Soft Errors

A curse from the heavens

Smruti R. Sarangi

Department of Computer Science
Indian Institute of Technology
New Delhi, India

Outline

Curse from the Heavens

Soft Error

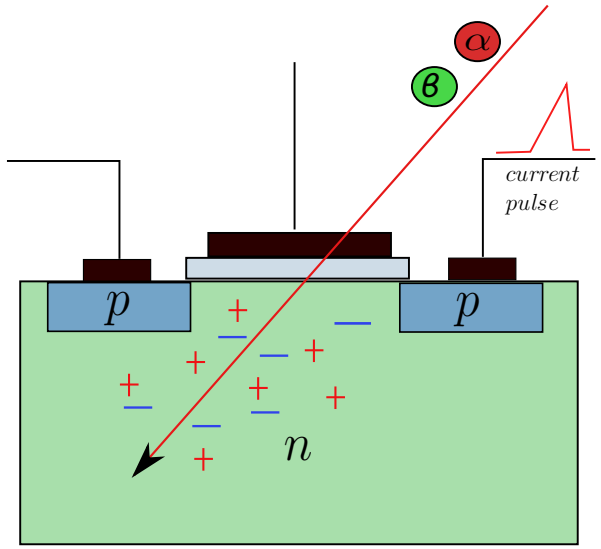


Figure 1: Current pulse after a particle strike

Definition

Soft Error: A soft error is any measurable or observable change in state or performance of a microelectronic device, component, subsystem, or system (digital or analog) resulting from a single energetic particle strike. The particle includes but is not limited to alpha particles, neutrons, and cosmic rays.

History of Research in Particle Strikes

- People recorded failures in above ground nuclear sites from 1954 to 1957. (Wallmark and Marcus, 1962)
- They started becoming important in space missions in the seventies.
- The first example of soft errors in circuits was observed in DRAMs. This was observed for the first time at sea level.
- In the early 80s most of the soft errors used to happen because of traces of radioactive elements like uranium and thorium in the packaging materials.
- Soft Errors gradually started affecting static RAMs. The failure rate is between 100 to 1000 FITs.
- By 2012, soft errors will begin affecting logic circuits. (adders, multipliers, and other complex units).

Types of Soft Errors

- Intrinsic
  - Power supply noise, cross coupling noise.
  - Temperature variations.
- Extrinsic
  - Cosmic rays.
  - alpha particles, neutrons, neutrinos, gluons

Radiation Mechanisms in Semiconductors

- Alpha Particles: In the 70s the were emitted by traces of uranium and thorium impurities in packaging materials. Gold used in the pins and lead based isotopes in solder bumps are mainly responsible for alpha particle emissions today. Their energy is between 4-9 MeV.
- Neutrons: These are produced by cosmic interactions in far away galaxies. They are able to penetrate the earth's atmosphere and ionize the silicon substrate. Their energy is about 1 MeV.
- Secondary radiation: Alpha particles and lithium nuclei are produced by the interaction of neutrons with the unstable isotope of boron, B^{10}, in boron doped silicon. Their energy is approximately 1 MeV. They were the major source of soft errors in 25 and 18 μ technologies. However, B^{10} is nowadays filtered out in the fabrication process.

Dynamics of a Strike

- In CMOS circuits the transistors in an “off” state are the most sensitive to particle strikes.
- Sensitive areas.
  - Channel region of the nmos transistor.
  - Drain region of the pmos transistor.
- The particles typically have an LET greater 20 MeV – cm^2/mg.

Definition

Linear Energy Transfer (LET) It is the amount of energy that a particle dissipates per unit distance. It is typically divided by the density of the target material.

What Happens on a Strike

- The particle displaces electrons and holes, thus ionizing a part of the silicon substrate.

What Happens on a Strike

- The particle displaces electrons and holes, thus ionizing a part of the silicon substrate.
- The displaced electrons and holes begin to recombine. This creates a current pulse.

What Happens on a Strike

- The particle displaces electrons and holes, thus ionizing a part of the silicon substrate.
- The displaced electrons and holes begin to recombine. This creates a current pulse.
- The current pulse propagates to other parts of the circuit. When the displaced charge, Q_{coll} , is more than Q_{crit}, the pulse is large enough to create a change in state.

What Happens on a Strike

- The particle displaces electrons and holes, thus ionizing a part of the silicon substrate.
- The displaced electrons and holes begin to recombine. This creates a current pulse.
- The current pulse propagates to other parts of the circuit. When the displaced charge, Q_{coll} , is more than Q_{crit}, the pulse is large enough to create a change in state.
- Q_{coll} is a function of the ionizing particle's energy, trajectory, point of impact, and the local electric field.

What Happens on a Strike

- The particle displaces electrons and holes, thus ionizing a part of the silicon substrate.
- The displaced electrons and holes begin to recombine. This creates a current pulse.
- The current pulse propagates to other parts of the circuit. When the displaced charge, Q_{coll} , is more than Q_{crit}, the pulse is large enough to create a change in state.
- Q_{coll} is a function of the ionizing particle's energy, trajectory, point of impact, and the local electric field.
- The current transient lasts for around 200 picoseconds. (NOTE: A clock cycle is 500 ps on a 2 GHz processor). Most of the impact is within 2-3 microns of the impact site.

Shape of the Pulse

- The current pulse typically has a sharp rise, and a very gradual fall.

$$I(t) = \frac{Q_{coll}}{\tau_\alpha - \tau_\beta}\left(e^{-\frac{t}{\tau_\alpha}} - e^{-\frac{t}{\tau_\beta}}\right)$$

- τ_α is the collection time constant, which is process dependent.
- τ_β is the ion-track establishment time constant. This is independent of technology.
- Typical values : τ_α = 164 ps, τ_β = 50 ps
- The displaced charge is about 0.65 pC.

Shape of the Pulse-II

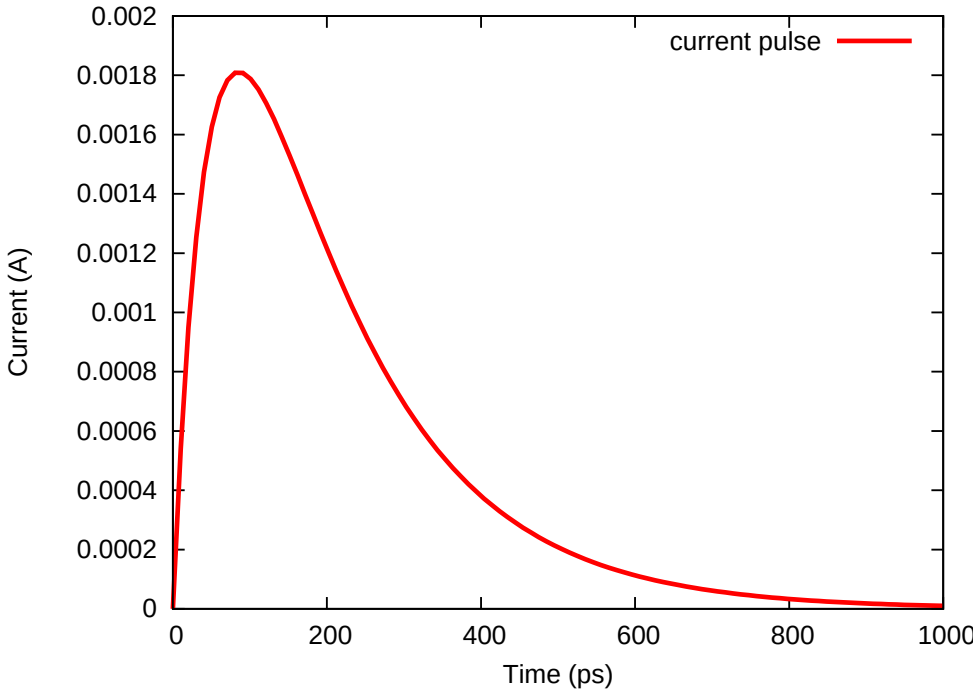


Figure 2: A typical current pulse

- Any kind of heavy tailed distribution can be used to model it.
  - Pareto, Log-Normal, Weibull, Double Exponential, Levy

Hazucha-Svensson Model

- Let us define the term *SER* as the number of times a current pulse capable of flipping a bit is generated per second.
- The Hazucha-Svensson model defines the SER to be

$$SER = F * CS$$

  - F is the neutron flux. The number of neutrons hitting an unit area per second.
  - CS : Critical Section. This is the area that is susceptible to particle strikes.
- The critical section, CS, is proportional to the drain area and is an inverse exponential function of Q_{crit}

$$CS \propto A * e^{-\frac{Q_{crit}}{Q_S}}$$

Hazucha-Svensson Model II

- Q_S is the called the *collection slope*
- It depends on the supply voltage and the doping profile.

The Hazucha-Svensson model proposes a one parameter model for the shape of the pulse.

$$I(t) = \frac{2}{T\sqrt{\pi}}\sqrt{\frac{t}{T}}e^{-\frac{t}{T}}$$

- T is called the effective parameter.

General Approaches

- Device Level Solutions
- Circuit Level Solutions
- Architecture Level Solutions

Outline

Purification of the Silicon

- Use low alpha packaging materials.
  - Uranium and Thorium impurities are reduced to less than 100 parts per trillion.
  - Purify the gold connectors. Use low alpha based lead isotopes for the soldering.

Purification of the Silicon

- Use low alpha packaging materials.
  - Uranium and Thorium impurities are reduced to less than 100 parts per trillion.
  - Purify the gold connectors. Use low alpha based lead isotopes for the soldering.
- Reduced the incidence of B^{10}.
  - Check all dopants for the unstable isotope.
  - Replace Boron Phosphate Silicate Glass (use as an insulator between metal layers) with other insulators.

Use Radiation Hardened Processes

- There are two broad approaches to solving this problem: Reduce Q_{coll} or increase Q_{crit}.
- Reduce Q_{coll}
  - Use a triple well process
  - Silicon on insulator process
- Increase Q_{crit}
  - Increase the supply voltage of the transistor
  - Increase the size of the transistor

Triple Well Process

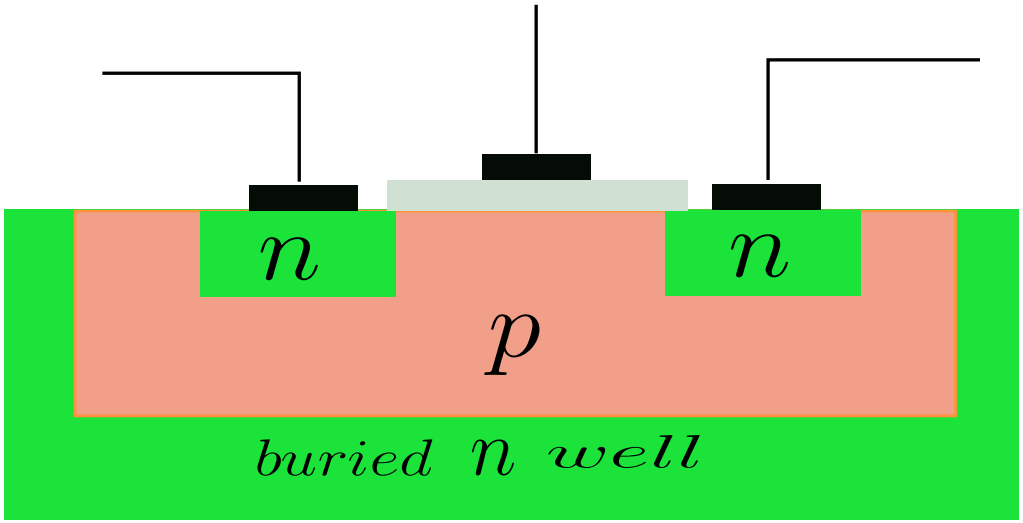


Figure 3: Triple-well process for a NMOS transistor

- An extra n-layer is added to isolate the substrate from electrical interference.
- It is also very effective in reducing the displaced charge.

Do you know the name of this stone?

Do you know the name of this stone?

Sapphire

Silicon on Insulator(SOI)

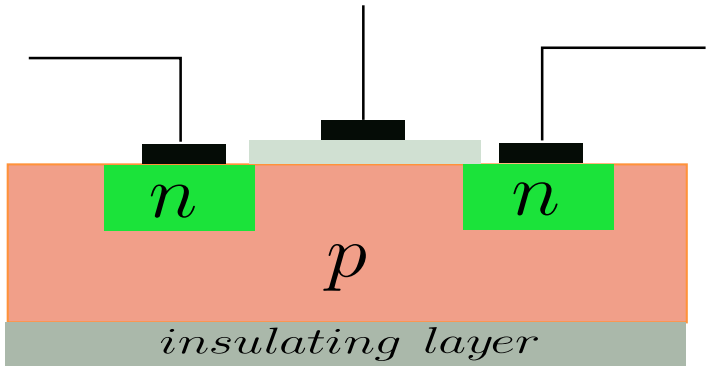


Figure 4: SOI based NMOS transistor

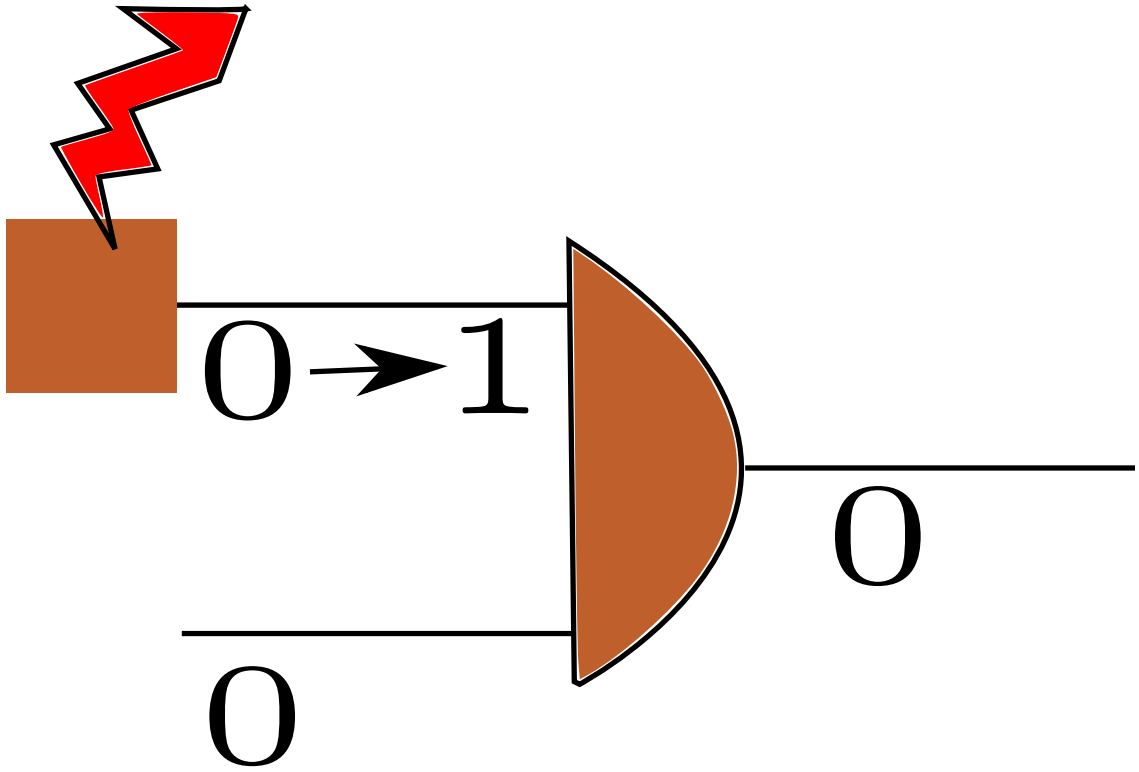
- The insulator is sapphire if we desire a radiation hardened process.
- The insulator shields the substrate from an external influence.
- It also decreases its net volume, thus decreasing Q_{coll} in the process.

All about Q_{crit}

- Q_{crit} primarily depends on four factors
  - Transistor size
  - Supply voltage
  - Output capacitance
  - Doping Density
- Q_{crit} decreases almost linearly with an increasing W/L ratio.
- Q_{crit} decrease very sharply with an increase in supply voltage.

Outline

Logical Masking



Electrical Masking

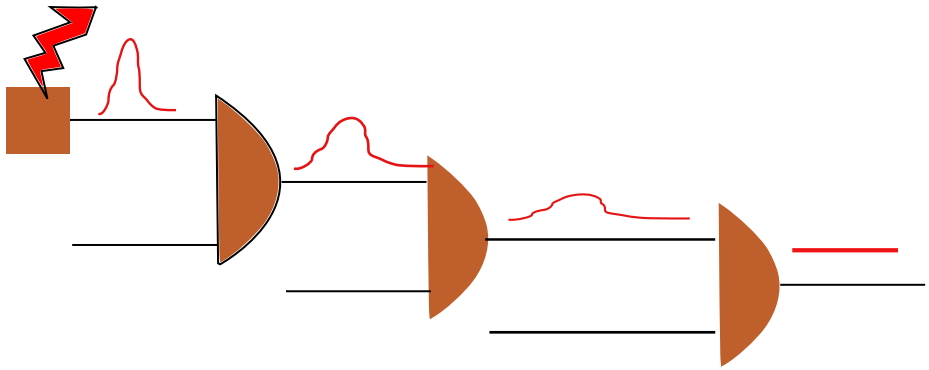
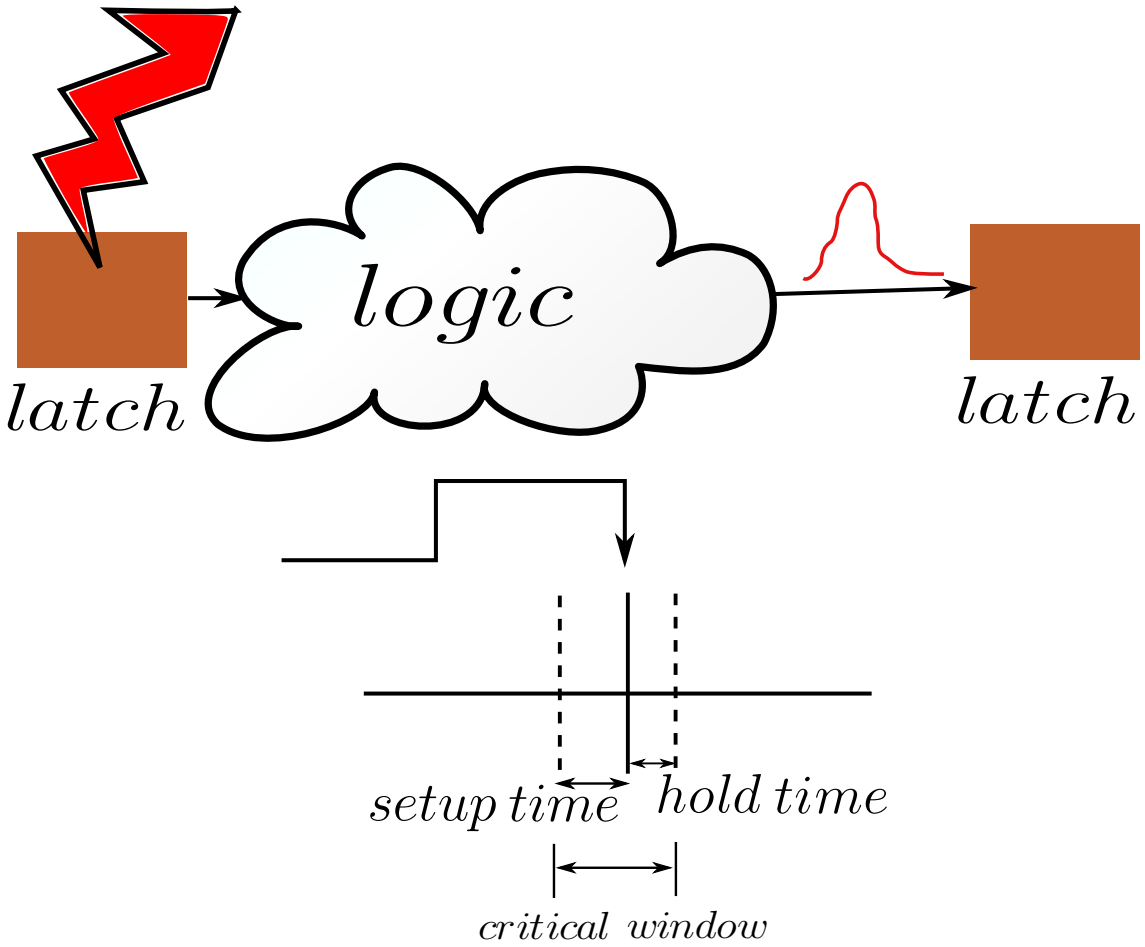


Figure 5: Electrical masking : Pulse attenuation

- A pulse gets severely attenuated as it passes through multiple gates.
- It gradually loses all of its energy.

Timing Window Masking



Finding Sensitive Latches and Gates

- Find the set of latches that are on sensitive paths. A "sensitive path" is a path of logic gates that can propagate a soft error with high probability.
  - Increase the size of the transistors of the latch.
  - Increase the output capacitance of the latch.

Finding Sensitive Latches and Gates

- Find the set of latches that are on sensitive paths. A "sensitive path" is a path of logic gates that can propagate a soft error with high probability.
  - Increase the size of the transistors of the latch.
  - Increase the output capacitance of the latch.
- Find the set of logic gates that are on sensitive paths.
  - Increase the chances of electrical masking by increasing the size of the transistors in the gate.
  - Or, connect those to a higher supply voltage line.

Outline

ECC and Redundancy

- ECC : Error Correction Code
  - We typically have a SECDED code (single error correction, double error detection)
  - Almost all the memory elements are protected
    - Main Memory (since the 70s)
    - L2 and L1 caches (since 2000)
    - Register Files (since early 2000)
    - Pipeline Latches (Fujitsu introduced it about 15 years ago)

ECC and Redundancy

- ECC : Error Correction Code
  - We typically have a SECDED code (single error correction, double error detection)
  - Almost all the memory elements are protected
    - Main Memory (since the 70s)
    - L2 and L1 caches (since 2000)
    - Register Files (since early 2000)
    - Pipeline Latches (Fujitsu introduced it about 15 years ago)
- Redundancy
  - Redundant threads: Use another thread to check the results of the current thread. (IBM G-5)
  - Checker processors : Use a smaller processor to check the results of a larger processor.
  - Extra cores : Use an extra core on a multi-core machine to check the results.

Architectural Vulnerability Factor (AVF)

Definition

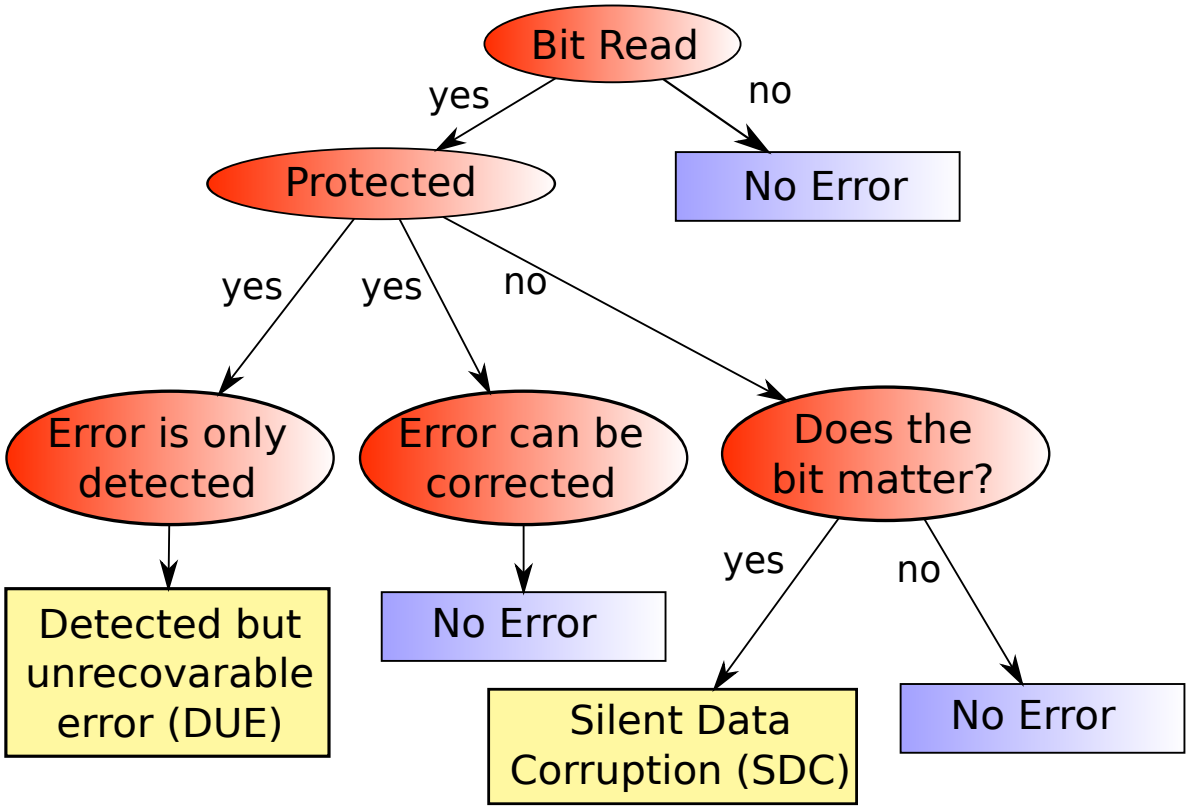
Architectural Vulnerability Factor (AVF): AVF is the probability that a soft error results in a failure.

The failure rate due to soft errors can be defined as follows:

$$F = SER * TVF * AVF$$

SER is the soft error rate. TVF is the timing vulnerability factor i.e, the fraction of time, the unit is used. AVF is the probability that the error results in an erroneous output.

Dissecting AVF



- The error rate is a combination of SDC and DUE
- SDC is potentially more harmful

Examples

Example

Example of SDC: Bit flips in functional units that are not protected, e.g., ALU, decode logic, pipeline latches.

Example

Example of DUE: Multiple SER events in units that are protected like the register file or the caches.

When is there no error?

Examples

Example

Example of SDC: Bit flips in functional units that are not protected, e.g., ALU, decode logic, pipeline latches.

Example

Example of DUE: Multiple SER events in units that are protected like the register file or the caches.

When is there no error?

Instructions that don't affect correctness

- Dynamically dead instructions
- Wrong path instructions
- Performance instructions
- Prefetch instructions
- No ops

Examples

Example

Example of SDC: Bit flips in functional units that are not protected, e.g., ALU, decode logic, pipeline latches.

Example

Example of DUE: Multiple SER events in units that are protected like the register file or the caches.

When is there no error?

Instructions that don't affect correctness

- Dynamically dead instructions
- Wrong path instructions
- Performance instructions
- Prefetch instructions
- No ops

Functional units that don't affect correctness

- Branch predictor
- Performance counters

AVF for Functional Units

- AVF is typically calculated on a per functional unit basis
- It takes into cognizance the effect of redundant instructions, and characteristics of the functional unit.
  - The AVF for the branch predictor is zero.
  - For the latches is about 50%.
  - It varies widely from 10% to 70% for all other units.
- How is AVF calculated?
  - For units like the branch predictor, it can be calculated theoretically
  - Otherwise, it is estimated with profiling runs for a set of benchmarks.
    - Inject a fault
    - Observe if the fault causes a failure in the program

Some More Definitions

Definition

ACE : Architecturally Correct Execution. These are instructions that determine the program output. Their erroneous exeuction will lead to a failure.

Some More Definitions

Definition

ACE : Architecturally Correct Execution. These are instructions that determine the program output. Their erroneous exeuction will lead to a failure.

Definition

Dynamically Dead Instruction : These are instructions whose values don't propagate to the final output of the program.

Some More Definitions

Definition

ACE : Architecturally Correct Execution. These are instructions that determine the program output. Their erroneous exeuction will lead to a failure.

Definition

Dynamically Dead Instruction : These are instructions whose values don't propagate to the final output of the program.

Definition

Ex-ACE instruction : Let us consider an ACE instruction in the instruction queue. After it is issued, it is still in the queue till it gets evicted by a newer instruction. After an ACE instruction leaves a functional unit, and is not required anymore by the unit, it becomes an Ex-ACE instruction.

AVF for the Instruction Queue

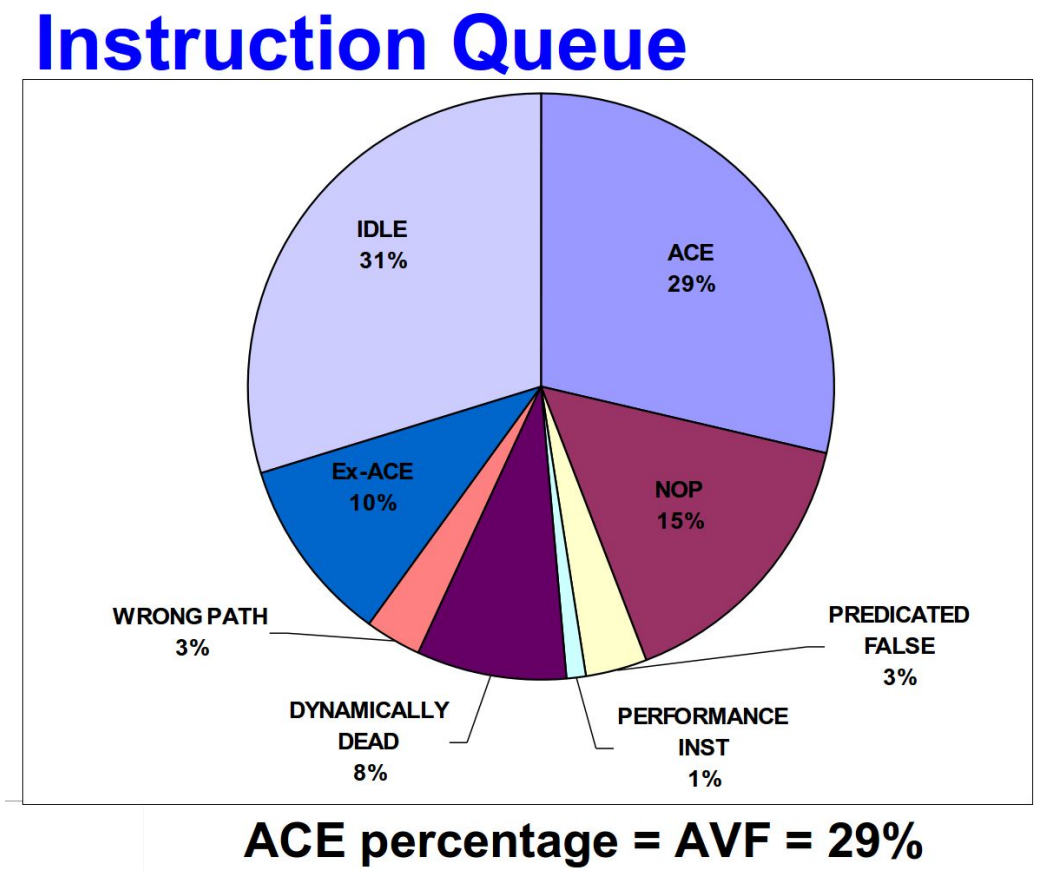


Figure 6: AVF for the instruction queue (courtesy Shubu Mukherjee Intel)

Shubhendu S. Mukherjee, Christopher T. Weaver, Joel S. Emer, Steven K. Reinhardt, Todd M. Austin: A Systematic Methodology to Compute the Architectural Vulnerability Factors for a High-Performance Microprocessor. MICRO 2003
http://portal.acm.org/citation.cfm?doid=956417.956570

Fan Wang, Agrawal, V.D. : Single Event Upset: An Embedded Tutorial, VLSI Design 2008
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4450538