EE 330
Lecture 44

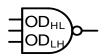## Digital Circuits

- Logic Effort
- Elmore Delay
- Power Dissipation
- Other Logic Styles
- Dynamic Logic Circuits

Propagation Delay in Multiple-Levels of Logic with Stage Loading

Asymmetric-sized gates

C_{IN}/C_{REF}	Equal Rise/Fall	Equal Rise/Fall (with OD)	Minimum Sized	Asymmetric OD (OD_{HL}, OD_{LH})
Inverter	1	OD	1/2	$\dfrac{OD_{HL}+3 \bullet OD_{LH}}{4}$
NOR	$\dfrac{3k+1}{4}$	$\dfrac{3k+1}{4} \bullet OD$	1/2	$\dfrac{OD_{HL}+3k \bullet OD_{LH}}{4}$
NAND	$\dfrac{3+k}{4}$	$\dfrac{3+k}{4} \bullet OD$	1/2	$\dfrac{k \bullet OD_{HL}+3 \bullet OD_{LH}}{4}$
Overdrive				
Inverter HL	1	OD	1	OD_{HL}
LH	1	OD	1/3	OD_{LH}
NOR HL	1	OD	1	OD_{HL}
LH	1	OD	1/(3k)	OD_{LH}
NAND HL	1	OD	1/k	OD_{HL}
LH	1	OD	1/3	OD_{LH}
t_{PROP}/t_{REF}	$\displaystyle\sum_{k=1}^{n} F_{l(k+1)}$	$\displaystyle\sum_{k=1}^{n} \dfrac{F_{l(k+1)}}{OD_k}$	$\dfrac{1}{2}\displaystyle\sum_{k=1}^{n} F_{l(k+1)}\left(\dfrac{1}{OD_{HLk}} + \dfrac{1}{OD_{LHk}}\right)$	$\dfrac{1}{2}\displaystyle\sum_{k=1}^{n} F_{l(k+1)}\left(\dfrac{1}{OD_{HLk}} + \dfrac{1}{OD_{LHk}}\right)$

$$t_{PROP} = t_{REF} \bullet \left(\frac{1}{2}\sum_{k=1}^{5} F_{l(k+1)}\left(\frac{1}{OD_{HLk}} + \frac{1}{OD_{LHk}}\right)\right)$$

Propagation Delay in "Logic Effort" approach

$$t_{PROP} = \sum_{k=1}^{n} f_k = \sum_{k=1}^{n} g_k h_k = \sum_{k=1}^{n} \frac{F_{l(k+1)}}{OD_k}$$

- **Note with the exception of the t_{REF} scaling factor, this expression is identical to what we have derived previously**

- **Probably more tedious to use the "Logical Effort" approach**

- **Extensions to asymmetric overdrive factors may not be trivial**

- **Extensions to include parasitics may be tedious as well**

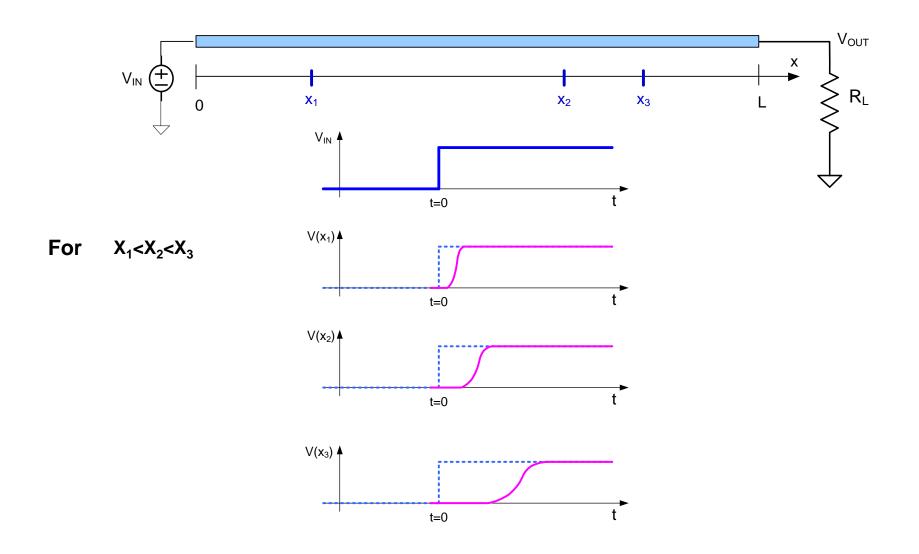- **Logical Effort is widely used throughout the industry**

Elmore Delay Calculations

- **Interconnects have a distributed resistance and a distributed capacitance**
  - **Often modeled as resistance/unit length and capacitance per unit length**

- **These delay the propagation of the signal**

- **Effectively a transmission line**
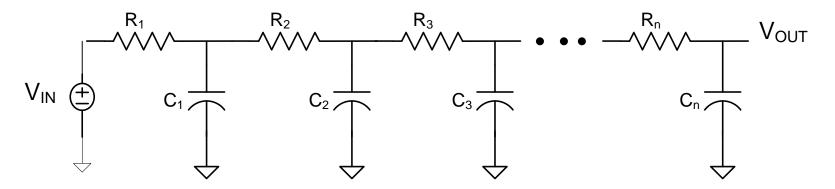 - **analysis is really complicated**

- **Can have much more complicated geometries**

Elmore Delay Calculations

For $X_1 < X_2 < X_3$

Elmore Delay Calculations

Elmore delay:
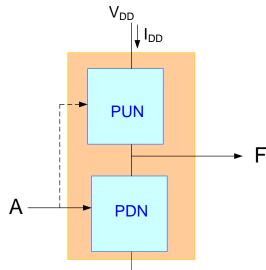$$t_{PD} = \sum_{i=1}^{n} \left(C_i \sum_{j=1}^{i} R_j \right)$$

- **It can be shown that this is a reasonably good approximation to the actual delay**
- **Numbering is critical** (resistors and capacitors numbered from input to output)
- **As stated, only applies to this specific structure**

Power Dissipation in Logic Circuits

Types of Power Dissipation

- **Static**
- **Pipe**
- **Dynamic**
- **Leakage**
    - **Gate**
    - **Diffusion**
    - **Drain**

Static Power Dissipation
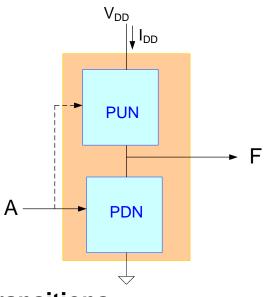
V_{DD}

$\downarrow I_{DD}$

PUN

F

A

PDN

If Boolean output averages H and L 50% of the time

$$P_{STAT,AVG} = \frac{P_H + P_L}{2}$$

$$P_{STAT,AVG} = \frac{V_{DD}(I_{DDH} + I_{DDL})}{2}$$

- **Generally decreases with V_{DD}**
- **$I_{DDH} = I_{DDL} = 0$ for static CMOS gates so $P_{STAT} = 0$**
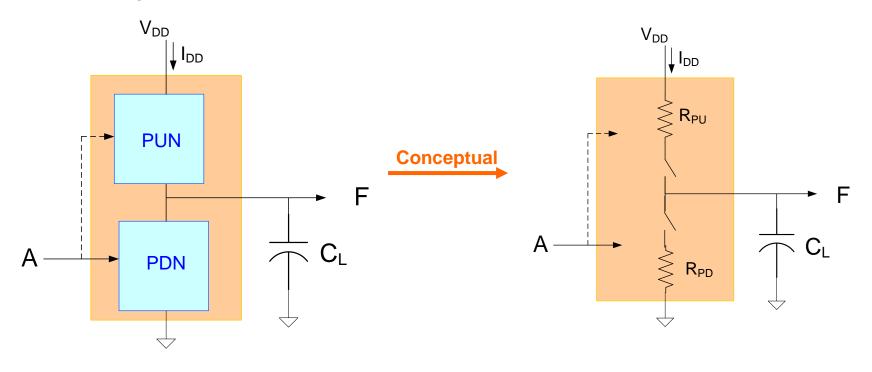- **A major source of power dissipation in ratio logic circuits and the major reason CMOS is so widely used**

Pipe Power Dissipation

V$_{DD}$

I$_{DD}$

PUN

F

A

PDN

Due to conduction of both PUN and PDN during transitions

- **Can be made small if transitions are fast**

- **Usually negligible in Static CMOS circuits**

Dynamic Power Dissipation

Due to charging and discharging C$_L$ on logic transitions

C$_L$ dissipates no power but PUN and PDN dissipate power during charge and discharge of C$_L$
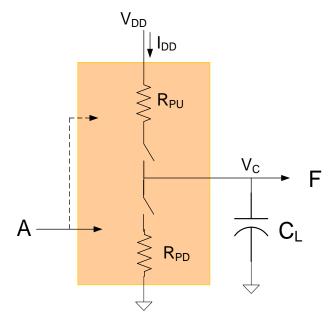
C$_L$ includes all gate input capacitances of loads and interconnect capacita

Dynamic Power Dissipation

Energy supplied by V_{DD} and dissipated in R_{PU} when C_L charges

$$E_{DIS} = \frac{1}{2} C_L V_{DD}^2$$

Energy stored on C_L after L-H transition

$$E_{STORE} = \frac{1}{2} C_L V_{DD}^2$$

Thus, energy from V_{DD} for one L-H: H-L output transition sequence is

$$E = E_{DIS} + E_{STORE} = C_L V_{DD}^2$$

When the output transitions from H to L, energy stored on C_L is dissipated in PDN

If f is the average transition rate of the output, determine P_{AVG}

Dynamic Power Dissipation
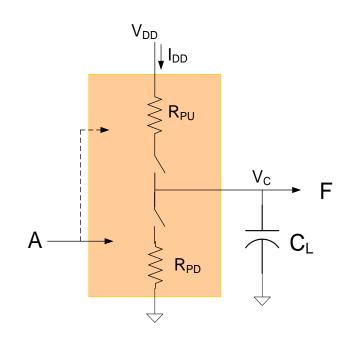
Energy from V_{DD} for one L-H: H-L output transition sequence is

$$E = C_L V_{DD}^2$$

If f is the average transition rate of the output, determine P_{AVG}

$$P_{AVG} = \frac{E}{T} = Ef$$

$$P_{DYN} = f C_L V_{DD}^2$$

If a gate has a transition duty cycle of 50% with a clock frequency of f_{CL}
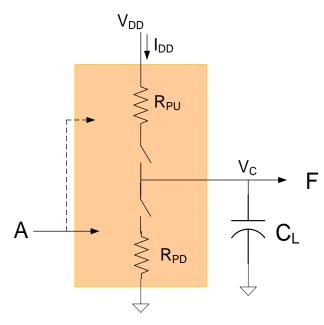
$$P_{DYN} = \frac{f_{CL}}{2} C_L V_{DD}^2$$

Note dependent on the square of V_{DD} ! …. Want to make VDD small !!!

Major source of power dissipation in many static CMOS circuits for $L_{min} > 0.1u$

Dynamic Power Dissipation

Energy dissipated with clock signal itself
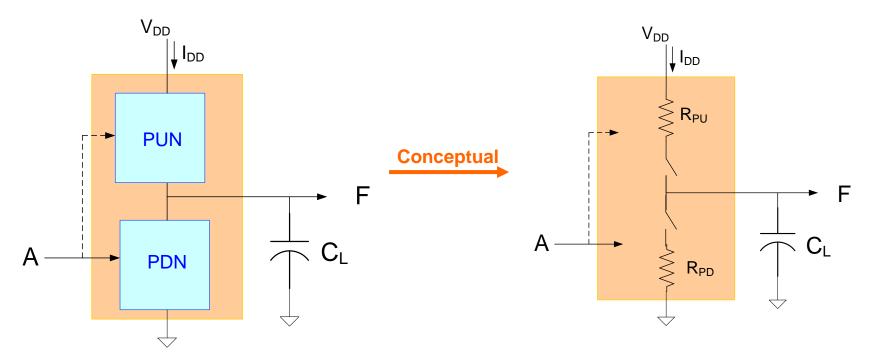
$$P_{DYN} = f_{CL} V_{DD}^2$$

The clock transitions on every clock cycle (i.e. it has a transition duty cycle of 100%)

Clock distribution can cause significant power dissipation

But if a gate has a transition duty cycle of 50% with a clock frequency of f_{CL}

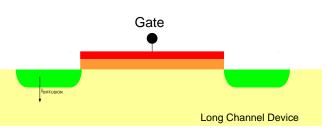$$P_{DYN} = \frac{f_{CL}}{2} C_L V_{DD}^2$$

Power Dissipation

- **All power is dissipated in pull-up and pull-down devices**

- C_L **dissipates no power but PUN and PDN dissipate power when charging and discharging** C_L

- **Dynamic power dissipation reduced by more (often much more) than a factor of 2 if minimum sizing strategy is used**

Leakage Power Dissipation

- Gate

- with very thin gate oxides, some gate leakage current flows
- major concern in 60nm and smaller processes
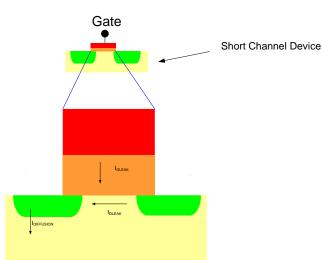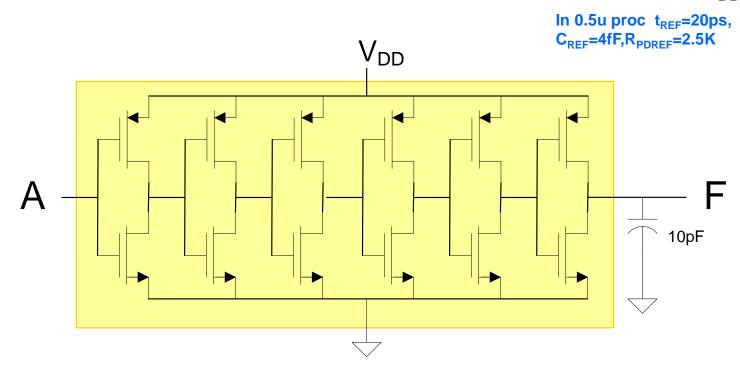- actually a type of static power dissipation

-Diffusion

- Leakage across a reverse-biased pn junction
- Dependent upon total diffusion area
- May actually be dominant power loss on longer-channel devices
- Actually a type of static power dissipation

-Drain

- channel current due to small V_{GS}-V_T
- of significant concern only with low V_{DD} processes
- actually a type of static power dissipation

Gate

$I_{DIFFUSION}$

Long Channel Device

Gate

Short Channel Device

I_{GLEAK}

I_{DLEAK}

$I_{DIFFUSION}$

Example: Determine the dynamic power dissipation in the last stage of a 6-stage CMOS pad driver if used to drive a 10pF capacitive load if clocked at 500MHz. Assume pad driver with OD of $\theta=2.5$ and $V_{DD}=3.5V$
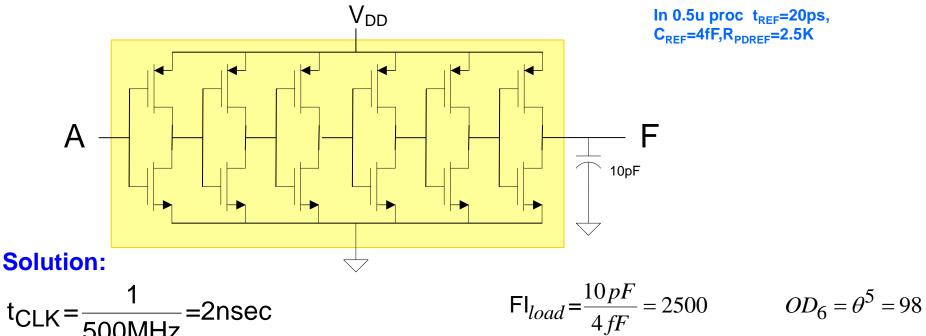
Solution: (assume output changes with 50% of clock transitions)

$$P_{DYN}=\frac{f_{CL}}{2}C_L V_{DD}^2 \quad =5E8 \bullet 10pF \bullet 3.5^2 = 61mW$$

Note this solution is independent of the OD and the process

Example: Will the CMOS pad driver actually be able to drive the 10pF load at 500MHz in the previous example in the 0.5u process?
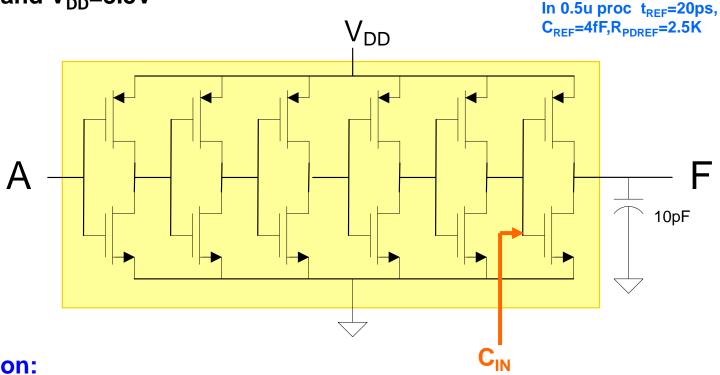
Solution:

$$t_{CLK} = \frac{1}{500\text{MHz}} = 2\text{nsec}$$

$$FI_{load} = \frac{10\,pF}{4\,fF} = 2500$$

$$OD_6 = \theta^5 = 98$$

$$t_{PROP} = 5\theta \bullet t_{REF} + \frac{FI_{load}}{OD_6} t_{REF}$$

$$\frac{FI_{load}}{OD_6} = \frac{2500}{98} \cong 25$$

$$t_{prop} = 5 \bullet 2.5 \bullet 20\text{psec} + 25 \bullet 20\text{psec} = (12.5 + 25)\,20\text{psec} = 0.75\text{nsec}$$

since $t_{CLK} > t_{PROP}$, this pad driver can drive the 10pF load at 500MHz

Example: Determine the dynamic power dissipation in the <u>next to the last stage</u> of a 6-stage CMOS pad driver if used to drive a 10pF capacitive load if clocked at 500MHz. Assume pad driver with OD of $\theta=2.5$ and $V_{DD}=3.5V$
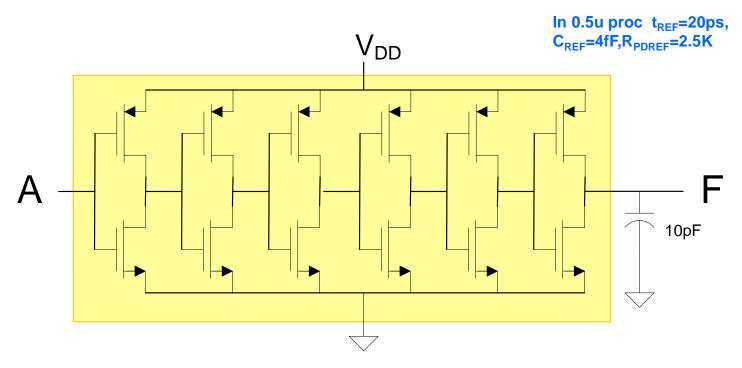
Solution:

$$C_{IN}=\theta^5 C_{REF}=2.5^5 \bullet 4fF=390fF$$

$$P_{DYN}=f_{CL}C_L V_{DD}^2 \quad =5E8 \bullet 390fF \bullet 3.5^2 = 2.4mW$$

Example: Is the 6-stage CMOS pad driver adequate to drive the 10pF capacitive load as fast as possible? Assume pad driver with OD of θ=2.5 and V_{DD}=3.5V

In 0.5u proc t_{REF}=20ps, C_{REF}=4fF, R_{PDREF}=2.5K

Solution:

$$n_{OPT} = \ln\left(\frac{C_L}{C_{REF}}\right) = \ln\left(\frac{10pF}{4fF}\right) = 7.8$$

No – an 8-stage pad driver would drive the load much faster (but is not needed If clocked at only 500MHz)

Example: Determine the power that would be required in the last stage of a CMOS pad driver to drive a 32-bit data bus off-chip if the capacitive load on each line is 2pF. Assume the clock speed is 500MHz and that each bit has an average 50% toggle rate. Assume $V_{DD}=3.5V$

In 0.5u proc $t_{REF}=20ps$, $C_{REF}=4fF, R_{PDREF}=2.5K$

Solution:

$$P_{DYN}=32 \bullet \frac{f_{CL}}{2} C_L V_{DD}^2 \quad =32 \bullet \frac{5E8}{2} \bullet 2pF \bullet 3.5^2 = 196mW$$

Note: A very large amount of power is required to take a large bus off-chip if bus has a high rate of activity.

Digital Circuit Design
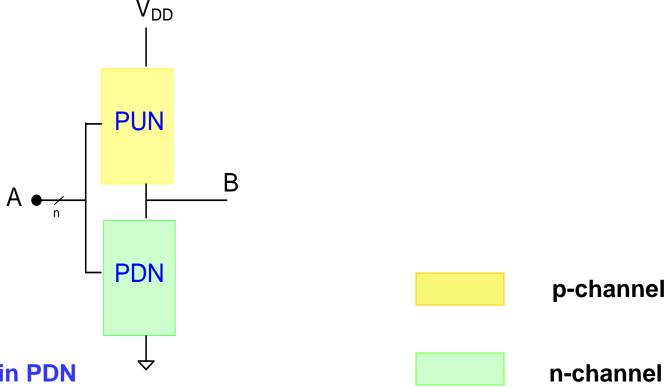
→ Hierarchical Design

→ Basic Logic Gates

→ Properties of Logic Families

→ Characterization of CMOS Inverter

→ Static CMOS Logic Gates

- Ratio Logic

→ Propagation Delay

→ Simple analytical models

→ FI/OD

→ Logical Effort

→ Elmore Delay

→ Sizing of Gates

→ **done**

→ **partial**

→ Propagation Delay with Multiple Levels of Logic

→ Optimal driving of Large Capacitive Loads

→ Power Dissipation in Logic Circuits

→ Other Logic Styles

- Array Logic

- Ring Oscillators

Logic Styles

- Static CMOS
- Complex Logic Gates
- Pass Transistor Logic (PTL)
- Pseudo NMOS
- Dynamic Logic
  - Domino
  - Zipper

Complex Logic Gates
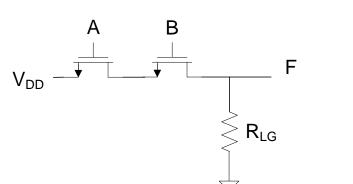
- **Implement $\overline{B}$ in PDN**
- **Implement B in PUN with complimented input variables**
- **Zero static power dissipation**
- **$V_H=V_{DD}$, $V_L=0V$ (or V_{SS})**
- **Complimented input variables often required**

Have implemented the logical function twice (once in PU, again in PD) and this is a major contributor to increased area and dynamic power dissipation

Pass Transistor Logic

$$F = A \cdot B$$

Observations about PTL

- **Low device count implementation of non inverting function (can be dramatic)**
- **Logic Swing not rail to rail**
- **Static power dissipation not 0 when F high**
- **R_{LG} may be unacceptably large**
- **Slow t_{LH}**
- **Signal degradation <u>can</u> occur when multiple levels of logic are used**
- **Widely used in some applications**
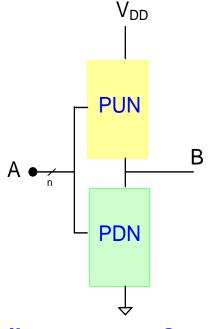- **Implements basic logic function only once!**
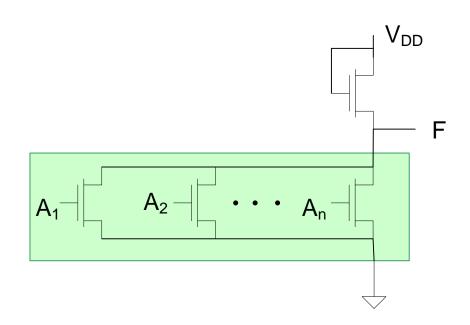
Pseudo NMOS Logic

n could be several hundred or even several thousand

Dynamic Logic

- **PTL reduced complexity of either PUN or PDN to single "resistor"**
- **PTL relaxed requirement of all n-channel or all p-channel devices in PUN/PDN**

V_{DD}

PUN

A •──/── ──┤ B
   n

PDN
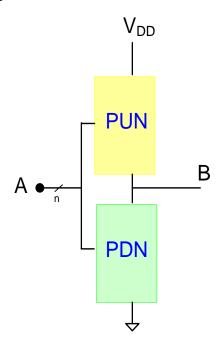
What is the biggest contributor to area? PUN **(3X active area for inverter, more for NOR gates, and Well)**

What is biggest contributor to dynamic power dissipation ?

PUN and is responsible for approximately 75% of the dynamic power dissipation in inverter, more in NOR gates !

Can the PUN be eliminated W/O compromising signal levels and power dissipation?

Dynamic Logic

V_{DD}

PUN

A • —n—

B

PDN
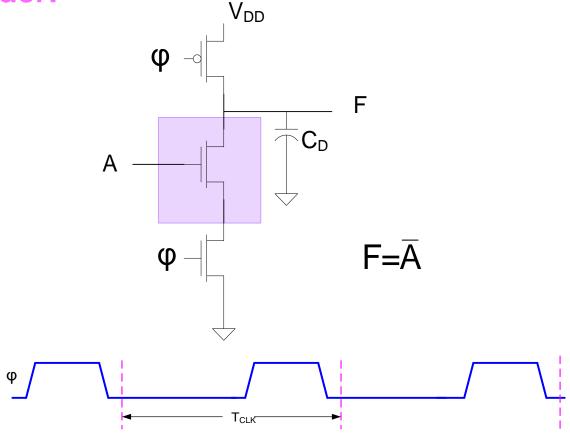
Can the PUN be eliminated W/O compromising signal levels and power dissipation?
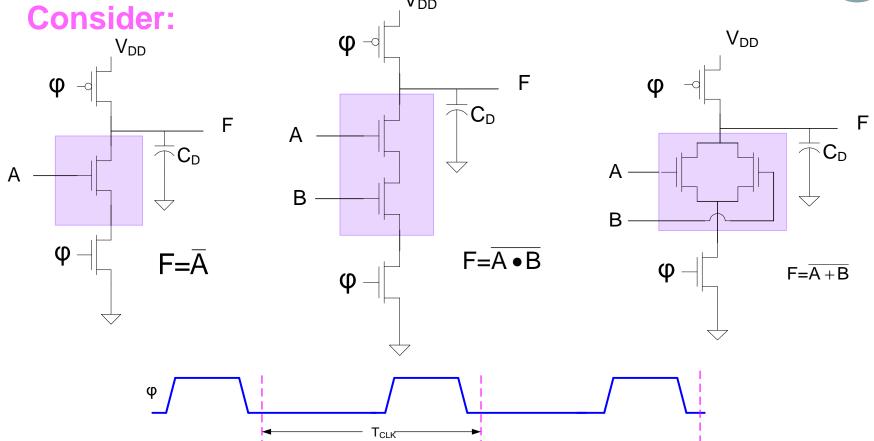
Benefits could be most significant !

Dynamic Logic

Consider:

Precharges F to "1" when φ is low
F either stays high if output is to be high or changes to low on evaluatio

Dynamic Logic
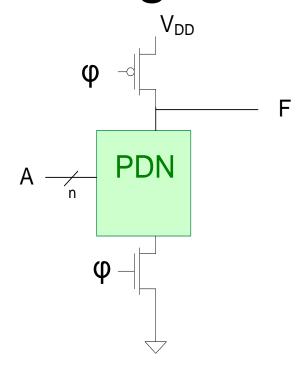
Consider:

$$F=\overline{A}$$

$$F=\overline{A \cdot B}$$

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

- **Termed Dynamic Logic Gates**
- **Parasitic capacitors actually replace C$_D$**
- **If Logic Block is n-channel, will have rail to rail swings**
- **Logic Block is simply a PDN that implements $\overline{F}$**

Dynamic Logic

Basic Dynamic Logic Gate

V_{DD}

φ

F

PDN

A

n

φ
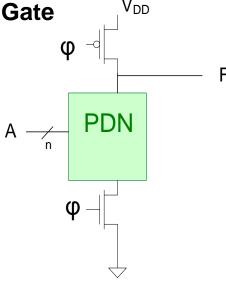
Any of the PDNs used in complex logic gates would work here !

- **Have eliminate the PUN !**
- **Ideally will have a factor of 4 or more reduction in C_{IN}**
- **Ideally will have a factor of 4 or more reduction in dynamic power dissipation relative to that of equal rise/fall !**
- **Ideally will have a factor of 2 reduction in dynamic power dissipation relative to that of minimum size!**

In integrated circuit design, **dynamic logic** (or sometimes **clocked logic**) is a design methodology in combinatory logic circuits, particularly those implemented in MOS technology. It is distinguished from the so-called static logic by exploiting temporary storage of information in stray and gate capacitances.[1] It was popular in the 1970s and has seen a recent resurgence in the design of high speed digital electronics, particularly computerCPUs. Dynamic logic circuits are usually faster than static counterparts, and require less surface area, but are more difficult to design. Dynamic logic has a higher toggle rate than static logic[2] but the capacitive loads being toggled are smaller[3] so the overall power consumption of dynamic logic may be higher or lower depending on various tradeoffs. When referring to a particular logic family, the dynamic adjective usually suffices to distinguish the design methodology, e.g. *dynamic CMOS*[4] or *dynamic SOI* design.[2]

Dynamic logic is distinguished from so-called *static logic* in that dynamic logic uses a clock signal in its implementation of combinational logic circuits. The usual use of a clock signal is to synchronize transitions in sequential logic circuits. For most implementations of combinational logic, a clock signal is not even needed.
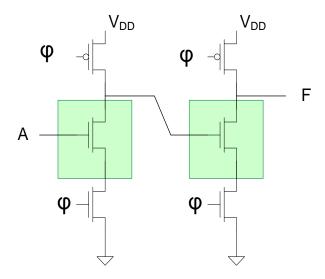
Dynamic Logic

Basic Dynamic Logic Gate

Advantages:

- **Lower dynamic power dissipation (Ideally 4X)**
- **Improved speed (ideally 4X)**

Limitations:

- **Output only valid during evaluate state**
- **Need to route a clock**
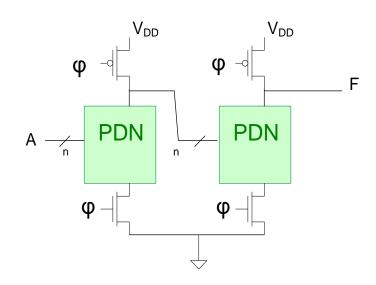   - **(and this dissipates some power)**
- **Premature Discharge !**
- **More complicated**
- **Charge storage on internal nodes of PDN**
- **No Static hold if output H**

Dynamic Logic

Premature Discharge Problem

If A is high, then F may go low at the start of the evaluate cycle and there is no way to recover a high output later in the evaluate phase - i.e. there may be a boolean error!.

Can not reliably cascade dynamic logic gates !

Dynamic Logic
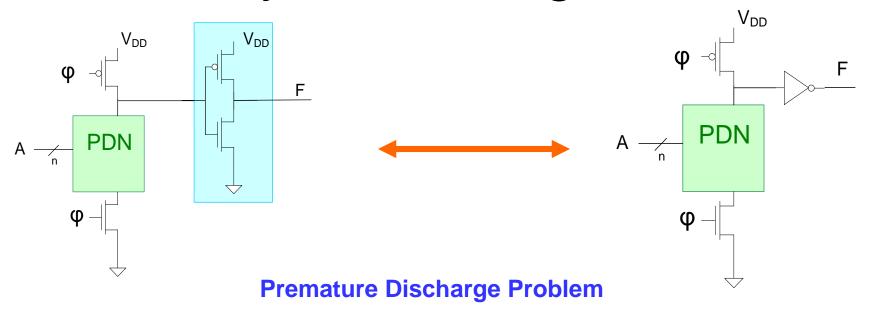
Premature Discharge Problem

This problem occurs when any inputs to an arbitrary dynamic logic gate create an R_{PD} path in the PDN during at the start of the evaluate phase that is not to pull do

How can this problem be fixed?
wn later in that evaluate phase

Precharging to the low level all inputs to a PDN that may change to the high state later in the evaluate cycle   (called domino)

Alternating gates with n-channel and p-channel pull networks (Zipper Logic)

Dynamic Logic

Premature Discharge Problem

Adding an inverter at the output will cause F to precharge low so it can serve as input to subsequent gate w/o causing premature discharge
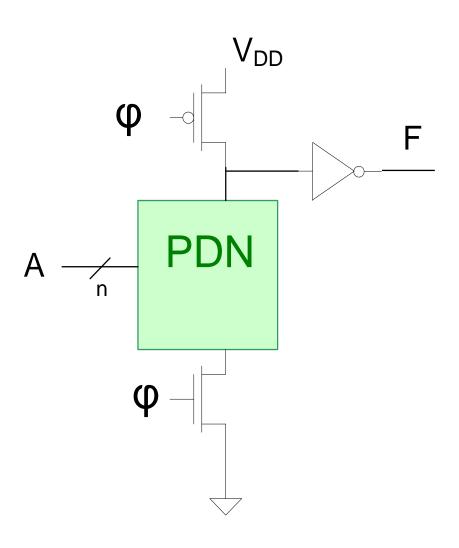
Implement F instead of $\bar{\bar{F}}$ in the PDN

Termed Domino Logic
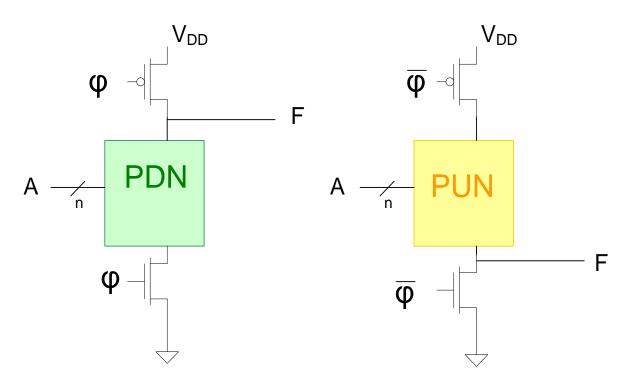
Some additional dynamic power dissipation in the inverter

Some additional delay during the evaluate state in inverter

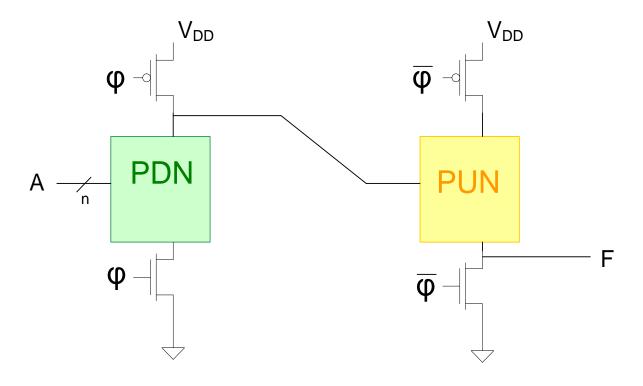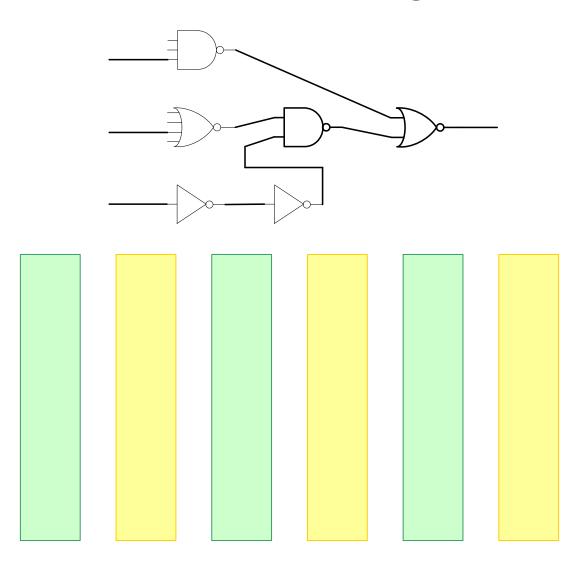# Domino Logic

Dynamic Logic

- **p-channel logic gate will pre-charge low**
- **Phasing of PUN and PDN networks is reversed**
- **Some performance loss with p-channel logic devices**
- **Direct coupling between alternate type dynamic gates is possible without causing a premature discharge problem**

Dynamic Logic
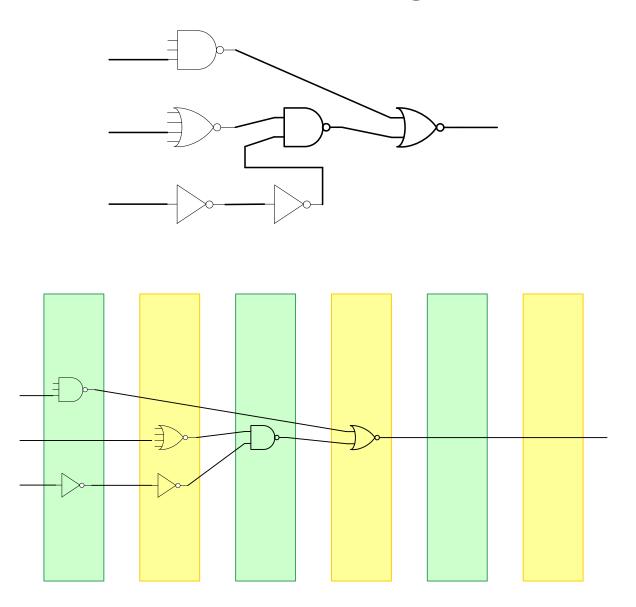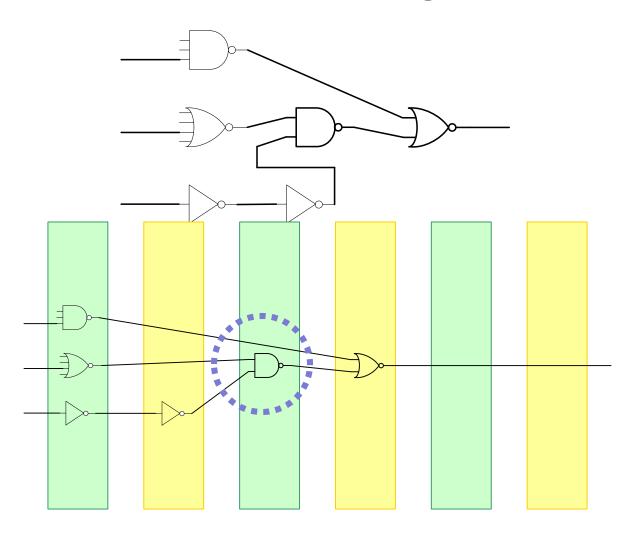

Direct coupling between alternate type dynamic gates

Zipper Logic

Map gates to appropriate precharge type
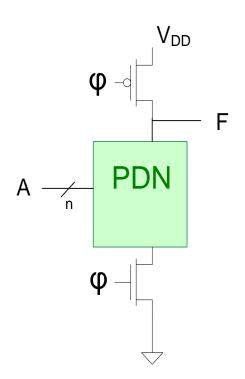
Zipper Logic

Acceptable Implementation in Zipper
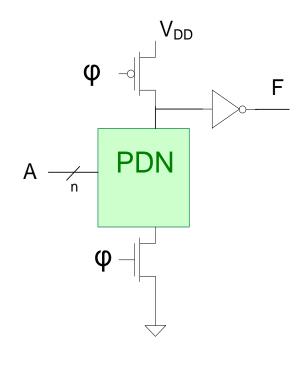
Zipper Logic

Unacceptable Implementation in Zipper
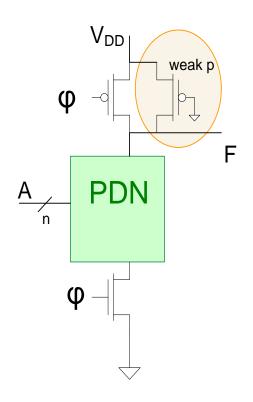**- Premature discharge at output of 2-input NAND**
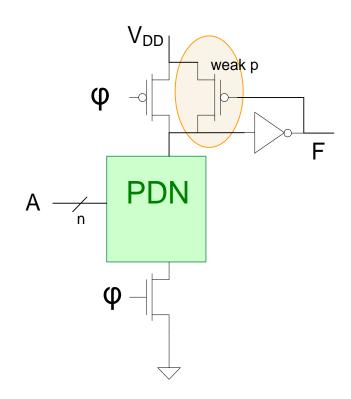
Static Hold Option

If not clocked, charge on upper node of PDN will drain off causing H output to degrade
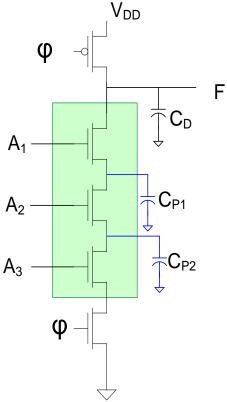
Static Hold Option

- **weak p will hold charge**
- **size may be big (long L)**
- **some static power dissipation**
- **can use small current source**
- **sometimes termed "keeper"**

- **weak p will hold charge**
- **size may be big (long L)**
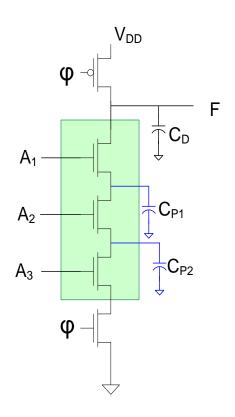- **can eliminate static power with domino**
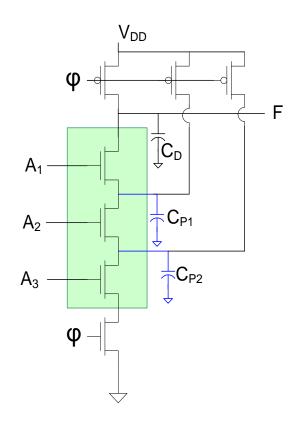- **sometimes termed "keeper"**

Charge stored on internal nodes of PDN

If voltage on C_{P1} and C_{P2} was 0V on last evaluation, these may drain charge (charge redistribution) on C_P if output is to evaluate high (e.g. On last evaluation $A_1=A_2=A_3=H$, on next evaluation $A_3=L$, $A_1=A_2=H$.)

Charge stored on internal nodes of PDN

Can precahrge internal nodes to eliminate undesired charge redistribution

Dynamic Logic

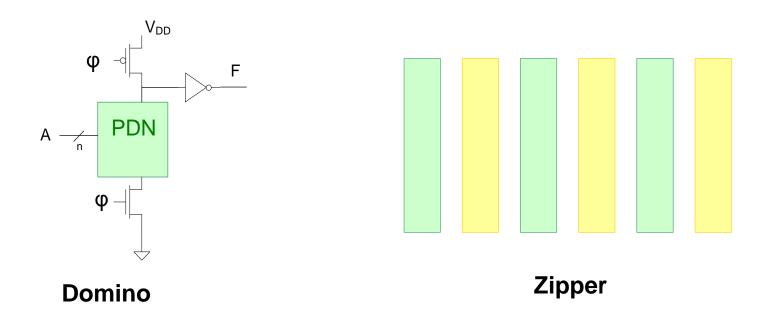**Many variants of dynamic logic are around**

- **Domino**
- **Zipper**
- **Ratio-less 2-phase**
- **Ratio-less 4-phase**
- **Output Prediction Logic**
- **Fully differential**
-

Benefits disappear, however, when interconnect (and diffusion) capacitances dominate gate capacitances

Future of Dynamic Logic

Domino

Zipper

Dynamic logic will likely disappear in deep sub-micron processes because interconnect parasitics will dominate gate parasitics

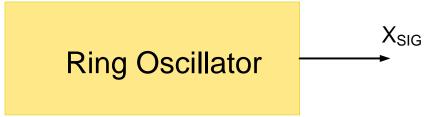# Other types of Logic (list is not complete and some have many sub-types)

From Wikipedia:

B
BiCMOS
C
CMOS
Cascode Voltage Switch Logic
Clocked logic
Complementary Pass-transistor Logic
Current mode logic
Current steering logic
D
Differential TTL
Diode logic
Diode–transistor logic
Domino logic
Dynamic logic (digital logic)
E
Emitter-coupled logic
F
Four-phase logic
G
Gunning Transceiver Logic

H
HMOS
HVDS
High-voltage differential signaling
I
Integrated injection logic
L
LVDS
Low-voltage differential signaling
Low-voltage positive emitter-coupled logic
M
Multi-threshold CMOS
N
NMOS logic
P
PMOS logic
Philips NORbits
Positive emitter-coupled logic
R
Resistor-transistor logic
S
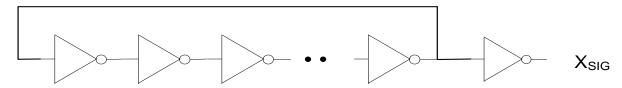**Static logic (digital logic)**
T
Transistor–transistor logic

Digital Building Blocks

- Shift Registers
- Sequential Logic
- Shift Registers (stack)
- Array Logic
- Memory Arrays

Ring Oscillators

Ring Oscillator $\rightarrow$ X_{SIG}

X_{SIG}

- **Odd number of stages will oscillate (even will not oscillate)**
- **Waveform nearly a square wave if n (number of stages) is large**
- **Output will slightly imbalance ring and device sizes can be compensated if desired**
- **Usually use a prime number (e.g. 31)**
- **Number of stages usually less than 50 (follow by dividers)**
- **Frequency highly sensitive to process variations and temperature**
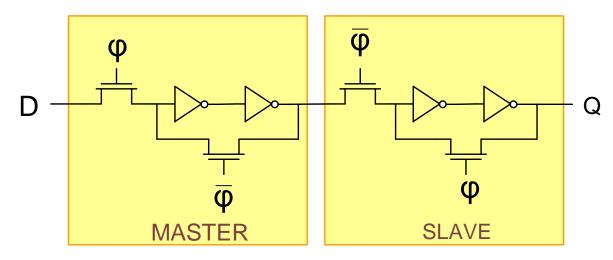
$$f_{OSC} \cong \frac{1}{n t_{PROP}}$$

- **n is the number of stages**
- **t_{PROP} is the propagation delay of a single stage (all assumed identical)**
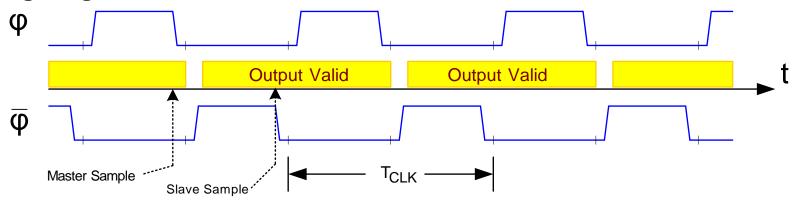
Sequential Logic Circuits

- **Flip Flops needed for sequential logic circuit**

- **Only one type of flip flop is required**

- **Invariably require clocked edge-triggered master-slave flop flops**

- **Flip flop circuits can be very simple**

- **Flip flops are part of Standard Cell Libraries**
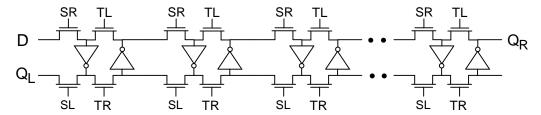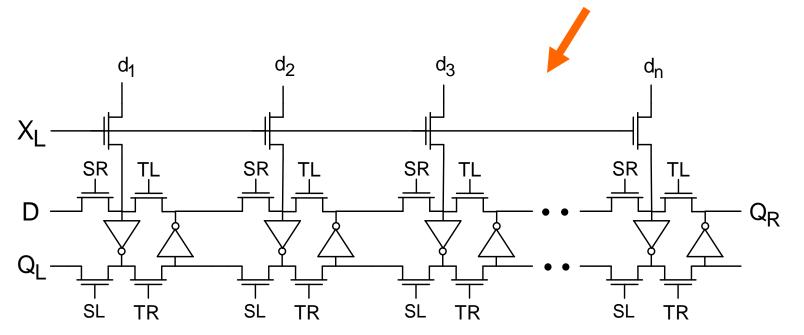
Flip Flops

Master-Slave Edge-triggered D Flip Flop

Timing Diagram

- **12 transistors (but will work with 10)**
- **Many other simple D Flip-flops exist as well**

Shift Registers

Dynamic Shift Register

n-bit Parallel-Load, Parallel-Read Bidirectional Dynamic Shift Register

- **Useful for Parallel to Serial and Serial to Parallel Conversion**
- **Can be put in static hold state if T_L and T_R replaced with H👆TL and H👆TL**

End of Lecture 44