# Exploring the Performance/Power/Resilience (PPR) Co-Design Options for Throughput-Oriented Architecture

**Xin Fu**[†]
Jingweijia Tan[†]
Shuaiwen Leon Song[*]

University of Houston[†]
Pacific Northwest National Laboratory[*]

UNIVERSITY of HOUSTON

Pacific Northwest
NATIONAL LABORATORY

# Throughput-Oriented Architecture: GPGPUs

- Graphics Processing Units (GPUs)
  - Support thousands of concurrent threads
  - High computational throughput
  - GPGPUs become a popular platform for ***general-purpose*** HPC applications


Biomedical engineering

Space

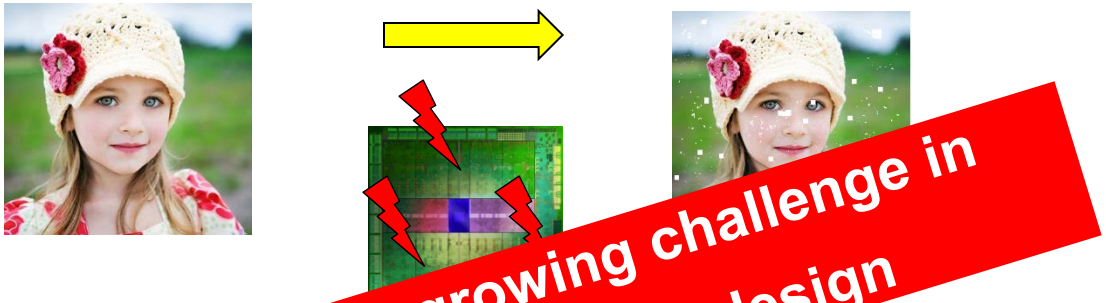
Oil exploration


Finance

Automotive

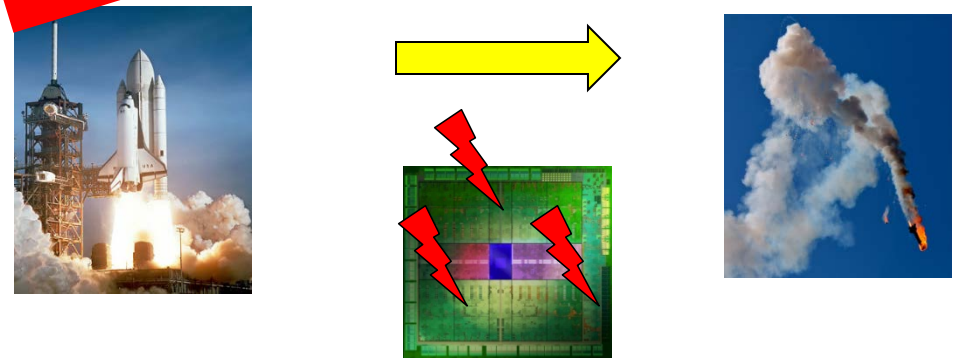
# Reliability Challenges in GPGPUs

- GPUs is originally designed for graphics applications
  - Error detection and fault tolerance are ignored!



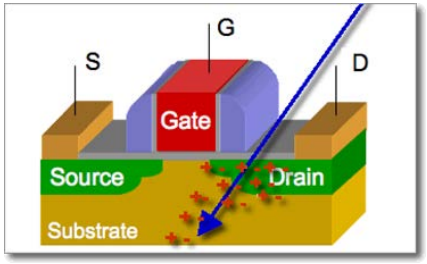
- General-p grams require strict execution correct

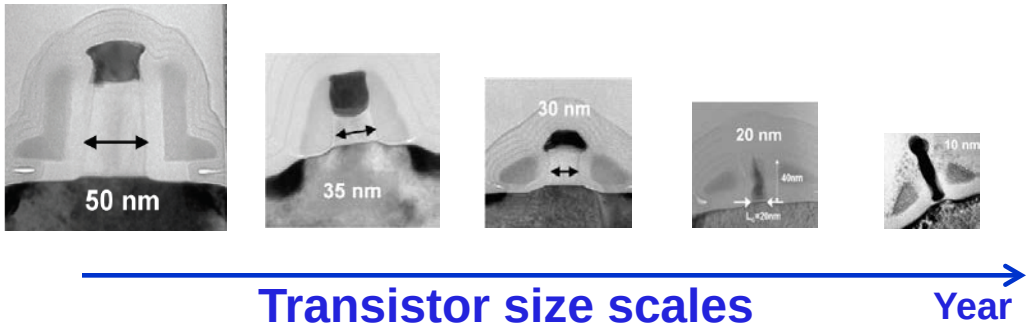
**Reliability is a growing challenge in GPGPU architecture design**

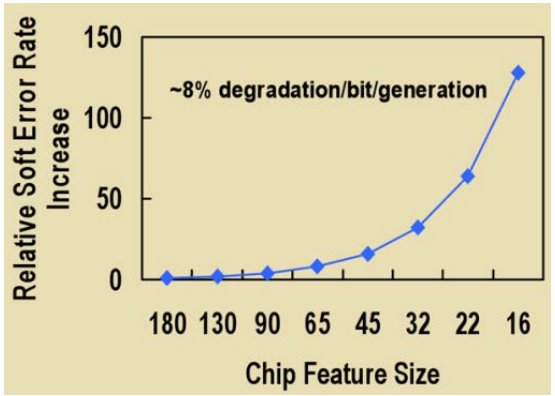


# Hardware Reliability Challenge: Soft Error

- One of the most severe reliability threats
- Caused by energetic particles
  - Flip the device state (0→1 or 1→0)
- Leads to data corruption
  - Circuits are not damaged



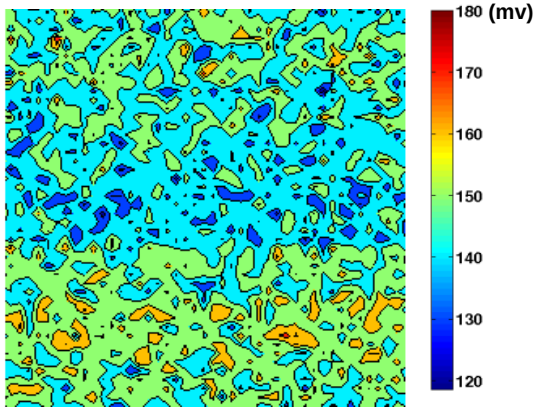



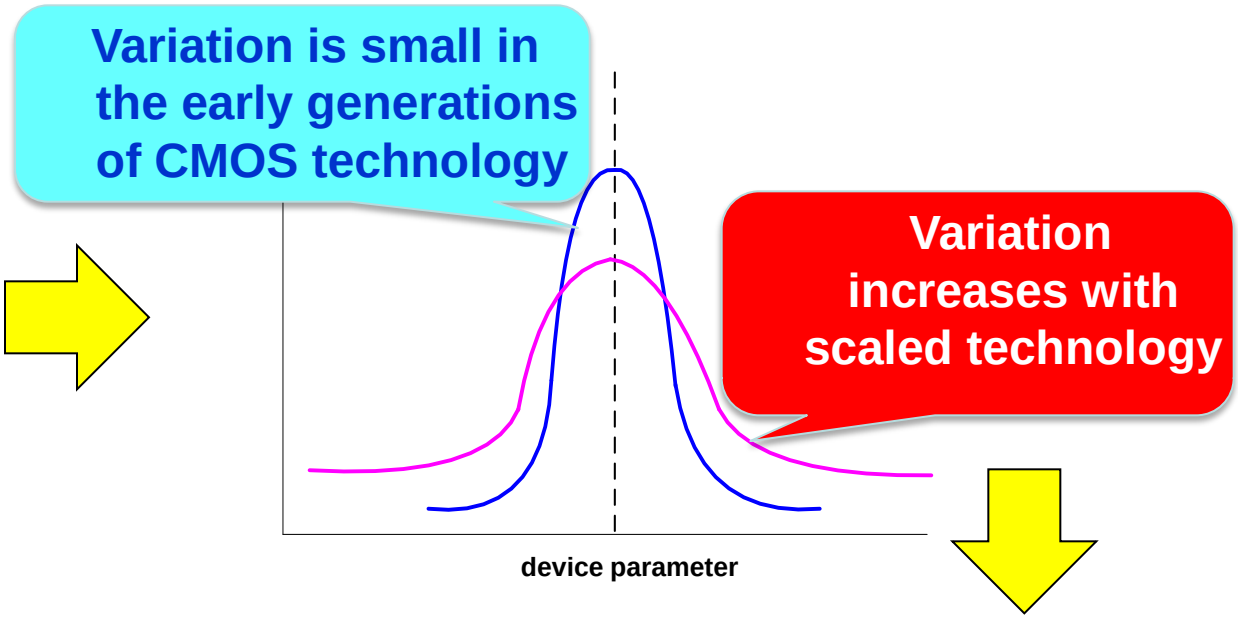

S. Borkar, Designing reliable systems from unreliable components: the challenges of transistor variability and degradation, IEEE Computer Society, 2005

**Soft error rate increases significantly as technology scales down**

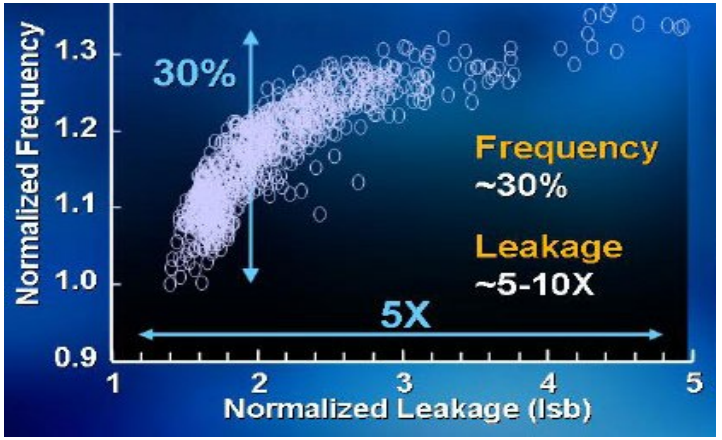
# Hardware Reliability Challenge: Process Variation (PV)




Device parameter varies across the chip

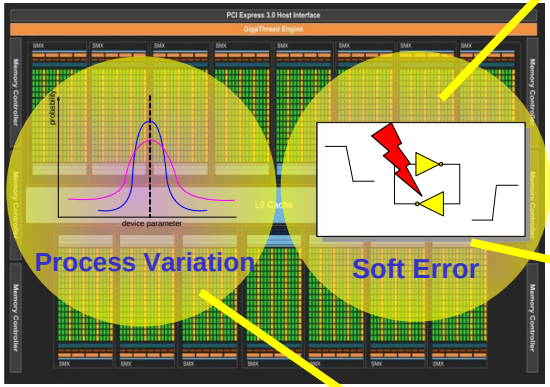



- PV dramatically reduce GPUs $F_{max}$
  - $F_{max}$ is limited by the slowest critical path
  - Numerous critical path in GPUs




S. Borkar, et al. Parameter variations and impact on circuits and microarchitecture, DAC'03

# Performance/Power/Resilience Co-Design




**RISE:** ***R***ecycling the streaming processors ***I***dle time for ***S***oft-***E***rror detection

**=> performance/resilience co-design**

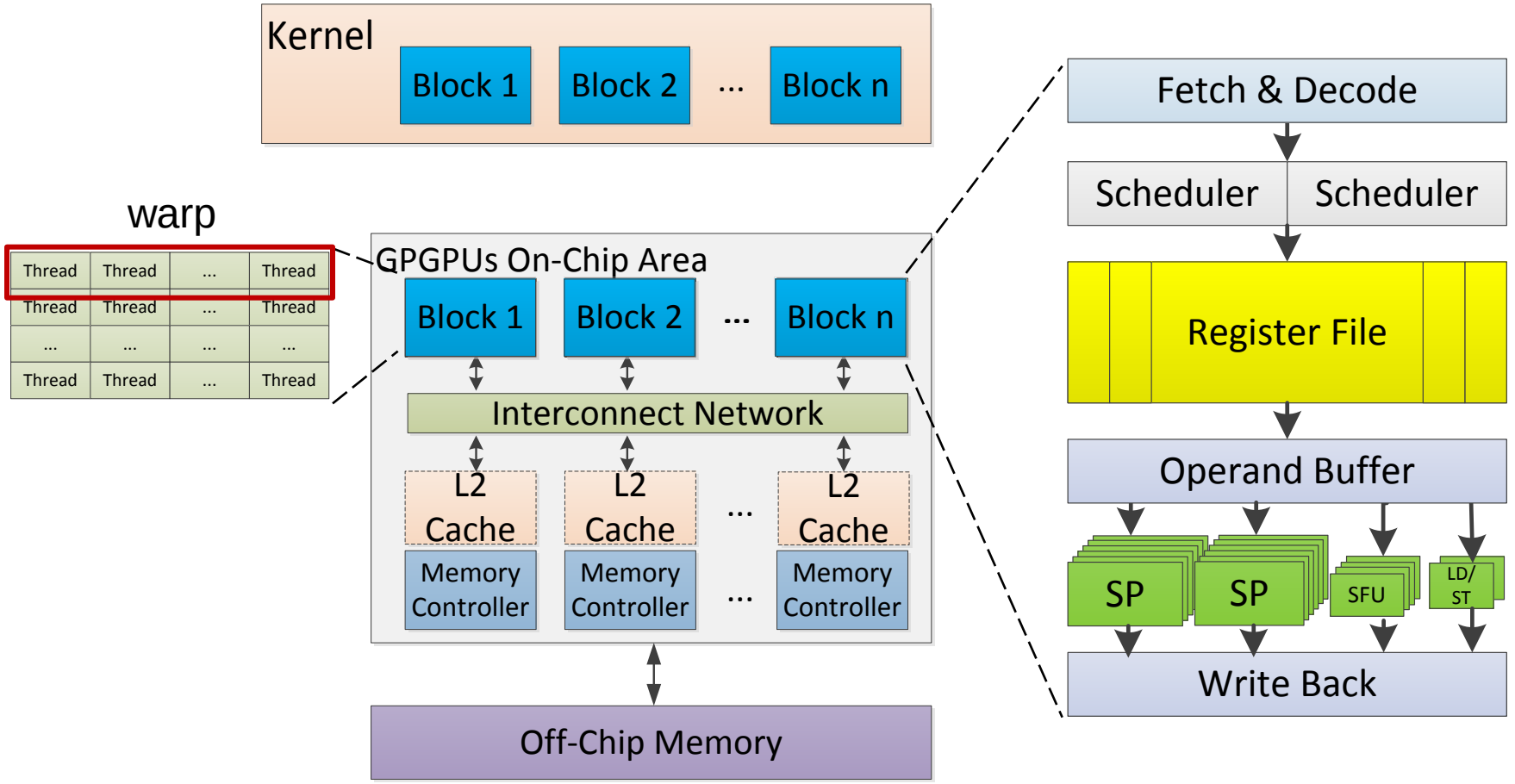
***LESS: LE***verage re***S***istive memory to enhance the ***S***oft-error robustness

**=> performance/power/resilience co-design**

Mitigating the Susceptibility of GPGPUs Register File to Process Variations

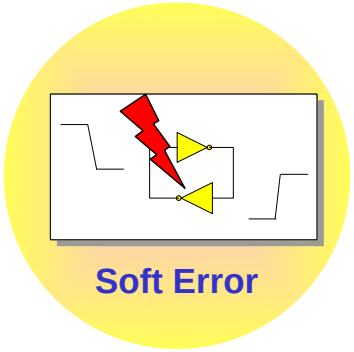
**=> performance/resilience co-design**

# Background: GPGPUs Architecture




SM: Streaming Multiprocessor

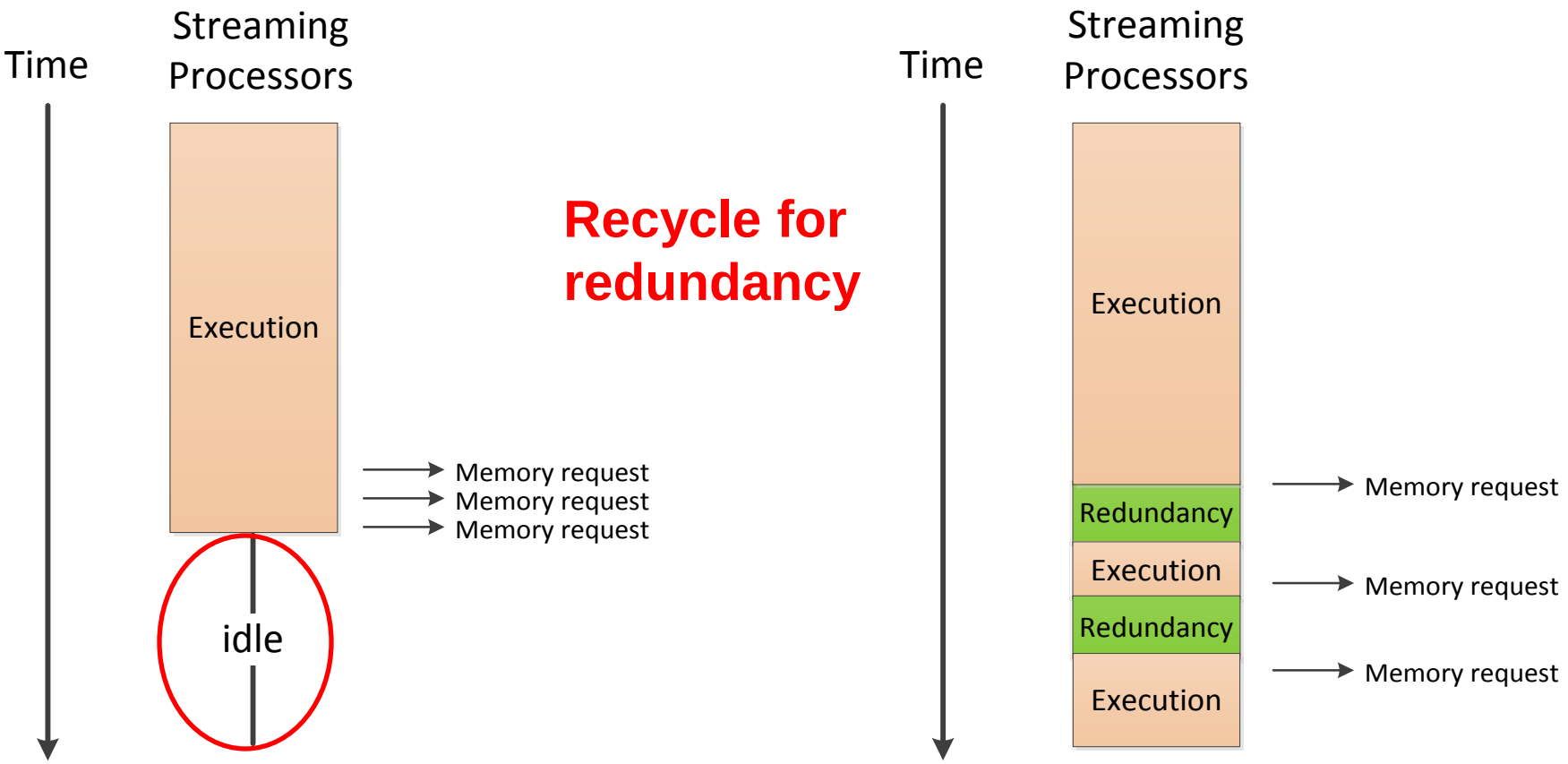
# Outline




- Introduction and Background
- RISE: ***R***ecycling the streaming processors ***I***dle time for ***S***oft-***E***rror detection
- ***LESS: LE***verage re***S***istive memory to enhance the robustness
- Mitigating the Susceptibility of GPGPUs Register File to Process Variations
- Conclusions

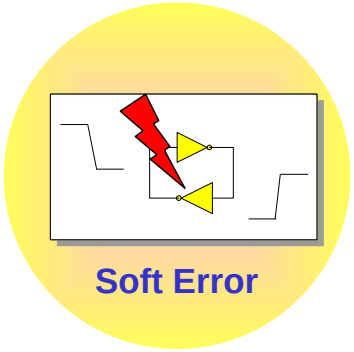
# RISE

- Warps in the same SM exhibit similar execution progress
  - Memory requests are sent at similar time




RISE: Recycle the SPs idle time caused by long-latency memory accesses for redundancy

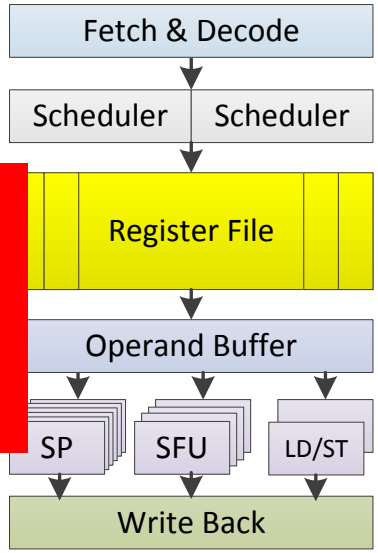
# Outline




- Introduction and Background
- RISE: ***R***ecycling the streaming processors ***I***dle time for ***S***oft-***E***rror detection
- ***LESS: LE***verage re***S***istive memory to enhance the robustness
- Mitigating the Susceptibility of GPGPUs Register File to Process Variations
- Conclusions

# Register File (RF) in GPGPUs

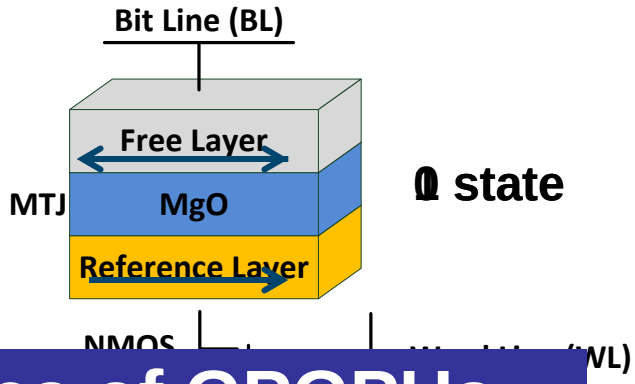
- The extreme multithreading requires a sizeable register file
  - 3.75MB in NVIDIA Kepler [NVIDIA, 2012]
  - 6MB in AMD Cayman [Kanter, 2010]
- **We focus on the soft-error reliability and power co-optimization for GPGPUs RF**
  - Exposes them to the particle strikes




- RF is *power-hungry* [Leng, 2013]
  - Large dynamic and leakage power due to large size

# Resistive Memory

- Spin-Transfer Torque RAM (STT-RAM)
  - Uses Magnetic Tunnel Junction (MTJ) as the data storage device




- Benefits:
  - high density, non-volatile
  - low leakage
  - Immune to soft errors

**Explore unique characteristics of GPGPUs applications to combat the negative impact caused by write**

- Disadvantages: long write delay and high write energy

**All STT-RAM RF=> 70% performance ↓, 14% energy ↑**

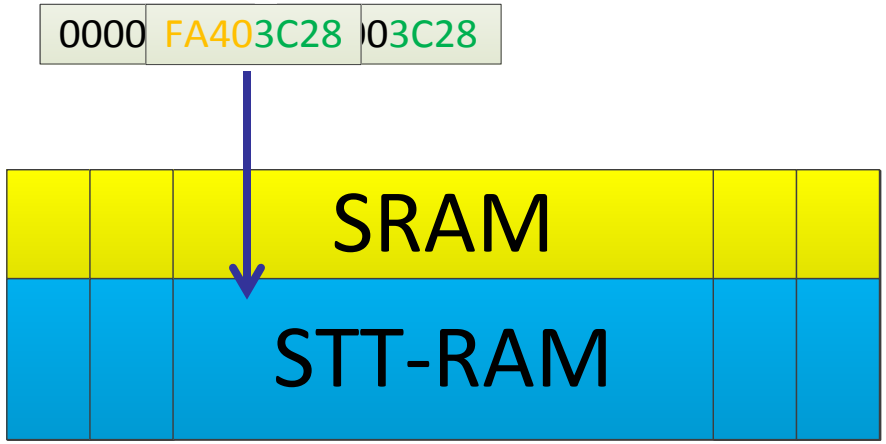
# LESS

- **LESS: co-optimizes soft-error reliability and power consumption for GPGPUs RF**
  - **A small amount of long lifetime registers are the major vulnerability contributor**

**Lifetime-Aware LESS**

  - **A lot of STT-RAM writes are narrow-width**

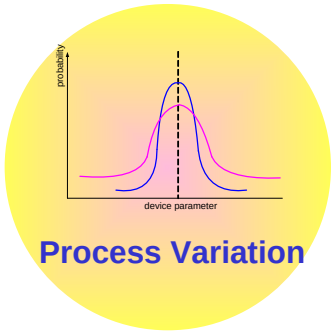
**Narrow-Width-Aware LESS**




**<- Short lifetime register values**

**<- Long lifetime register values**

# Outline

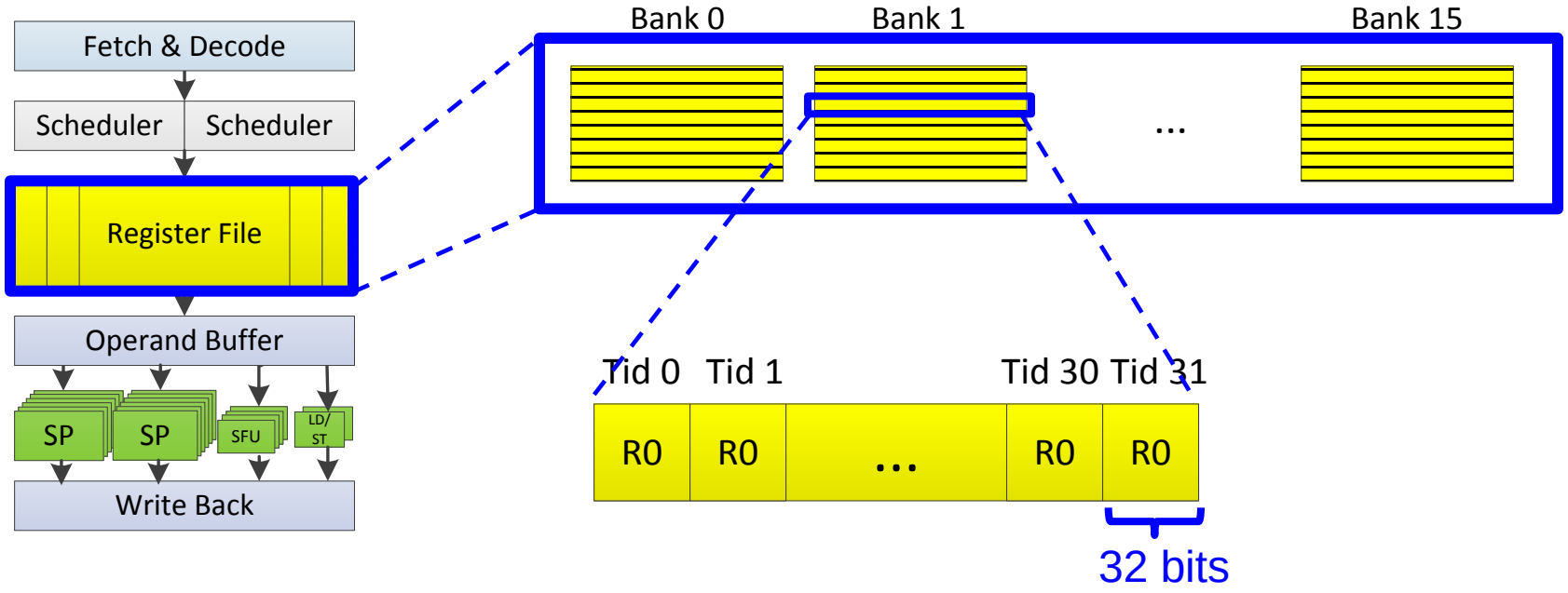




- Introduction and Background
- RISE: ***R***ecycling the streaming processors ***I***dle time for ***S***oft-***E***rror detection
- ***LESS: LE***verage re***S***istive memory to enhance the robustness
- Mitigating the Susceptibility of GPGPUs Register File to Process Variations
- Conclusions

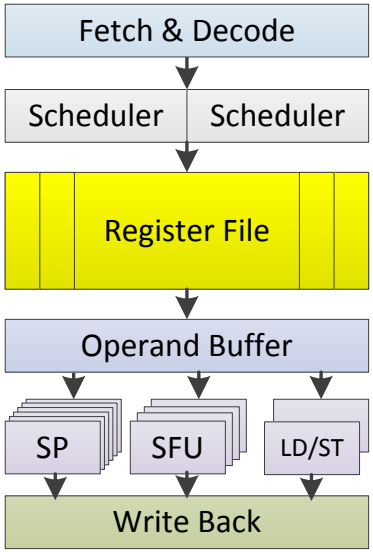
# Background: Multi-Banked RF in GPGPUs

- Highly-banked (e.g., 16) to provide high bandwidth
  - Multiple register operands required by one instruction can be accessed simultaneously
- Each entry is composed of 32 same-named registers
  - Threads execute in SIMD mode
  - 1 warp = 32 threads

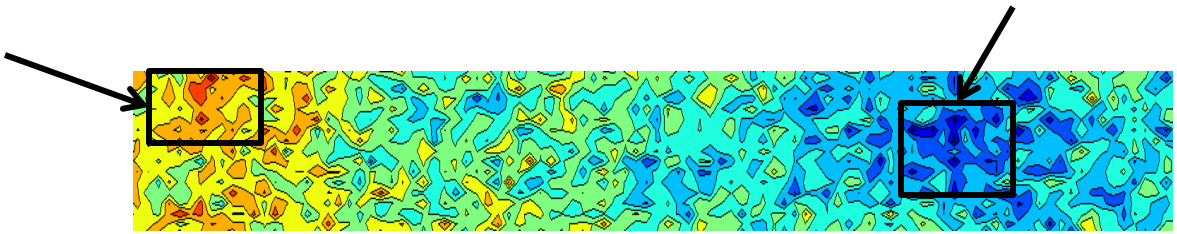


# PV Impact on GPGPUs Register File (RF)

- GPGPUs RF
  - Large size
  - Highly banked
  - Contains numerous critical paths



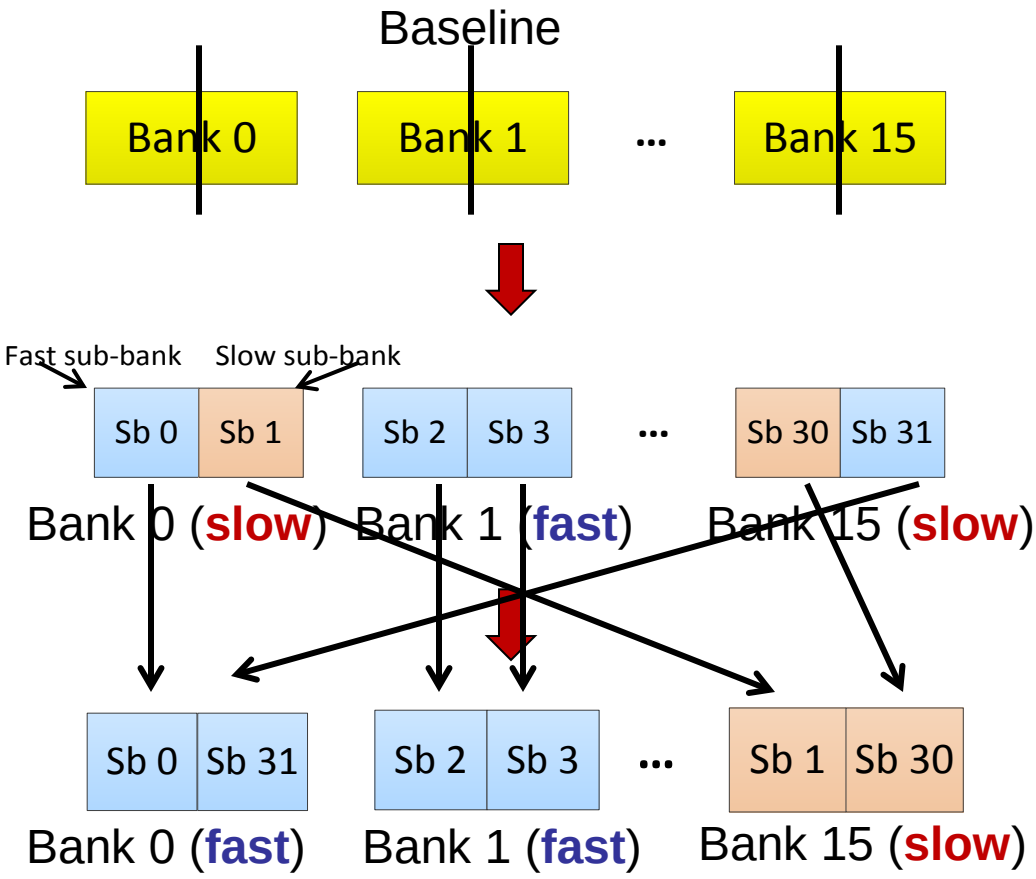

- **RF is vulnerable to PV impact**



$V_{th}$ variation map of GPGPUs RF

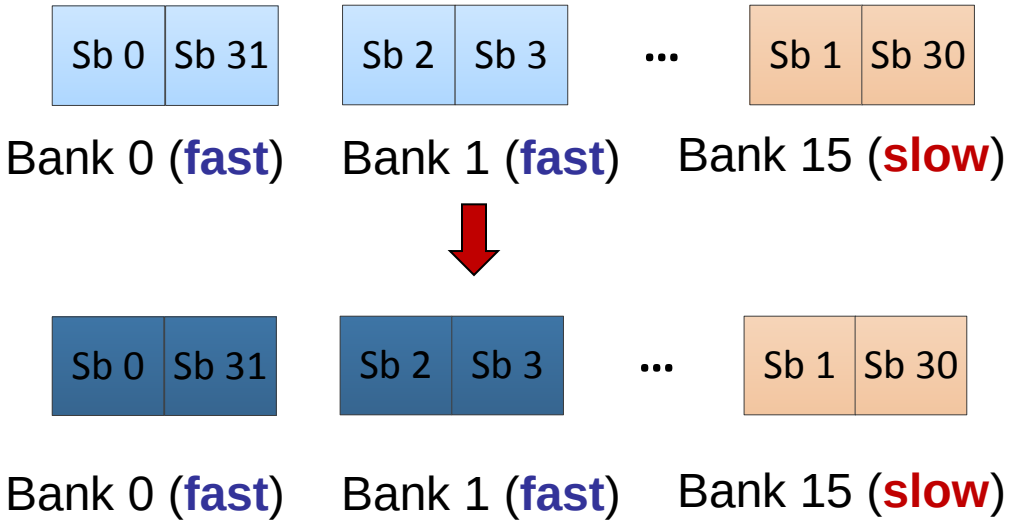
# Frequency Optimization under PV

- ***Frequency*** **Optimizations for GPGPUs RF under PV**
  - Variable-Latency Sub-Bank
  - RF Bank Re-Organization
  - 15% frequency improvement



# IPC improvement under PV

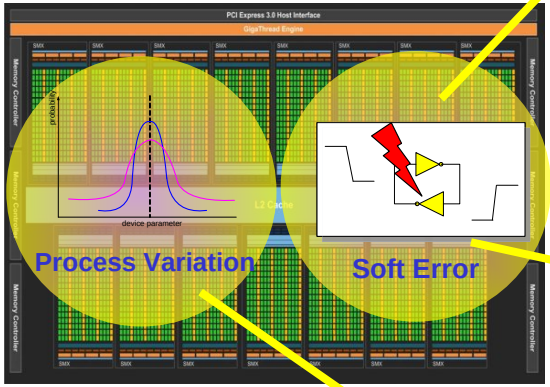
- **Mitigating the *IPC* Degradation under PV**
  - Fast-Warp Aware Scheduling
  - IPC degradation is almost 0%
  - 15% overall performance improvement



# Outline

- Introduction and Background
- RISE: ***R***ecycling the streaming processors ***I***dle time for ***S***oft-***E***rror detection
- ***LESS: LE***verage re***S***istive memory to enhance the robustness
- Mitigating the Susceptibility of GPGPUs Register File to Process Variations
- Conclusions

# Performance/Power/Resilience Co-Design




**RISE:** ***R***ecycling the streaming processors ***I***dle time for ***S***oft-***E***rror detection

**=> performance/resilience co-design**

***LESS: LE***verage re***S***istive memory to enhance the ***S***oft-error robustness

**=> performance/power/resilience co-design**

Mitigating the Susceptibility of GPGPUs Register File to Process Variations

**=> performance/resilience co-design**