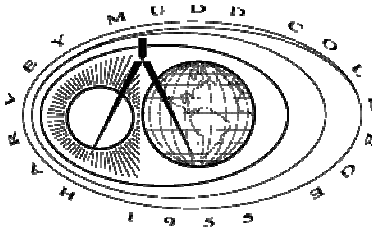


# **Advanced Domino Circuit Design**

## **Part I: Gates & Sequencing**

David Harris



Harvey Mudd College

# Outline

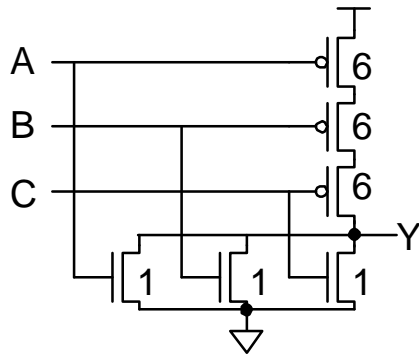
---

- ❑ Domino Circuits
- ❑ Domino Sequencing
- ❑ Nonmonotonic Dynamic Techniques

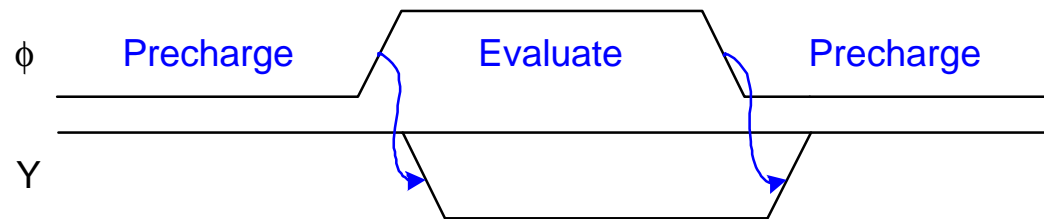
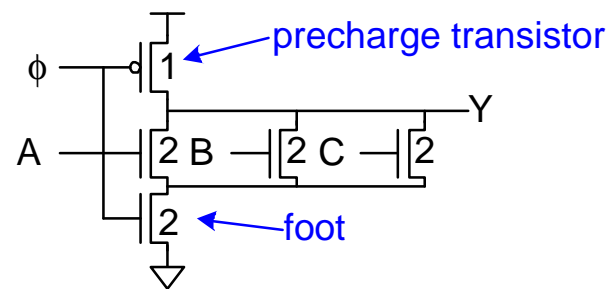
# Dynamic Logic

- ❑ Static CMOS is slow from big input transistors
- ❑ *Dynamic* gates use clocked precharge transistor
- ❑ Operate in two steps: *precharge* and *evaluate*

Static NOR3

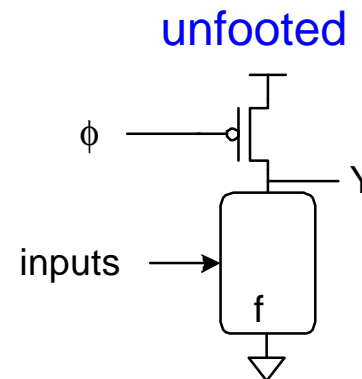
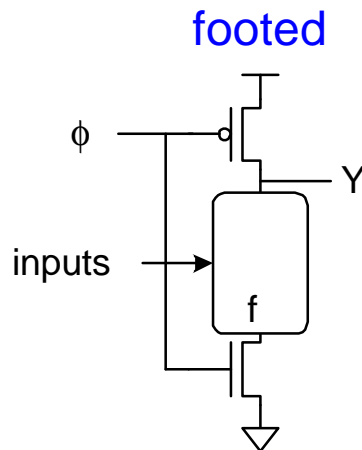


Dynamic NOR3



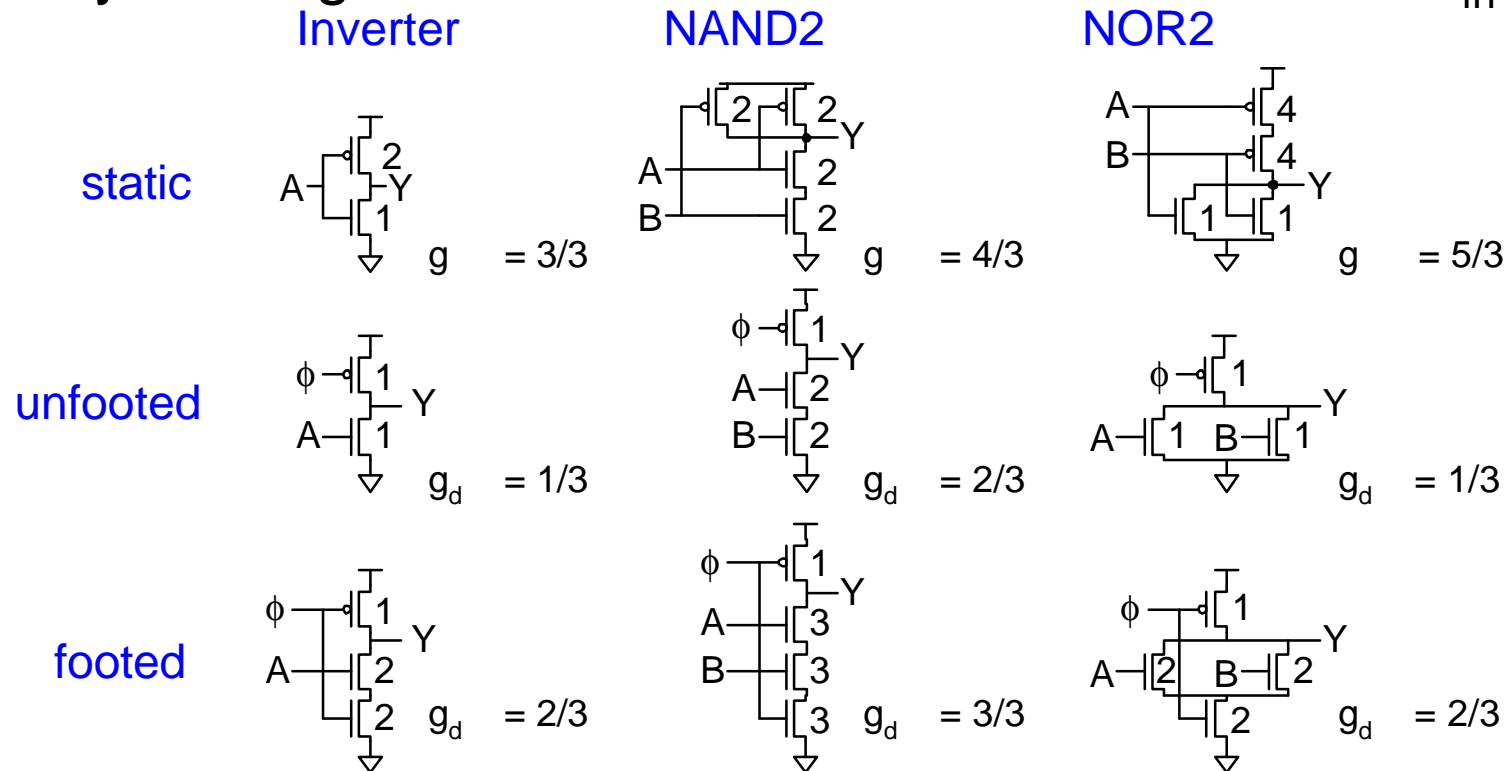
# Feet

- ❑ Foot transistor prevents contention between precharge and evaluation
- ❑ Can be left off if inputs are low during precharge



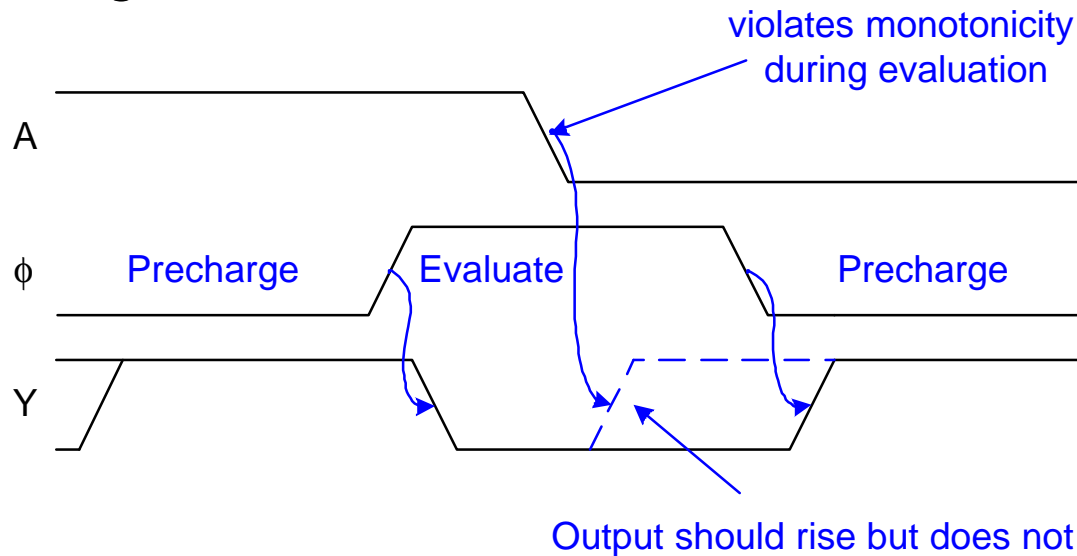
# Logical Effort

- Dynamic gates are 1.5-2x faster because of low  $C_{in}$



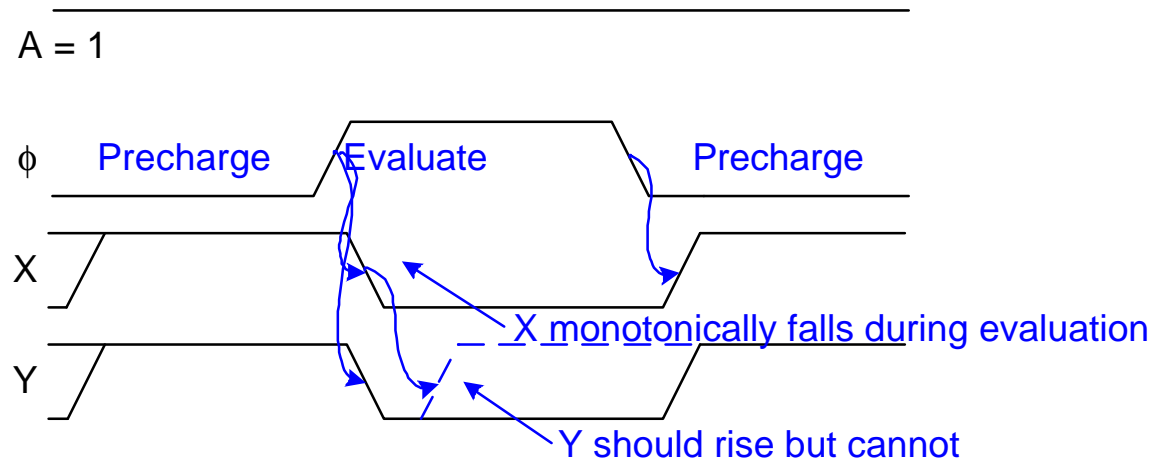
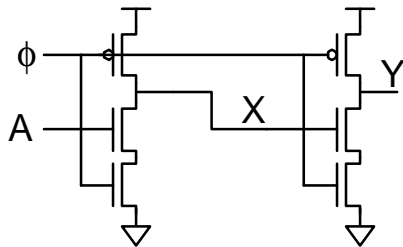
# Monotonicity

- ❑ Inputs to dynamic gates must be monotonically rising while gate is in evaluation
- ❑ A can go 0  $\rightarrow$  0, 0  $\rightarrow$  1, 1  $\rightarrow$  1 but not 1  $\rightarrow$  0



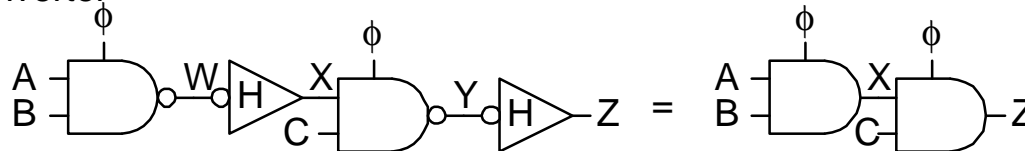
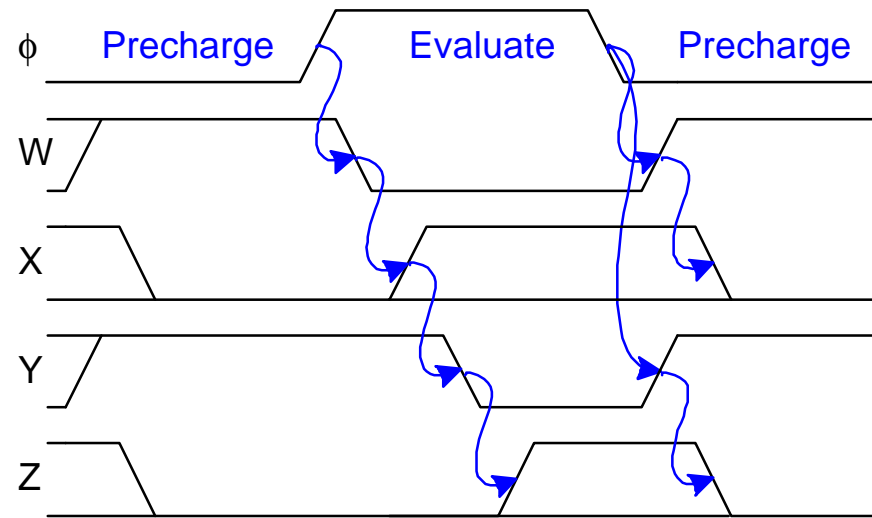
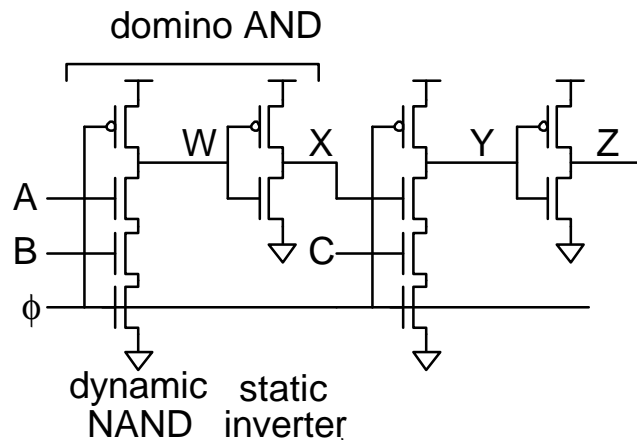
# Cascading Dynamic Gates

- ❑ But dynamic outputs are monotonically falling
- ❑ Can't cascade dynamic gates with same clock



# Domino Logic

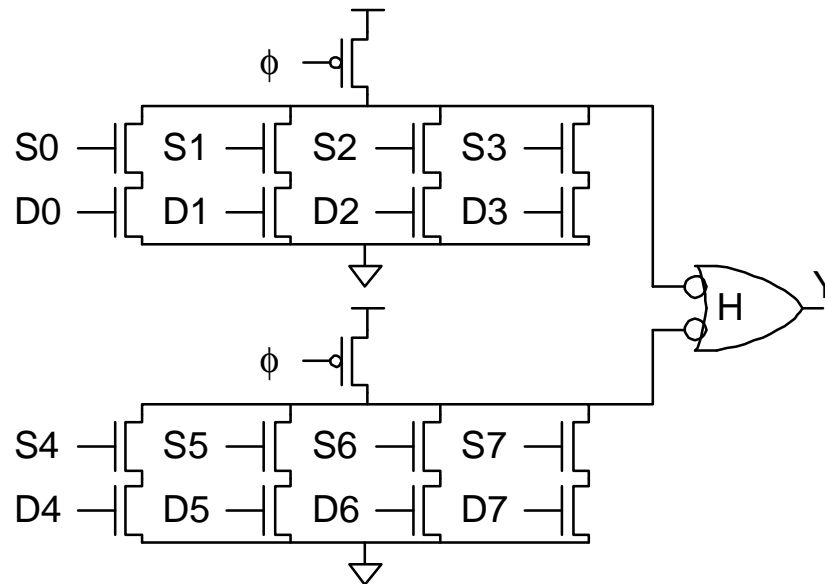
- ❑ Alternate dynamic gates with static inverters
- ❑ Skew inverters HI to favor critical rising outputs





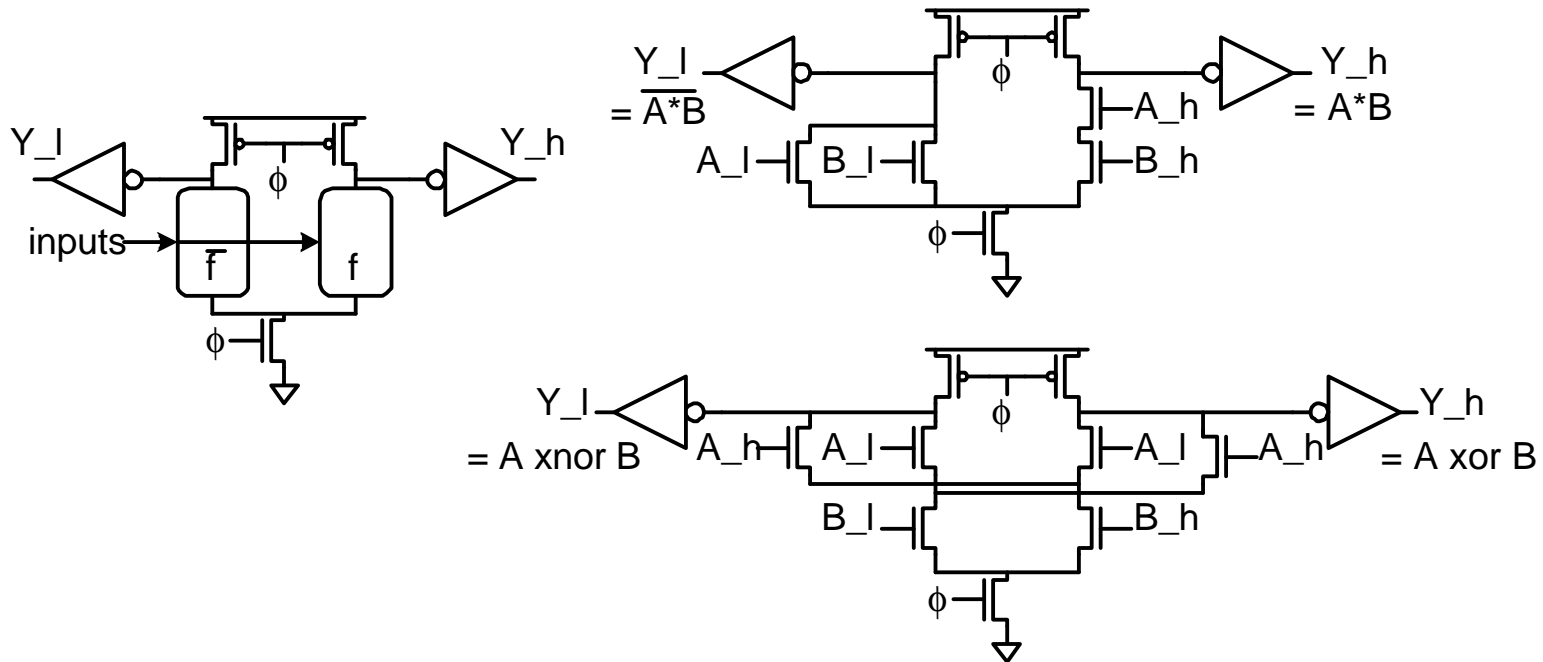
# Logic in the Static Stage

- ❑ Domino gate = dynamic gate + static gate
- ❑ Static gate may do logic as well



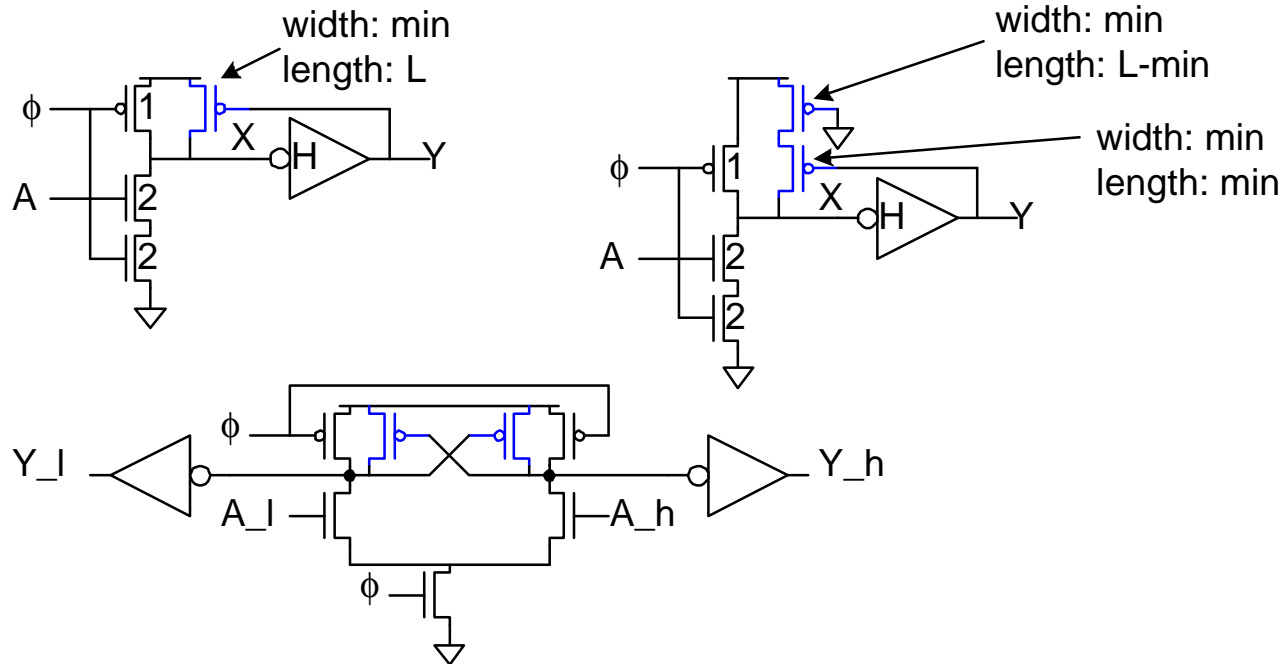
# Dual-Rail Domino

- ❑ Domino computes noninverting functions (AND, OR)
- ❑ Dual-rail domino can compute all functions



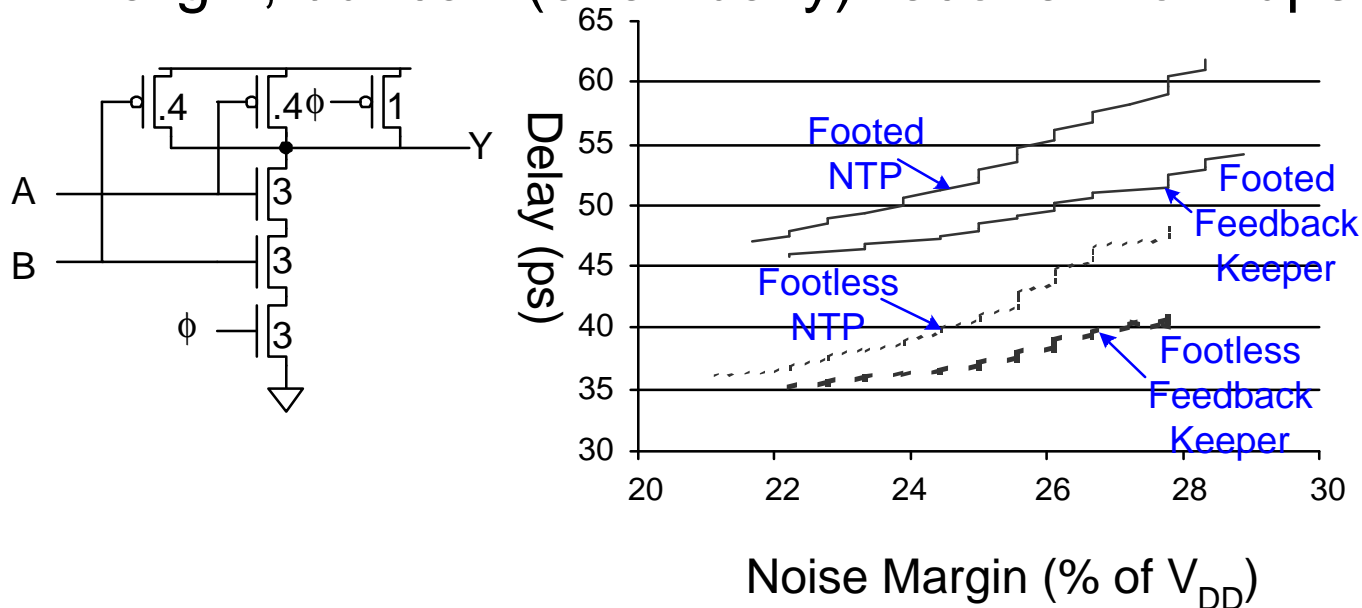
# Keepers

- ❑ Dynamic node floats when 1 during evaluation
- ❑ Keeper provides weak feedback to hold state



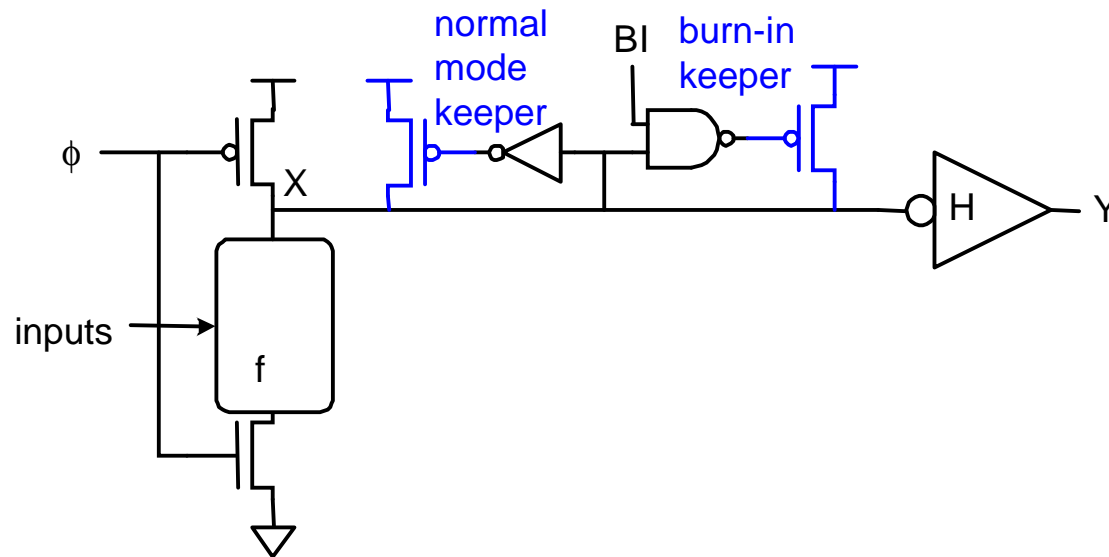
# Noise-Tolerant Precharge

- ❑ NTP uses very small pMOS input transistors
- ❑ Compared to keepers, NTP is slower for same noise margin, but can (eventually) recover from upsets



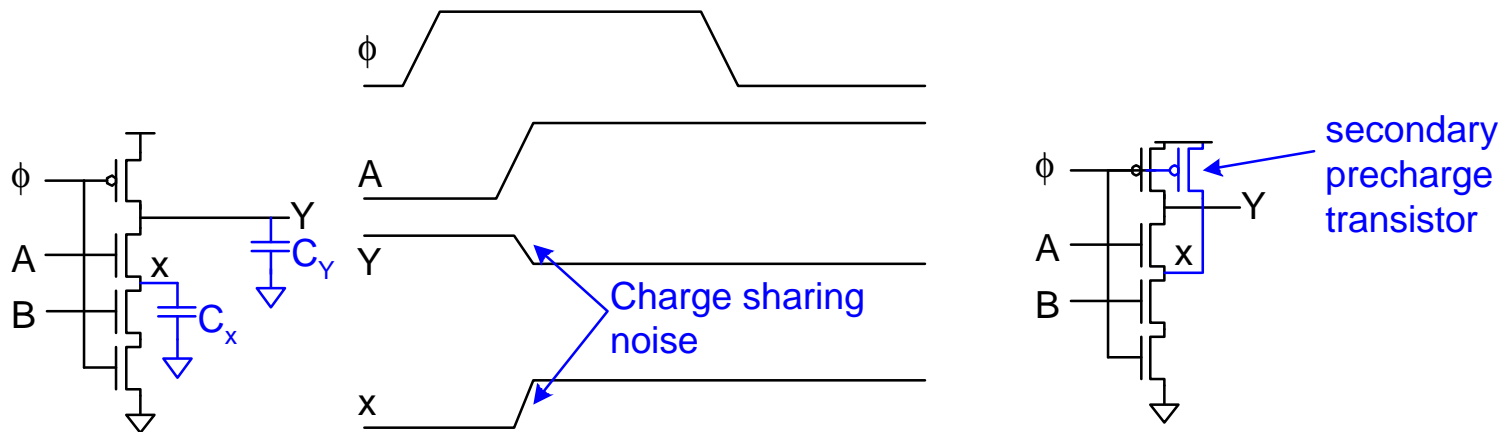
# Burn-In Keepers

- ❑ Leakage is a problem during burn-in @ high temp
- ❑ Need strong keeper for burn-in but don't want to slow gate during normal operation



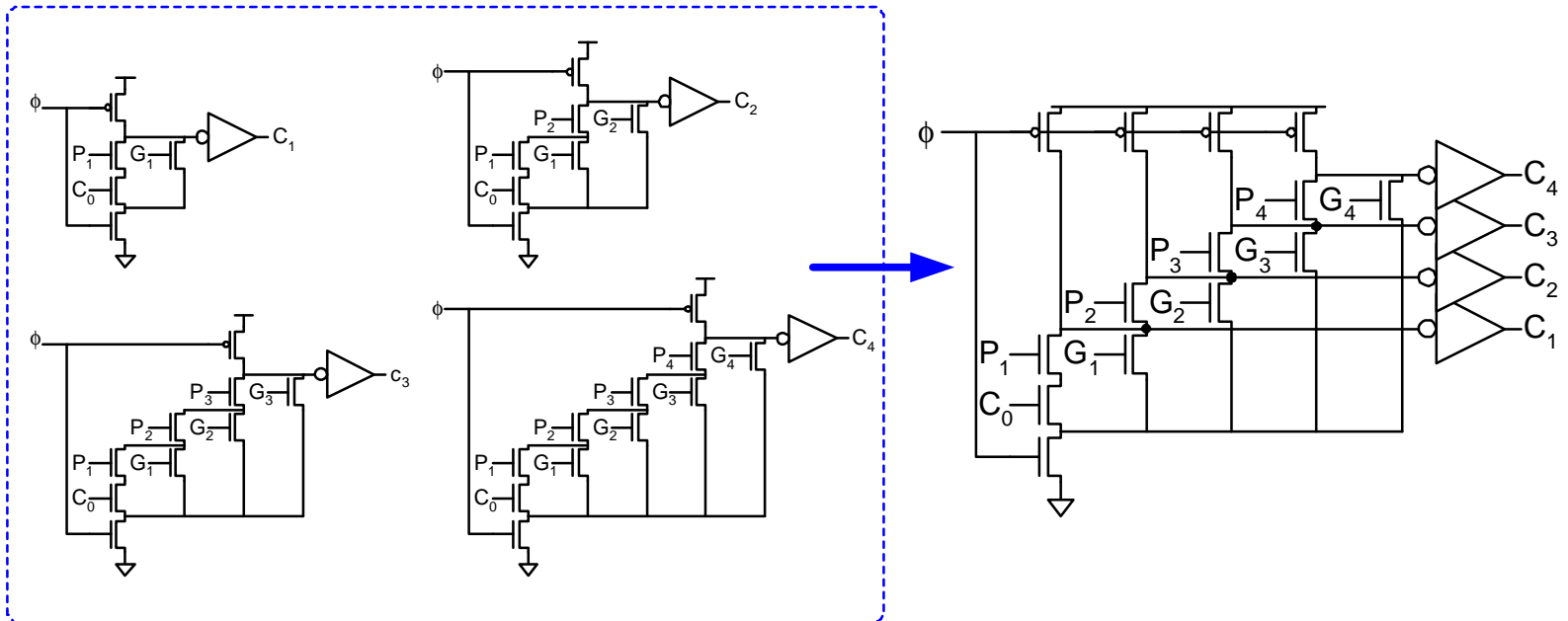
# Charge Sharing

- ❑ Dynamic gates are prone to charge sharing noise
- ❑ Secondary precharge transistor solves problem



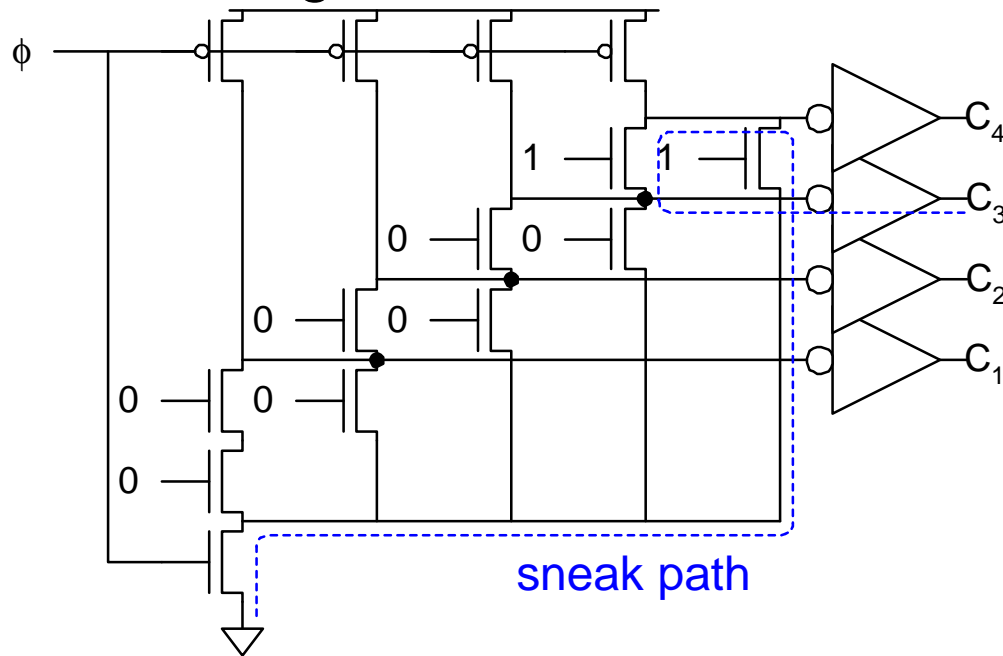
# Multiple Output Domino Logic

- ❑ One dynamic gate may drive multiple outputs
- ❑ Especially useful for adder carry chains



# Sneak Paths

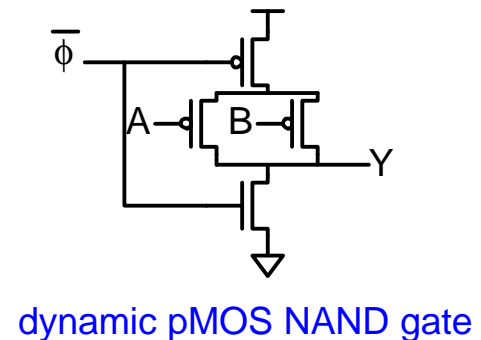
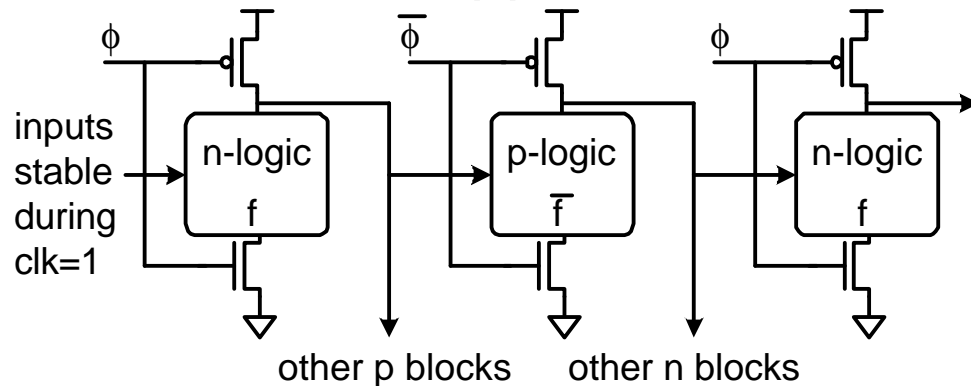
- ❑ Sneak paths can cause improper evaluation
- ❑ Prevented through mutual exclusion





# NORA & Zipper

- ❑ NP Domino / NO-Race alternate dynamic nMOS and pMOS stages
- ❑ Usually a bad idea: footed pMOS worse than static
- ❑ Also very sensitive to noise. AT&T CRISP  $\mu$ Proc
- ❑ Similar to Zipper domino



# Noise

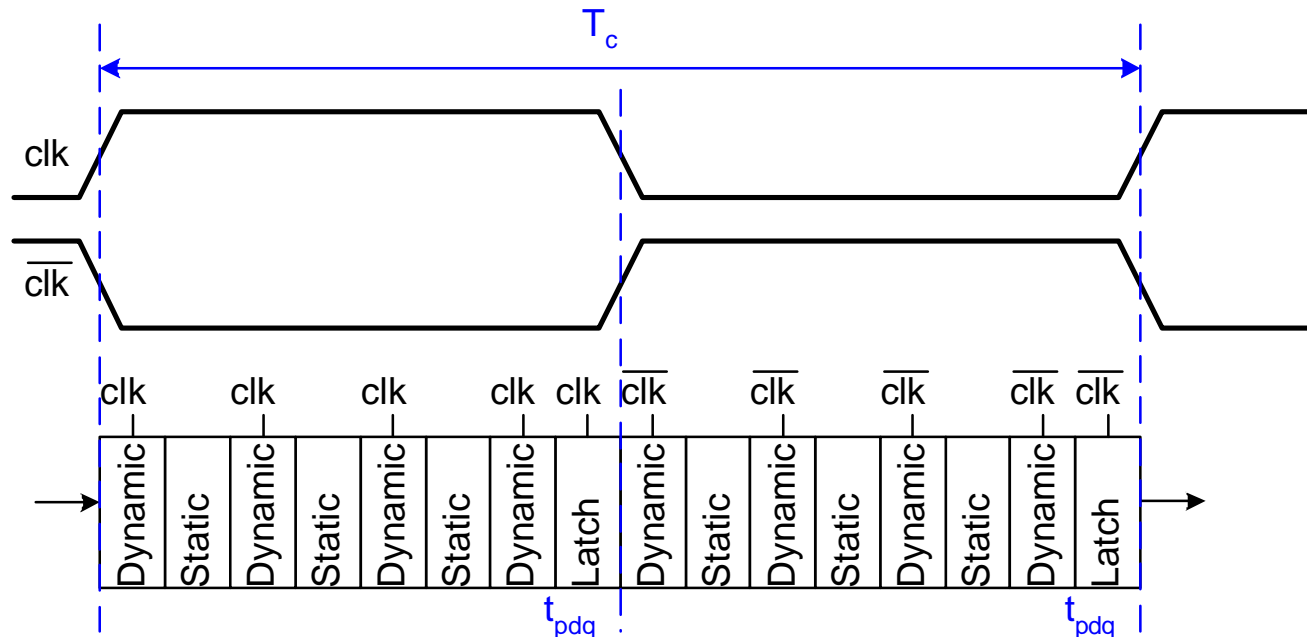
- ❑ Domino is sensitive to many noise sources including
  - Leakage keepers
  - Charge sharing secondary precharge
  - Capacitive coupling spacing & shielding
  - Back gate coupling circuit design
  - Power supply noise good supply grid
  - Minority carrier injection avoid injectors
  - Soft errors adequate capacitance

# Outline

- ❑ Domino Circuits
- ❑ **Domino Sequencing**
- ❑ Nonmonotonic Dynamic Techniques

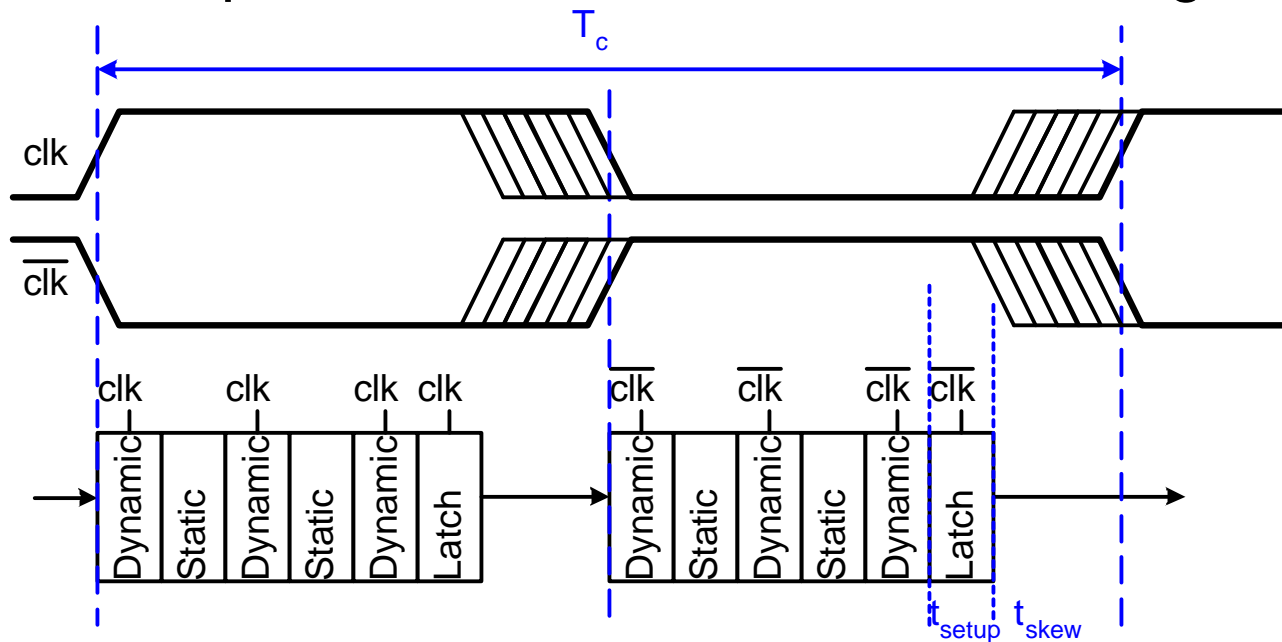
# Traditional Domino Sequencing

- ❑ One half-cycle evaluates while other recharges
- ❑ Latches hold results of stage that recharges



# Clock Skew

- ❑ Path starts on latest skewed rising edge of clock
- ❑ Must complete before earliest skewed falling edge

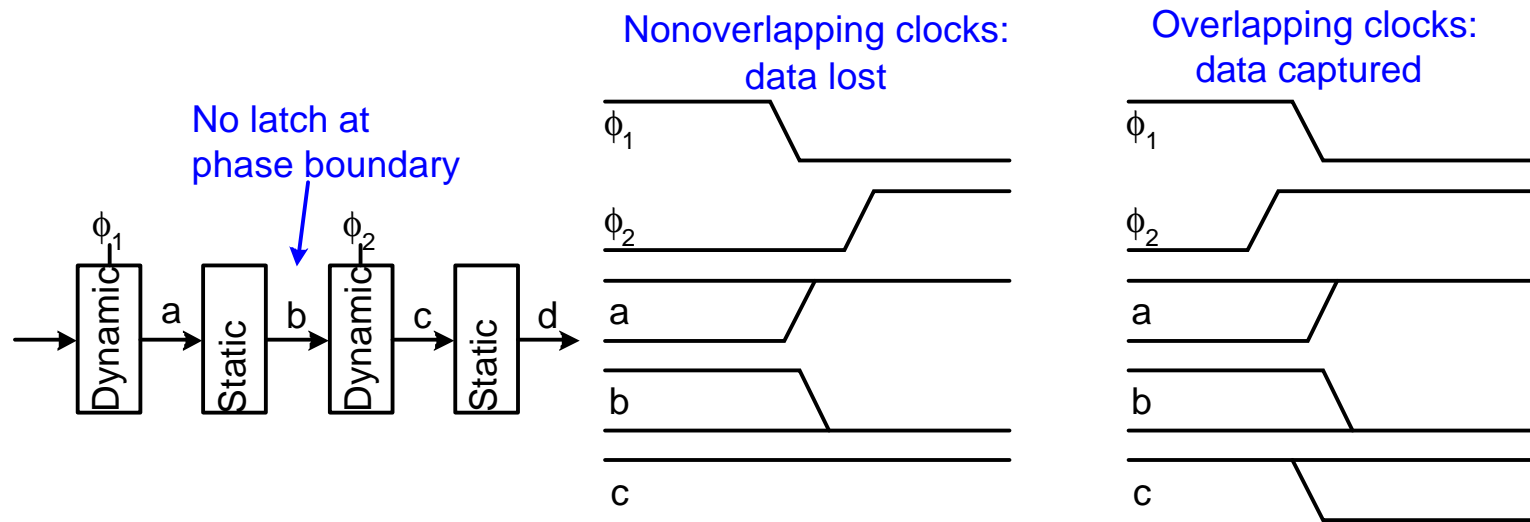


# Sequencing Overhead

- ❑ Latch and clock skew overhead in each half-cycle
- ❑  $T_{\text{logic}} = T_c - \{2t_{\text{setup}} + 2t_{\text{skew}}\}$
- ❑ Also unable to borrow time between half-cycles to balance paths
- ❑ Traditional domino sequencing has too much overhead to be practical
- ❑ Most companies have developed some skew-tolerant alternative to eliminate latches and overhead

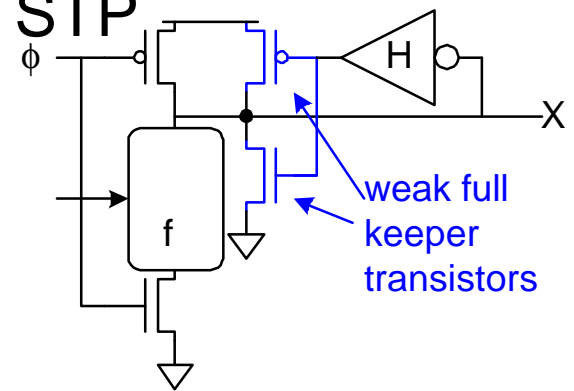
# Eliminating Latches

- ❑ If clocks overlap, latches aren't required to hold the data when a half-cycle precharges



# Skew-Tolerant Domino Sequencing

- ❑ Skew-tolerant domino techniques use multiple overlapping clock phases and eliminate latches to achieve zero sequencing overhead
- ❑ Many ways to do this:
  - OTB, N-phase, Delayed Reset, Self-Resetting, Postcharge, SRCMOS, Global STP
- ❑ Full keeper holds state when input precharges

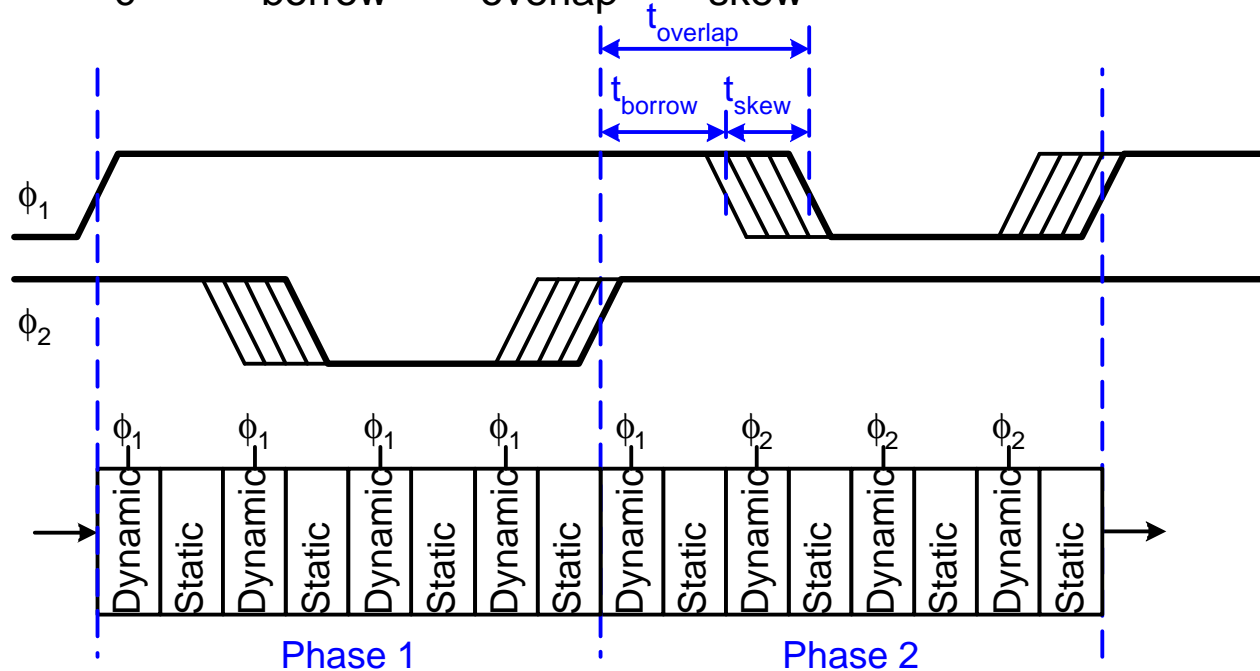




# Time Borrowing

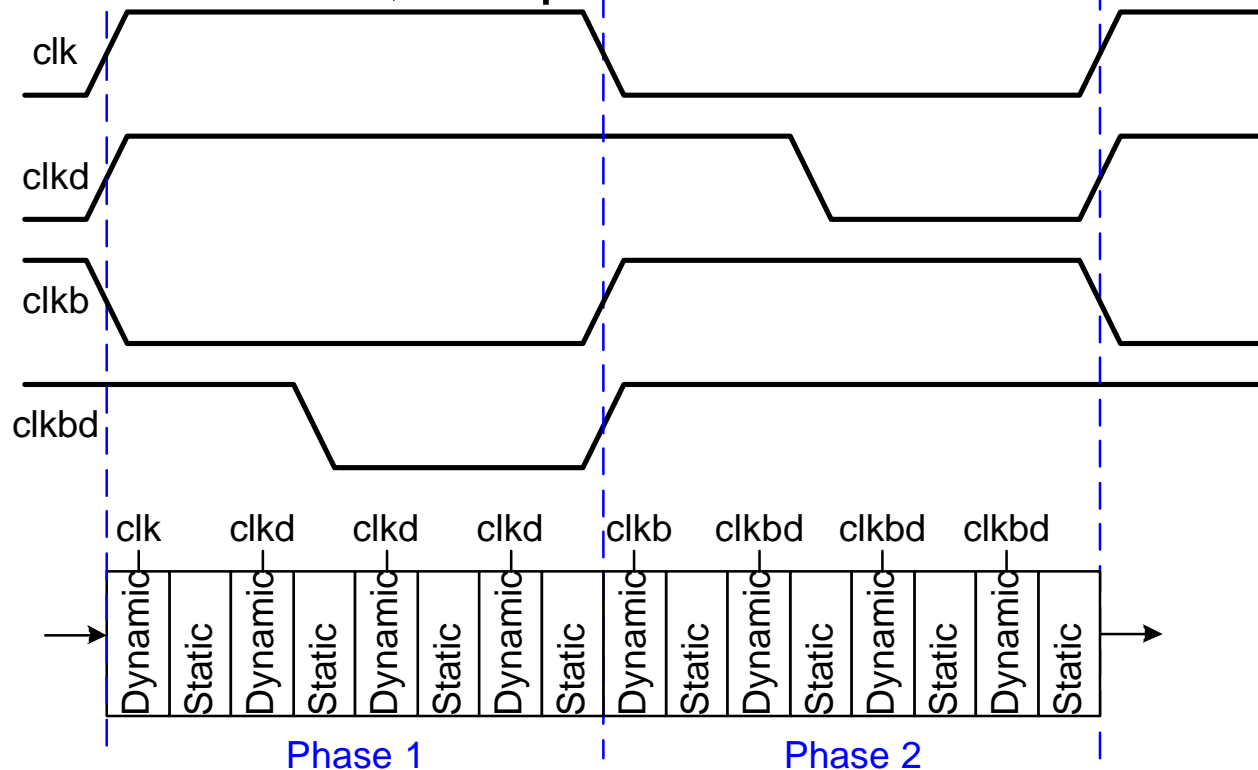
❑ Skew-tolerant domino permits time borrowing

❑  $t_{\text{logic}} = T_c$ ;  $t_{\text{borrow}} = t_{\text{overlap}} - t_{\text{skew}}$



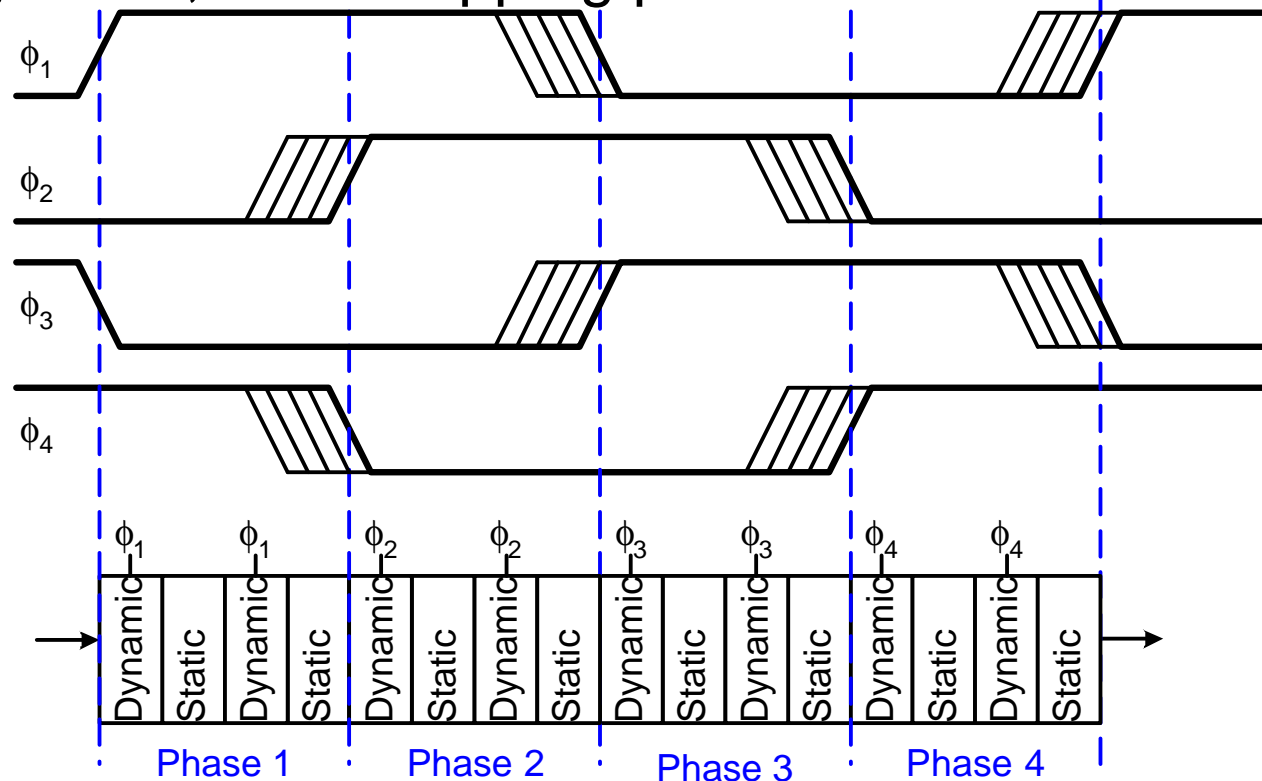
# Opportunistic Time Borrowing Domino

- ❑ Used on Itanium; two phases but no latches



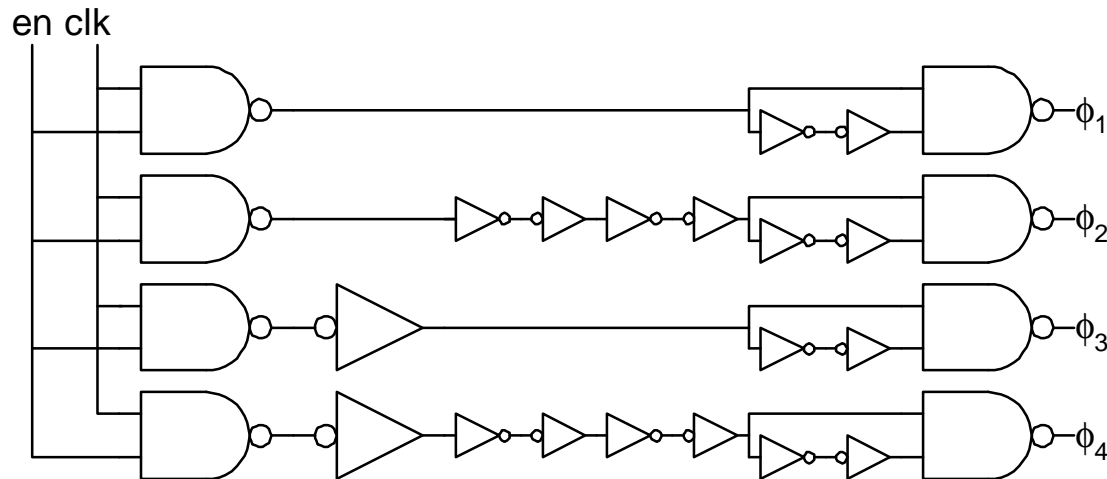
# N-Phase Domino

- In general, N overlapping phases for more overlap



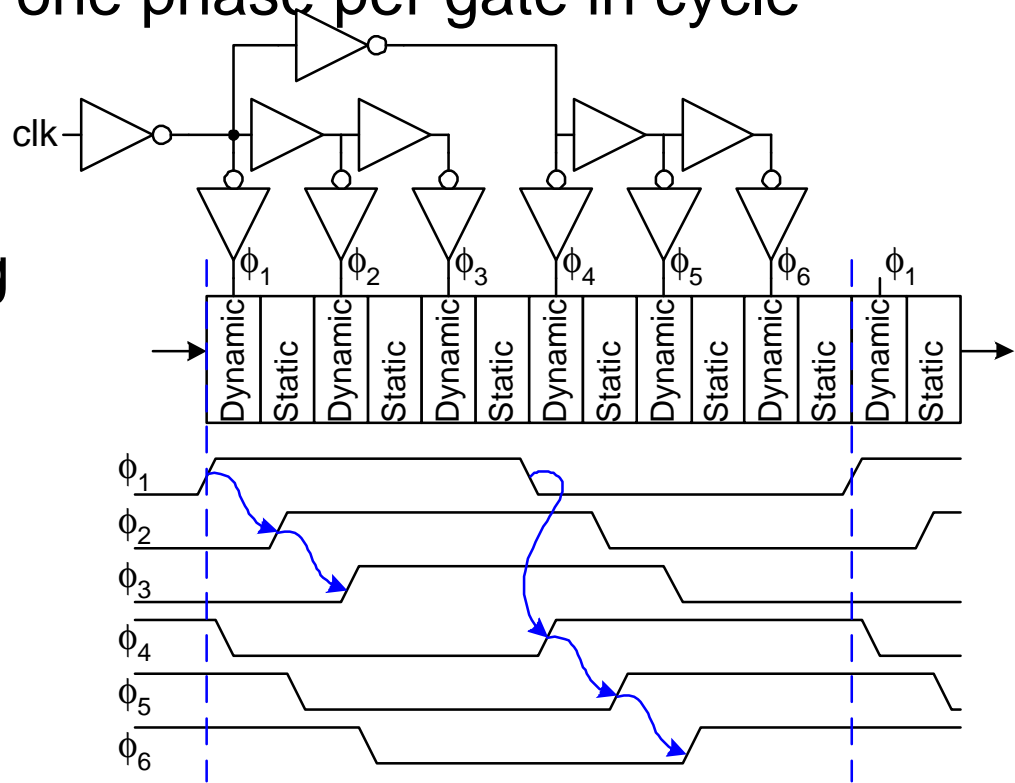
# Four-Phase Domino

- ❑ Itanium 2 uses four-phase Skew-Tolerant Domino
- ❑ Simple clock generation at clock gaters
- ❑ Delay each phase by  $\frac{1}{4}$  cycle
- ❑ Optional clock choppers can increase overlap



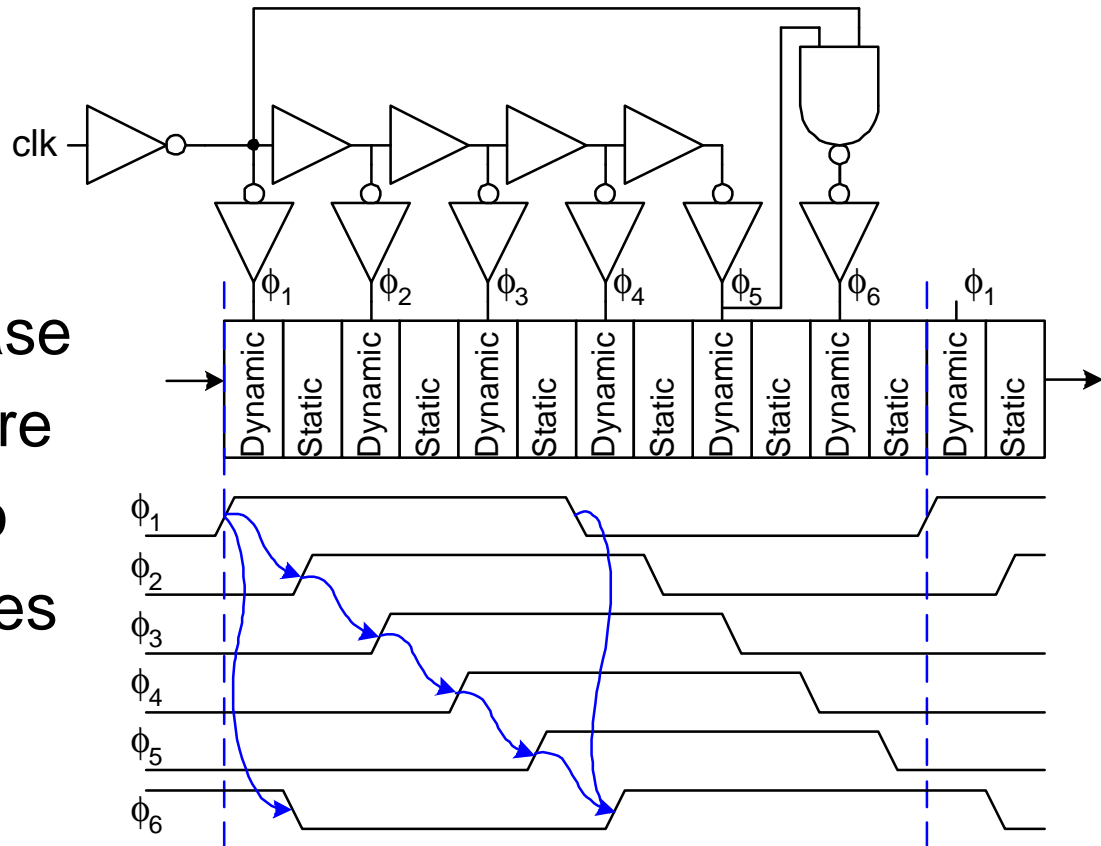
# Delayed Reset

- Alternatively, use one phase per gate in cycle
- Delayed Reset, Cascaded Reset
- Delayed Clocking (IBM & Sun)
- Can use both clock edges



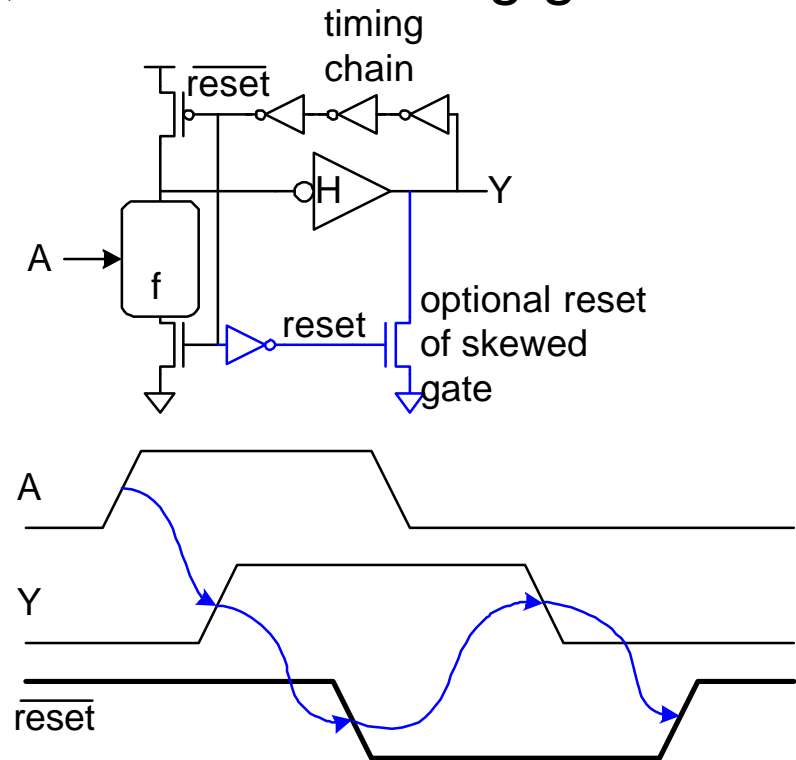
# More Delayed Reset

- ❑ Or just one
- ❑ These N-phase techniques are well-suited to unfooted gates



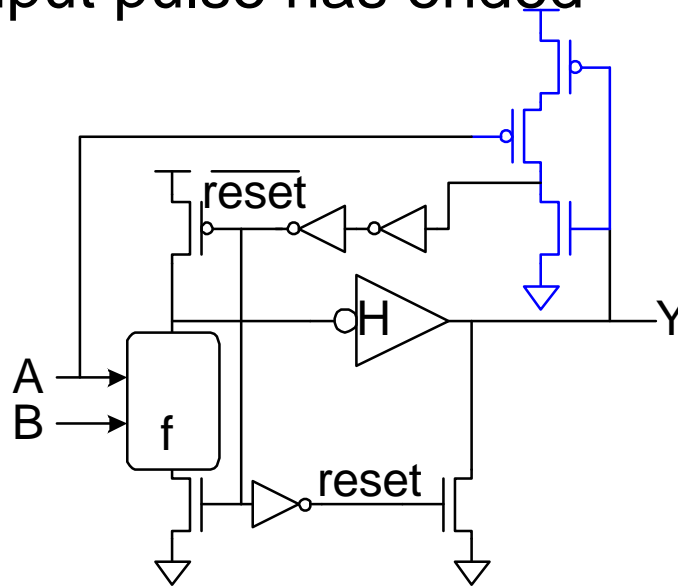
# Self-Resetting Domino

- ❑ Instead of supplying clock, use self-resetting gate
- ❑ Gate precharges itself five gate delays after evaluating
- ❑ No power consumed when gate is idle, but complicated timing analysis
- ❑ Ideal for RAM decoders



# Predicated Self-Reset

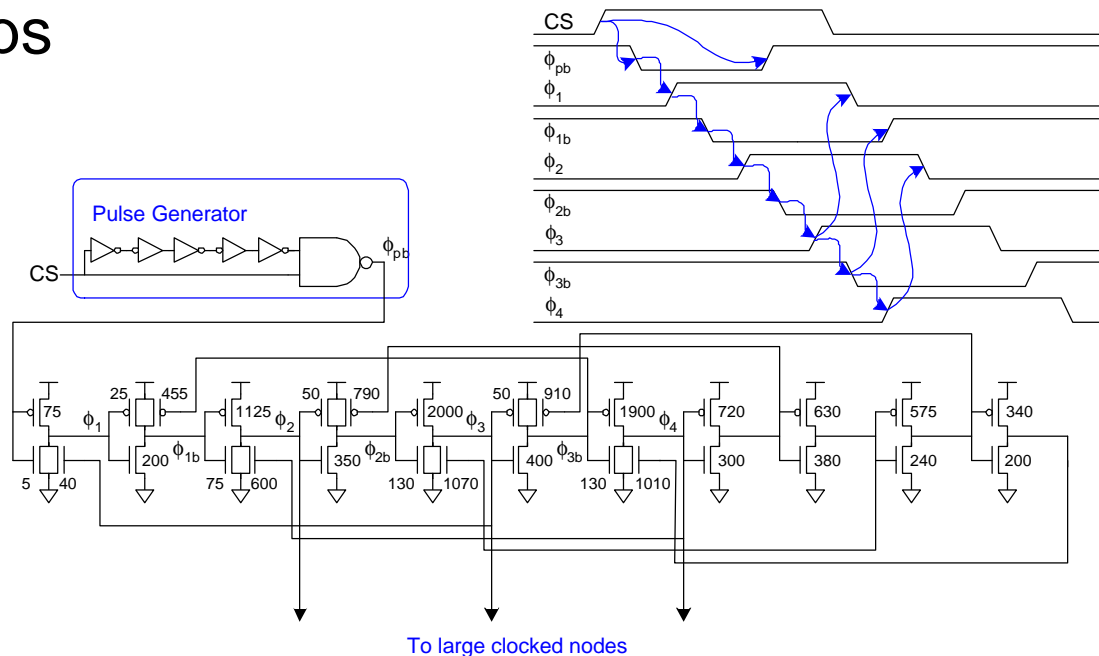
- ❑ Self-resetting domino requires pulsed inputs
- ❑ Predicated self-resetting domino stretches output pulse until input pulse has ended





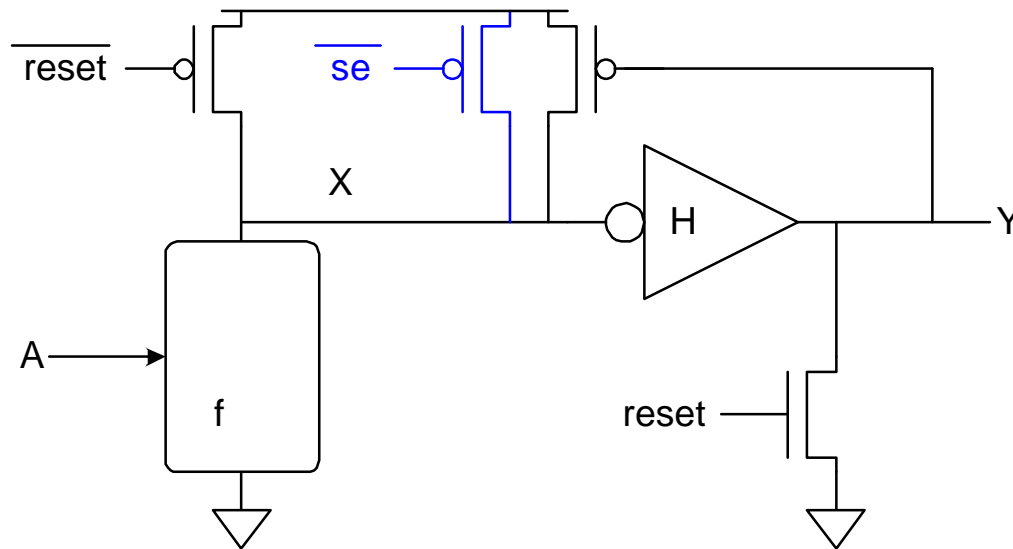
# Postcharge Logic

- ❑ Unfooted self-resetting n and p dynamic gates
- ❑ Used to rapidly amplify leading edge of chip select in RAM chips



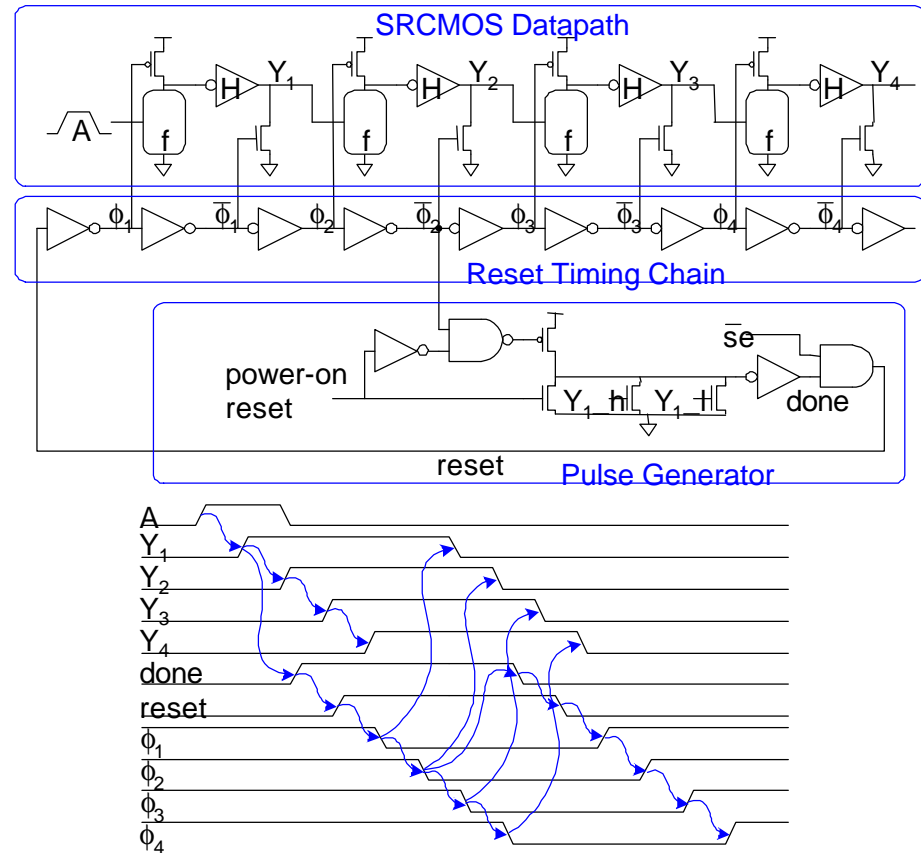
# SRCMOS

- ❑ IBM variant of self-resetting gates with static eval.
- ❑ Turns gate into pseudo-nMOS during low-speed test



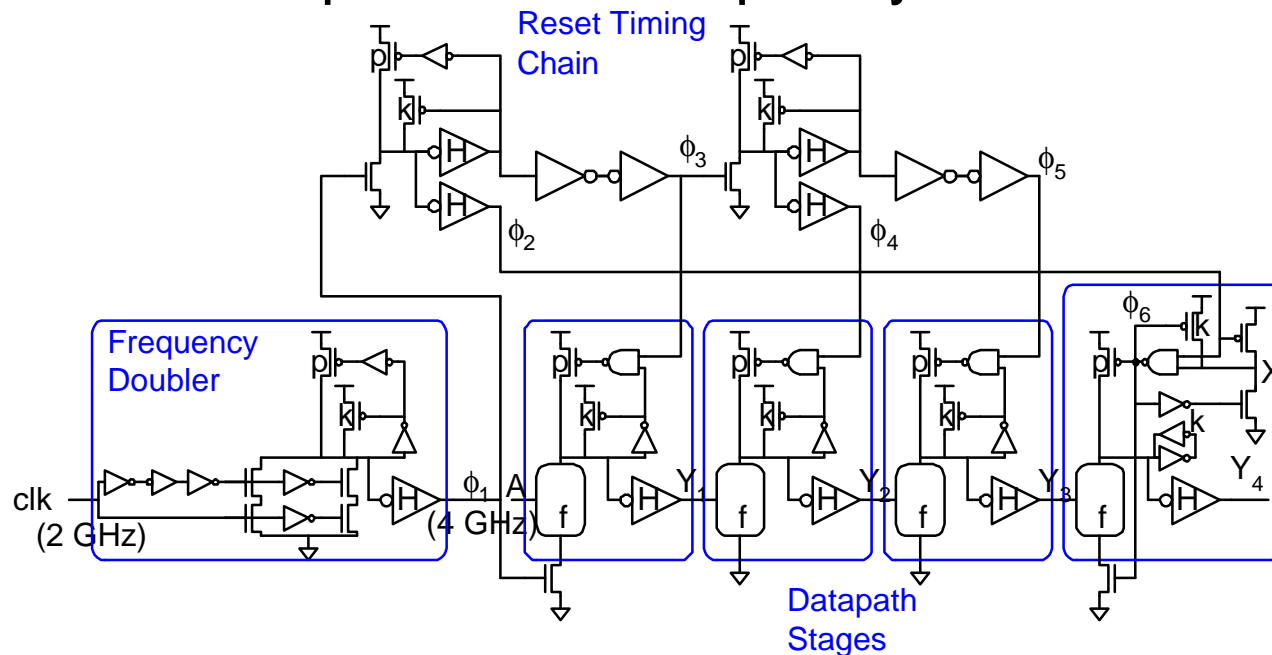
# SRCMOS Example

- ❑ Amortize cost of self-resetting pulse generator across many gates.
- ❑ Use a timing chain to produce delayed clocks.



# Global Self-Terminating Precharge

- ❑ Intel variant of self-resetting gates from Pentium 4
- ❑ Derives initial pulse from frequency doubler



# Summary

- ❑ Skew-tolerant domino uses overlapping phases to eliminate latches and sequencing overhead
- ❑ More overlap permits more time borrowing
- ❑ How to generate clock:
  - Global number of fixed phases
    - Simplest option, easy to analyze
  - Inverter chain with one gate per stage
    - More convenient for unfooted gates
  - Self-resetting pulse generator
    - Saves clock power, very complicated design

# Outline

- ❑ Domino Circuits
- ❑ Domino Sequencing
- ❑ Nonmonotonic Dynamic Techniques

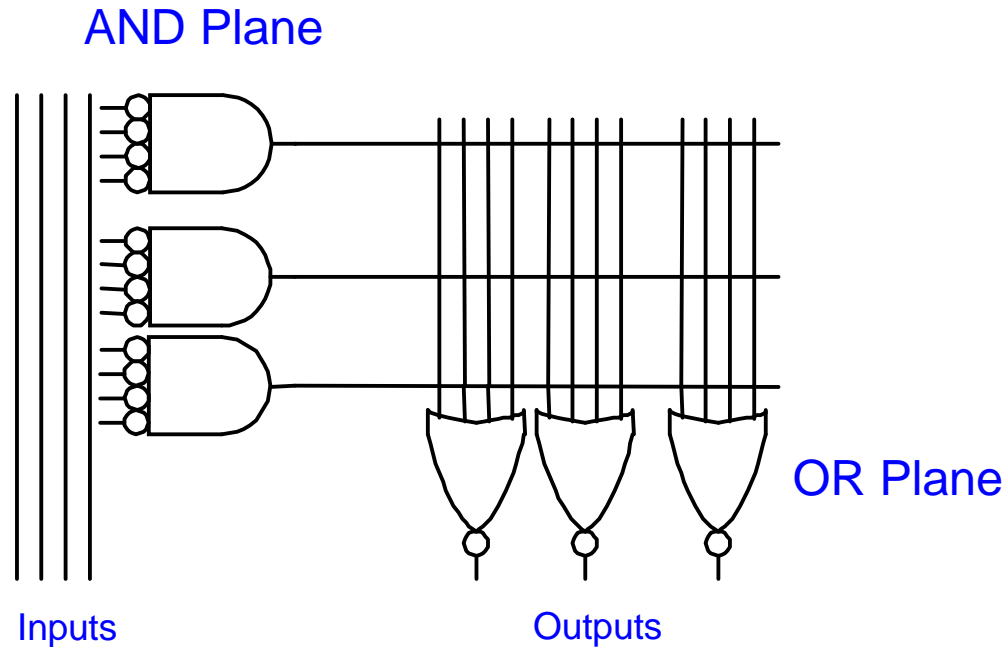
# Nonmonotonic Dynamic Techniques

- ❑ Dynamic gates require monotonically rising inputs during evaluation so dynamic gates with same clock cannot be cascaded
- ❑ Domino provides these but only computes noninverting functions
- ❑ Dual-rail domino computes any function but is costly
- ❑ Nonmonotonic dynamic techniques cascade dynamic gates with delayed clocks so first gate settles before second evaluates

# NOR-NOR Functions

- ❑ Dynamic gates make very fast NORs
- ❑ NOR – NOR cascade attractive (= AND – OR)
- ❑ Require

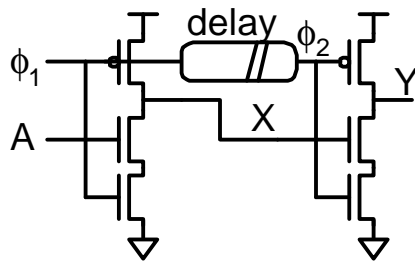
nonmonotonic  
techniques



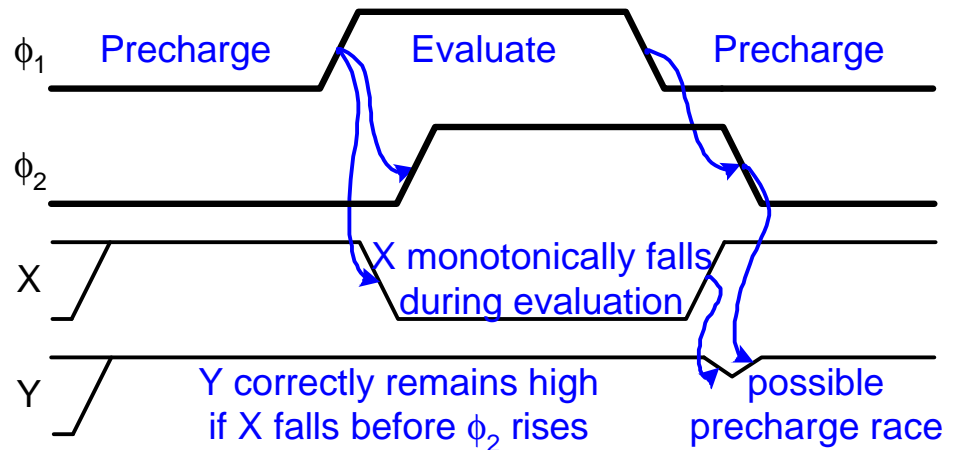


# Clock-Delayed Domino

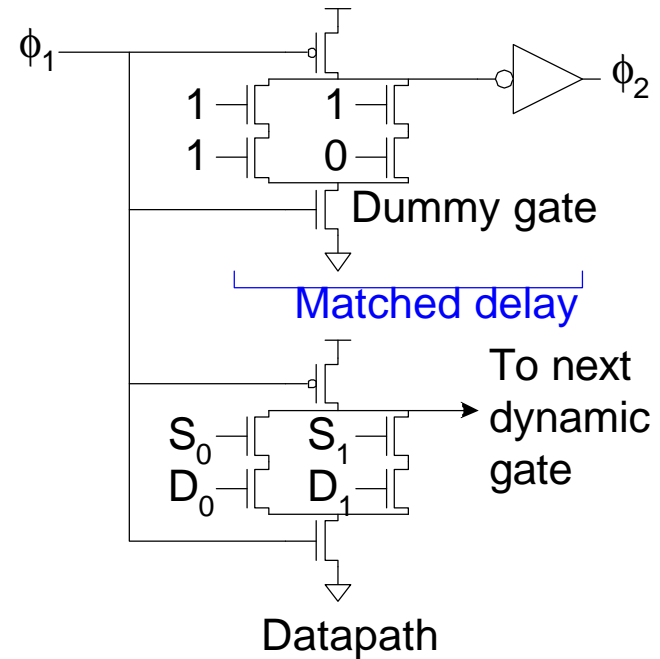
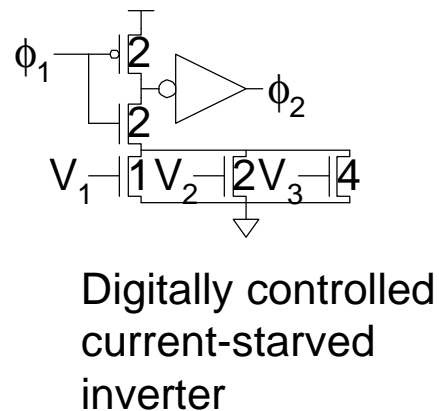
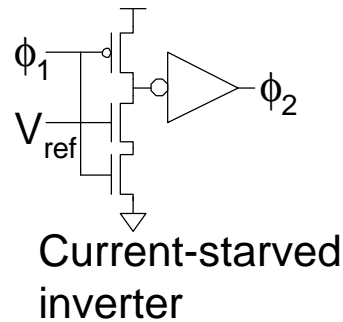
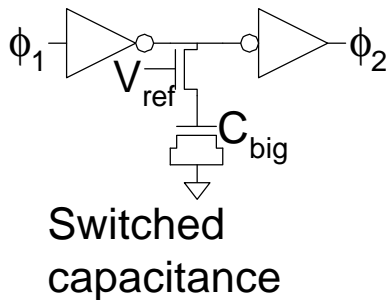
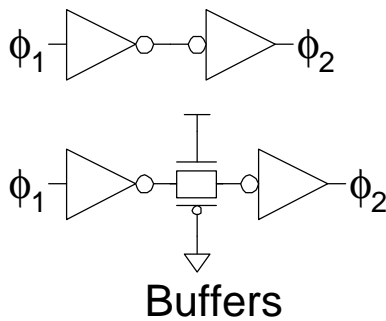
- ❑ Delay the clock to the second dynamic gate
- ❑ Add 30% margin for process/environmental variation
- ❑ Less margin with replica delay lines



$A = 1$



# Delay Elements

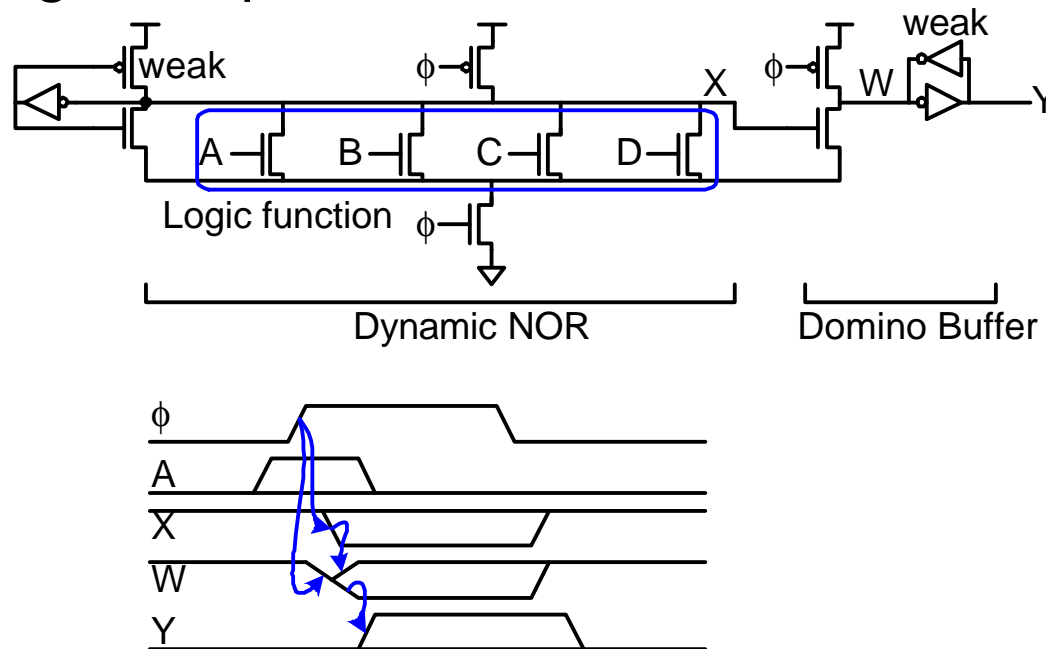


# Race-based Logic

- ❑ Another common problem is a fast AND function (e.g. memory decoder)
- ❑ NAND uses series transistors. NOR uses parallel.
- ❑ Prefer to recast as NOR of inverted inputs
- ❑ But need a monotonically rising output to drive subsequent domino stages
- ❑ Several tricky circuits depend on races:
  - Annihilation gates (Itanium2)
  - Latched domino
  - Complementary Signal Generator (Intel)

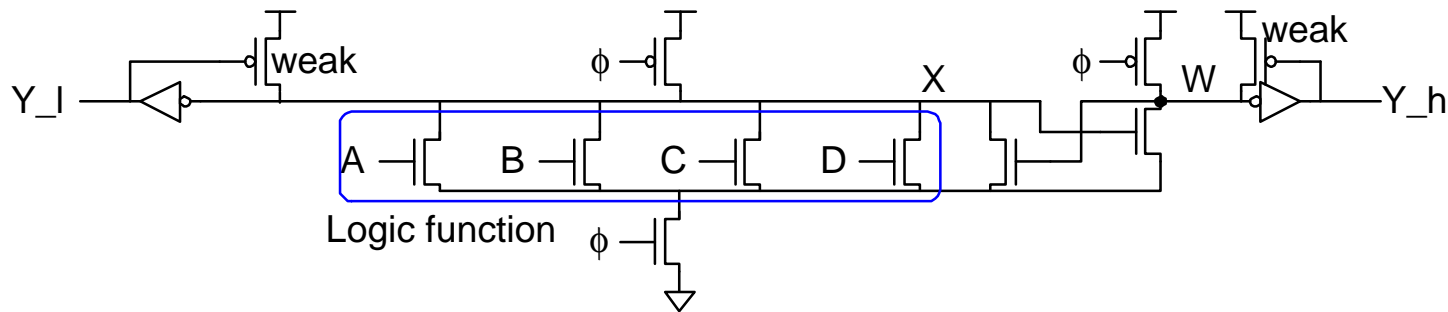
# Annihilation Gates

- ❑  $X = \sim(A + B + C + D)$ ;  $Y = \bar{A}\bar{B}\bar{C}\bar{D}$
- ❑ W begins to pull low, but recovers if X falls rapidly



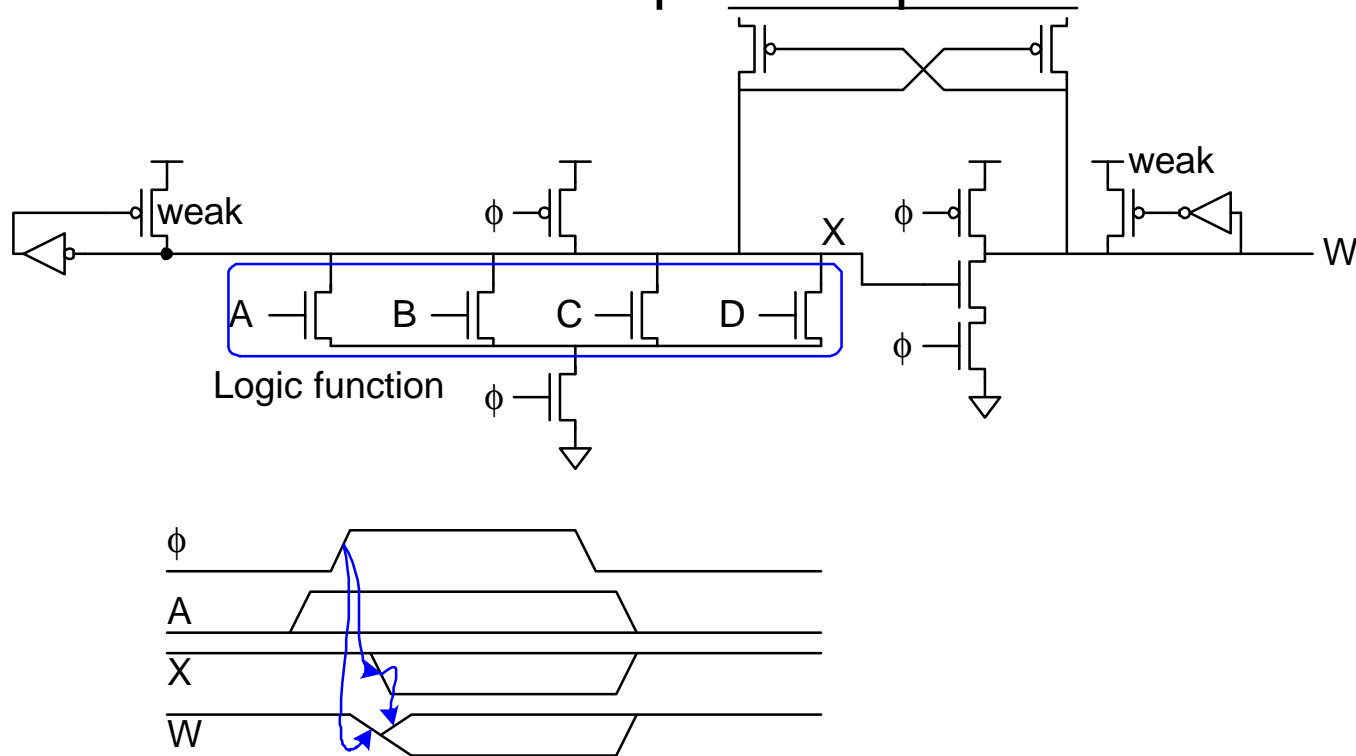
# Latched Domino

- ❑ Latched Domino uses a different keeper structure



# Complementary Signal Generator

- ❑ CSG uses a cross-coupled keeper

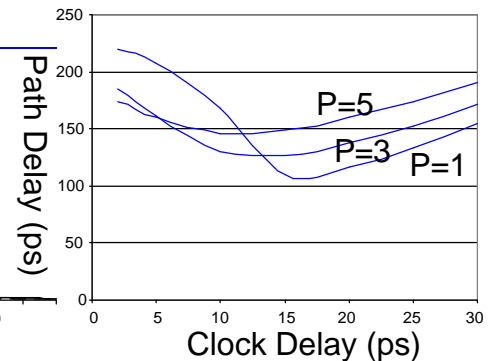
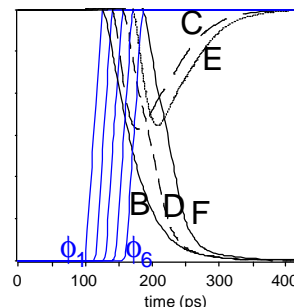
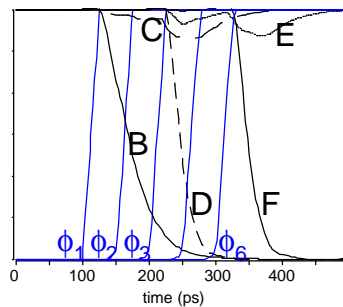
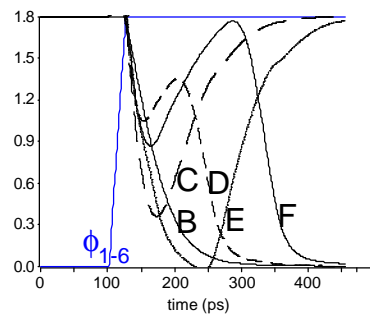
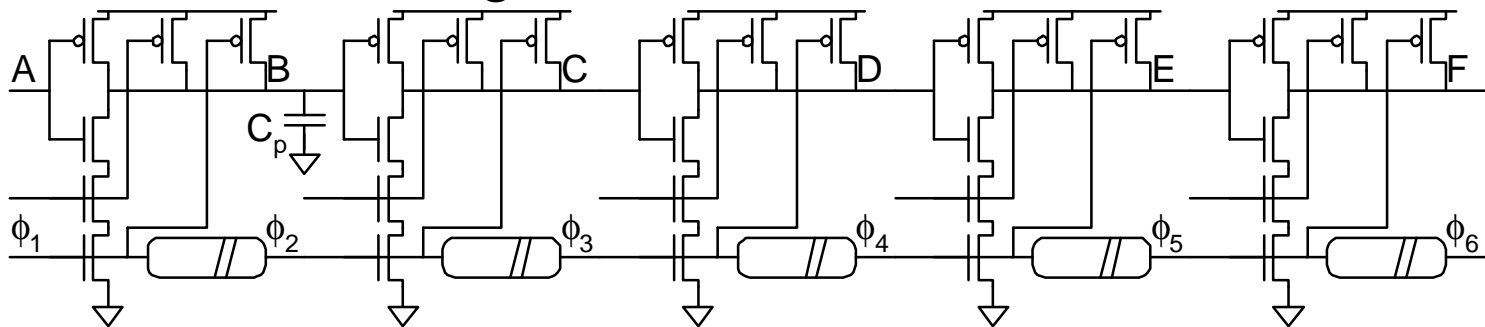


# Output Prediction Logic

- ❑ Race-based logic suggests that inputs need not be monotonically rising so long as keeper can recover.
- ❑ Output Predication Logic directly cascades dynamic gates, as in Clock-Delayed Domino.
- ❑ But clock delays are short enough that stages will glitch. NTP gates are used to recover well.
- ❑ If the delays are too short, the glitches will flip the gate and recovery is very slow. If the delays are too long, the circuit behaves as CD Domino. If the glitches are just right, the circuit could be very fast.

# OPL Waveforms

- Path delay depends on delay line length and pMOS widths in NTP gates.





# OPL Summary

- ❑ OPL seems attractive because it is extremely fast
- ❑ But...
  - The best delays are very short. How do we really generate them?
  - How much margin must be provided for process and environmental variations?
- ❑ OPL advantages have yet to be convincingly demonstrated in silicon

# Conclusion

- ❑ Domino is attractive for 1.5-2x speedup
- ❑ Careful design required for noise and monotonicity
- ❑ Traditional domino sequencing has much overhead
- ❑ Skew-tolerant domino eliminates this overhead
- ❑ Many flavors of skew-tolerant domino with clocked and self-timed precharge
- ❑ Nonmonotonic structures exploit very fast dynamic NORs

---

# **Part 2**

# **A Domino Methodology and Some Common Pitfalls**

Tom Grutkowski  
Intel

# Outline of remainder of tutorial.

---

## □ Goals

- A Detail discussion of a complete domino methodology in use on a production microprocessor.
- Outline some of the common pitfalls found in domino design.
- Illustrate some actual silicon bugs.
- Inspire a *little* fear.

# Itanium 2 Background

- ❑ Co developed by Intel and HP.
- ❑ Implement EPIC ISA
- ❑ Code Names:
  - McKinley: 180 nm product.
  - Madison: 130 nm product.
- ❑ Runs up to 1.5 GHz.
- ❑ 130 Watts limit for both 180 nm and 130 nm products.
- ❑ Area.
  - McKinley: 421 mm<sup>2</sup>
  - Madison: 374 mm<sup>2</sup>

# Itanium 2 Domino Circuitry

- ❑ Integer execution unit
  - 6 pipes, single cycle bypass.
- ❑ Multi-Media pipes.
  - 6 pipes, two cycle latency.
- ❑ 2 Floating Point FMAC/FMISC units
- ❑ Much of the pipe control.
- ❑ Out of order control issue logic for 2<sup>nd</sup> level cache.
- ❑ Register file.
  - Integer and Floating Point Register File
  - 60+ miscellaneous register files.

# Itanium 2 Dynamic Methodology

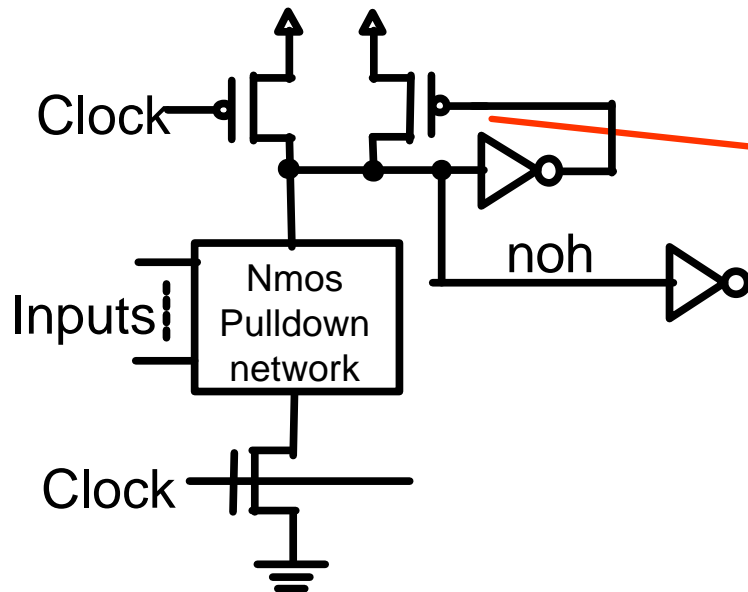
- ❑ How can we improve on OTB?
- ❑ OTB Features:
  - Provides for removal of latches.
  - Allows time borrowing across clock phases.
- ❑ What would we like?
  - Small, flexible, and **robust**.
  - Scan capability on dynamic “latches”.
  - Standard interface from dynamic to static.
  - Standard interface from static to dynamic.
  - Limit number of clocks.

# Why Scan Capable Domino?

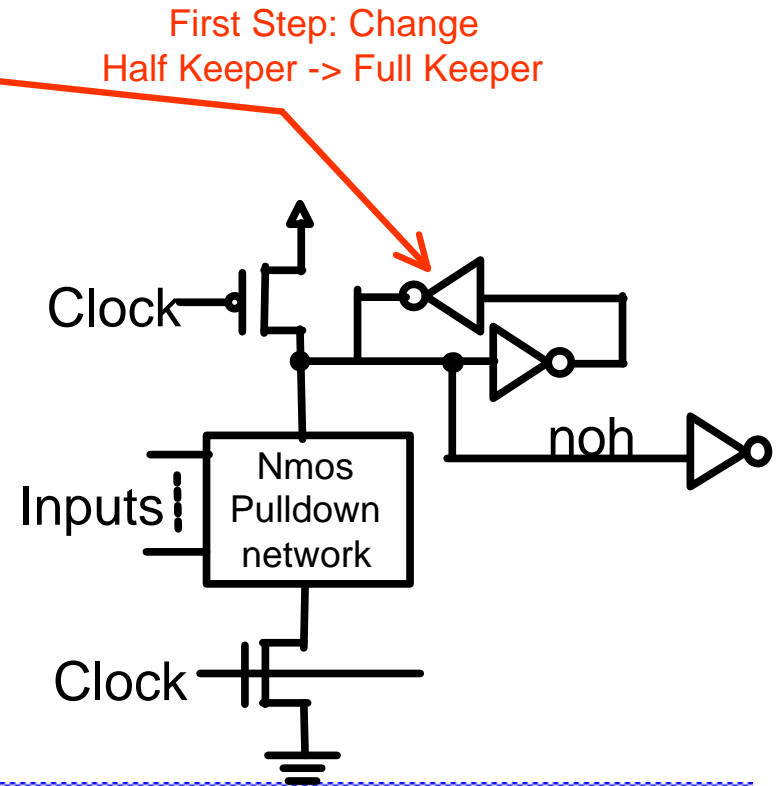
- ❑ What is scan?
  - Ability to **observe** and **control** state elements through a serial chain controlled by the Test Access Port (TAP).
  - Enables small portions of the design to be tested and debugged in isolation.
  - Two varieties
    - Destructive: Data in state elements destroyed during scan operation.
    - Non-Destructive: Data in state elements preserved.
  - Definitions:
    - Full Scan=> **All** State Elements are scanned.
    - ROSL => Read Only Scan Latch; no controls
- ❑ Scan base testing is used at wafer sort to isolate manufacturing defects before packaging.
- ❑ Itanium 2™ example:
  - Single Cycle: Integer Execution: 100% domino.
  - Four Cycle Floating Point: 100% domino.
- ❑ Conclusion: Without a scan-capable domino methodology, scan vector coverage can be severely limited.



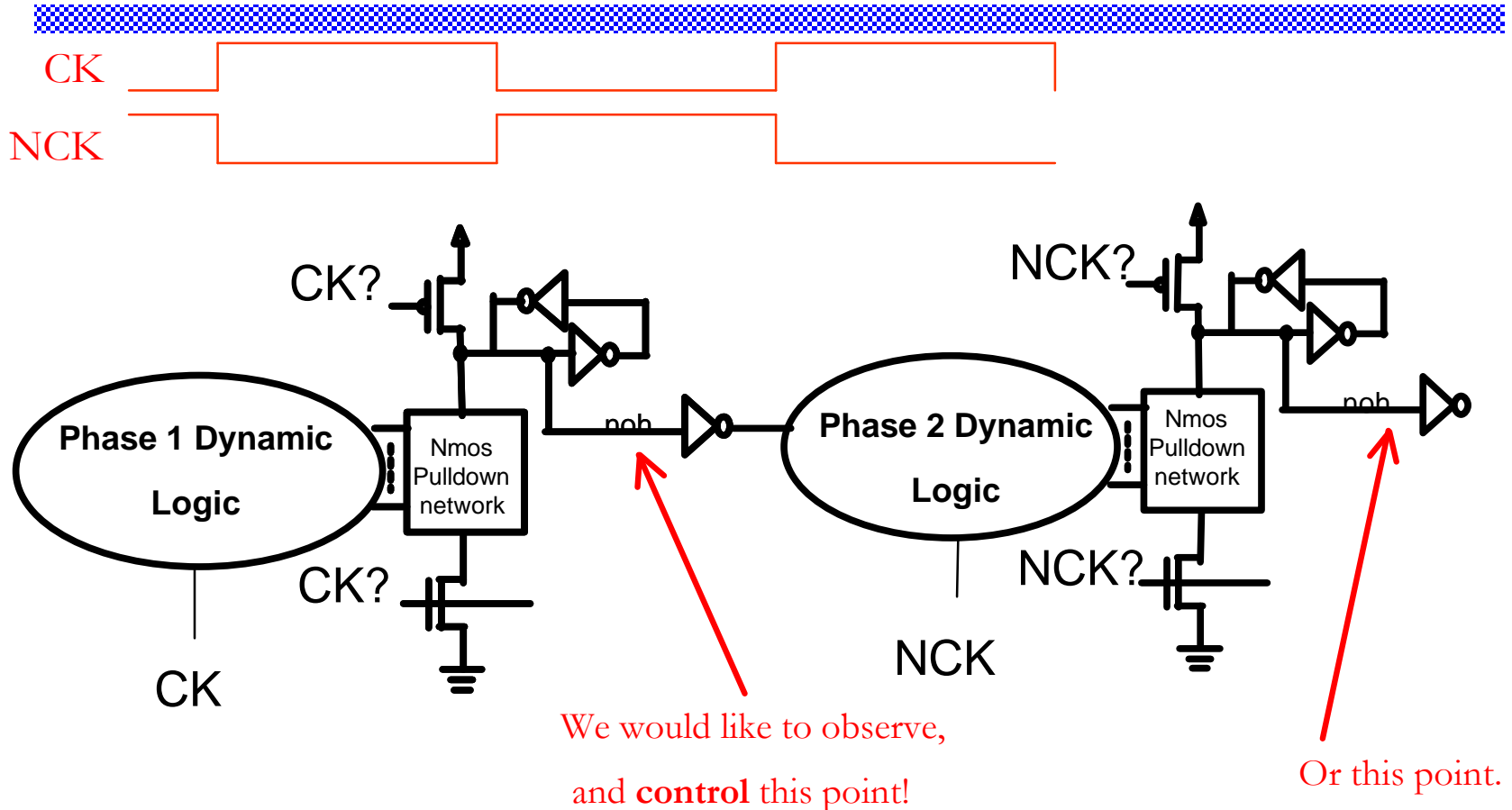
# Scan Capable Domino 'latch'.



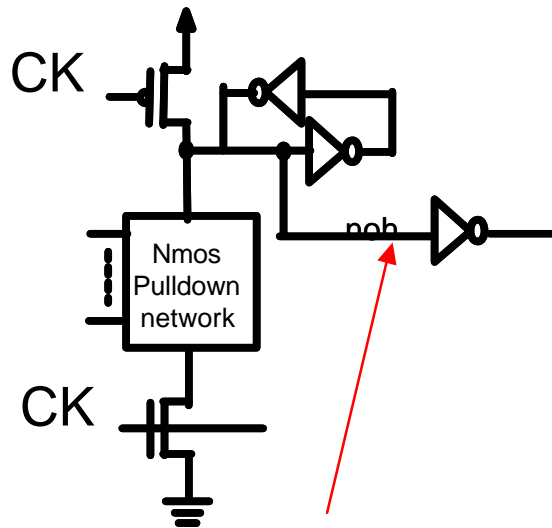
Full Holder allows noh to hold a low arbitrarily long after inputs have 'precharged' away.



# Where do we scan?



# Scanning Data In

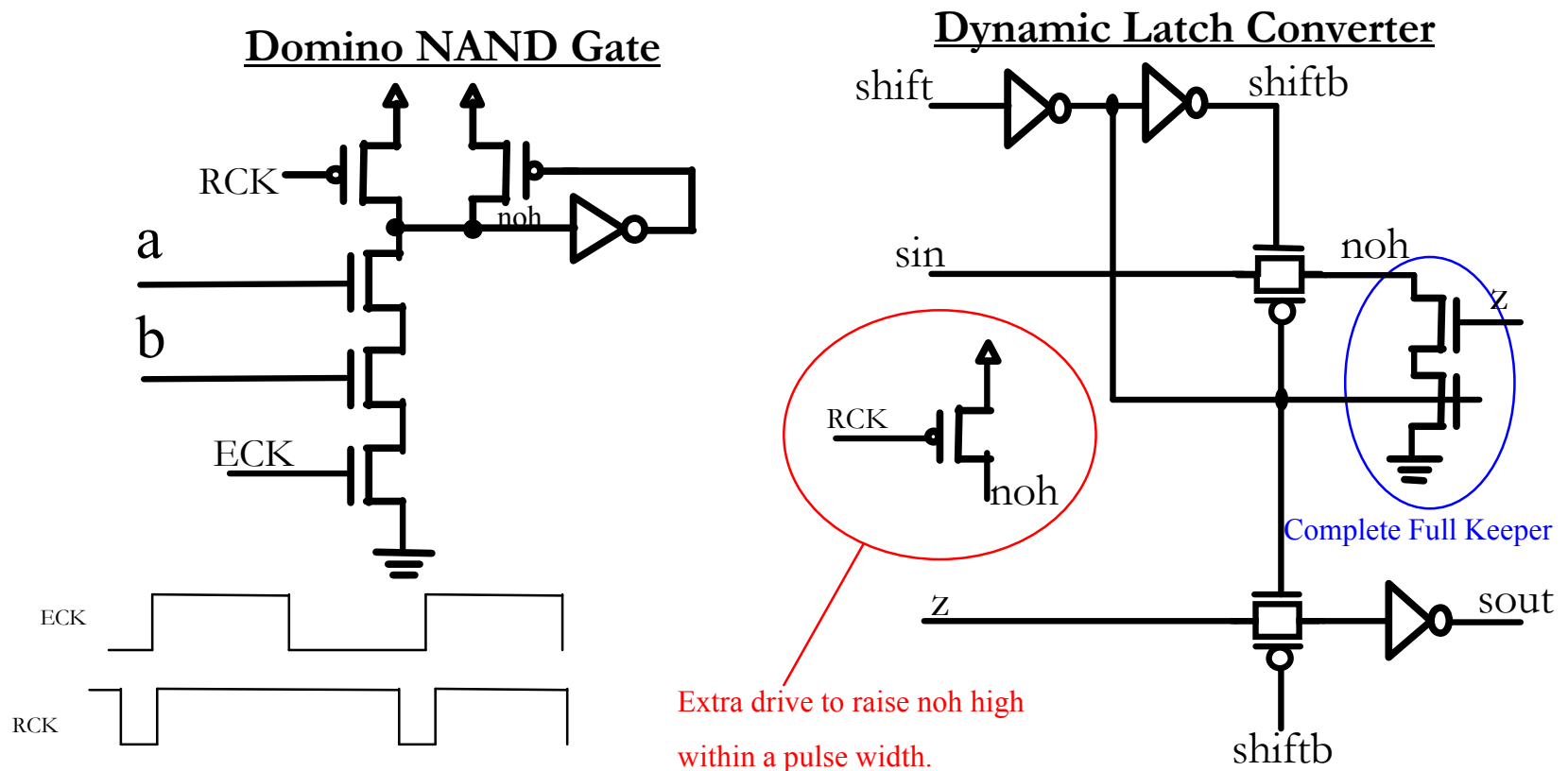


- ❑ Observing state element relatively easy.
  - Attach ROSL to node.
- ❑ Impossible to scan data into noh without significant drive fight.
  - When CK is low, noh is being pulled high by precharge FET.
  - When CK is high, would fight against evaluation stack.

---



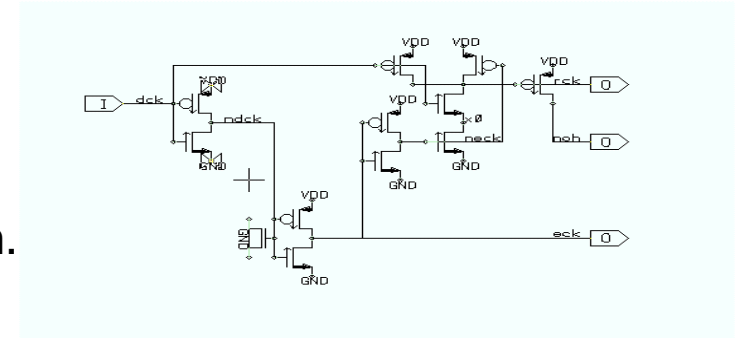
# Bolting on a DLC



# Generating RCK and ECK

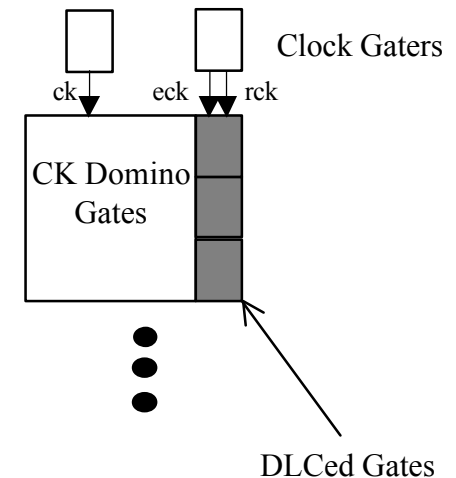
## Local Generation

- No significant RC concerns.
- Area expensive.
  - Need a generator for each latch.
- Used in domino control.



- ❑ “Global” generation

- Used in data path applications
- Need to control RC
  - Especially on RCK
  - Tolerable Skew: ~2-3% of cycle time
- Area efficient.
- One generator for each ‘register’.



# Concerned about Pulse?

---

## ❑ Pulse Concerns.

- Must be sufficient long enough to properly pre-charge node.
- Must also have limited overlap with ECK to avoid excessive short-circuit current, and delayed evaluation.

## ❑ Itanium 2 already uses **Pulse Latches** for Static Flip Flops.

- Establish methodology for distributing pulse clocks.
  - RCK add only incremental risk.
-

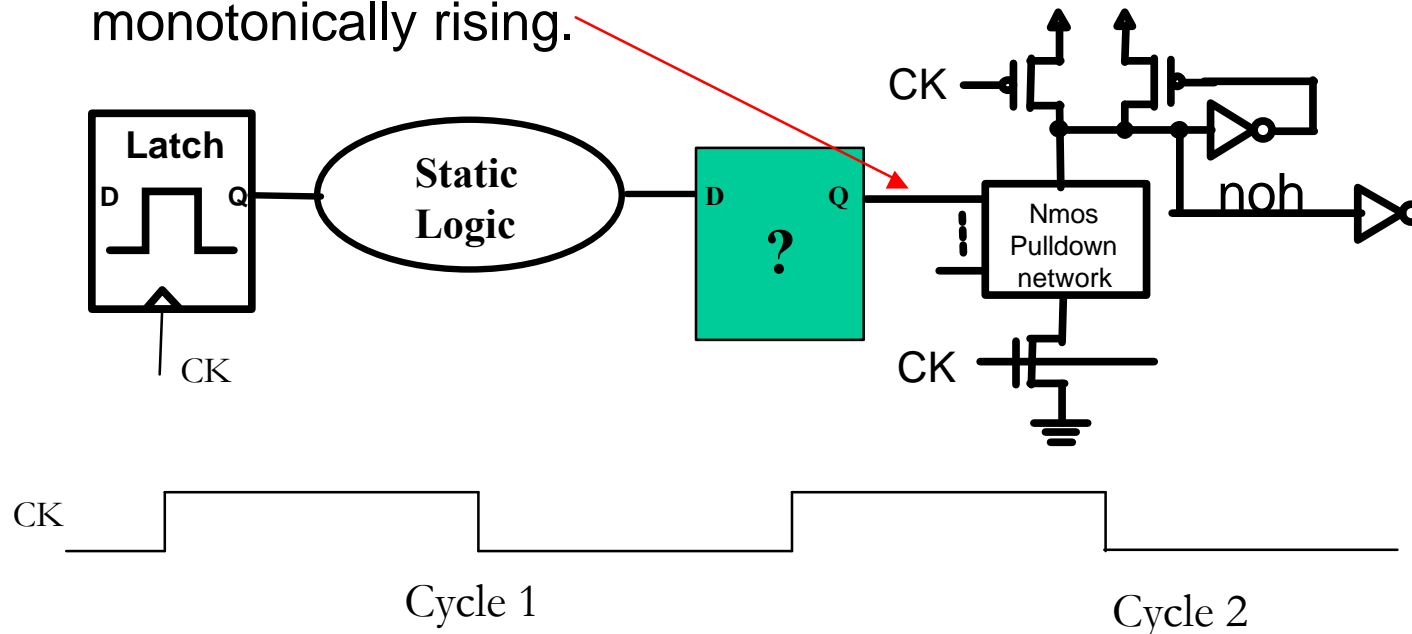
# Summary of DLC Benefits

- ❑ Flexibility
  - Any standard dynamic gate can be converted DLC.
- ❑ Small Overhead
  - A DLC consist as few as 12 transistors.
- ❑ Creates a static output.
  - A signal generated by a first phase DLC will remain stable throughout the second phase.
  - This saves on the need for extra latches or catchers.
- ❑ Scan Capable.
  - Enables *nearly* full scan across the design.
  - Tremendous benefit in silicon debug. Sequential depth ~ 1.



# Static to Dynamic Interface

- ❑ Problem: How do we take a static signal and introduce it to a dynamic circuit?
- ❑ Domino input must either be stable on CK rising edge, or monotonically rising.



# Some Options

---

## ❑ CK Flip Flop

- Just **doesn't** work! Not stable, not monotonic.

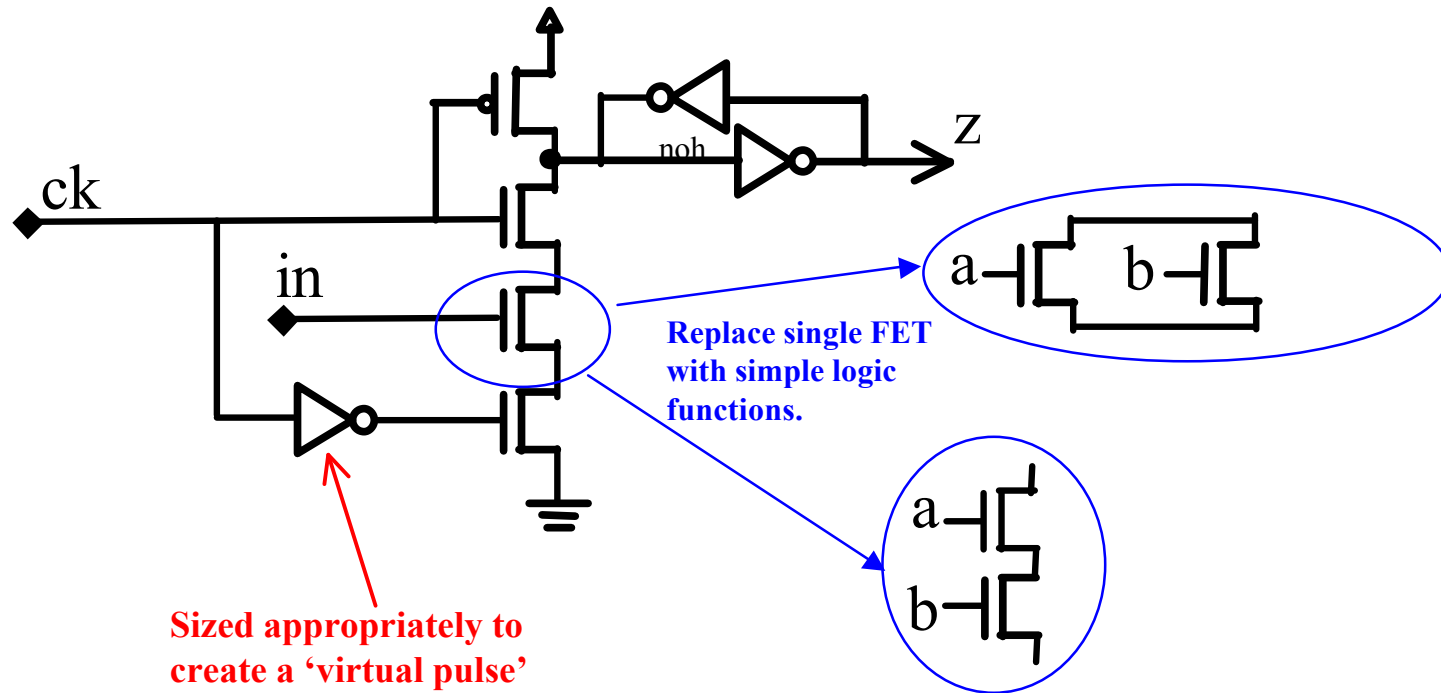
## ❑ NCK Flip Flop

- Domino input will be stable on rising CK.
- Previous static stage limited to a single phase.
- Waste phase of logic.

## ❑ NCK Transparent Latch

- Provides stable input to CK dynamic
- Cost: area and insertion delay.

# Enter the Entry Latch



- ❑ Small and simple.
- ❑ Ability to do simple logic, helps negate any insertion delay.

# Domino Pit Falls?

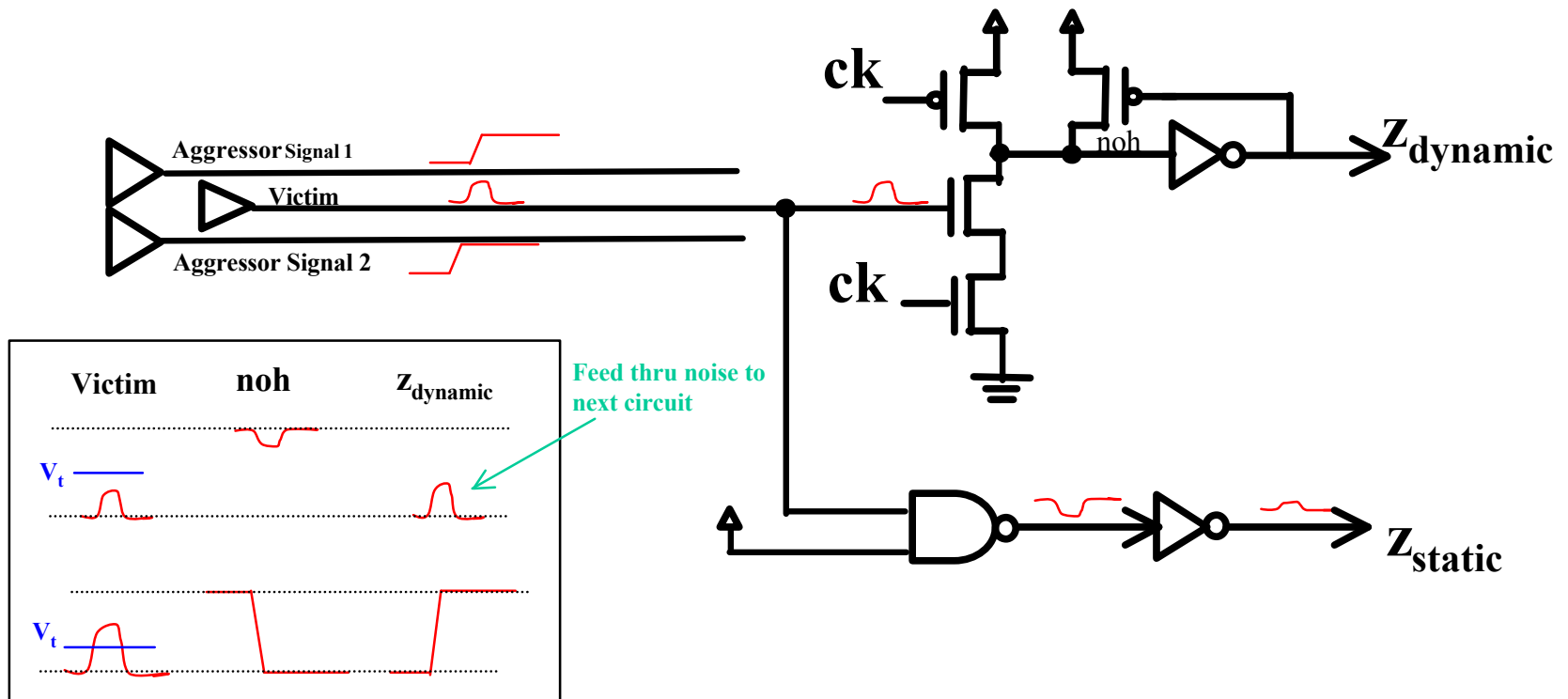
- ❑ Domino is sensitive to many noise sources including, we look at a few real world examples:

- |                              |                      |
|------------------------------|----------------------|
| – Leakage                    | keepers              |
| – Charge sharing             | secondary precharge  |
| – Capacitive coupling        | spacing & shielding  |
| – Back gate coupling         | circuit design       |
| – Power supply noise         | good supply grid     |
| – Minority carrier injection | avoid injectors      |
| – Soft errors                | adequate capacitance |

# Noise Margin Sensitivities

- ❑  $V_t$  sensitivity.
  - That which makes it fast, also makes it more dangerous.
  - Static CMOS has a 'trip point' which higher than a single  $V_t$ .
- ❑ Flip once, and lose!
  - Once a dynamic circuit has switched, there is no recovery mechanisms.
  - In a static circuit, noise 'glitches' only cause failure when they are captured by latches.
    - Static noise event normally results in frequency degradation, not a dead chip.

# Capacitive Coupling

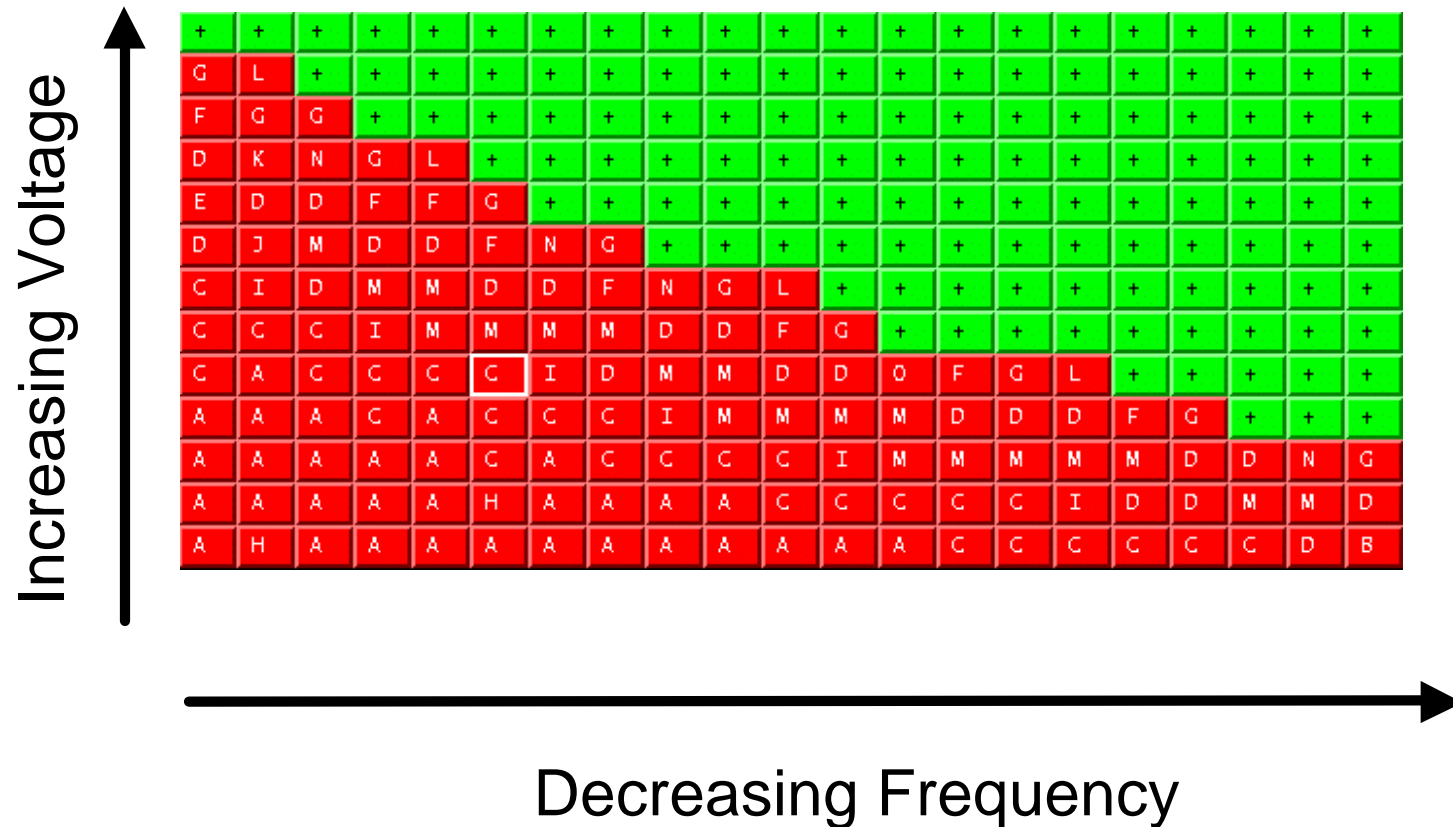


# Aside: The Shmoo

- ❑ Before looking at real world failures, we need to be familiar with this very important tool.
- ❑ A graphical representation of the performance characteristics of an IC
- ❑ The *shmoo* is named after creatures in the Lil' Abner cartoon strip
- ❑ Believe it or not, shmoo is now a registered trademark!



# Standard Speed path Shmoo





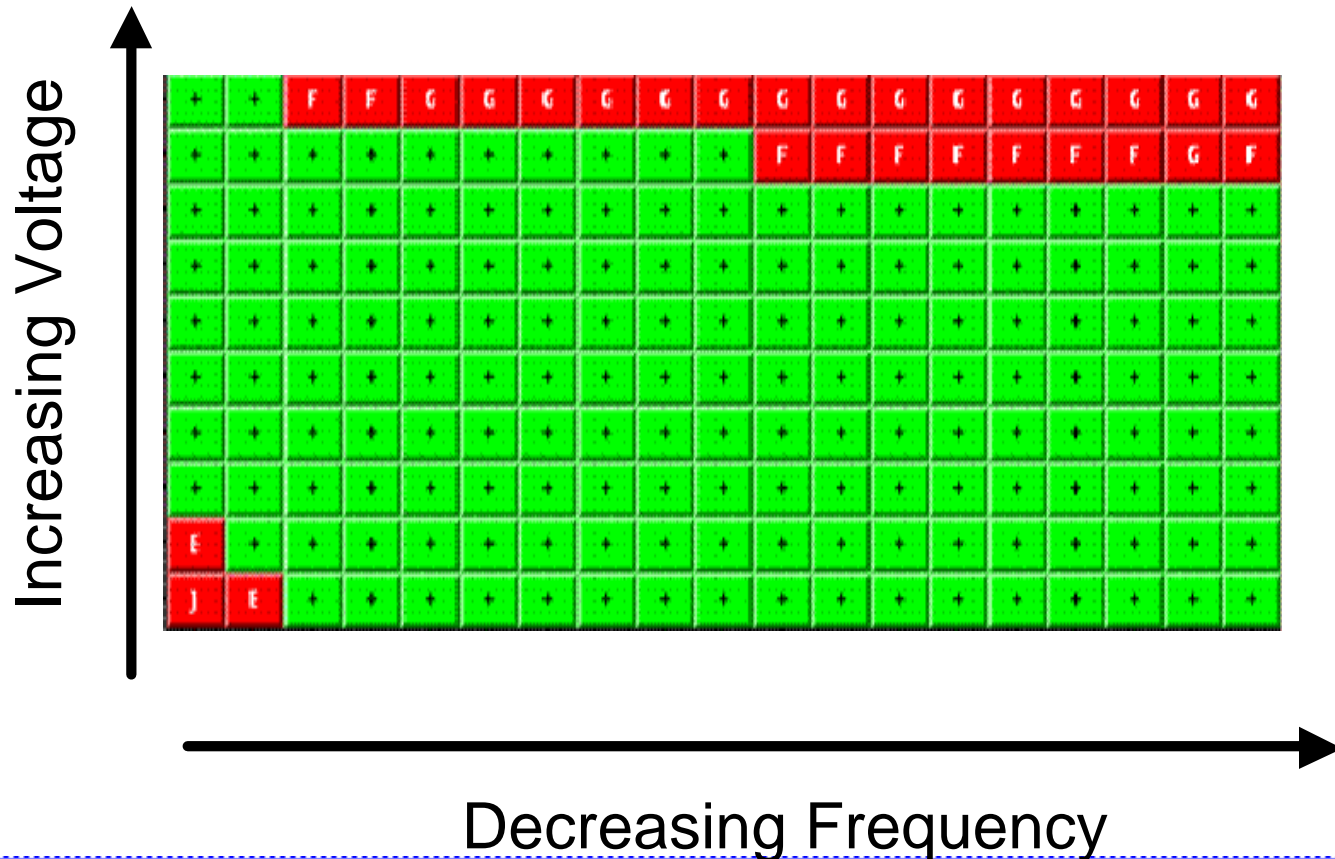
# Shmoo Usage

- ❑ Normally varies frequency and/or voltage
  - See how chip responds at different operating points.
  - Green is good =>passing; Red is failing
  - The shape of the shmoo should be the first clue to the nature of the silicon failure.
- ❑ Can also vary other operating parameters, examples:
  - Frequency vs. Temp;
  - I/O Voltage versus Core voltage

# Register File Failure

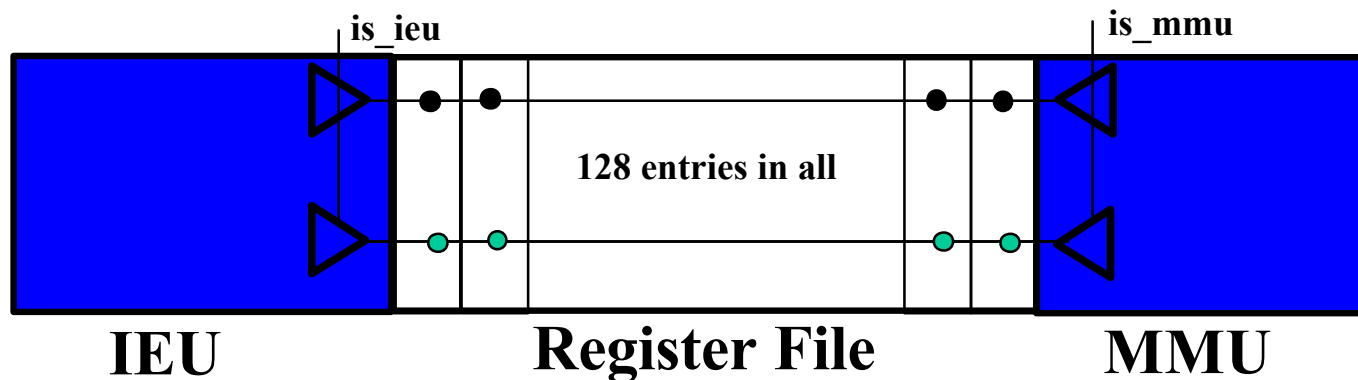
- ❑ Failure in general purpose register file.
  - Seen only at High Voltage. (Shmoo)
  - 1<sup>st</sup> seen in system test. Appears coupling related.
  - Test case transferred to stored response tester:
    - Scan collected.
    - Clearly indicated coupling issue.
    - Bits only failing in single direction
    - Reads are good. Writes are failing!

# RF Write Failure Shmoo

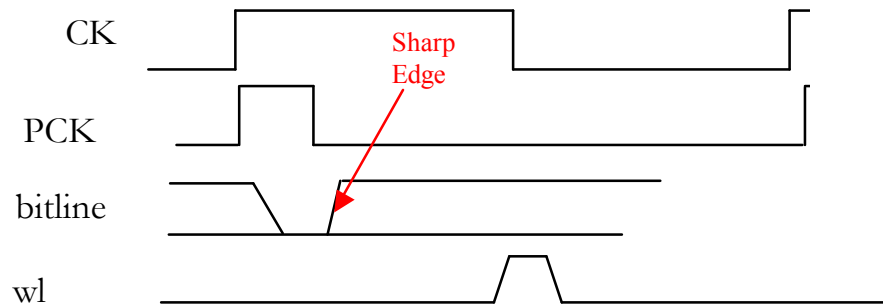
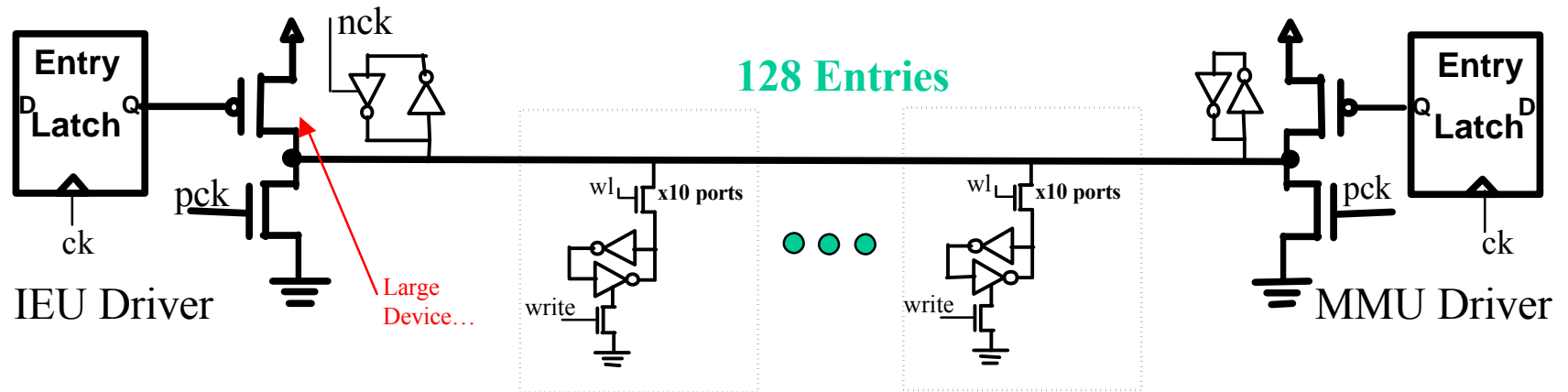


# Register File Design

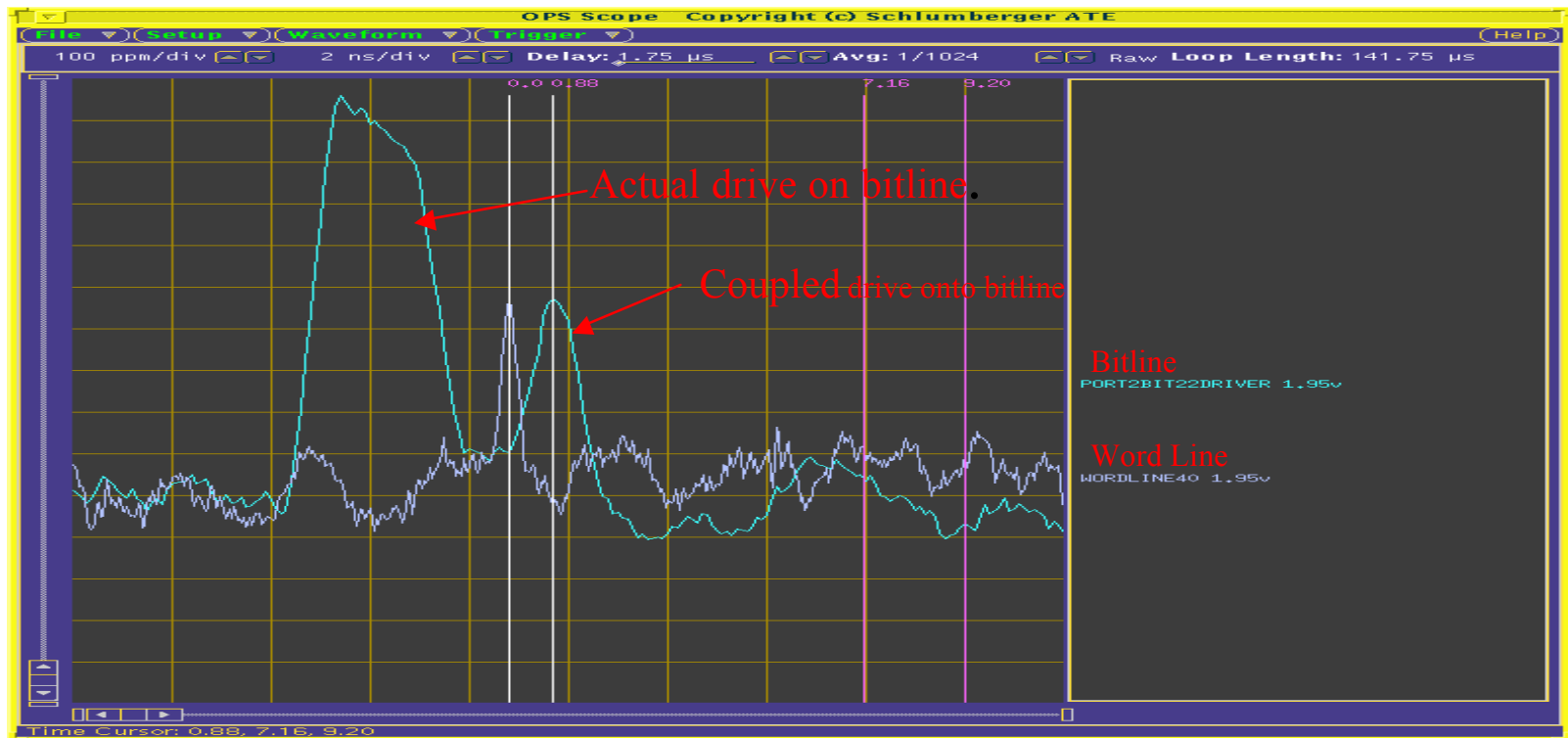
- ❑ Register File design challenges
  - 128 entry x 65 bits
  - 12 read ports, 8 write ports
  - Write ports may be written by either the IEU or MMU.
  - Must be capable of performing write back stage bypassing.
  - Needs to be area efficient.



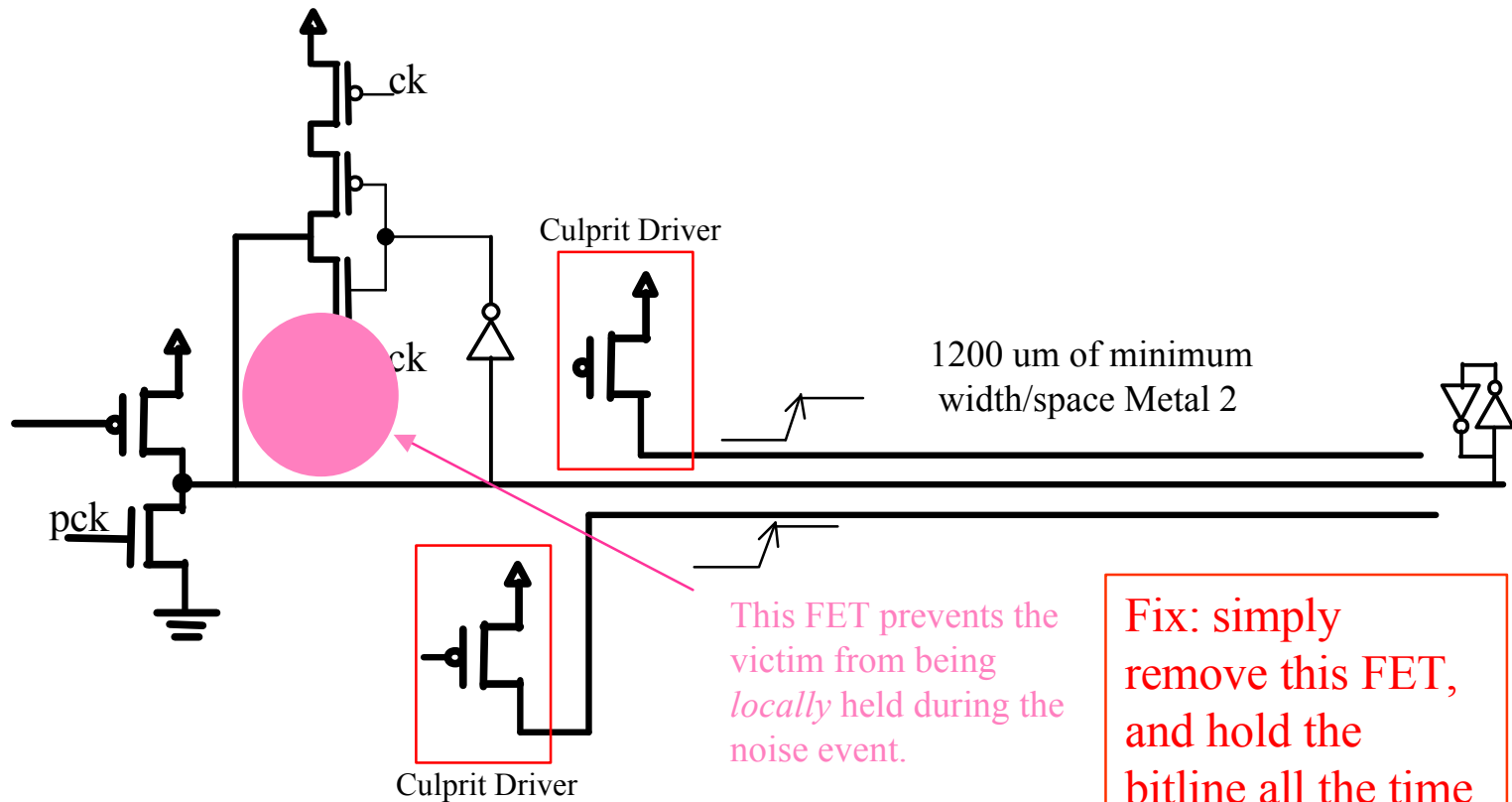
# Register Bit Line Writes



# Backside Probe Waveforms



# A Closer Look.



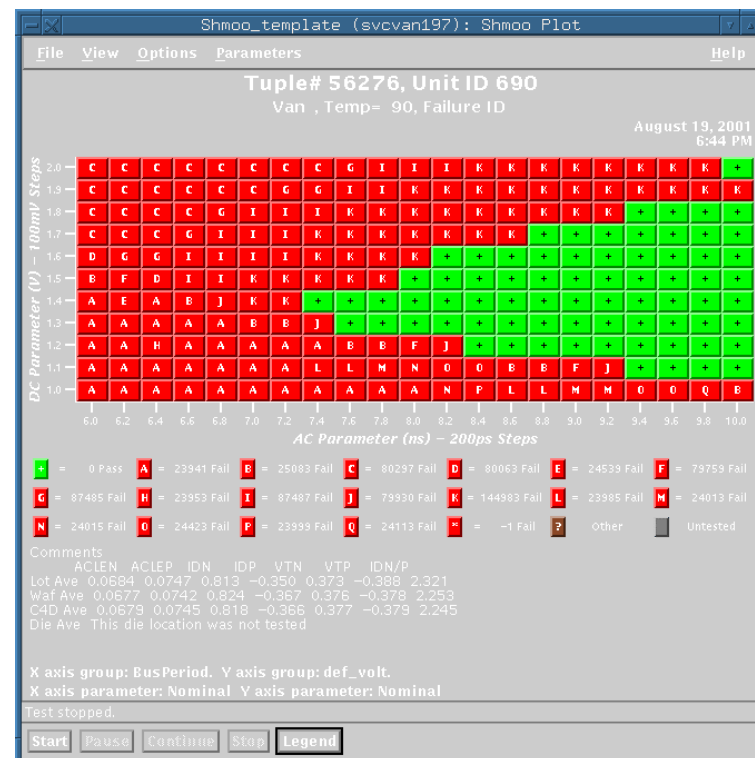
# Lessons Learned

- ❑ Circuit Design
  - Open Drain Buses are subject to failure.
    - Usually feed skewed receivers.
    - No substantial drivers on victim line during noise events (weakly held)
  - When designing risky circuits, design team must stay current with all process file changes.
- ❑ Engineering Tradeoffs
  - Odd are the circuit you're working on will NOT limit the speed of the chip
    - Robustness is **much** more important than speed of any particular circuit!
    - How much is that "little tweak" buying you in frequency. Is it worth it?
  - Always ask the question: "Can I make this circuit more robust?"



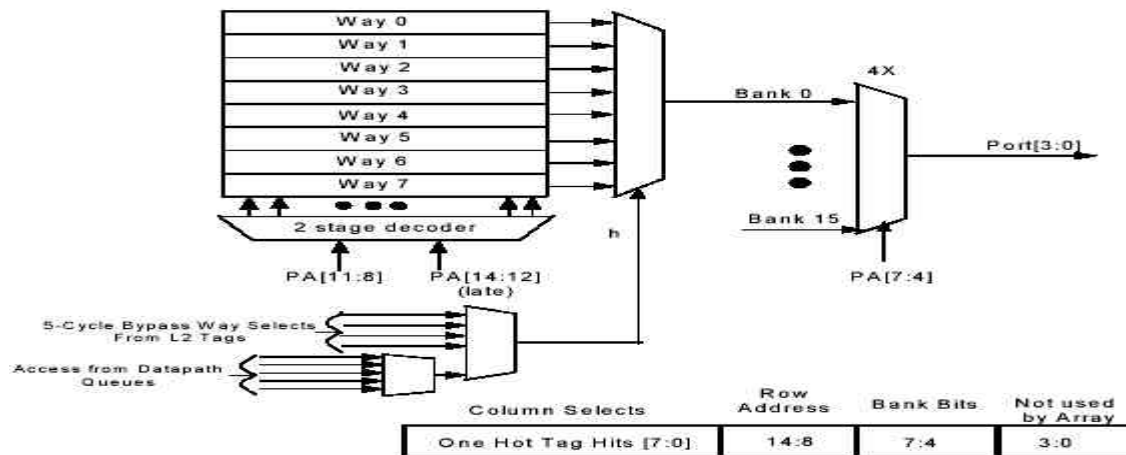
# 2<sup>nd</sup> Level Cache Coupling Failure

- ❑ Failure seen on almost all patterns.
- ❑ Shmoo is characterized as a “half flying saucer”.
- ❑ Not seen on first silicon, only seen on new stepping which was ‘tweaked’ for speed.
- ❑ Shmoo shape indicates:
  - High voltage failure. Again noise issue is suspected.
  - Reverse speed path nature indicates a frequency dependency for the coupling event.
- ❑ Straight forward debug process bring debug team to the read out of the 2<sup>nd</sup> level Cache.

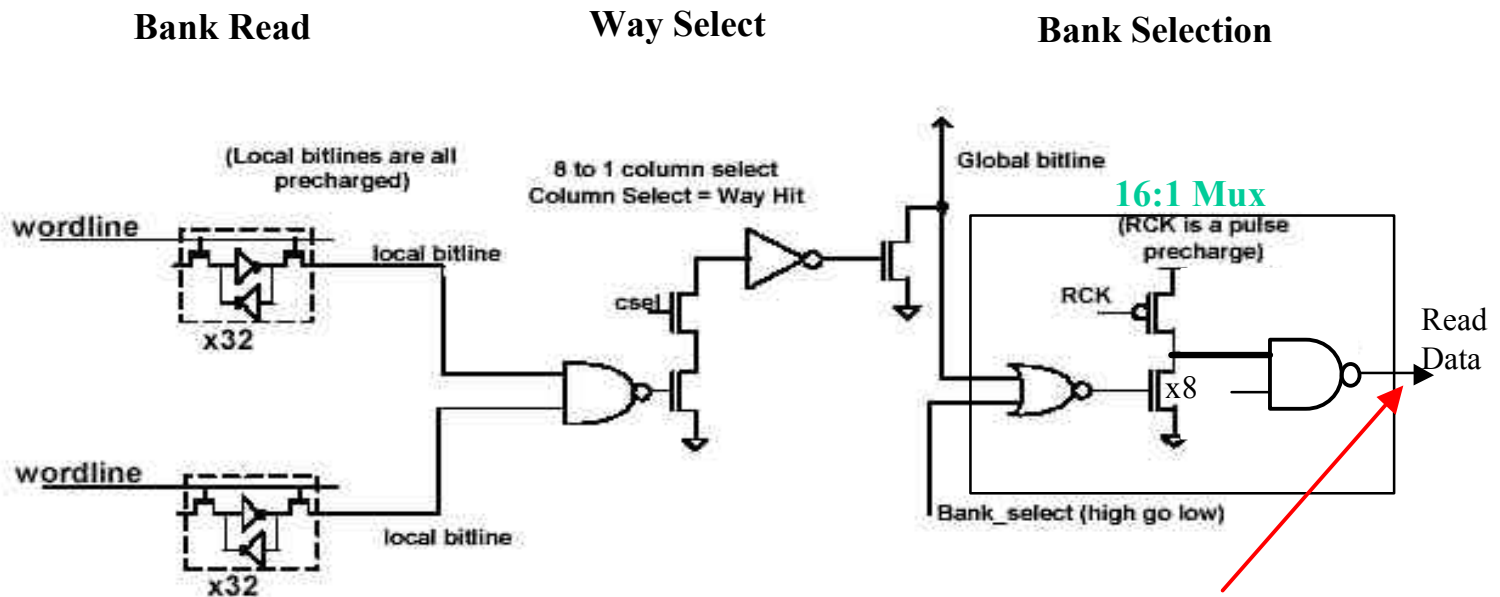


# 2<sup>nd</sup> Level Cache design.

- ❑ 256K Byte + ECC
- ❑ 8 Way Cache
- ❑ Pseudo 4 ported.
  - If each port is accessing a separate bank than all 4 ports
  - Control logic prevents bank conflicts
  - 16 banks

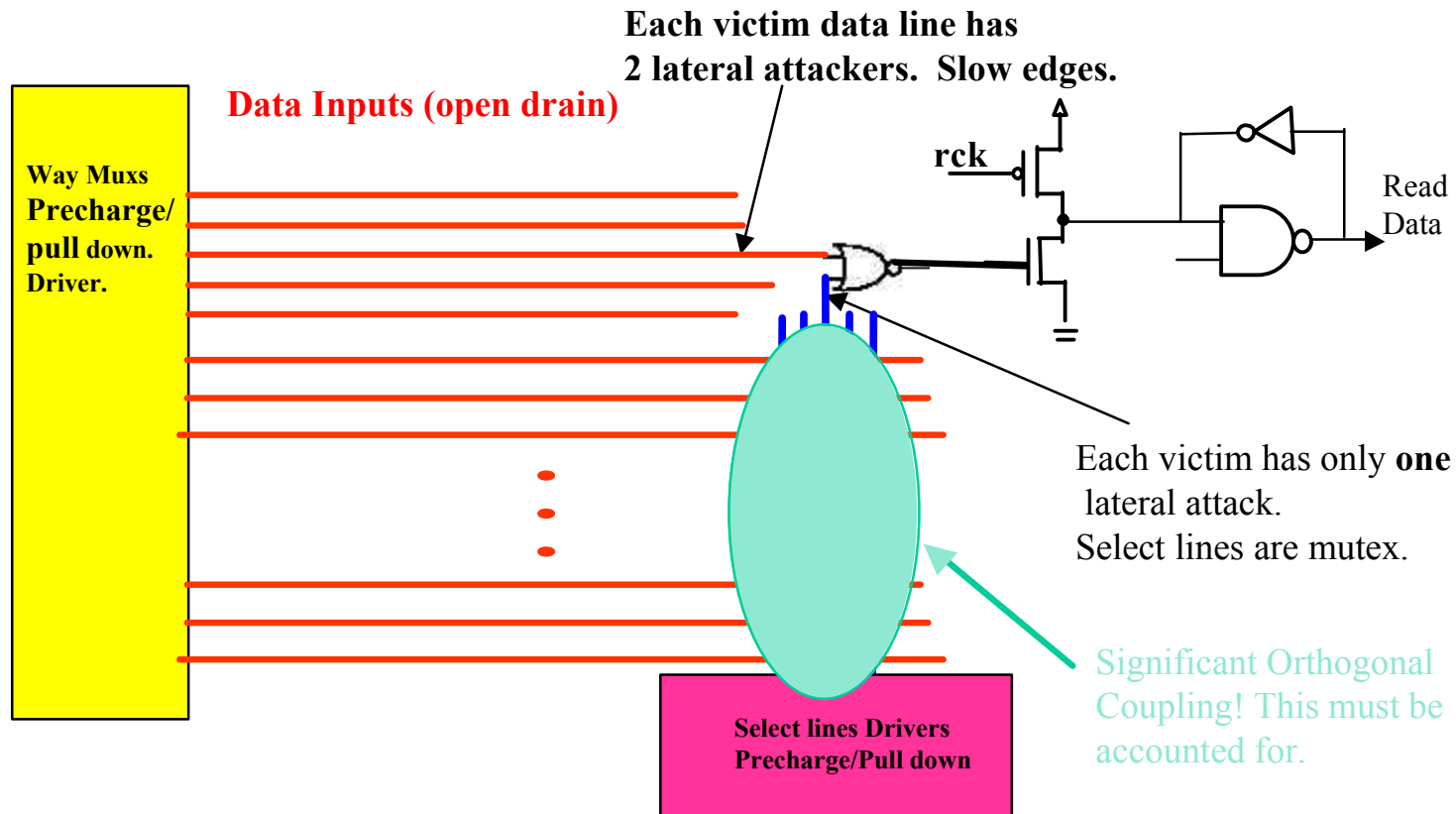


# Cache Read Circuitry

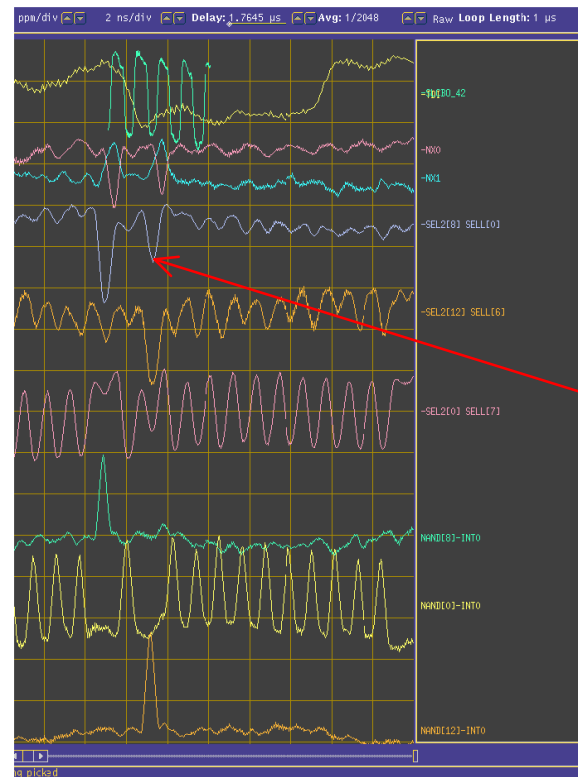


Downstream Scan Data clearly indicates that this signal is corrupted. 0s are getting transformed into 1s.

# A Closer Look.



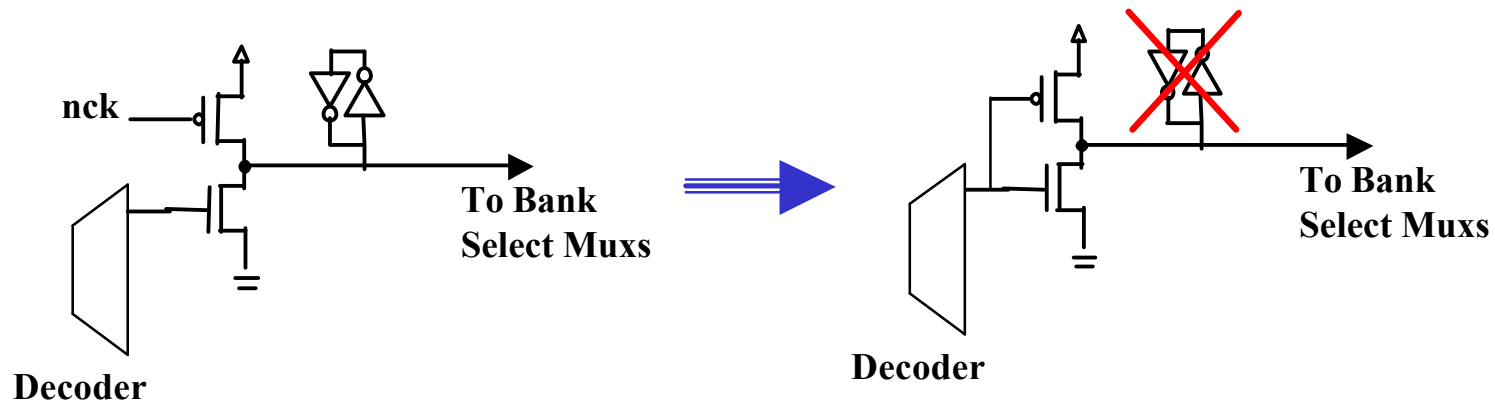
# LVP Waveforms



**Coupling  
Event**

# Fix and Lessons

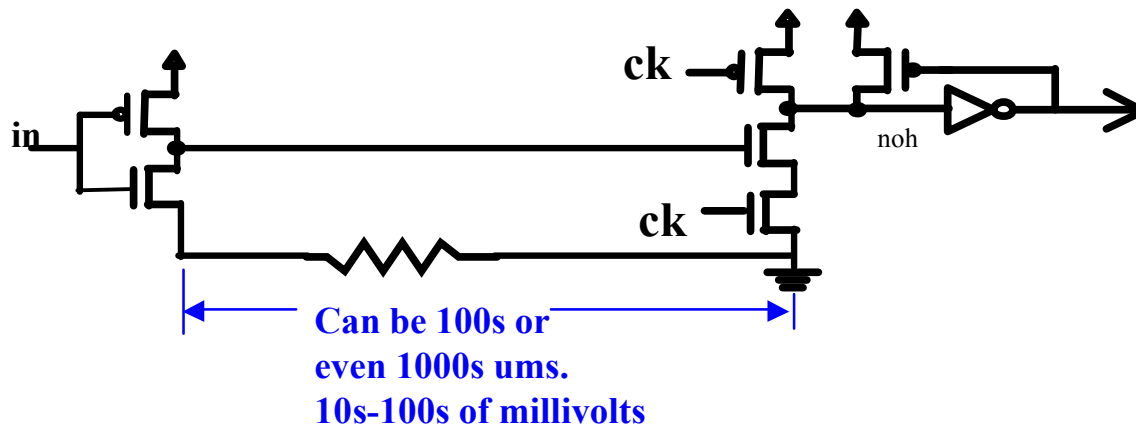
- ❑ Select line drivers changed to static drivers.



- ❑ Solution gives up a little speed for a robust design.
- ❑ Orthogonal metal coupling not properly accounted for.
  - Three dimensional geometries need to be considered
  - ‘Reasonably pessimistic’ initial assumptions for all noise analysis.

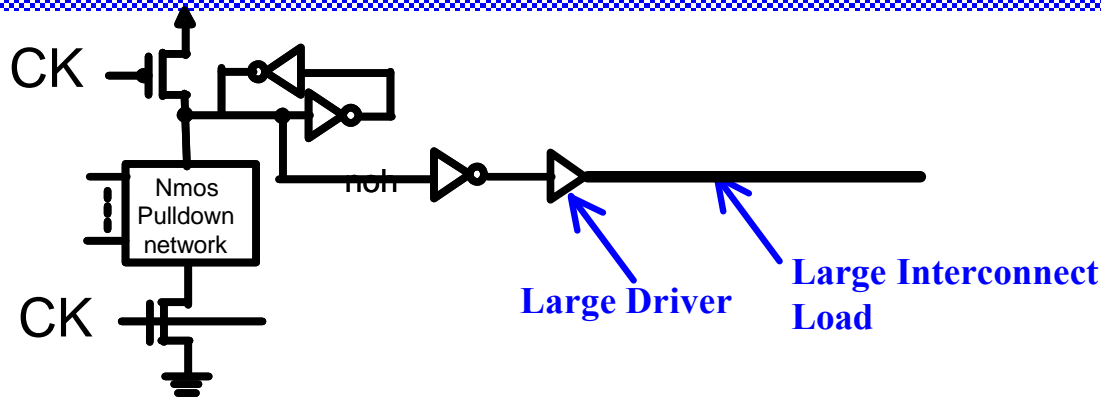
# Power Grid and Coupling

- ❑ Supply/ground on a chip is not equipotential.
- ❑ Any circuit is only as good as its underlying power grid!
  - Especially true for domino circuitry.



- ❑ Power grid most often **cannot** be verified by inspection. 'Simulations' are required.

# Domino and Large Drivers.



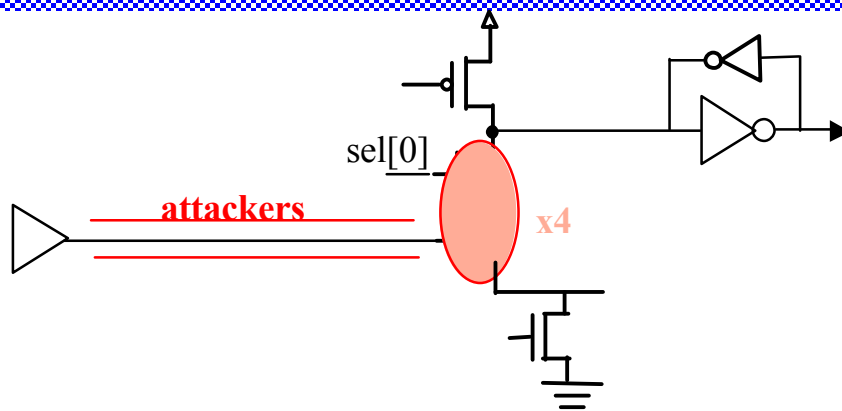
- ❑ Large drivers will dump transient current into the power grid.
  - Affects the apparent trip point of local domino circuitry.
  - Making circuit more sensitive to noise events.
- ❑ Avoid this practice, if forced:
  - Smear out evaluation: have adjacent domino drivers evaluate on skewed clocks.
  - Bypass Cap.



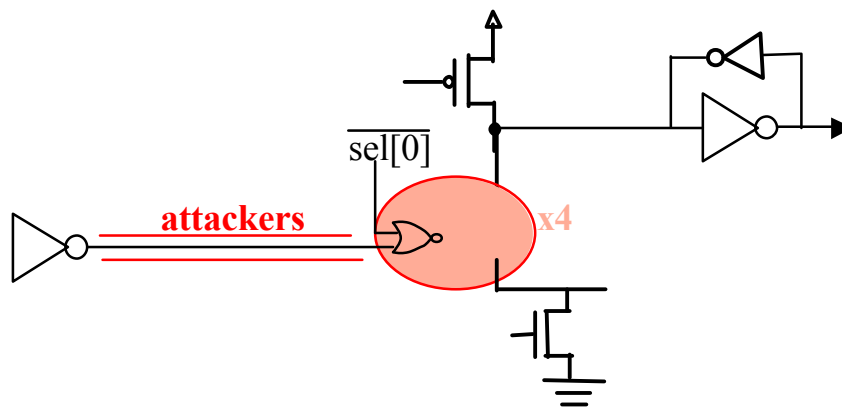
# Fighting Coupling

- ☐ Use caution with highly ratioed gates
  - Performance gain vs. noise margin reduction flattens out at somewhere between 4:1 and 6:1 for high skew devices.
- ☐ Avoid receiving long routes directly into domino gates.
- ☐ Will a static design work?
  - In most case you will save power.
- ☐ Pseudo NMOS?
  - Better noise immunity
- ☐ Anti-Miller Devices.
- ☐ 'Statizing' a domino node.
- ☐ Orthogonal Metal Fill.
- ☐ Robust Power Grid.

# Long Interconnect

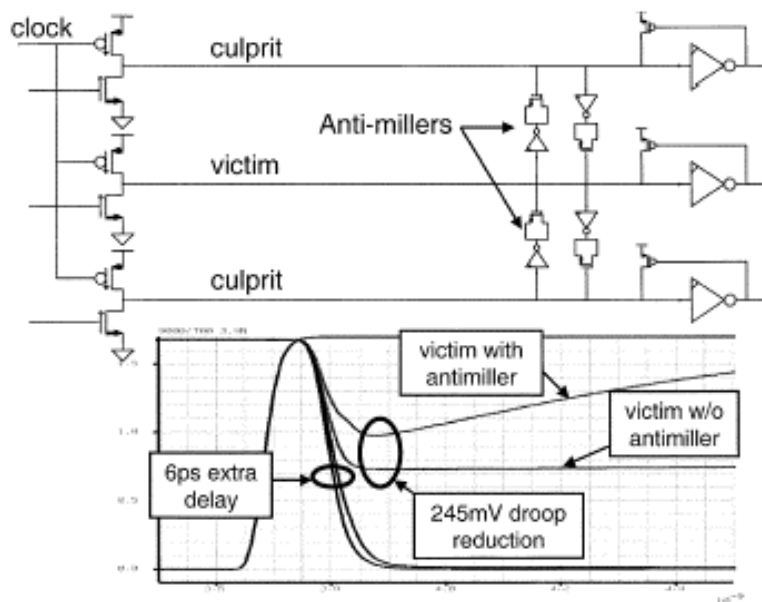


**4:1 Mux**



**4:1 Mux with one  
static gate of noise  
rejection**

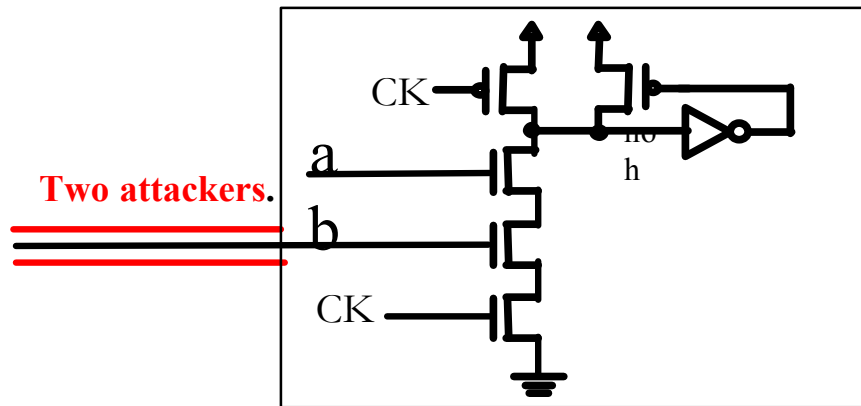
# Anti-Miller Devices



**“A Full Bypassed Six-Issue Data path and Register File on the Itanium-2 Microprocessor”;**  
E.S.Fetzer, et al.

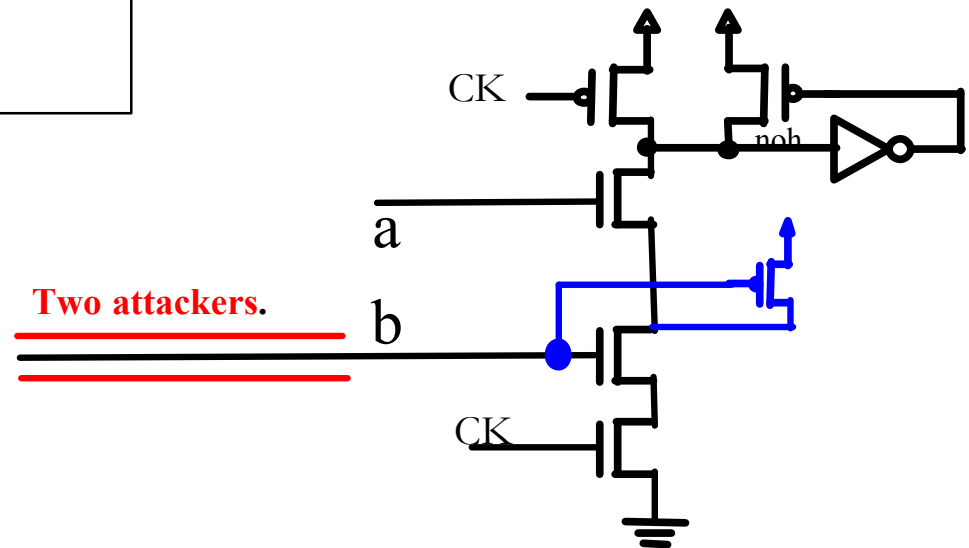
- ❑ Simple and small.
  - Inverter in series with a capacitor.
  - Capacitor is formed using an NMOS FET with shorted source-drain.
- ❑ Any coupling event is offset by charge of opposite sense being dumped on victim line.
- ❑ Minimal frequency impact.
- ❑ Enabled fully packed metal routing on Itanium 2.

# “Statizing” a Domino Circuit

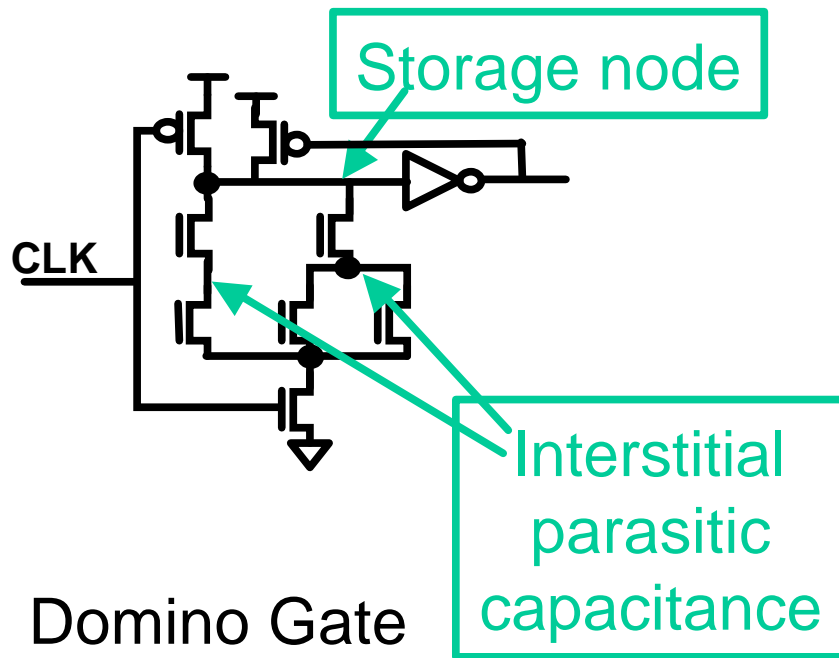


- B Input Failing for noise check.

- Adding small PFET adds significant noise immunity while minimally reducing circuit speed.

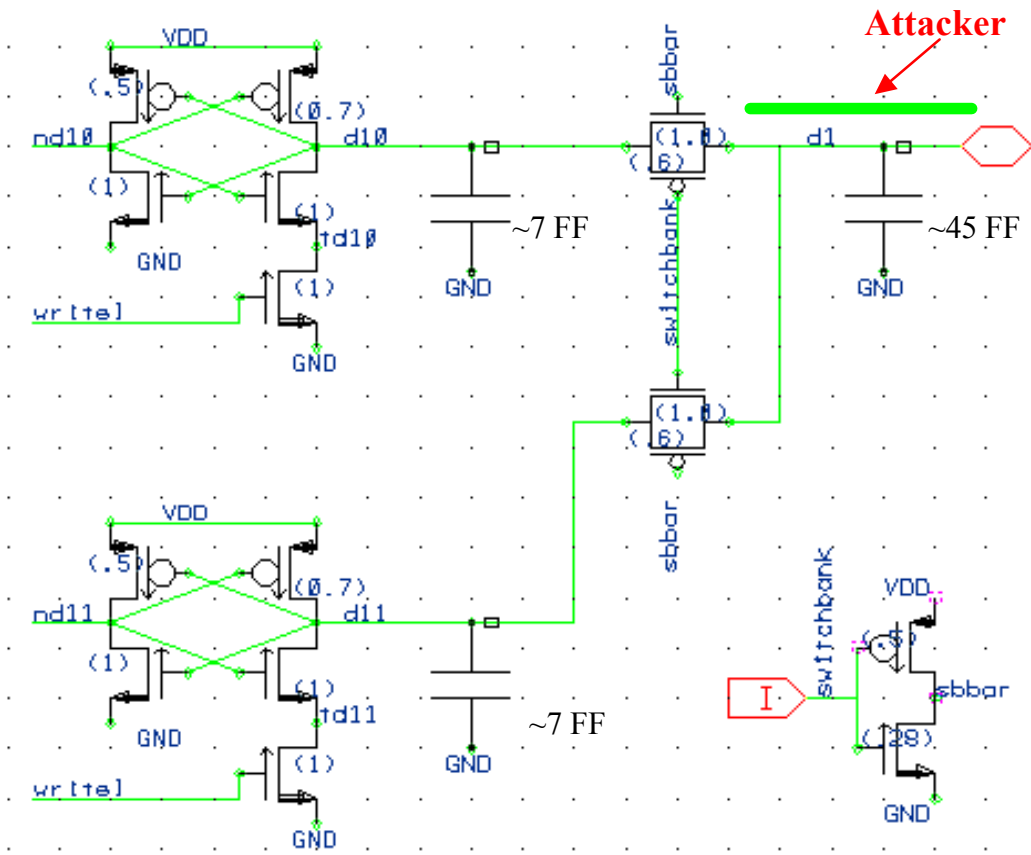


# Typical Charge Share Issue.



- ❑ High capacity node dumps charge from lower capacity node forcing domino gate to flip.
- ❑ Easy to avoid
  - Interstitial precharges device.
  - Place one hot signals on top FET of stacks.

# Charge Share + Coupling



- ❑ Circuit switches between two banked registers.
- ❑ If switchback signal is slow enough, memory cell easily overcomes charge share.
- ❑ At the fastest corner of newest process, the circuit fails.
- ❑ Coupling onto d1 node combines to make problem much worse.

# Leakage Failures

- ❑ Leakage is becoming a more significant concern as we move to tighter geometries.
  - Leakage grows 2-4x per generation
  - Biggest effect is on power dissipation.
    - 180nm: < 5% of total power.
    - 130nm: 10-30%
    - 90nm: 50% ?
  - Circuits need to work in the face of this reality.
- ❑ Keeper sizing
  - 180nm keeper sizing: 1%
  - 90nm keeper sizing: 6%
- ❑ Burn-in exacerbates the situation.
  - Temperature saturated at high end of spec.
  - 1.2x to 1.3x use voltage : DIBL effects

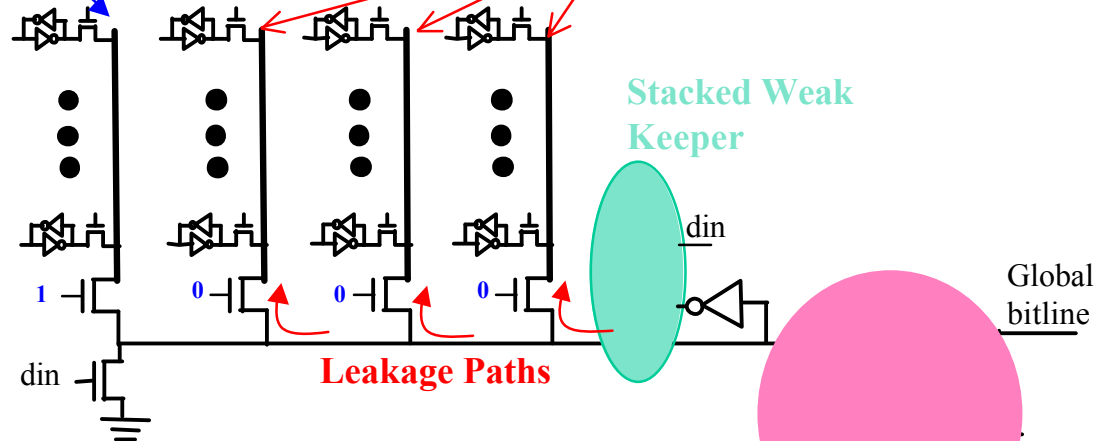
\_\_\_\_\_



# Cache Dump Circuit

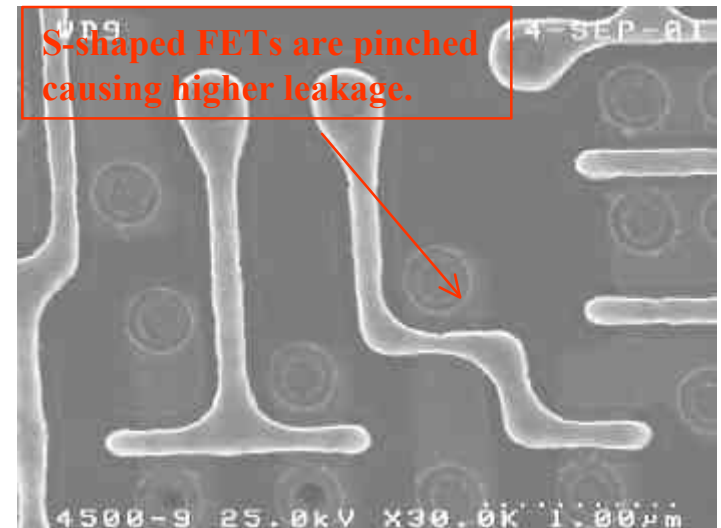
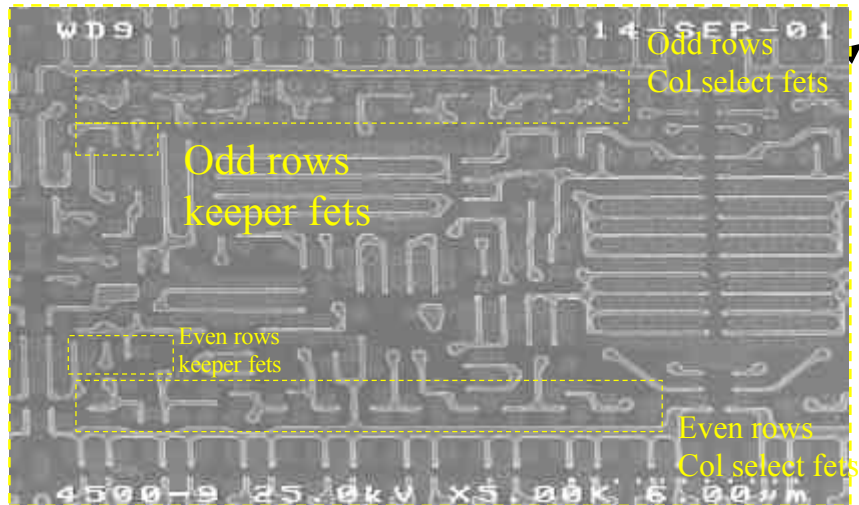
This bitline is read = 1

One or more of these maybe 0.



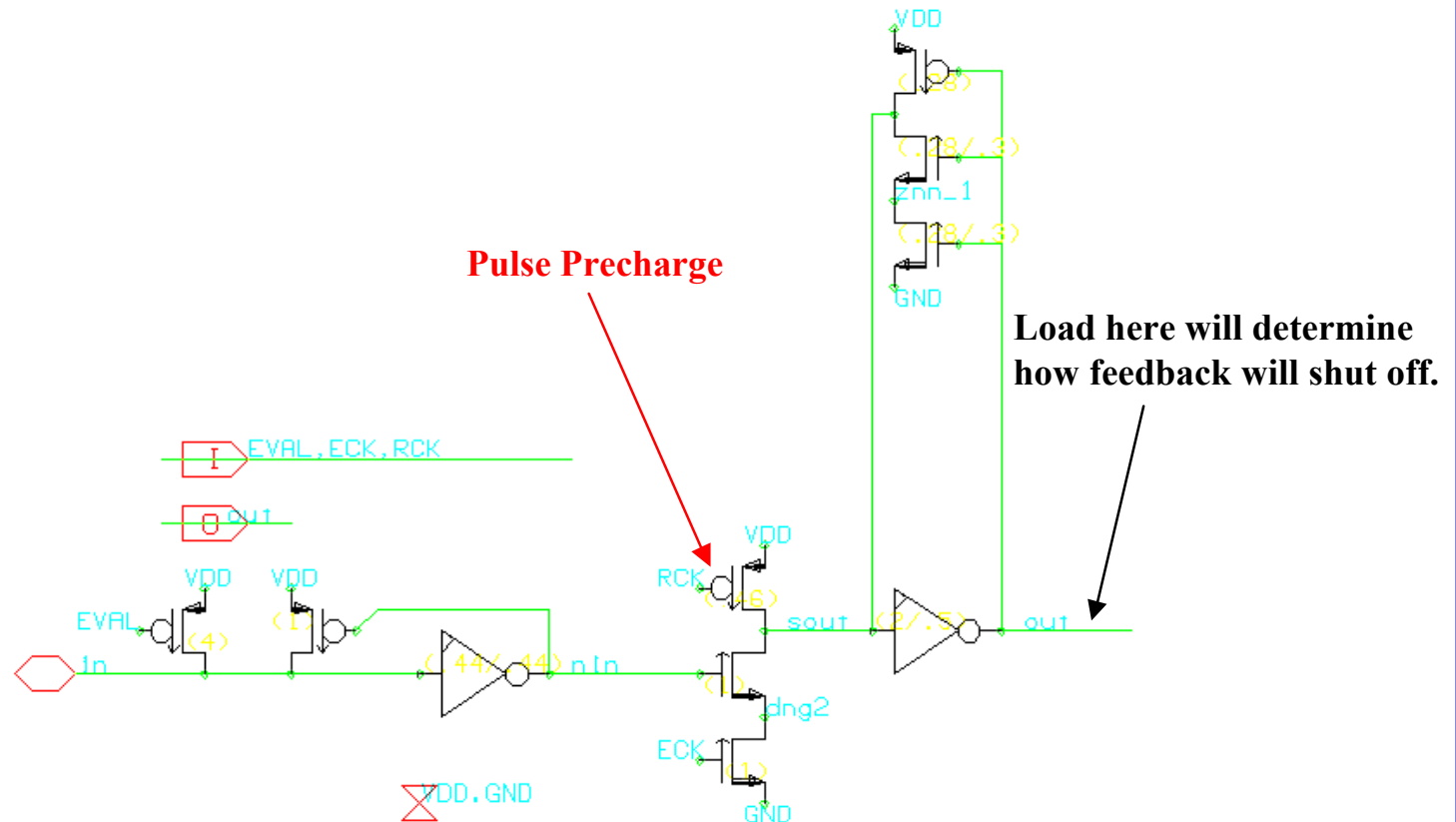
- ❑ Combines 4-16 local bit lines into a global bit line.
- ❑ This is a highly utilized circuit.
  - 10s of thousands of usages.
  - Similar versions found in nearly all caches.

# Layout Geometries

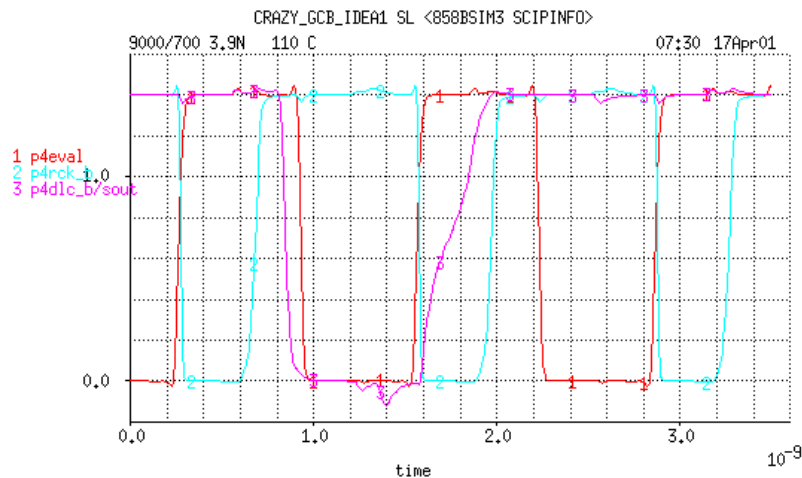


- ❑ Highly reproduced circuits need to have unquestioned robustness.
  - Cache Circuits.
  - Statistical Analysis -> Monty Carlo Simulation.
- ❑ Unexpected processing issues **will** cause a marginal design to fail.
- ❑ Risk Reward Assessment. (e.g. predischage logic)

# Pulse Precharge



# Pulse Precharge



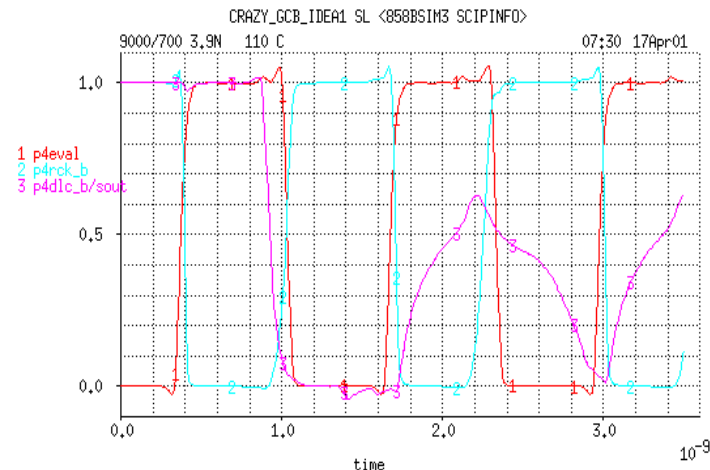
## ☐ No Load Case

- **sout** node easily make it to full rail with RCK pulse.

## ☐ Worst Case Load

- **sout** is unable to fully precharge within pulse.

## ☐ Experience designer includes a switch to get wide pulse!



# Some Final Words.

- ❑ Domino is here to stay.
  - High performance designs demand the performance.
  - Density benefits.
    - Register Files.
    - Large CAMs
    - Muxes.
- ❑ Methodology
  - Robustness is Job #1.
  - Standardization .
    - At most, solve a problem once per product.
    - Minimize silicon debug issues.
- ❑ When to use domino?
  - Consider static or pseudo-NMOS.
  - Make the proper tradeoffs.

# A few more words.

- ❑ Multiple factors often combine to cause a silicon failure.
  - Power grid, charge sharing, noise events, layout geometries, leakage, etc.
  - Develop tools and an **understanding** that address the interplay of these factors.
- ❑ Problems need to found in pre-silicon.
  - Post silicon failure **are** very expensive.
  - Make the proper choices.
  - Each risk should be balanced by sufficient benefit.
  - Simulate, re-Simulate, and then Question.
- ❑ Future process implications
  - Increased Leakage.
  - Coupling (faster edge, tighter geometries, hopefully low-K dielectric)
  - Increased Process Variability
  - Design for the future.
- ❑ Good Luck!

---

# Backup

# Itanium 2 Die Photo

