EE371
Debug Examples

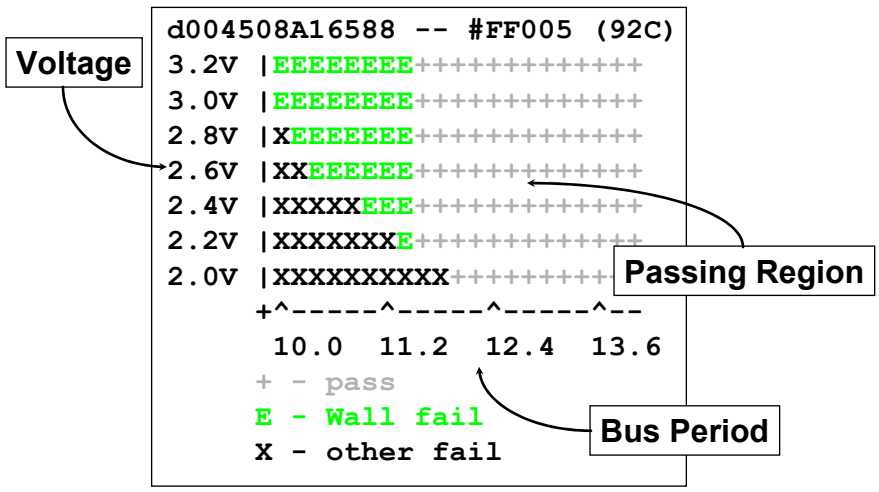
Intel Corporation
jstinson@mipos2.intel.com

Agenda

- Speedpath Failure
- Circuit Marginality: Noise
- Functional Failure
- Circuit Marginality: Multiple
- PowerUp Problems

Speedpath Failure

Speedpath Example: The Wall Shmoo



Skew Insensitive wall

```
d004508A16588 -- #FF005 (92C)
3.2V |EEEEEEEE++++++++++++
3.0V |EEEEEEEE++++++++++++
2.8V |XEEEEEEE++++++++++++
2.6V |XXEEEEEE++++++++++++
2.4V |XXXXXEEE++++++++++++
2.2V |XXXXXXXE++++++++++++
2.0V |XXXXXXXXXX++++++++++
     +^-----^-----^-----^--
      10.0  11.2  12.4  13.6
     + - pass
     E - Wall fail
     X - other fail
```

Fast Transistor Part

```
d004508A16588 -- #SS131 (92C)
3.2V |XXXXXEEE++++++++++++
3.0V |XXXXXXEE++++++++++++
2.8V |XXXXXXXE++++++++++++
2.6V |XXXXXXXXX++++++++++++
2.4V |XXXXXXXXXXX++++++++++
2.2V |XXXXXXXXXXXXXXX++++++
2.0V |XXXXXXXXXXXXXXXXXXX++
     +^-----^-----^-----^--
      10.0  11.2  12.4  13.6
     + - pass
     E - Wall fail
     X - other fail
```

Slow Transistor Part

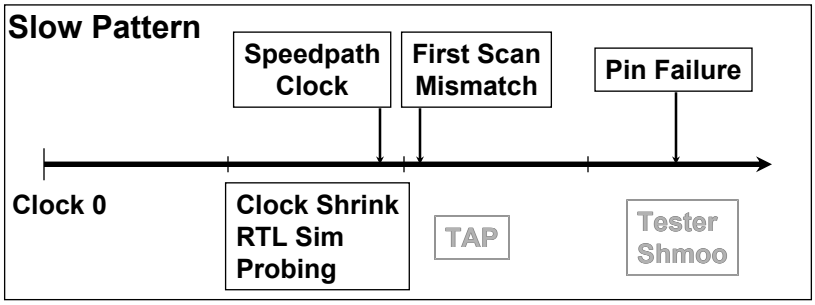
The Wall Debug

- Production test platform suspected
  - A timing setup problem
  - How could silicon act this way?

However...

- Debug test platform confirmed
  - Unlikely two diff't platforms had same timing error
  - Now we had to do the debug...

Pattern Timeline

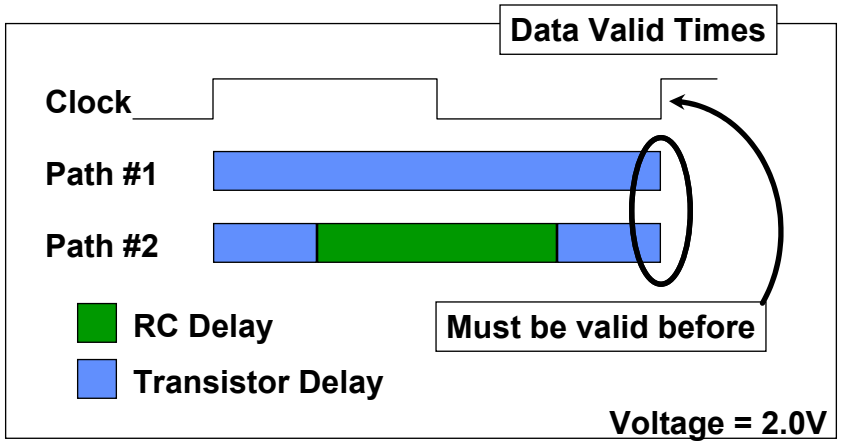


Debug Process

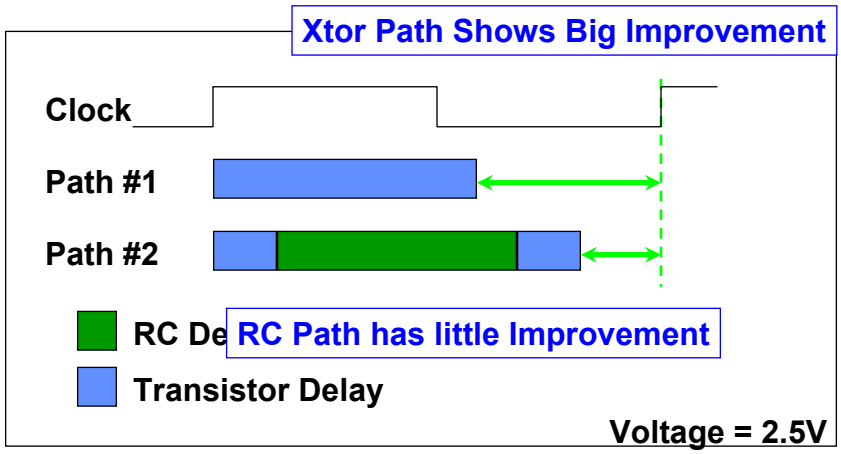
Why was it a wall?

- Long Interconnect Line
  - RC Delay less sensitive to driver strength
  - Voltage/process only improve driver

Interconnect Effect

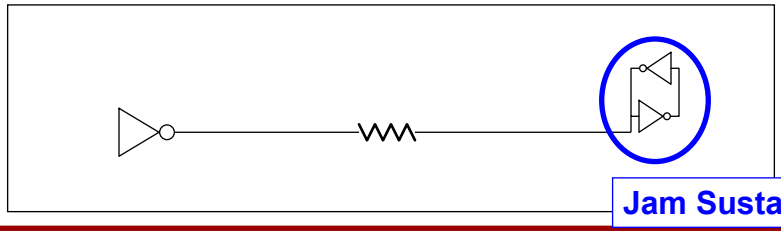


Interconnect Effect



Why was it a wall?

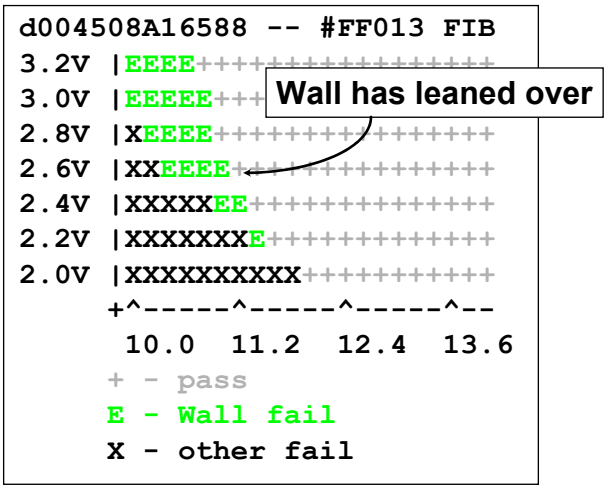
- Jam sustainer at end of the line
  - Fights transition of signal
  - Sustainer gets stronger with voltage/skew
  - Adds to “wall” characteristics



Wall Follow-up

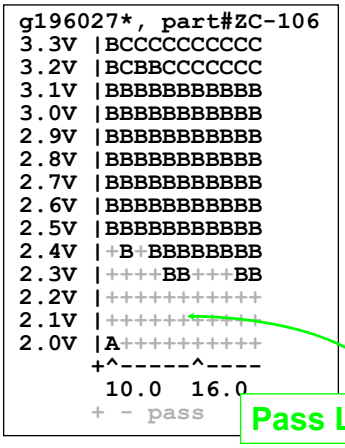
- Two FIB experiments
  - Driver speedup – wall moved
  - Cut sustainer – wall “leaned”

Shmoo with Cut Sustainer



Circuit Marginality: Noise

Noise Example



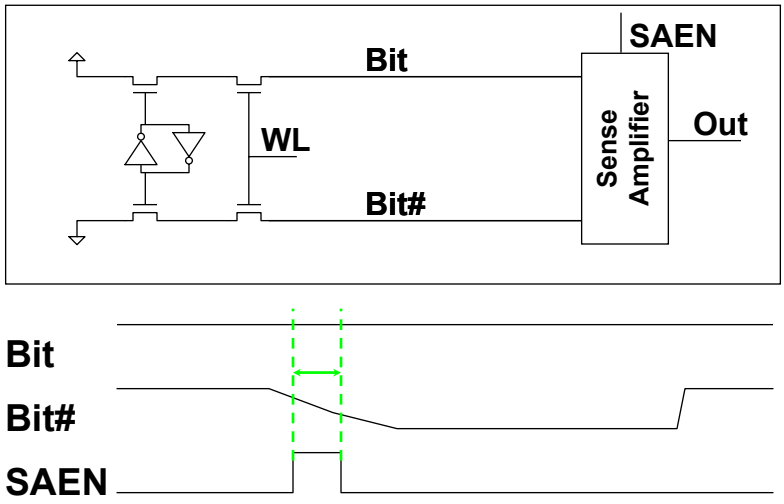
- High Voltage Failure
  - Only one FAB showed signature
    - Second FAB seemed clean
  - Scan pointed to branch memory array

Pass Low Voltage Only

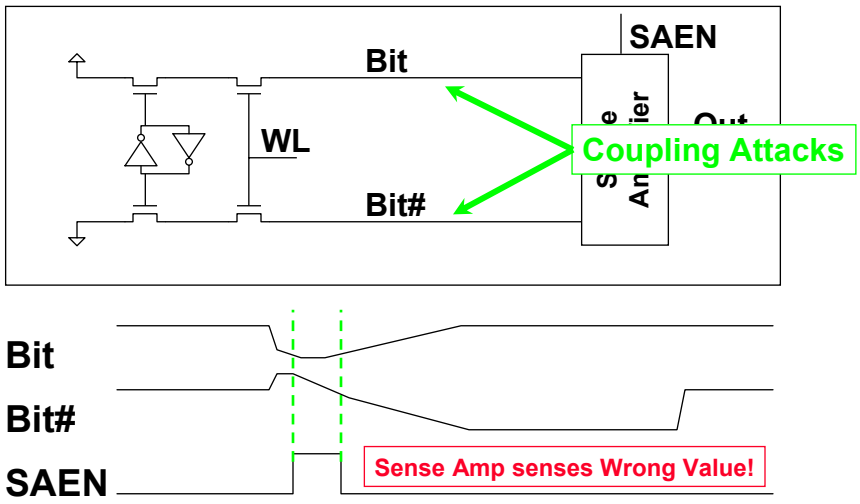
Noise Debug

- EBeam confirmed branch array read
  - Visibility limited in array
- Bit 4 resolved later than other bits
  - Based on EBeam waveforms
- Signals on either side of read lines transistioned in opposite direction
  - Suspected coupling problem

Coupling Schematics



Coupling Schematics



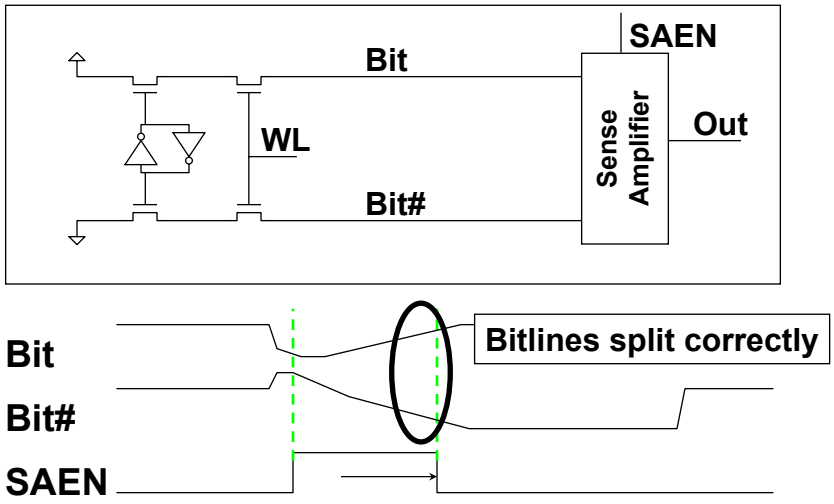
BTB Coupling Debug

- Parameters data checked at problematic FAB
  - M2 CD's wider than normal
  - ILD1 and ILD2 thicker than normal
  - More sensitive to coupling
- Audit of original design
  - Simulations ignored some coupling
  - New simulations showed failure

BTB Coupling Validation: FIB experiments

- Deposit extra capacitance on read line
  - Resists coupling from neighbors
- Extend sense amp pulse width
  - Gives more time for read to resolve

Coupling Schematics



Functionality Failure

Functionality Problem

- "Dash stepping" first silicon non-functional
  - Stepping was supposed to fix a min-delay race
- Suspected inadequate race fix
  - Scandiff confirmed same circuitry
  - EBeam also confirmed…
  - But visibility was limited

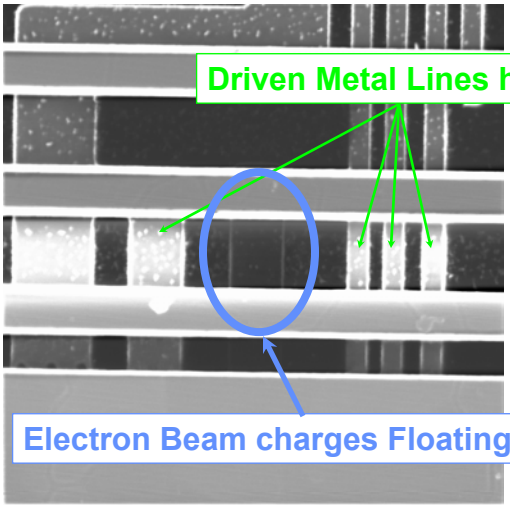
Functionality Debug

- Design team was confident in fix, so…
- Plan to strip back the entire block
  - Look for possible mask defect
  - Takes 4-10 days in FIB

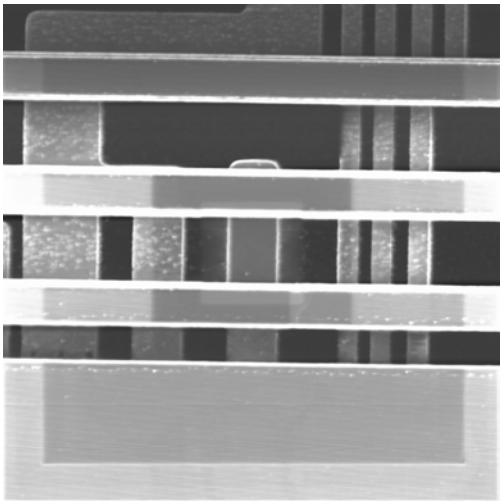
However...

- Noticed a floating node in EBeam scope

Floating Node



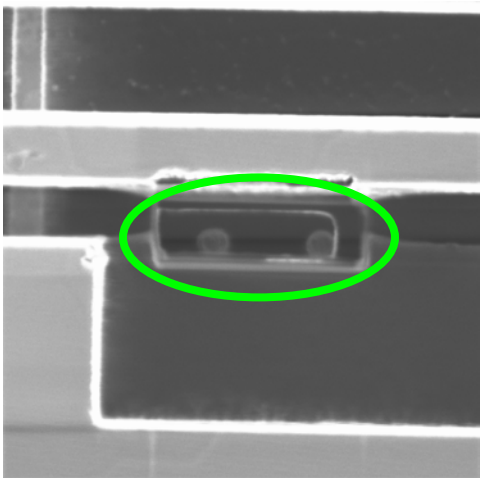
Floating Node



Floating Node Debug

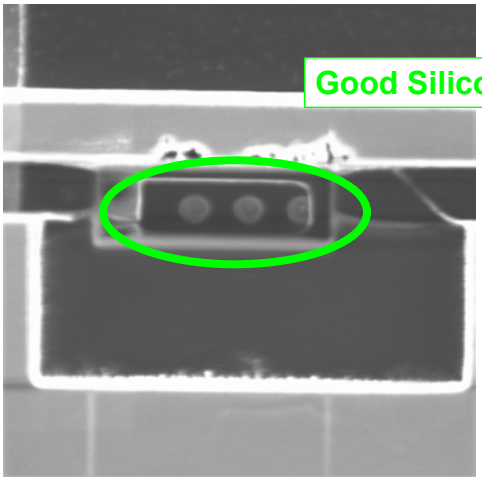
- Node should NOT have been floating
- A0 and A1 layout compared
  - Via1 or M1 could cause error
- FIB strip back focused on this node

FIB Stripback Results



- **Should be 3 via1's**
- **FAB contacted**
  - **Accidentally used A0 via1 mask**
- **Problem fixed**
  - **New silicon arrived shortly**

FIB Stripback Results



- **Fully functional with correct via1 mask**

Functionality Summary

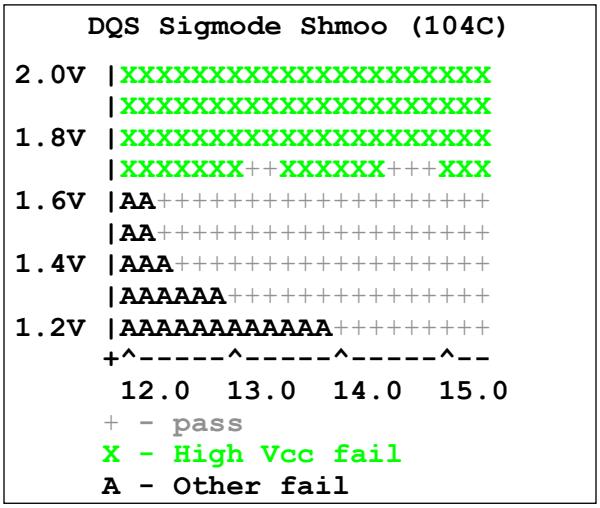
- Notice details
  - Focused stripback saved days of work
  - Very important during time critical debug

Circuit Marginality: Multiple Sources

Circuit Marginality

- Observed High Vcc failures
  - Frequency Insensitive
- TDO only failure
  - All signature mode tests were failing
  - Turning off signature mode allowed test to pass

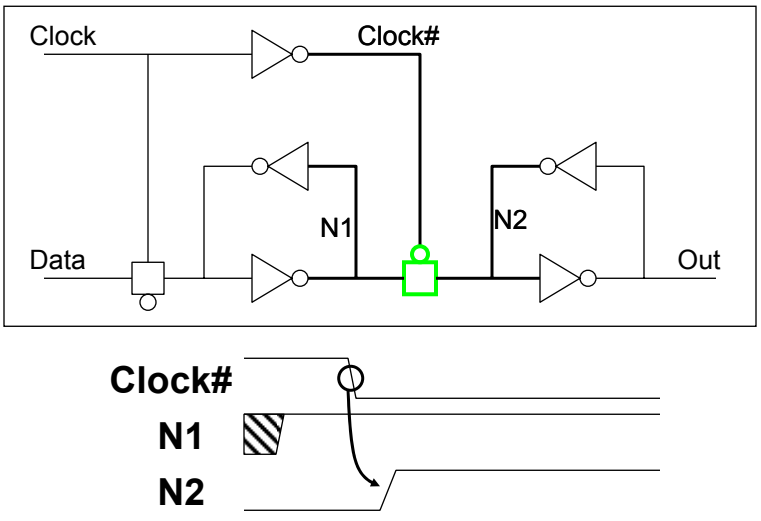
High Vcc Shmoo



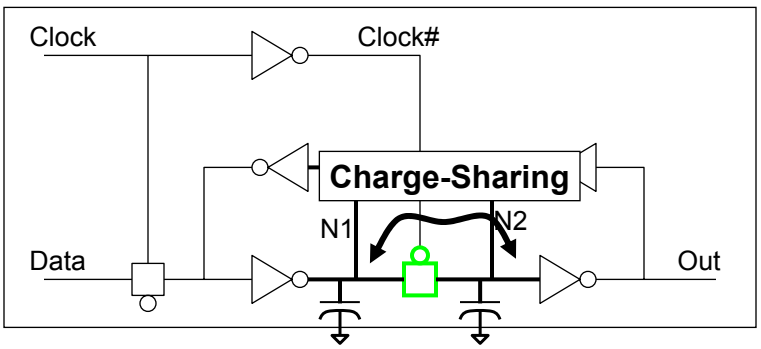
Marginality Root Cause

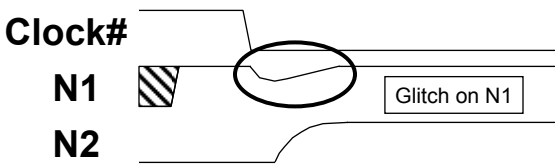
- Scanout stopped working in failure region
  - Deduce scan chain itself was broken
- Probing was only way to root cause
  - Laser Voltage Probe was able to narrow failure down to Scan MSFF
  - Three different mechanisms observed

Scan MSFF Analysis: Pass

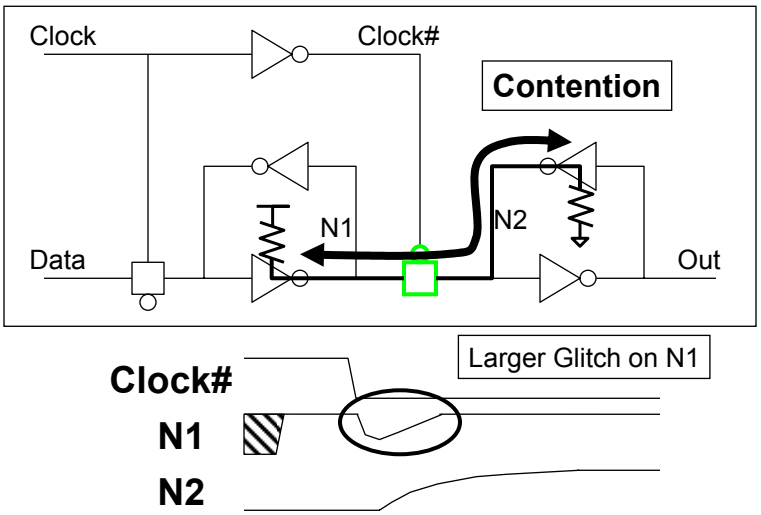


Problem #1: Charge-Share

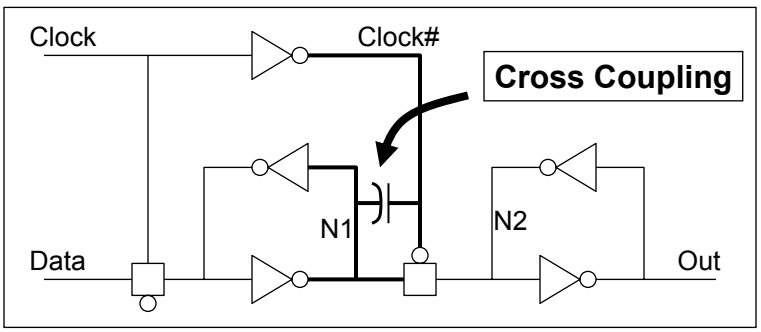


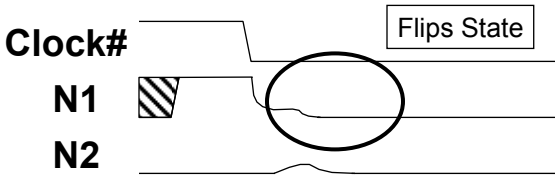


MSFF Problem #2



MSFF Problem #3





Scan MSFF "Backwriting"

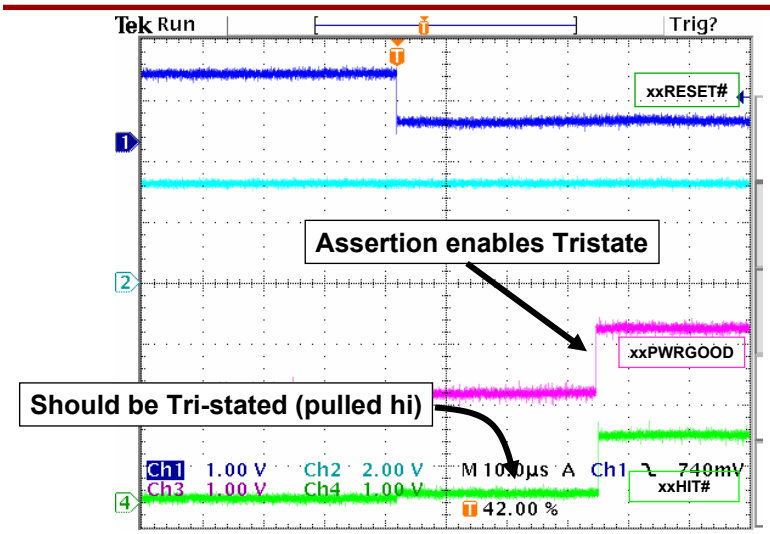
- Slave "backwrites" value into Master
  - Combination of three mechanisms to cause failure
- Re-simulated all standard cell MSFF's
  - Two other cells flagged with same problem
- Circuit was a direct "shrink" from a previous process
  - Discovered same issue on prior process—but at a MUCH higher voltage

PowerUp and Initialization

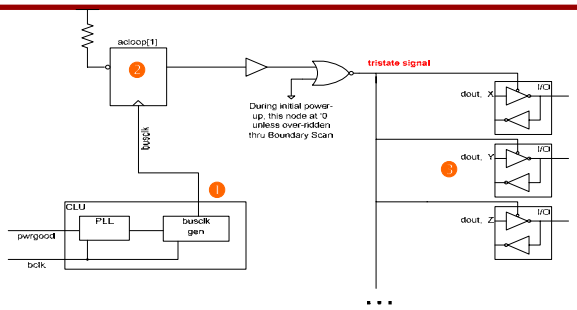
PowerUp Issue

- Observed *some* systems wouldn't boot
  - Toggling RESET always enabled boot
  - Toggling power did not guarantee boot
- Nasty problem to debug
  - System level issue (not seen on tester)
  - Intermittent failure (occurred 1 out of 100 times)
  - Debug tools not enabled (part hasn't booted)
- Started with oscilloscope waveforms…

Oscope Waveforms



Why is TriState determined by PWRGOOD?



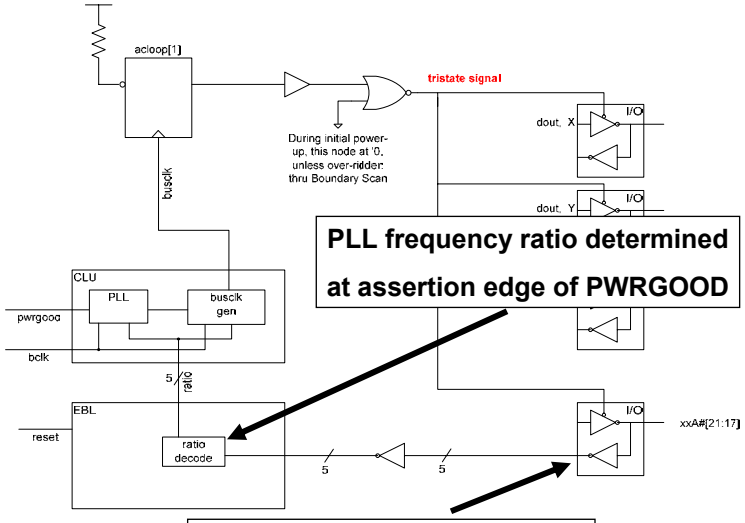
- Discovered busclk dependency @ ❷
  - ACLOOP[1] directly controls I/O tristate signal
    - Depends upon busclk for proper initialization
  - While !PWRGOOD, busclk is not generated
    - Power-up initialization @ ❶ may generate a busclk ➔ no issue
    - Otherwise, must depend on power-up initialization of ACLOOP[1] (❷)
    - "Driven value" on I/O pins will depend on power-up initialization at ❸

Why wasn't the part booting?

- PWRGOOD will always clear the ACLOOP
  - Eventually the pins should tristate
  - So, why was the part still not booting?

- Further characterization: Power levels were very low
  - When the part failed to boot, the power was very low
  - Potentially indicated that the PLL wasn't running
  - Discovered secondary effect of ACLOOP initialization problem

PLL Ratio Depends on PWRGOOD



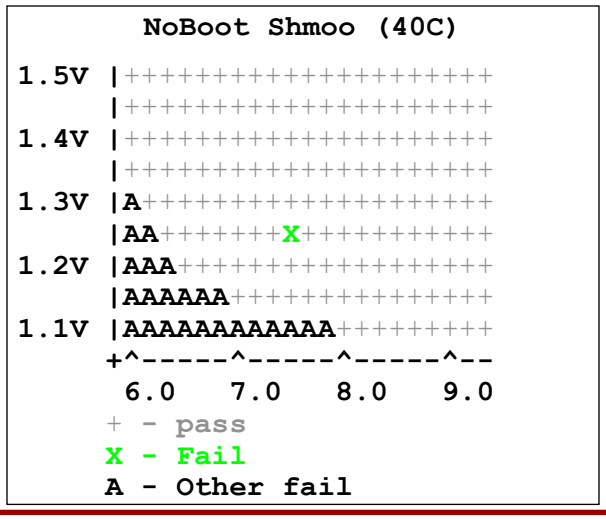
Final Root Cause

- System drives address pins at PWRGOOD assertion
  - Sets internal PLL frequency
  - Address pins are *supposed* to be tristated by the processor
- If ACLOOP powers up incorrectly, contention can occur
  - Processor is driving a '0 on address pin; system is driving a '1
  - The processor will always win
- PWRGOOD assertion tristates the address bus
  - Too late! It's already been sampled by PWRGOOD assertion
  - Only "illegal" bus fractions will cause failure
    - Only 7 out of 32 possible bus fractions are "illegal"
- Failure requires a confluence of diff't events
  - ACLOOP powers up "on"
  - Bus clock does NOT glitch during power up
  - Address pins power up driving an "illegal" bus fraction

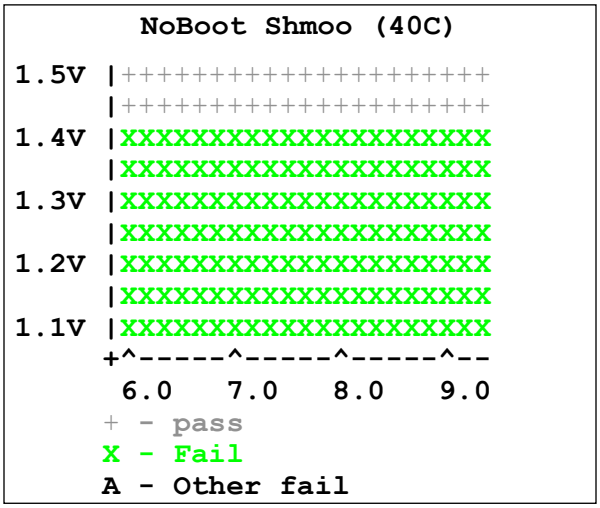
2nd PowerUp Issue

- Observed *some* systems wouldn't boot
  - Toggling RESET never enabled boot
  - Toggling power usually enabled boot
- Nasty problem to debug
  - Intermittent failure (occurred 1 out of 1000+ times)
- Some bright spots
  - Able to demonstrate on tester
    - Enabled "deterministic" behavior
    - Enabled debug tools (scan)

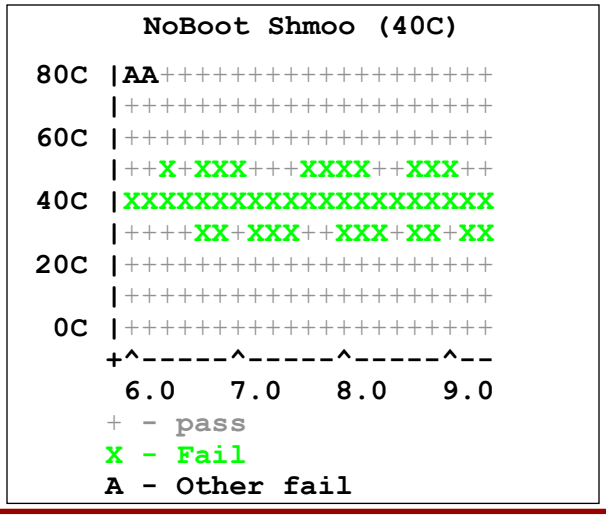
Vcc Shmoo (100x repeat)



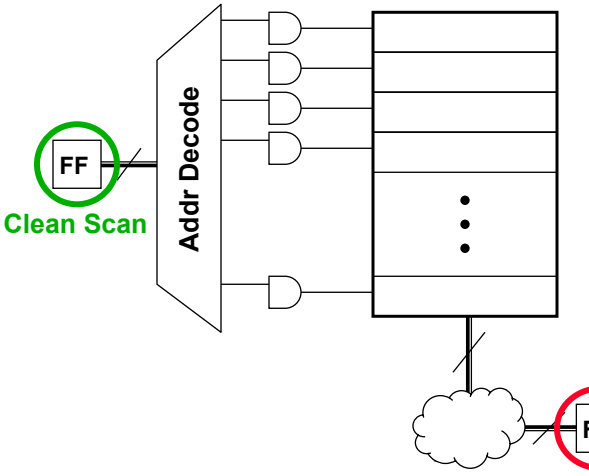
Vcc Shmoo (10000x repeat)



Temperature Shmoo (10000x repeat)



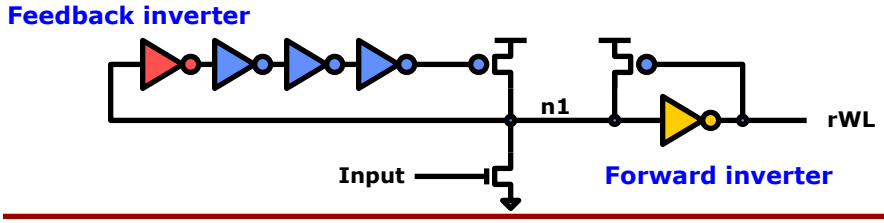
Scan Analysis



- Scan failure looked like OR of two entries
  - Common for multiple WL firing (dynamic read)
  - Uncommon for random logic
- Address decode was simple CMOS

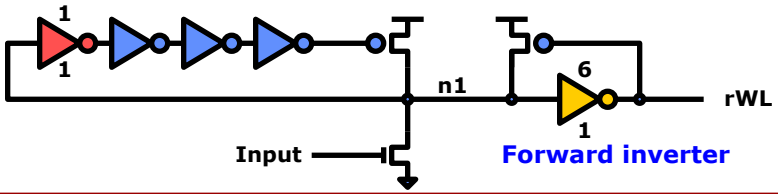
Wordline Driver

- Used a fancy self-resetting mechanism
  - Self-reset WL prevented read→write min-delay
  - Pulsed WL read array for short period of time



Wordline Driver: Problem

- Self-reset sized diff't than forward path
  - Initial state could flip forward inverter but not feedback (pseudo-metastable state)
- Resolving pseudo-meta state
  - Access WL
  - High temp
  - Low temp



Summary

Summary

- Debug requires a lot of detective work
  - Review all the evidence
  - Develop experiments to eliminate possible problems
  - Develop theory of failure
  - Validate theory
- Can't ignore ANY evidence
  - If something doesn't fit, you're missing something
- EVERY problem is different
  - Need to constantly think about alternative methods of validation
    - The Norwegian capacitor
    - The Kleveland voltmeter