



## *EE141-Spring 2010 Digital Integrated Circuits*

### Lecture 19 Timing

EECS141

Lecture #19

1

### *Administrativa*

- ❑ Project Phase 2 now on the web-site.
- ❑ Hw 6 due today.
- ❑ New homework to be posted in a week.
- ❑ Cory Hall closed on Monday (Power Outage)
  - Instructional computers in 353 Cory should come back on line on Tu.
- ❑ Enjoy Spring Break!

EECS141

Lecture #19

2

## *Class Material*

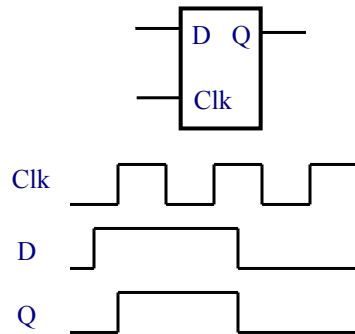
- Last lecture
  - Registers
- Today's lecture
  - Timing
- Reading (Ch 10)



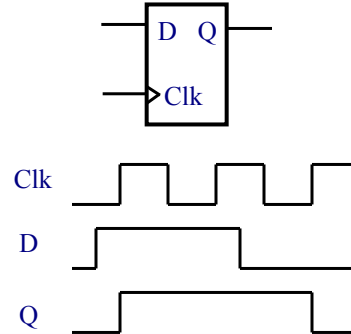
## *Timing*

## Latch versus Register (Flip-flop)

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



- ◆ **Register: edge-triggered**  
stores data when  
clock rises

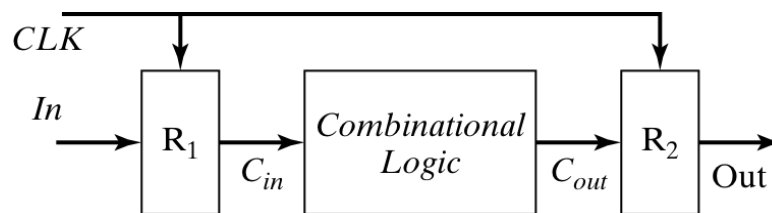


EECS141

Lecture #16

5

## Synchronous Timing

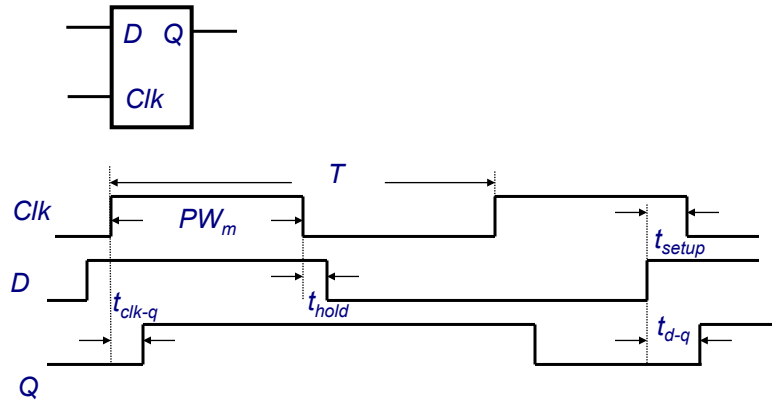


EECS141

Lecture #16

6

## Latch Parameters



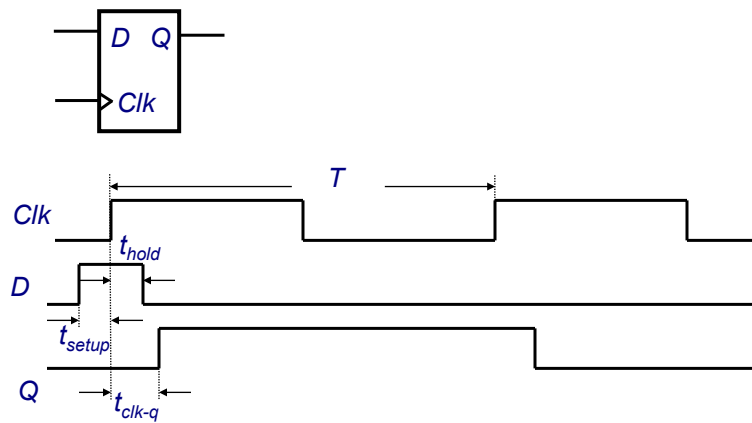
*Delays can be different for rising and falling data transitions*

EECS141

Lecture #16

7

## Register Parameters



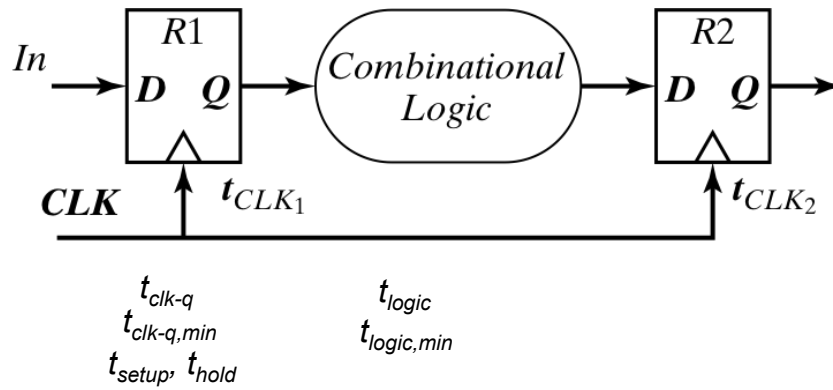
*Delays can be different for rising and falling data transitions*

EECS141

Lecture #16

8

## Timing Constraints

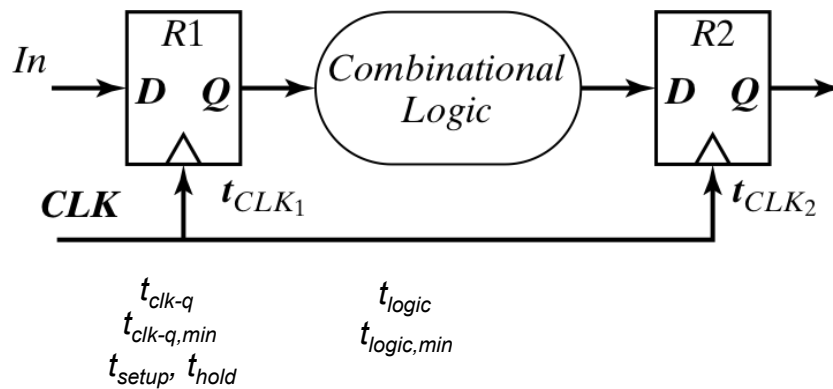


EECS141

Lecture #16

9

## Timing Constraints



Cycle time (max):  $T_{Clk} > t_{clk-q} + t_{logic} + t_{setup}$

Race margin (min):  $t_{hold} < t_{clk-q,min} + t_{logic,min}$

EECS141

Lecture #16

10

## Clock Nonidealities

### □ Clock skew

- Spatial variation in temporally equivalent clock edges; deterministic + random,  $t_{SK}$

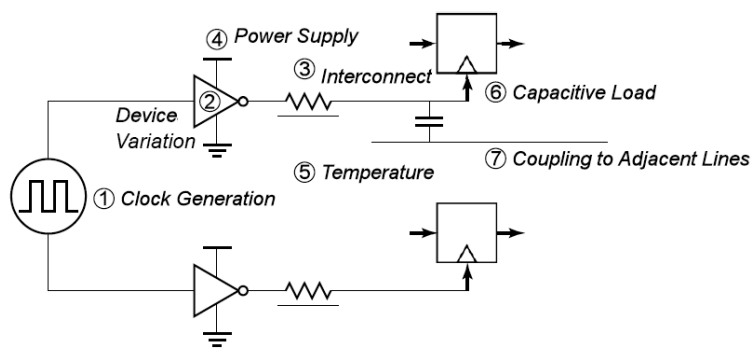
### □ Clock jitter

- Temporal variations in consecutive edges of the clock signal; modulation + random noise
- Cycle-to-cycle (short-term)  $t_{JS}$
- Long term  $t_{JL}$

### □ Variation of the pulse width

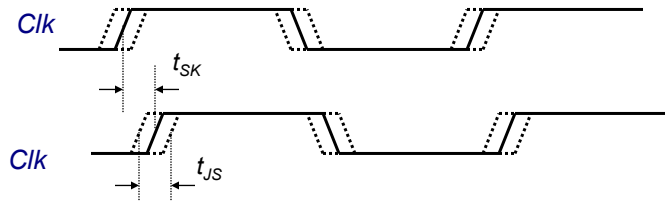
- Important for level sensitive clocking

## Clock Uncertainties



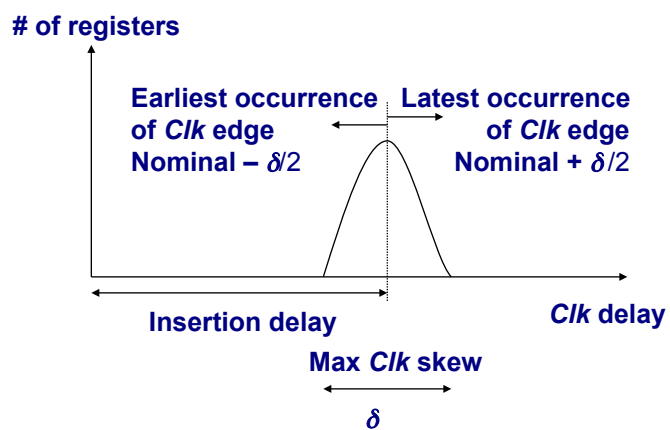
*Sources of clock uncertainty*

## Clock Skew and Jitter

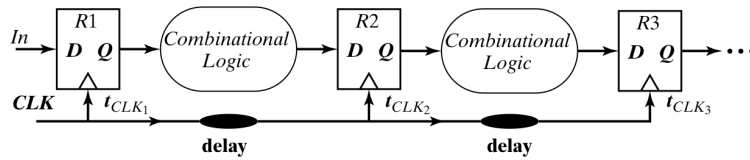


- Both skew and jitter affect the effective cycle time and the race margin

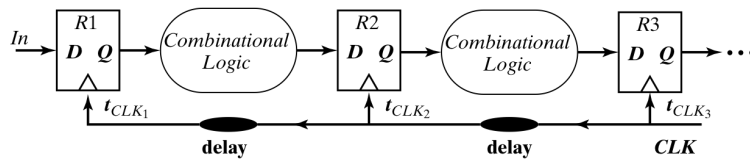
## Clock Skew



## Positive and Negative Skew

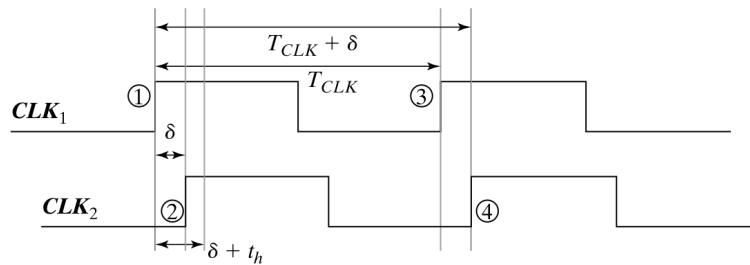


(a) Positive skew



(b) Negative skew

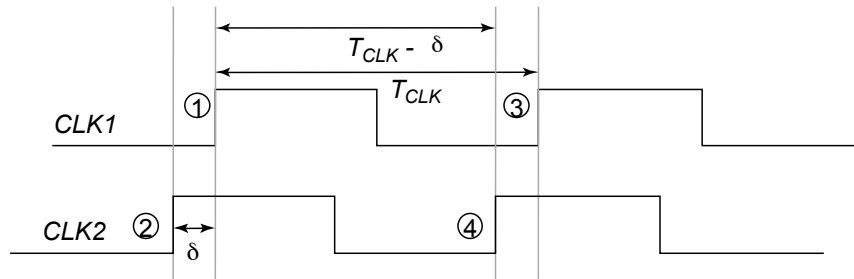
## Positive Skew



*Launching edge arrives before the receiving edge*

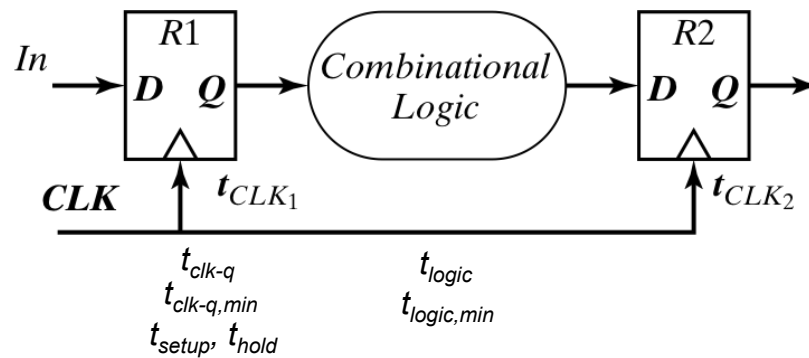


## Negative Skew



*Receiving edge arrives before the launching edge*

## Timing Constraints

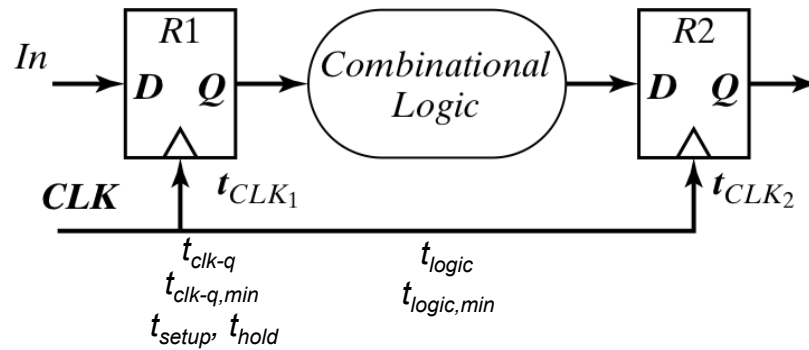


*Minimum cycle time:*

$$T_{clk} + \delta = t_{clk-q} + t_{setup} + t_{logic}$$

Worst case is when receiving edge arrives early (negative  $\delta$ )

## Timing Constraints



*Hold time constraint:*

$$t_{(clk-q,min)} + t_{(logic,min)} > t_{hold} + \delta$$

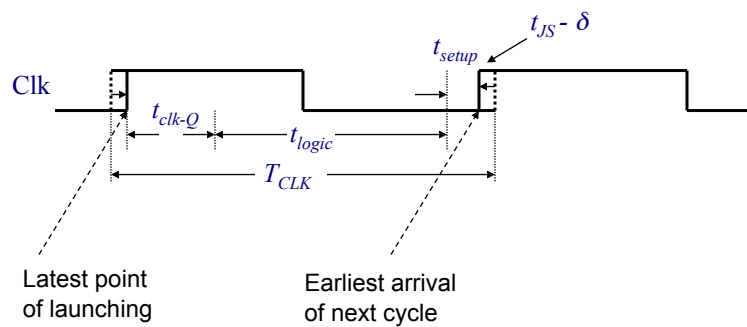
Worst case is when receiving edge arrives late  
Race between data and clock

EECS141

Lecture #16

19

## Longest Logic Path in Edge-Triggered Systems



EECS141

Lecture #16

20

## Clock Constraints in Edge-Triggered Systems

If launching edge is late and receiving edge is early, the data will not be too late if:

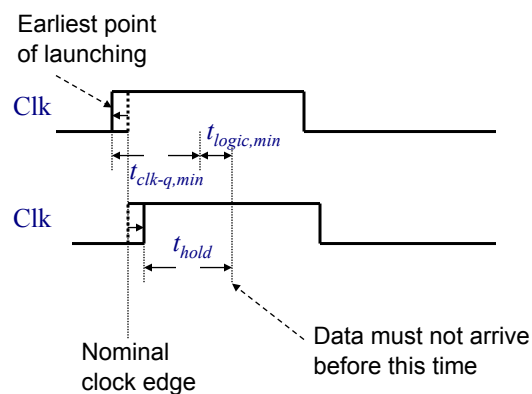
$$t_{clk-q} + t_{logic} + t_{setup} < T_{CLK} - t_{JS,1} - t_{JS,2} + \delta$$

Minimum cycle time is determined by the maximum delays through the logic

$$t_{clk-q} + t_{logic} + t_{setup} - \delta + 2t_{JS} < T_{CLK}$$

**Skew can be either positive or negative**

## Shortest Path



## Clock Constraints in Edge-Triggered Systems

If launching edge is early and receiving edge is late:

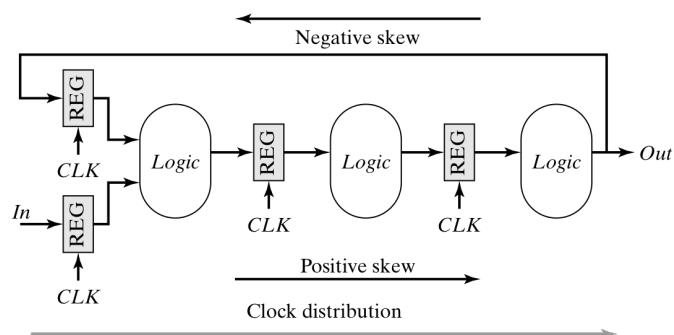
$$t_{clk-q,min} + t_{logic,min} - t_{JS,1} > t_{hold} + t_{JS,2} + \delta$$

Minimum logic delay

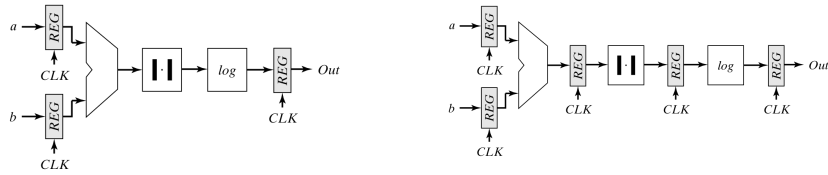
$$t_{clk-q,min} + t_{logic,min} > t_{hold} + 2t_{JS} + \delta$$

(This assumes jitter at launching and receiving clocks are independent – which usually is not true)

## Datapath with Feedback



## Pipelining



Reference

Clock Period	Adder	Absolute Value	Logarithm
1	$a_1 + b_1$		
2	$a_2 + b_2$	$ a_1 + b_1 $	
3	$a_3 + b_3$	$ a_2 + b_2 $	$\log( a_1 + b_1 )$
4	$a_4 + b_4$	$ a_3 + b_3 $	$\log( a_2 + b_2 )$
5	$a_5 + b_5$	$ a_4 + b_4 $	$\log( a_3 + b_3 )$

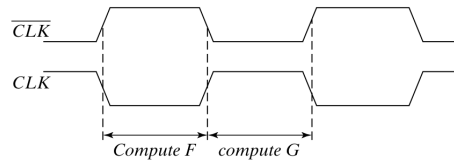
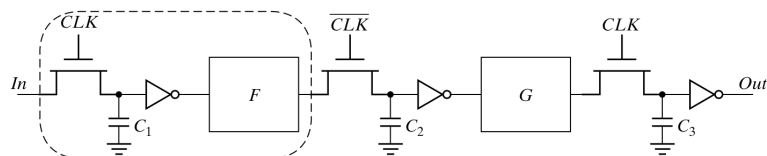
Pipelined

EECS141

Lecture #16

25

## Latch-Based Clocking



(Domino logic almost always uses latch-based clocking)

EECS141

Lecture #16

26

## Latch vs. Flip-flop

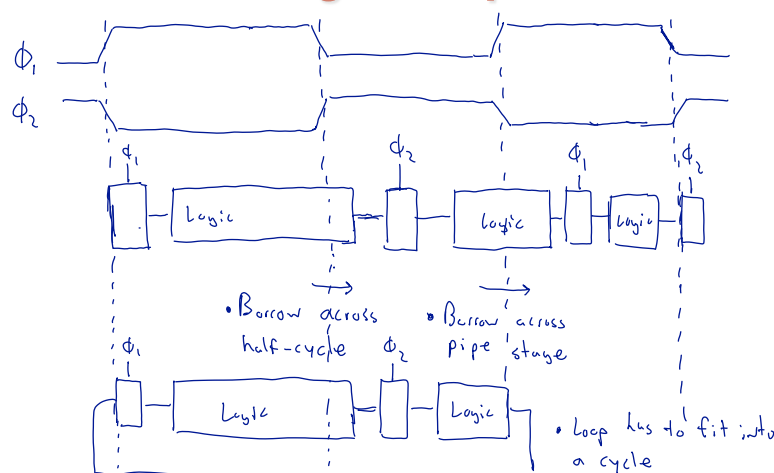
- In a flip-flop based system:
  - Data launches on one rising edge
    - And must arrive before next rising edge
  - If data arrives late, system fails
    - If it arrives early, wasting time
  - Flip-flops have hard edges
- In a latch-based system:
  - Data can pass through latch while it is transparent
  - Long cycle of logic can borrow time into next cycle
    - As long as each loop finished in one cycle

EECS141

Lecture #16

27

## Time Borrowing Example



EECS141

Lecture #16

28

## *Latch vs. Flip-flop Summary*

- ❑ Flip-flops generally easier to use
  - Most digital ASICs designed with register-based timing
- ❑ But, latches (both pulsed and level-sensitive) allow more flexibility
  - And hence can potentially achieve higher performance
  - Latches can also be made more tolerant of clock un-certainty
  - More in EE241

## *CMOS Transistor Scaling*



## Goals of Technology Scaling

- ❑ Make things cheaper:
  - Want to sell more functions (transistors) per chip for the same money
  - Or build same products cheaper
  - Price of a transistor has to be reduced
- ❑ But also want to be faster, smaller, lower power...

EECS141

Lecture #16

31

## Technology Scaling

- ❑ Benefits of 30% “Dennard” scaling (1974):
  - Double transistor density
  - Reduce gate delay by 30% (increase operating frequency by 43%)
  - Reduce energy per transition by 65% (50% power savings @ 43% increase in frequency)
- ❑ Die size used to increase by 14% per generation (not any more)
- ❑ Technology generation spans 2-3 years

EECS141

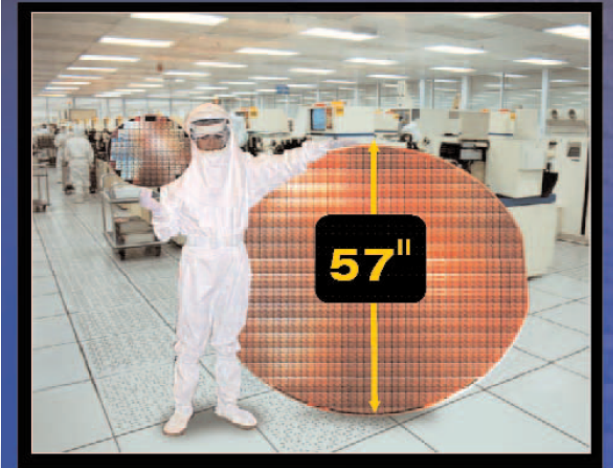
Lecture #16

32



## *Moore was not always accurate*

### **Projected 2000 Wafer, circa 1975**

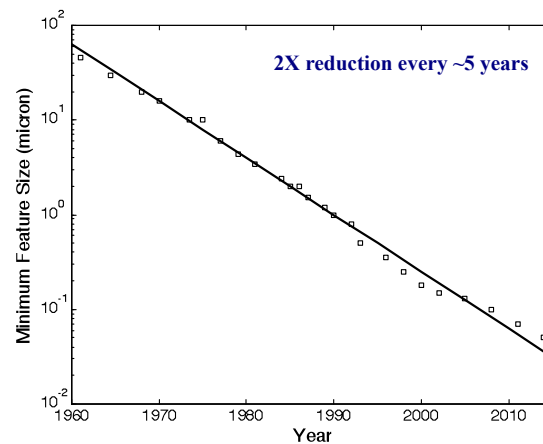


EECS141

Lecture #16

33

## *Technology Scaling (1)*

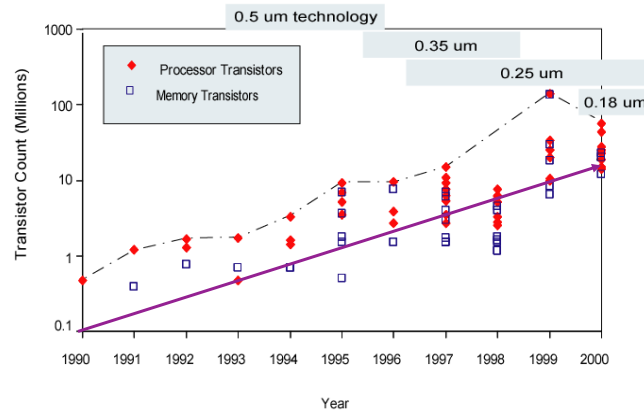
**Minimum Feature Size**

EECS141

Lecture #16

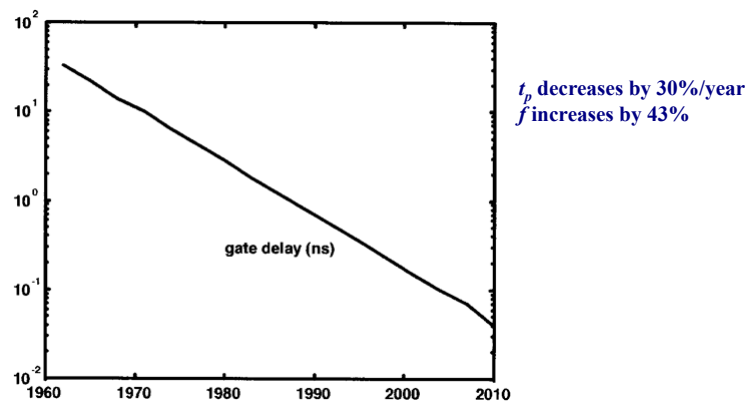
34

## Technology Scaling (2)



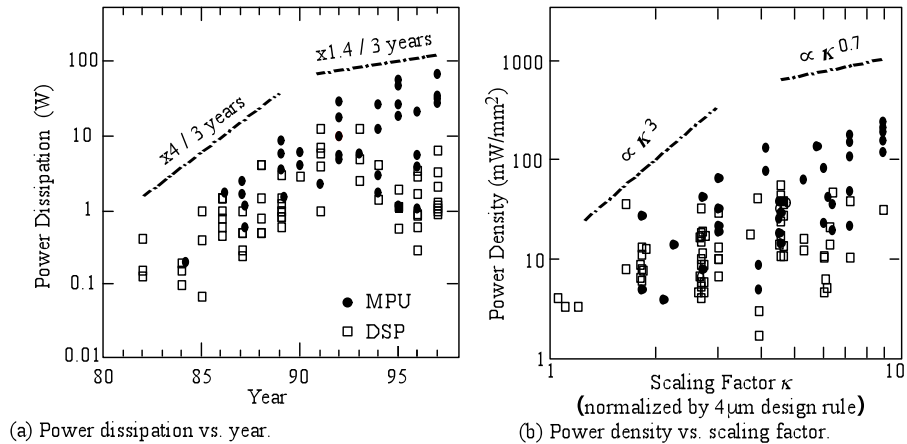
**Number of components per chip**

## Technology Scaling (3)



**Propagation Delay**

## Technology Scaling (4)



From Kuroda

EECS141

Lecture #16

37

## Technology Scaling Models

- **Full Scaling (Constant Electrical Field)**

ideal model — dimensions and voltages scale together by the same factor  $S$

- **Fixed Voltage Scaling**

most common model until 1990's  
only dimensions scale, voltages remain constant

- **General Scaling**

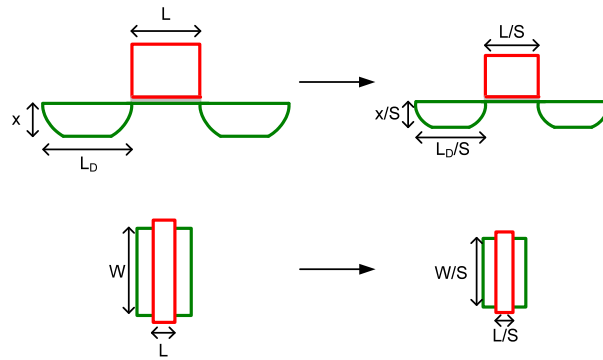
most realistic for today's situation —  
voltages and dimensions scale with different factors

EECS141

Lecture #16

38

## Scaling



EECS141

Lecture #16

39

## Full Scaling (Dennard, Long-Channel)

- $W, L, t_{ox}$ :  $1/S$
- $V_{DD}, V_T$ :  $1/S$
- Area:  $WL$
- $C_{ox}$ :  $1/tox$
- $C_L$ :  $C_{ox}WL$
- $I_D$ :  $C_{ox}(W/L)(V_{DD}-V_T)^2$
- $R_{eq}$ :  $V_{DD}/I_{DSAT}$

EECS141

Lecture #16

40

## Full Scaling (Dennard, Long-Channel)

- $W, L, t_{ox}: 1/S$
- $V_{DD}, V_T: 1/S$
- $t_p: R_{eq} C_L$
- $P_{avg}: C_L V_{DD}^2 / t_p$
- $P_{avg}/A: C_{ox} V_{DD}^2 / t_p$

## Scaling Relationships for Long Channel Devices

Parameter	Relation	Full Scaling	General Scaling	Fixed Voltage Scaling
$W, L, t_{ox}$		$1/S$	$1/S$	$1/S$
$V_{DD}, V_T$		$1/S$	$1/U$	$1$
$N_{SUB}$	$V/W_{depl}^2$	$S$	$S^2/U$	$S^2$
Area/Device	$WL$	$1/S^2$	$1/S^2$	$1/S^2$
$C_{ox}$	$1/t_{ox}$	$S$	$S$	$S$
$C_L$	$C_{ox}WL$	$1/S$	$1/S$	$1/S$
$k_n, k_p$	$C_{ox}W/L$	$S$	$S$	$S$
$I_{av}$	$k_{n,p} V^2$	$1/S$	$S/U^2$	$S$
$t_p$ (intrinsic)	$C_L V / I_{av}$	$1/S$	$U/S^2$	$1/S^2$
$P_{av}$	$C_L V^2 / t_p$	$1/S^2$	$S/U^3$	$S$
PDP	$C_L V^2$	$1/S^3$	$1/SU^2$	$1/S$

## *Full Scaling (Dennard, Short-Channel)*

- $W, L, t_{ox}: 1/S$
- $V_{DD}, V_T: 1/S$
  
- Area:  $WL$
- $C_{ox}: 1/tox$
- $C_L: C_{ox}WL$
- $I_D: WC_{ox}v_{sat}(V_{DD}-V_T-V_{SAT}/2)$
- $R_{eq}: V_{DD}/I_{DSAT}$

EECS141

Lecture #16

43

## *Full Scaling (Dennard, Short-Channel)*

- $W, L, t_{ox}: 1/S$
- $V_{DD}, V_T: 1/S$
  
- $t_p: R_{eq}C_L$
- $P_{avg}: C_L V_{DD}^2/t_p$
- $P_{avg}/A: C_{ox} V_{DD}^2/t_p$

EECS141

Lecture #16

44

## Transistor Scaling (Velocity-Saturated Devices)

Parameter	Relation	Full Scaling	General Scaling	Fixed-Voltage Scaling
$W, L, t_{ox}$		$1/S$	$1/S$	$1/S$
$V_{DD}, V_T$		$1/S$	$1/U$	1
$N_{SUB}$	$V/W_{dop}^2$	$S$	$S^2/U$	$S^2$
Area/Device	$WL$	$1/S^2$	$1/S^2$	$1/S^2$
$C_{ox}$	$1/t_{ox}$	$S$	$S$	$S$
$C_{gate}$	$C_{ox}WL$	$1/S$	$1/S$	$1/S$
$k_n, k_p$	$C_{ox}W/L$	$S$	$S$	$S$
$I_{sat}$	$C_{ox}WV$	$1/S$	$1/U$	1
Current Density	$I_{sat}/Area$	$S$	$S^2/U$	$S^2$
$R_{on}$	$V/I_{sat}$	1	1	1
Intrinsic Delay	$R_{on}C_{gate}$	$1/S$	$1/S$	$1/S$
$P$	$I_{sat}V$	$1/S^2$	$1/U^2$	1
Power Density	$P/Area$	1	$S^2/U^2$	$S^2$

EECS141

Lecture #16

45

## An interesting question

- ❑ What ~~will~~ did cause this model to break?
  - Leakage set by  $kT/q$ 
    - Temp. does not scale
    - $V_T$  set to minimize power
  - Power actually increased
    - Leakage increased drastically
    - $f$  increased faster than device speed
    - Hit cooling limit
  - Process Variation
    - Hard to build very small things accurately (less averaging)

EECS141

Lecture #16

46