EE 330
Lecture 44
Digital Circuits

• Other Logic Styles
• Dynamic Logic Circuits

Course Evaluation Reminder  -  All Electronic
Power Dissipation in Logic Circuits

Types of Power Dissipation

- Static
- Pipe
- Dynamic
  - Gate
  - Diffusion
  - Drain

Review from Last Time
Dynamic Power Dissipation

Energy from $V_{DD}$ for one L-H output transition sequence is

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

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

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

$$P_{DYN} = fC_L V_{DD}^2$$

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 $L_{min} > 0.1\mu$m

Review from Last Time
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
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

- 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

- done
- partial
Logic Styles

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

- Implement 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 can occur when multiple levels of logic are used
- Widely used in some applications
- Implements basic logic function only once!
Pseudo NMOS Logic

\[ n \text{ could be several hundred or even several thousand} \]
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

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?
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 $\phi$ is low.

$F$ either stays high if output is to be high or changes to low on evaluation.
Consider:

- **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 $F$
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!
Dynamic logic (properly designed) is over twice as fast as normal logic. It uses only fast N transistors, and is amenable to transistor sizing optimizations. Static logic is slower because it has twice the loading, higher thresholds, and actually uses slow P transistors to compute things. Domino logic may be harder to work with, but if you need the speed, there is no other choice. Anything you buy that runs over 2GHz in 2007 uses dynamic logic. Another advantage is low power. A dynamic logic circuit running at 1/2 voltage will consume 1/4 the power of normal logic. Also each rail can convey an arbitrary number of bits, and there are no power-wasting glitches. Also power-saving clock gating and asynchronous techniques are much more natural in dynamic logic.
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 down later in that evaluate phase.

How can this problem be fixed?

1. Precharging to the low level all inputs to a PDN that may change to the high state later in the evaluate cycle (called domino)
2. Alternating gates with n-channel and p-channel pull networks (Zipper Logic)
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 $\overline{F}$ 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

![Diagram of 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
Map gates to appropriate precharge type
Zipper Logic

Acceptable Implementation in Zipper
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 $C_{P_1}$ and $C_{P_2}$ 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 precharge 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

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

- 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
  \[
  f_{\text{OSC}} \approx \frac{1}{nt_{\text{PROP}}}
  \]
- \( n \) is the number of stages
- \( t_{\text{PROP}} \) 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

- 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 $H\triangleq TL$ and $H\triangleq TL$