Outline

• Announcements
• Review: Static, Ratioed, and Pass Gate CMOS Logic
• Dynamic Logic
• Impact of Internal Circuit Nodes
• Sizing
• Clocking Styles Overview
• Static Latches and Flip-Flops
Outline

• Announcements
• Review: Static, Ratioed, and Pass Gate CMOS Logic
• Dynamic Logic
• Impact of Internal Circuit Nodes
• Sizing
• Clocking Styles Overview
• Static Latches and Flip-Flops
• **Pull-Up network consists of PMOS devices connected complementary to NMOS Pull-Down network**

R. Amirtharajah, EEC216 Winter 2008
Complex Gate Example

\[ F = D + A \cdot (B + C) \]
Static CMOS for Low Power

- Dynamic power and short circuit current applies
  - Mismatched delays can lead to glitching, increased dynamic power
  - Dynamic logic eliminates glitches, short circuit
- Only static power due to leakage
- Fully complementary design has high noise margin
  - $V_{OH} = V_{DD}$, $V_{OL} = GND$
  - Design style more scalable to lower supply voltages
  - Implies lower threshold voltages can be used also
- **PMOS devices may degrade performance**
  - High input capacitance, slow series P-stacks
• Pull-Up network replaced by simple (often resistive) load
Ratioed Logic for Low Power

- Dynamic power and static current applies
  - Mismatched delays can lead to glitching, increased dynamic power
  - Conducts current as long as output is low
- Reduced noise margin because of resistance ratios
  - $V_{OH} = V_{DD}$, $V_{OL} = V_{DD}R_{PDN} / (R_L + R_{PDN})$
  - Could increase leakage in load gates whose NMOS gates are at $V_{OL}$ instead of ground
- Reduced transistor count decreases input capacitance
- Low-to-High transition speed determined by load (could be faster or slower than series PMOS)
- Most useful for high fan-in gates
• PDN1 ON implies PDN2 OFF pulls Out low, turning on PMOS which pulls complement high
DCVSL Summary for Low Power

• Differential logic style
  – Generating both polarities of output can improve speed (eliminates inverters)
  – Extra noise immunity to common-mode noise
  – Convenient for self-timed (asynchronous) logic design

• Still a ratioed logic style, even though outputs transition rail-to-rail
  – PMOS must be sized carefully to ensure functionality
  – Pulldown networks must overcome PMOS on other side

• Short circuit current flows while outputs are switching (pulldown fighting opposite side PMOS)

• Twice the number of NMOS inputs compared to single-ended ratioed logic styles, higher input capacitance
• Since complementary signals needed anyway, can create a fully differential version of pass gate logic
CPL Basic Gates: AND / NAND

\[ F = A \cdot B \]

\[ \overline{F} = \overline{A} \cdot \overline{B} \]
CPL Summary for Low Power

• Fully differential signals
  – Requires more devices, but simplifies complex gates like XOR, full adder; eliminates extra inverters

• Static logic style
  – Output nodes always have a low impedance path to $V_{DD}$ and GND
  – Improves resilience to noise events

• Very modular design style
  – All gates share same fundamental topology: only inputs are permuted to implement other logic functions

• Energy is low (good PDP) but speed is poor (bad EDP)
  – Series transistors have high resistance
  – Circuit techniques necessary to overcome $V_{Th}$ drops
Outline

• Announcements
• Review: Static, Ratioed, and Pass Gate CMOS Logic
• Dynamic Logic
  • Impact of Internal Circuit Nodes
• Sizing
• Clocking Styles Overview
• Static Latches and Flip-Flops
Summary of CMOS Logic Styles

• Nonclocked Logic
  – Does not require clock for proper logic operation (although clocks may be required for state operation)
  – Static CMOS, ratioed logic, DCVSL, Pass-Gate logic

• Clocked Logic
  – Periodic signal required for correct logic operation as well as for state (latches and flip-flops)
  – Dynamic logic (domino, NP-CMOS)

• Clocked styles faster in general, but also consume more power (can be observed in Power-Delay Product)
Dynamic CMOS Logic Concepts

• **Precharge Phase**
  – Output node charged to reference voltage and left floating before evaluation
  – Load capacitance “stores” the precharge value

• **Evaluation Phase**
  – Path to change the output node voltage energized by turning on evaluation transistor
  – Depending on inputs, load capacitance “written” with final value (changed from precharge value or left unchanged)

• **Inputs must make at most one transition during evaluation**

• **Output can be left high impedance, unlike static CMOS**
Dynamic CMOS Logic

`Clk` [Gate]

`In_0`, `In_1`, `In_2` [Inputs]

`Out` [Output]
Dynamic CMOS Two-Input NAND Gate

- PMOS precharges (Clk low), NMOS evaluates (Clk high)
Dynamic CMOS Logic Gate Properties

- Logic function implemented by NMOS pulldown network
  - Design of PDN identical to static CMOS
- Number of transistors for N-input dynamic gate is N+2 (2N for static CMOS gate)
  - Lower area, lower input capacitance
- Nonratioed logic family: sizing of PMOS device independent of sizing of PDN
  - Wider devices imply faster precharge, more clock power
- Gates only consume dynamic power
  - Ideally, no static current path exists when gate evaluates
- Fast! Reduced input capacitance, all current to output
Cascading Dynamic CMOS N-blocks

- Problem: late arriving inputs cause false output discharge
Variations on the Domino Theme

• **Multiple-Output Domino**
  – Exploit situation when certain outputs are subsets of other outputs to reduce area
  – Precharge intermediate nodes in PDN and follow with inverters to drive other N-block dynamic gates

• **Compound Domino**
  – Use complex static CMOS gates (NANDs, NORs) on outputs of multiple dynamic gates in parallel
  – Replaces large fanin domino gates with lower fanin gates
  – Capacitive coupling from static gate outputs to dynamic gate outputs an issue
Multiple Output Domino CMOS Logic

\[ Clk \]

\[ In_0 \quad In_1 \quad In_4 \quad In_5 \]

PDN

\[ Clk \]

\[ Out_0 \]

PDN

\[ Out_1 \]
Dynamic Gate Activity Factor

Static Gate Activity Factor Equation:

\[ \alpha_{0 \rightarrow 1} = p_0 p_1 = p_0 \left(1 - p_0 \right) \]

- Dynamic logic has higher activity due to periodic precharge and discharge
- Output transition probability independent of input state, dependent on input probabilities
- Output makes low-high transition if discharged during previous evaluate phase:

\[ \alpha_{0 \rightarrow 1} = p_0 = \frac{N_0}{2^N} \]
- \(N_0\) is the number of 0s in truth table output column
Dynamic Logic Design for Low Power

• Advantages for low power
  – Lower physical capacitance since fewer devices used to implement given logic function
  – Input loading lower since no dual PMOS devices
  – Gates *must* allow only one transition for correct operation (i.e., no glitching!)
  – No short circuit power since pullup path not enabled when evaluating output

• Disadvantages for low power
  – Higher clock power since guaranteed clock node transition
  – More than minimal number of devices for implementation
  – Higher switching activity as shown earlier
Signal Integrity Issues in Dynamic Design

- **Charge leakage**
  - High impedance nodes can lose charge due to leakage
  - Can be offset by adding feedback devices to trickle charge the nodes and maintain state

- **Charge sharing**
  - Parasitic capacitances on intermediate nodes in PDN can pull charge from output node, degrading output state

- **Capacitive coupling**
  - High impedance nodes sensitive to adjacent node transitions coupling in noise

- **Clock feedthrough**
  - Capacitive coupling between clock input and output node
Closing Thoughts on Dynamic Logic

- Dynamic logic is an aggressive design style
  - Performance can be very good, very fast circuits
  - Noise issues require care to guarantee correct operation
  - Sensitivity to noise impedes aggressive voltage scaling
  - Leakage may require “staticizing” gate anyway using feedback to prevent output level degrading
  - Cannot reduce clock frequency arbitrarily for testing
  - Must evaluate power tradeoffs carefully during design
Outline

• Announcements
• Review: Static, Ratioed, and Pass Gate CMOS Logic
• Dynamic Logic
• Impact of Internal Circuit Nodes
• Sizing
• Clocking Styles Overview
• Static Latches and Flip-Flops
Complex Gate Power Example

\[ F = D + A \cdot (B + C) \]
1. Suppose in Cycle 0, $A = B = C = D = 0$, $F = 1$, then $C_{i0}$, $C_{i1}$ charged to $V_{DD}$.

2. In Cycle 1, $A$ and $C$ transition high, other inputs stay low.

3. Capacitors $C_{i0}$ and $C_{i1}$ in addition to output load $C_L$ must all be discharged.

4. Depending on the state of the inputs in cycles preceding Cycle 0, $C_{i2}$ may need to be discharged as well (for example, if $A$ was the last input to transition low).
Power Consumption With Internal Nodes

- Gate represents a variable capacitance to power and ground rails
  - Capacitance depends on current state and history of input signals
- Optimal routing of equivalent inputs is to put signal with highest activity factor closest to output
  - Reduces the amount of switched capacitance and power for the gate
  - Similar to optimizing gate inputs for speed, but not necessarily the same
- Can incorporate internal node capacitances into power estimation methodology
Power Estimation With Internal Nodes

- Incorporate into internal node capacitances into dynamic power estimation:

\[ P = V_{DD}^2 f \left( C_L \alpha_L + \sum_{n_i} C_{n_i} \alpha_{n_i} \right) \]

- Where \( C_{n_i} \) are normalized internal node capacitances
  - Many internal nodes may typically charge only to \( V_{DD} - V_{Tn} \)
  - Can fold this voltage factor into either capacitance or activity, but capacitance makes more sense since it corresponds to charge

- Because input history affects internal nodes, computing activity is NP-hard (usually just estimate)
Final Words on Internal Nodes

- Data dependent capacitance affects speed as well as power
  - Try to minimize through layout: share sources and drains by implementing in the same diffusion whenever possible
- In addition to data dependence, source / drain depletion region capacitances depend on voltage also
- Consider both effects in extreme high performance and extreme low power designs
  - Late generation Alpha processors from Compaq considered data and voltage dependence in delay models for cells
  - Must have a lot of confidence in simulation and process characterization to optimize this way
Outline

- Announcements
- Review: Static, Ratioed, and Pass Gate CMOS Logic
- Dynamic Logic
- Impact of Internal Circuit Nodes
- Sizing
  - Clocking Styles Overview
  - Static Latches and Flip-Flops
Sizing for Speed and Low Power

• Lowest level of design optimization
  – Explore sizing for minimum delay and energy and compare optimal points

• Will see that device sizing combined with voltage reduction is very effective approach to reducing energy consumption

• Still an ongoing area of research
  – Optimal points can change quantitatively as device characteristics change (for example, due to velocity saturation)
  – Choice of metric to optimize (PDP vs. EDP) important
  – Increased leakage affects shifts optimal point to smaller devices
Intrinsic (Self-Load) and Extrinsic Capacitance

\[ C_g \quad C_i \quad C_{\text{wire}} \]
RC Switch Model for Inverter Sizing

\[ C_{ext} = C_{wire} + C_g \]

- Model delay using ideal switch and resistor for MOSFET
Unloaded Inverter Delay

• Estimate delay using ideal switch and resistor model (RC time constant):

\[ t_{pd} \propto R_{eq} \left( C_i + C_{ext} \right) \]

\[ \propto R_{eq} C_i \left( 1 + C_{ext} / C_i \right) \]

\[ \propto t_{p0} \left( 1 + C_{ext} / C_i \right) \]

• Define intrinsic inverter delay (with fudge factor):

\[ t_{p0} = 0.69 R_{eq} C_i \]

• \( C_i \) consists of source / drain and overlap capacitance
Fastest Loaded Inverter Sizing

• Decrease delay by enlarging transistor (increases current, decreases $R_{eq}$) by factor $S$:

$$t_{pd} = 0.69 \frac{R_{eq}}{S} SC_i \left(1 + \frac{C_{ext}}{SC_i}\right)$$

$$t_{pd} = t_{p0} \left(1 + \frac{C_{ext}}{SC_i}\right)$$

• Intrinsic delay independent of sizing
• Infinite $S$ yields fastest gate (eliminates external load), reducing delay to intrinsic in the limit
Relating Self-Load to Gate Capacitance

• Increasing transistor sizing enlarges self-load and gate input capacitance

• Convenient to relate them by a constant factor $\gamma$ ($\gamma$ around 1 in submicron processes)

$C_i = \gamma C_g$

$t_{pd} = t_{p0} \left( 1 + \frac{C_{ext}}{\gamma C_g} \right) = t_{p0} \left( 1 + \frac{k}{\gamma} \right)$

• $k$ is effective fanout of gate

• Delay depends only on ratio between external load capacitance and input capacitance
Inverter Chain Sizing for Minimum Delay

- Using inverter sizing, want to minimize delay of driving large load $C_L$
- Optimize using equivalent resistance delay equation derived in previous slides

\[ C_L = KC_{g1} \]
Total Inverter Chain Delay

- Delay of the jth inverter stage is (ignoring wiring):

\[ t_{pd,j} = t_{p0} \left( 1 + \frac{C_{g,j+1}}{\gamma C_{g,j}} \right) = t_{p0} \left( 1 + \frac{k_j}{\gamma} \right) \]

- Total delay is:

\[ T_{pd} = \sum_{j=1}^{N} t_{pd,j} = t_{p0} \sum_{j=1}^{N} \left( 1 + \frac{C_{g,j+1}}{\gamma C_{g,j}} \right) \]

where

\[ C_{g,N+1} = C_L = KC_{g,1} \]
Optimal Inverter Sizing for Minimum Delay

- Minimize delay by taking partial derivatives wrt $C_{g,j}$, set them equal to 0
  - N-1 equations in N unknowns
  - Solution for jth inverter is geometric mean of its neighbors sizing:
    $$C_{g,j} = \sqrt{C_{g,j-1}C_{g,j+1}}$$
- Implies each inverter has constant scale-up factor $k_j$:
  $$k_j = k = \frac{N}{\sqrt{C_L/C_{g,1}}} = \frac{N}{\sqrt{K}}$$
- Minimum delay:
  $$T_{pd} = Nt_p^0\left(1 + \frac{N}{\sqrt{K}}/\gamma\right)$$
Optimal Inverter Stages for Minimum Delay

- Delay trade off in the number of stages $N$
  - Too many stages, intrinsic delay term dominates
  - Too few stages, extrinsic delay term due to fanout ratio dominates
- Taking derivative of $T_{pd}$ wrt $N$ and setting equal to zero yields scale up factor for optimal number of stages:
  $$k = e^{\left(1 + \frac{\gamma}{k}\right)}$$
- Closed form solution when $\gamma = 0$, $N = \ln(K)$
  $$k = e = 2.71828$$
- For more typical case of $\gamma = 1, k = 3.6$
- Often choose $k = 4$
Inverter Chain Sizing for Minimum Energy

- Using inverter sizing, want to minimize energy of driving large load $C_L$ while maintaining fixed delay
- Again, optimize using equivalent resistance delay equation derived in previous slides
Optimal Inverter Sizing for Minimum Energy

• Write delay equation for chain with two stages:

\[ T_{pd} = t_{p0} \left( \left( 1 + \frac{k}{\gamma} \right) + \left( 1 + \frac{K}{\gamma k} \right) \right) \]

• Delay for an individual stage (assuming velocity saturation):

\[ t_{p0} \approx \frac{\alpha (C_i + C_{ext}) V_{DD}}{V_{DD} - V_T - \frac{V_{DSAT}}{2}} \]

• Total energy dissipated depends on total capacitance:

\[ E = V_{DD}^2 C_g \left( (1 + \gamma)(1 + k) + K \right) \]
Optimal Inverter Sizing for Minimum Energy 2

• Goal is to choose \( k \) factor to minimize energy
• Obvious approach is to choose \( k \) to minimize delay, then reduce voltage until delay constraint is met
  – In two stage example, this approach yields \( k = \sqrt{K} \)
• Suppose supply voltage for \( k = 1 \) is \( V_0 \) and delay is \( T_0 \)
  – Assuming \( \gamma = 1 \), increase \( k \) and decrease \( V_{DD} \) until \( T_{pd} = T_0 \) again
• Ratio of total energy for reduced voltage case to energy for \( V_{DD} = V_0 \) is:
  \[
  \frac{E}{E_0} = \frac{V_{DD}^2}{V_0^2} \left( \frac{2 + 2k + K}{4 + K} \right)
  \]
• Increasing \( k \) allows greater reduction in \( V_{DD} \) (1st term) at expense of increased capacitance (2nd term)
Energy vs. Transistor Sizing Factor

Fig. 6. Plot of energy versus transistor sizing factor for various parasitic contributions.

- From Chandrakasan92, “Low-Power CMOS Digital Design”
Summary of Sizing for Minimum Energy

- Device sizing combined with voltage reduction is very effective approach to reducing energy consumption
  - For large fanouts, a factor of 10 reduction can be gained
  - $K = 1$ case is exception; minimum-size device optimal
- Overly large sizing can result in large power penalty
  - Typical of designs today, especially standard cells since cells designed for worst case load conditions to guarantee design meets timing
- Optimal sizing for minimum energy (at fixed delay) smaller than sizing for minimum delay
  - Example: for fanout $K = 20$, $k_{opt(energy)} = 3.53$ vs. $k_{opt(delay)} = 4.47$
  - Further increasing sizes leads to minimal voltage reductions
Caveats

• Leakage power ignored in analysis so far
  – Increase pressure for smaller devices, shifting optimal point to smaller scaling factors

• Sizing opportunities may be limited
  – Synthesized design, cannot customize each cell very finely
  – Best approach is not to overdesign cells by assuming pessimistic loading conditions
  – Have many device sizes (INV_1X, INV_2X, NAND2_4X) so tool can pick cells to meet timing-driven synthesis, place, and route constraints
Extending Sizing Optimization to Logic

• So far only looked at sizing of inverter chains
• Can extend results to explore sizing of more complex logic gates
  – RC model leads to design concept of logical effort
  – Logical effort is systematic way of sizing arbitrary logic gates in a critical path to minimize delay
• Choose appropriate metric and constraints to optimize for low power design
  – Previous examples focused on minimizing power for a fixed delay constraint (corresponds to Power-Delay Product)
  – Can minimize Energy-Delay Product instead to count performance more heavily
Outline

• Announcements
• Review: Static, Ratioed, and Pass Gate CMOS Logic
• Dynamic Logic
• Impact of Internal Circuit Nodes
• Sizing
• Clocking Styles Overview
• Static Latches and Flip-Flops
High Level Clocking Styles

• Timing classification of digital systems (Messerschmitt 1990) depends on relation to system clock
  – **Synchronous**: same frequency, known fixed phase offset
  – **Mesochronous**: same frequency, unknown phase offset
  – **Plesiochronous**: nominally same frequency, but slightly different (difference causes phase offset to drift in time)
  – **Asynchronous**: signals transition at arbitrary times relative to system clock

• Traditionally designed for synchronous operation

• As speeds get higher, clock skew and jitter increases, total synchronization less likely

• Mesochronous, Plesiochronous, Asynchronous styles more prevalent (for example, GALS)
Synchronous System With Global Clock

- Simple and convenient design style with minimal circuit overhead
- Challenge is creating and distributing clock with low skew and jitter (timing uncertainty) at high frequencies
Critical Path Replica Self-Timed System

• Similar to synchronous style except clock frequency directly correlated to circuit speed

• Robust to process, voltage, temperature variations

• Minimal circuit overhead for self-timing
Handshaking Between Pipeline Stages

- Truly asynchronous style with maximum performance
  - Each stage computes as fast as possible on each datum
  - Overhead between stages to guarantee information flows correctly through pipeline
- Also robust to process, voltage, temperature variations
- Circuit overhead implies more switched capacitance
Clocking Styles for Low Power

• No definitive conclusions on which is better … yet

• Asynchronous styles offer highest speed and greatest voltage reductions
  – Offset significantly by higher switched capacitance
  – Difficulty of designing circuits properly a barrier to widespread adoption
  – Functions reliably with respect to voltage variations, allows aggressive supply scaling and battery operation

• Synchronous benefits include ease of design and minimum circuit overhead

• Combination of styles likely in future
  – Globally Asynchronous, Locally Synchronous blocks
  – Mixed clocked and self-timed pipeline stages
Outline

• Announcements
• Review: Static, Ratioed, and Pass Gate CMOS Logic
• Dynamic Logic
• Impact of Internal Circuit Nodes
• Sizing
• Clocking Styles Overview
• Static Latches and Flip-Flops
Sequential Element Design for Low Power

• Clock power is often large component of chip dynamic power
  – Activity factor is high (charges every cycle)
  – Interconnect capacitance due to clock distribution wiring is very large
  – Clock inputs to latches and flip-flops can present significant capacitance

• Sequential element design style must balance several competing demands
  – Robustness wrt noise, timing uncertainty, leakage current
  – Speed since setup/hold times and clock-to-Q delays directly impact critical path delay
  – Power dissipation
Static vs. Dynamic Design Styles

• Tradeoffs are similar to static and dynamic combinational logic styles

• Static designs rely on feedback to maintain state
  – Internal nodes as well as outputs always driven to supply rails by low impedance path
  – Requires more devices, area, possibly power
  – Robust design style with wide noise margins, scales to lower frequencies and supply voltages

• Dynamic designs store state on parasitic capacitances
  – Very fast since fewer devices, no static current fights
  – Sensitive to noise, challenging to scale to lower frequencies / voltages
  – Power dissipation depends on specific circumstances
Static Latch Bistability

\[ V_{i1} = V_{o1} = V_{i2} = V_{o2} \]

\[ V_{i1} = V_{o2} \]

\[ V_{i2} = V_{o1} \]

C (metastable)
Multiplexer-Based Static Latches

• Latches are transparent during half of clock cycle
Transmission Gate Positive Latch

\[ D \rightarrow \text{Transistor} \rightarrow \text{Transmission Gate} \rightarrow Clk \rightarrow \text{Transistor} \rightarrow Q \]

\[ Clk \rightarrow \text{Transistor} \rightarrow \text{Transmission Gate} \rightarrow \overline{Clk} \rightarrow \text{Transistor} \rightarrow \text{Transmission Gate} \rightarrow Clk \rightarrow \text{Transistor} \rightarrow \text{Transmission Gate} \rightarrow Q \]
**NMOS Pass Gate Positive Latch**

- Fewer devices, less area, lower clock load
- Threshold drop on internal nodes implies more static power, less noise margin
Master-Slave Positive Edge-Triggered FF

- Connect two opposite phase transparent latches

R. Amirtharajah, EEC216 Winter 2008
Positive Edge-Triggered FF Clock Load

- Latch choice clearly impacts flip-flop power
- Full transmission gate latch has high loading
  - Requires 7 inverters (one extra to create local inverted clock, can be amortized over entire registers)
  - Clock touches 1 inverter and 8 transmission gate FETs
  - Transmission gate load less than inverter since PMOS can be sized same as NMOS (mobility ratio is overkill)
- NMOS only pass gate latches reduce loading
  - Requires 5 inverters including local clock inversion
  - Clock touches 1 inverter and 4 transmission gate FETs
  - Static power when storing 1 since internal nodes charged to $V_{DD} - V_{Tn}$
- Specific circumstances dictate which is better
Reduced Clock Load Static Positive FF

• Reduce clock load by directly cross-coupling inverters
• Ratioed circuit: must size transmission gate to overpower feedback inverters I1 and I3
  – Can make I1, I3 intentionally weak (> minimum length)
Reverse Conduction Problem

- Second stage output can affect first stage state
- Must size feedback inverter I3 to avoid contamination by making it weak enough
Clock Overlap Failures

1. Both high simultaneously, race condition from $D$ to $Q$
2. Node A can be driven simultaneously by $D$ and B

R. Amirtharajah, EEC216 Winter 2008
Race Through and Feedback Paths

1. Both high simultaneously, race condition from $D$ to $Q$
2. Node $A$ can be driven simultaneously by $D$ and $B$

R. Amirtharajah, EEC216 Winter 2008
Nonoverlapping Clocks Methodology

- Guarantee nonoverlap period long enough
- **Note:** internal nodes left high Z during nonoverlap
Closing Thoughts

• Rich design space of circuit styles, sizing methodology, and clocking strategies
  – Must balance large circuit capacitance (static CMOS) vs. higher activity factor (dynamic CMOS)
  – Dynamic logic is faster, so must be used if speed is priority
  – Can mitigate higher activity by sharing pulldown networks (multiple-output domino) or using complex static gates to eliminate dynamic gates and inverters (compound domino)

• Next topic: dynamic latch styles, self-timed circuits