UCLA COMPUTER SCIENCE DEPARTMENT

Securing Netlist-Level FPGA Design Through Exploiting Process Variation and Degradation

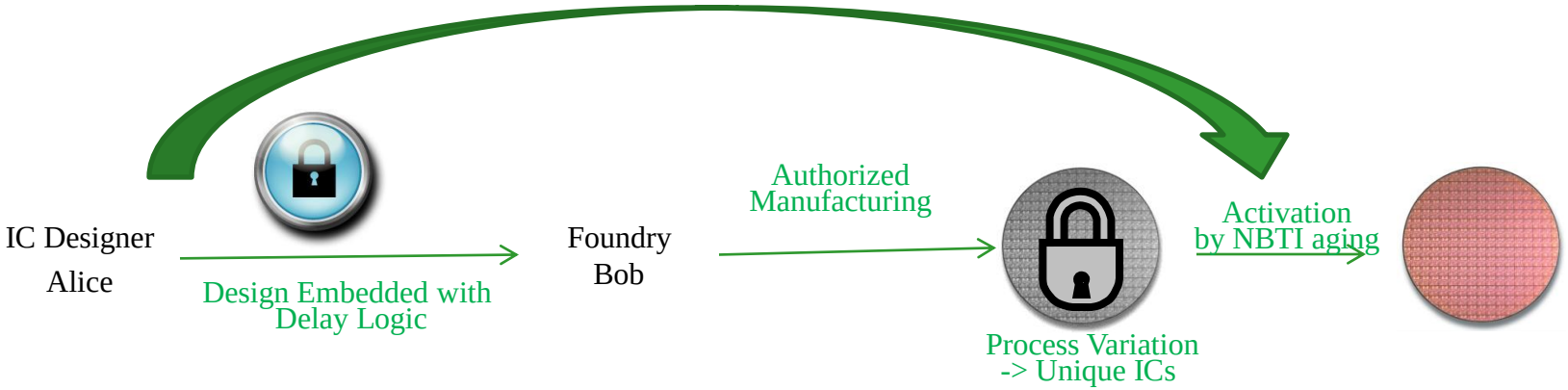
Jason Xin Zheng, Miodrag Potkonjak

Computer Science Department, UCLA
{jxzheng, miodrag}@cs.ucla.edu

Motivation

- The CMOS scaling trend of the past few decades have enabled us to build incredibly large and fast IC chips at low costs.
- At the same time, NRE costs (foundry, IC design, IC verification, etc.) have grown tremendously, which gives incentives to IC piracy.
- Goal: IP protection for both FPGA and ASIC
  - small overhead
  - resilient to cloning and reverse-engineering.

Key Concepts

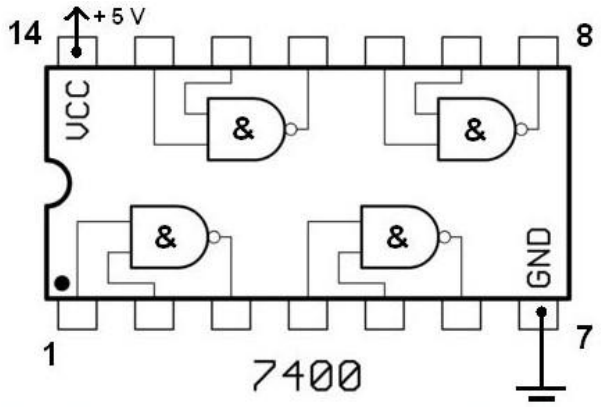


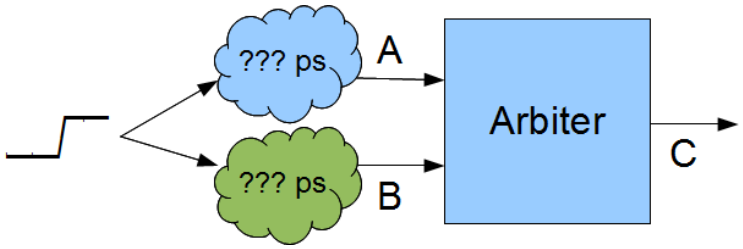
- The designer incorporates “locks” in the design.
  - Locks are implemented by Delay Logic.
  - Unlocked by correct keys: intrinsic delay features
- Every manufactured IC chip is initially “broken”.
  - Process Variation result in uniquely broken keys.
  - But at the same time structurally identical.
- A unique activation process “fixes” the IC chip.
  - Negative Bias Temperature Stability (NBTI) is used to selectively slow down some logic paths.
- In FPGA land, the same concepts apply.

Why Does it Work?

- **Each chip is activated through a unique procedure.**
  - **Activation procedure cannot be used on a different chip.**
- **No structural differences between locked and activated chips.**
  - **Structures/FPGA configurations can be cloned, but not process variations.**
- **Netlists can be obtained by reverse engineering, but locks can't be removed.**
  - **Requires high-level understanding.**

Concept: Delay Logic



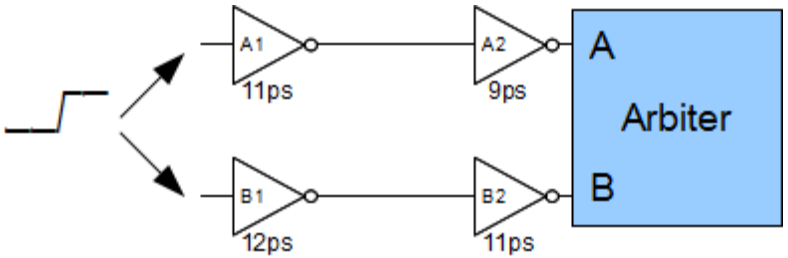


Delay Logic with Arbiter
C=1 if rising edge arrives at A first;
C=0 if rising edge arrives at B first;

- Combinatorial logic's outputs are determined by the static input states.
- In contrast, delay logic's outputs are determined by delay.
- In the lower left picture, if the rising edge arrives at A before B, the arbiter declares A the winner, and vice versa.

Delay Logic with "Identical" Paths

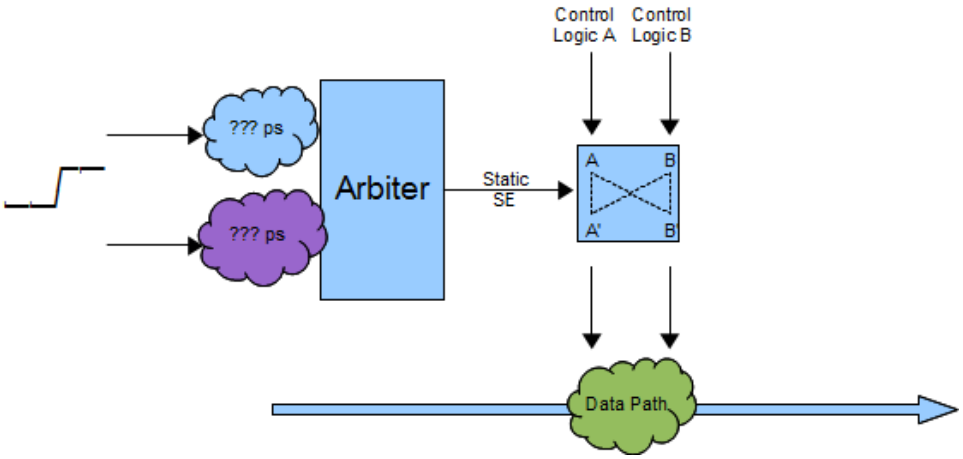
Path A wins the race!



Delay Logic with 4 Inverters with slightly different V_{th}

- Path A and B each has two inverters rated at 10ps nominal delay. A draw race on paper.
- However, process variation gives path A a slight speed advantage over path B.
- Output thus cannot be pre-determined prior to manufacturing.
- On a different die, A might be slower than B.
- Finally, if we age Path A such that A is slower, the output of the arbiter will change.

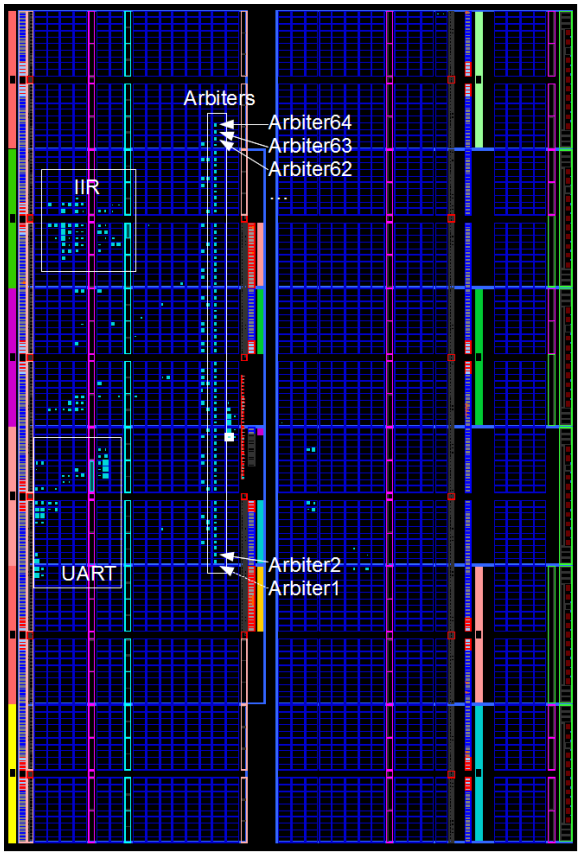
Static Shuffling



Static Shuffle Control

- In static shuffling, the shuffle controls are driven by a static/constant output from the delay logic.
- Although the delay logic outputs are static, they are dictated by unique process variation and therefore difficult to predict or measure.

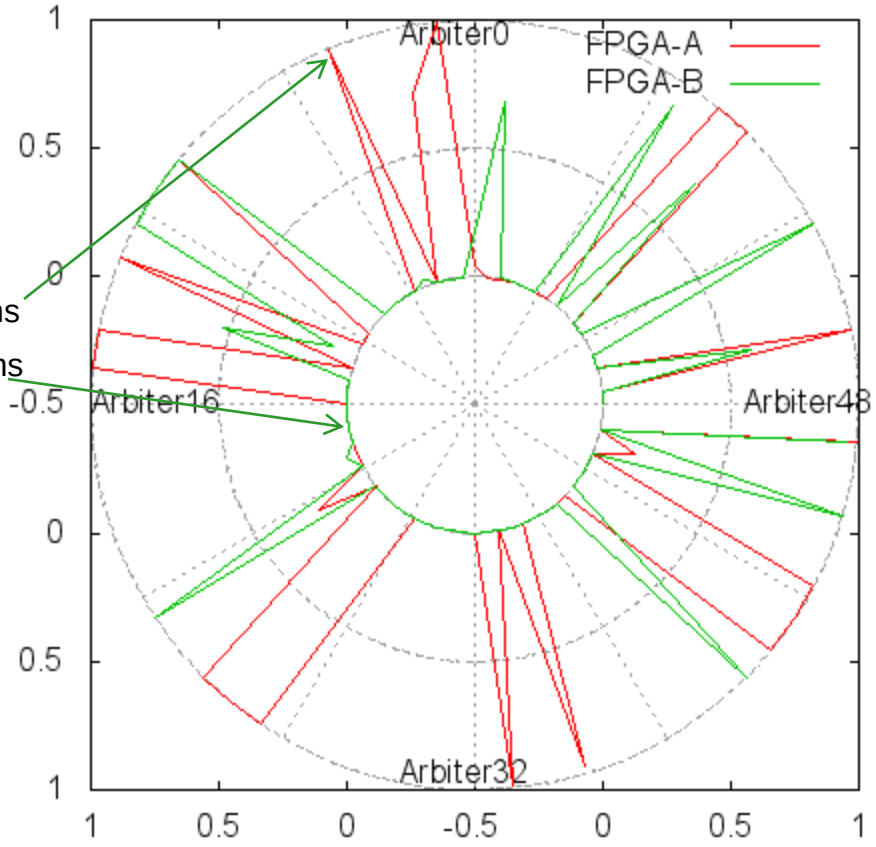
Results: Process Variation



64-Arbiter Floor Plan

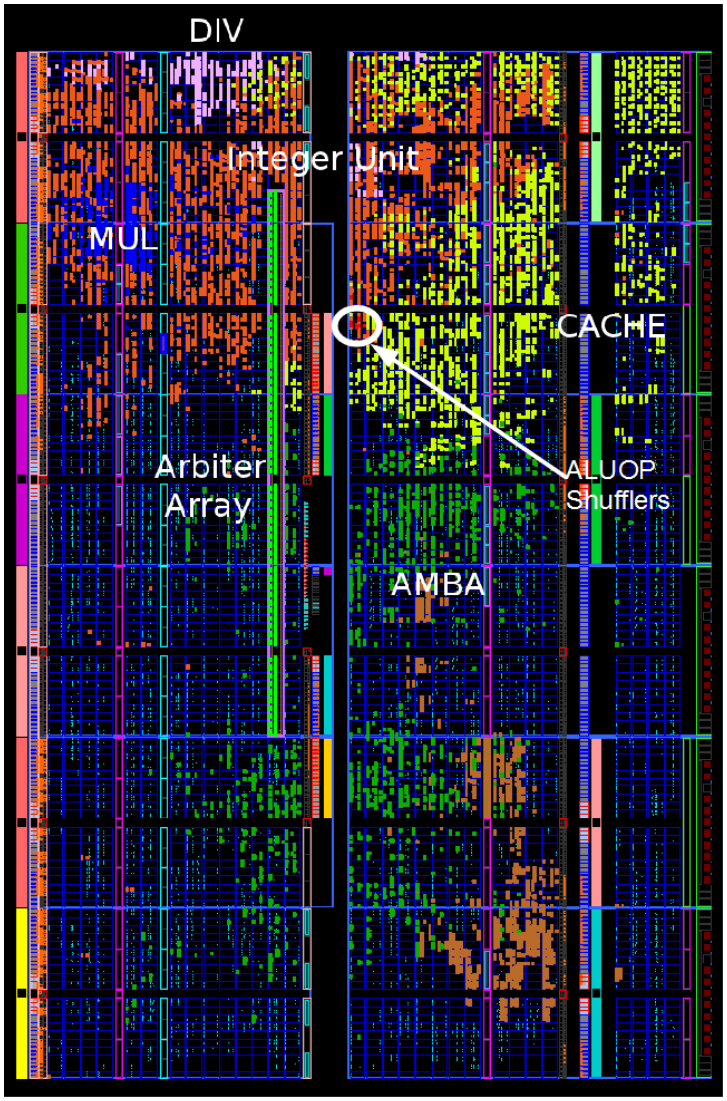
1 – Path A wins

0 – Path B wins



64-Arbiter Array Output Comparison between Two FPGAs with Identical Design

Results: LEON3 Locking Demonstration

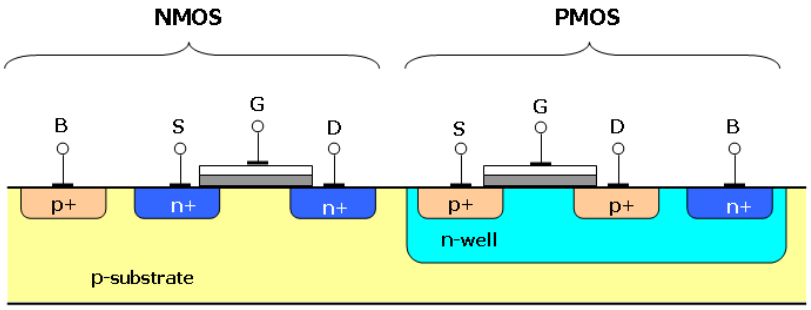


LEON3 Floor Plan

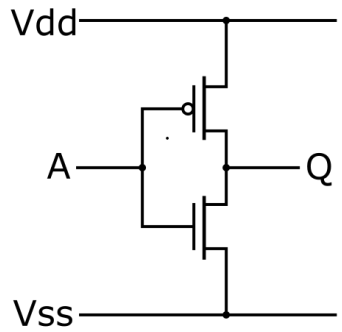
- LEON3 is an open-sourced SPARC V8 processor.
- 32-bit general purpose, 7-stage pipelined. Not state-of-art but modern architecture.
- ~8k flip-flops, ~15k logic cells.
- Modified LEON3 successfully ran on the FPGA whose arbiter outputs matches that of expected.

Negative Bias Temperature Instability (NBTI)

- NBTI is an aging process that primarily affects PMOS devices which are negatively biased during operation.
- Interface traps are created, and threshold voltage (Vth) and propagation delay are degraded.
- In essence, NBTI slows down CMOS devices.

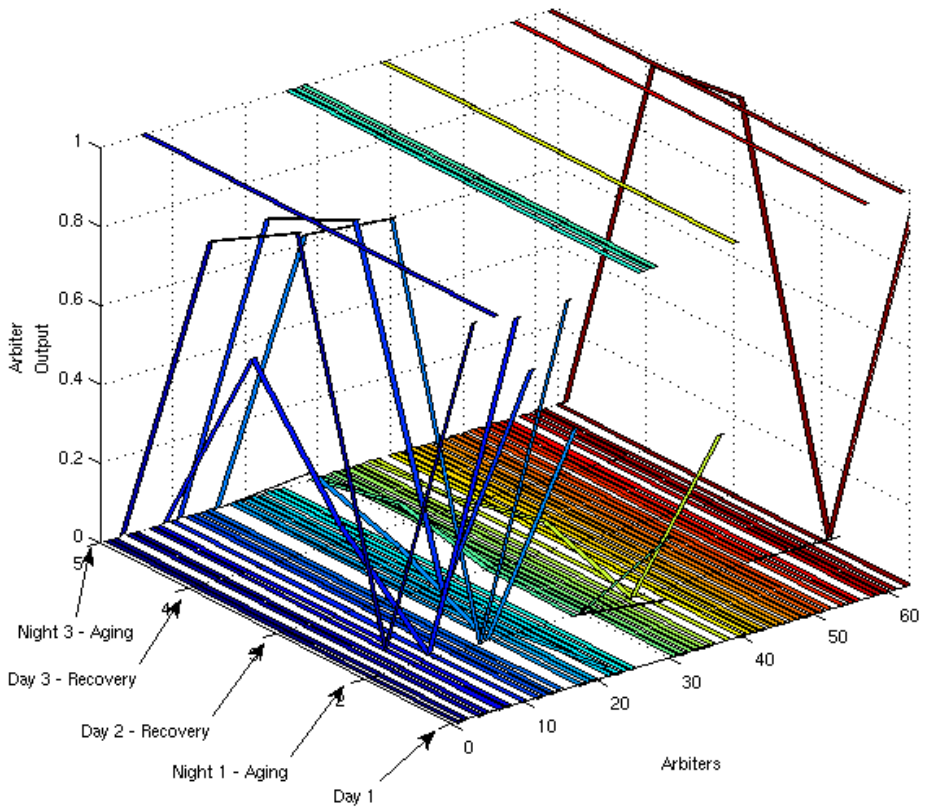


CMOS structures
Source: http://en.wikipedia.org/wiki/File:Cmos_impurity_profile.PNG



Example CMOS Inverter
Source: http://en.wikipedia.org/wiki/File:CMOS_Inverter.svg

Results: NBTI Aging



Aging/Recovery Results for FPGA A

- Static inputs are applied to the arbiters overnight.
- Recovery by changing the static inputs to dynamic.
- Two aging cycles are clearly visible.

Conclusion

- A new approach to IP security backed by process variation and NBTI aging is proposed.
- We have demonstrated how to lock down designs using static shuffling and delay logic.
- We have shown that process variation on 65nm FPGA can be observed using the delay logic.
- We have shown that NBTI aging and recovery effects can also be observed after just 10-14 hours of stress.