in TSPC Latch

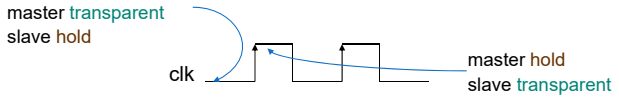


TSPC edge-triggered flip-flop



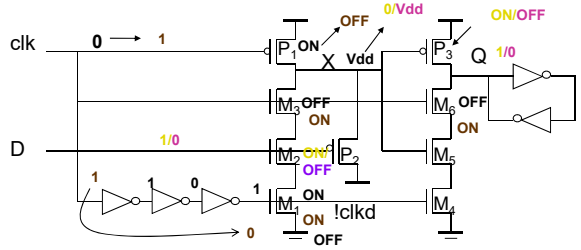
Simplified TSPC edge-triggered FF





Pulsed FF (AMD-K6)

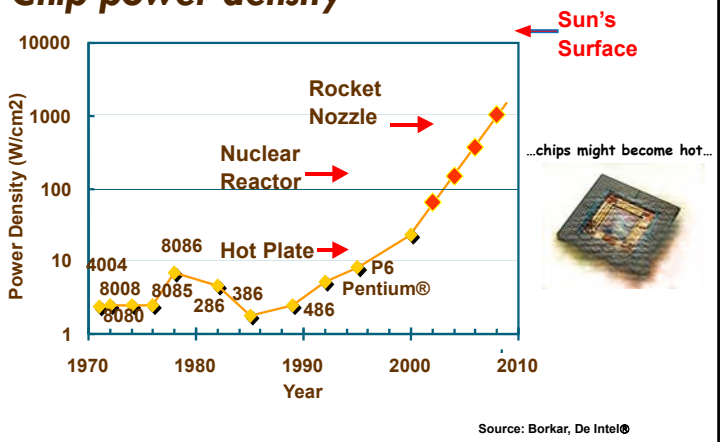
- Pulse registers - a short pulse is generated locally from the rising (or falling) edge of the system clock and is used as the clock input to the flip-flop
  - Race conditions avoided since transparent mode time very short (during the pulse only)
  - Advantage: reduced clock load;
  - Disadvantage: increase in verification complexity



Why power matters

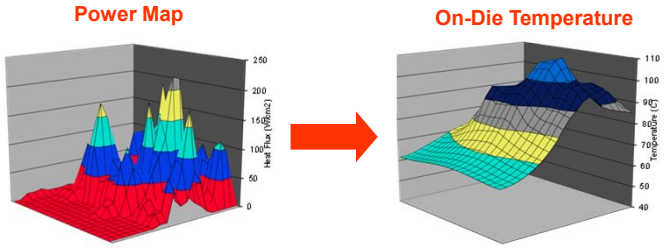
- Packaging costs
- Power supply rail design
- Chip and system cooling costs
- Noise immunity and system reliability
- Battery life (in portable systems)
- Environmental concerns

Chip power density



Source: Borkar, De Intel®

Chip power density distribution

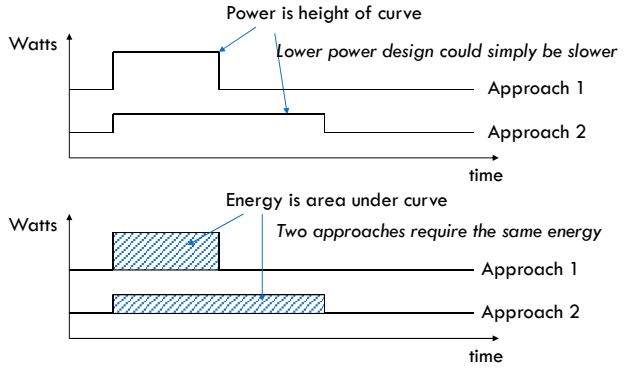


- Power density is not uniformly distributed across the chip
- Silicon is not a good heat conductor
- Max junction temperature is determined by hot-spots
  - Impact on reliability and packaging, w.r.t. cooling

Power and energy figures of merit

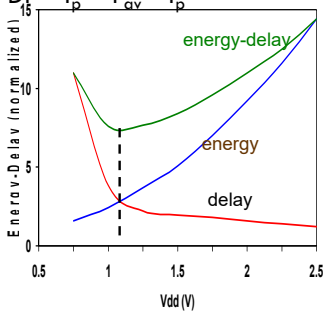
- Power dissipation in Watts
  - determines battery life in hours
  - rate at which energy is taken from Vdd and converted into heat
- Peak power
  - determines power ground wiring designs
  - sets packaging limits
  - impacts signal noise margin and reliability analysis
- Energy efficiency in Joules
  - rate at which power is consumed over time
- Energy = power * delay
  - Joules = Watts * seconds
  - lower energy number means less power to perform a computation at the same frequency

Power versus Energy



PDP and EDP

- Power-delay product (PDP) = $P_{av} * t_p = (C_L V_{DD}^2)/2$
  - PDP is the average energy consumed per switching event (Watts * sec = Joule)
  - lower power design could simply be a slower design
- Energy-delay product (EDP) = PDP $* t_p = P_{av} * t_p^2$
  - EDP is the average energy consumed multiplied by the computation time required
  - takes into account that one can trade increased delay for lower energy/operation (e.g., via supply voltage scaling that increases delay, but decreases energy consumption)



CMOS energy & power equations

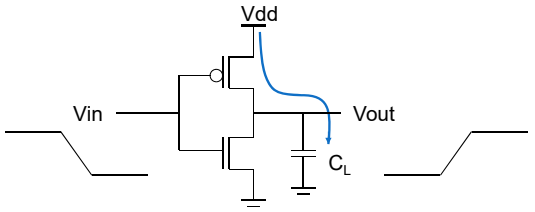
$$E = C_L V_{DD}^2 P_{0\to1} + t_{sc} V_{DD} I_{peak} P_{0\to1} + V_{DD} I_{leakage}$$

$$f_{0\to1} = P_{0\to1} * f_{clock}$$

$$P = C_L V_{DD}^2 f_{0\to1} + t_{sc} V_{DD} I_{peak} f_{0\to1} + V_{DD} I_{leakage}$$

Dynamic power | Short-circuit power | Leakage power

Dynamic power dissipation



Energy/transition = $C_L * V_{DD}^2 * P_{0\to1}$

P_{dyn} = Energy/transition * f = $C_L * V_{DD}^2 *$ $P_{0\to1}$ * f ($f_{0\to1}$)

$P_{dyn} = C_{EFF} * V_{DD}^2 * f$ where $C_{EFF} = P_{0\to1} C_L$

Data dependent → a function of switching activity!

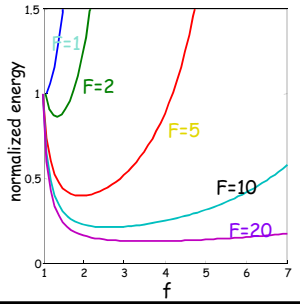
Dynamic power vs. device size

- Device sizing affects dynamic energy consumption
  - gain is largest for networks with large overall effective fan-outs ($F = C_L/C_{g,1}$)
- The optimal gate sizing factor (f) for dynamic energy is smaller than for performance,
  - e.g., for chain of 2 inv., F=20

$$f_{opt}(\text{perf.}) = \sqrt{F} = 4.47, \quad \text{while}$$

$$f_{opt}(\text{energy}) = 3.53$$

- Oversizing has worse implications for energy than for delay

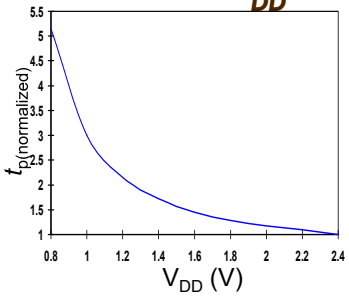


Dynamic power as a function of V_{DD}

- Decreasing V_{DD} decreases dynamic energy consumption (quadratically)
- But, increases gate delay (decreases performance)

$$I_D \approx k\left[\frac{V_{DS}^2}{2}\right] \quad \text{in saturation}$$

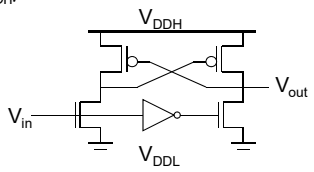
$$t_p = \frac{C_L V_{DD}/2}{I_{DS}} \Rightarrow t_p = \frac{C_L}{kV_{DD}}$$



Determine the critical path(s) at design time and use high V_{DD} for gates on those paths for speed. Use a lower V_{DD} on the other gates, especially those that drive large capacitances (since this yields the largest energy benefits).

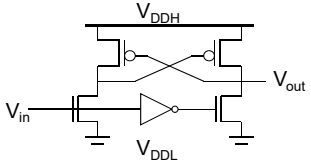
Multiple V_{DD} Considerations

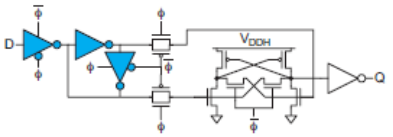
- How many different voltages V_{DDi}? → Two is relatively common
  - Many chips already have two supplies (core and I/O)
- When combining multiple supplies, level converters are required when a gate with lower V_{DD} drives a gate at higher V_{DD}
  - If gate at V_{DDL} drives a gate at V_{DDH}, the PMOS never turns off
    - cross-coupled PMOS transistors do the level conversion
    - NMOS transistors operate on a reduced supply
  - Level converters are not needed for a step-down change in voltage
  - Overhead of level converters can be mitigated by doing conversions at register boundaries and embedding conversion inside the flip-flop



Multiple V_{DD} Considerations

- cross-coupled PMOS transistors do the level conversion
- NMOS transistors operate on a reduced supply



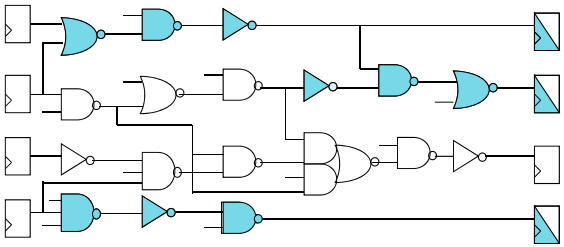


See section 10.4.4 (pg. 409) for examples of level-converter flip-flops

BROWN

Dual-supply inside a logic block

- Minimum energy consumption achieved if all logic paths are critical (have the same delay). *Why?*
- Clustered voltage-scaling
  - Each path starts with V_{DDH} and switches to V_{DDL} (gray logic gates) when delay slack is available
  - Level conversion is done in the flip flops at the end of the paths
  - Use standard cells to construct the dual-V_{DD} function block



BROWN

HW#4 assignment

- Design standard cells that can connect to VDDH or VDDL