Processor Shield for L1 Data Cache Software-Based On-line Self-testing

Authors: **Ching-Wen Lin** and Chung-Ho Chen

Inst. of Computer & Communication Engineering,

Department of Electrical Engineering,

National Cheng Kung University, Taiwan

Outline

- Introduction
- Challenge for on-line cache SBST
- Processor shield design
- Guardband calibration
- Case study : ARMv5 processor
- Conclusion

System Reliability

- Operational fault
  - On-line testing
    - Hardware built-in self-test (BIST)
    - Software-based self-test (SBST)
- Aging effect (NBTI)
  - Guardbanding
    - One-time worst-case guardband
    - Dynamically calibrated guardband

Purpose: Data Cache On-Line SBST

- OS-managed platform
  - Virtual memory system
- RAM and control logic on-line SBST
- Dynamic voltage frequency scaling system (DVFS)
  - Minimal required guardband calibration

Outline

- Introduction
- Challenge for on-line cache SBST
- Processor shield design
- Guardband calibration
- Case study : ARMv5 processor
- Conclusion

Challenges

- System memory mapping
  - Virtual address translation
  - Physical memory layout
- Alteration to current system
  - Required memory region
- Faulty effect isolation
  - Iterative SBST
  - DVFS system cooperation

March Algorithm

- Special addressing order
  - Ascending (from index 0 to index MAX)
  - Descending (from index MAX to index 0)
  - Either
- Data background

$$\Updownarrow(\mathrm{wDB});\ \Uparrow(\mathrm{rDB},\ \mathrm{wCDB});\ \Uparrow(\mathrm{rCDB},\ \mathrm{wDB});\ \Downarrow(\mathrm{rDB},\ \mathrm{wCDB});\ \Downarrow(\mathrm{rCDB},\ \mathrm{wDB});\ \Updownarrow(\mathrm{rDB})$$

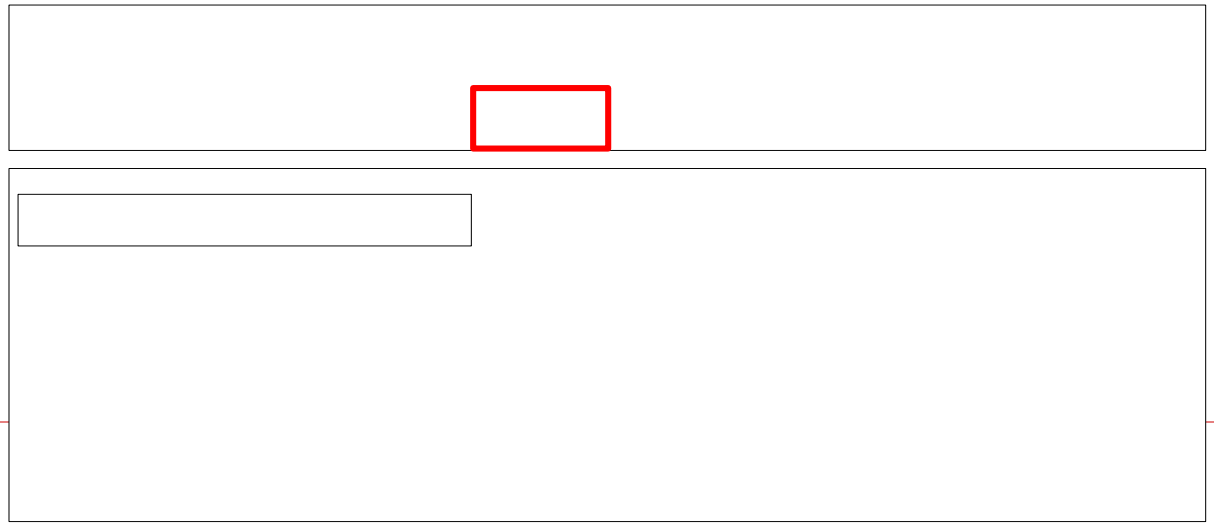
- Cache RAM cell test
  - Tag / Data RAM module

System Memory Mapping

- Virtual address translation
  - Access right
- Physical memory layout
  - Main memory
  - Memory-mapped I/O
  - Unused space
- Cache architecture
  - Virtual/physical index
  - Virtual/physical tag

Misalignment Problem

- Cause
  - Starting March sequence from a non-zero cache index
  - Cache size > page size
- OS-managed vs. Non-OS



Shielded Address

- Required for testing (high coverage)
  - March data background for tag RAM testing
  - Address for control logic testing
- Limited by system
  - Memory protection scheme
  - Physical memory layout
- Shielded page

System Protection

- Before SBST
  - Protect all shielded addresses
  - Minimize the alteration of system state
- During SBST
  - Detect and block all faulty effects
  - Prevent system from entering an unrecoverable status

Outline

- Introduction
- Challenge for on-line cache SBST
- Processor shield design
- Guardband calibration
- Case study : ARMv5 processor
- Conclusion

Processor Shield

- System level approach
  - Software framework
  - Hardware design for test (DFT)
- System protection
  - Current system states
  - Other processes
  - On-bus devices
- Shielded address redirection
- Faulty access block

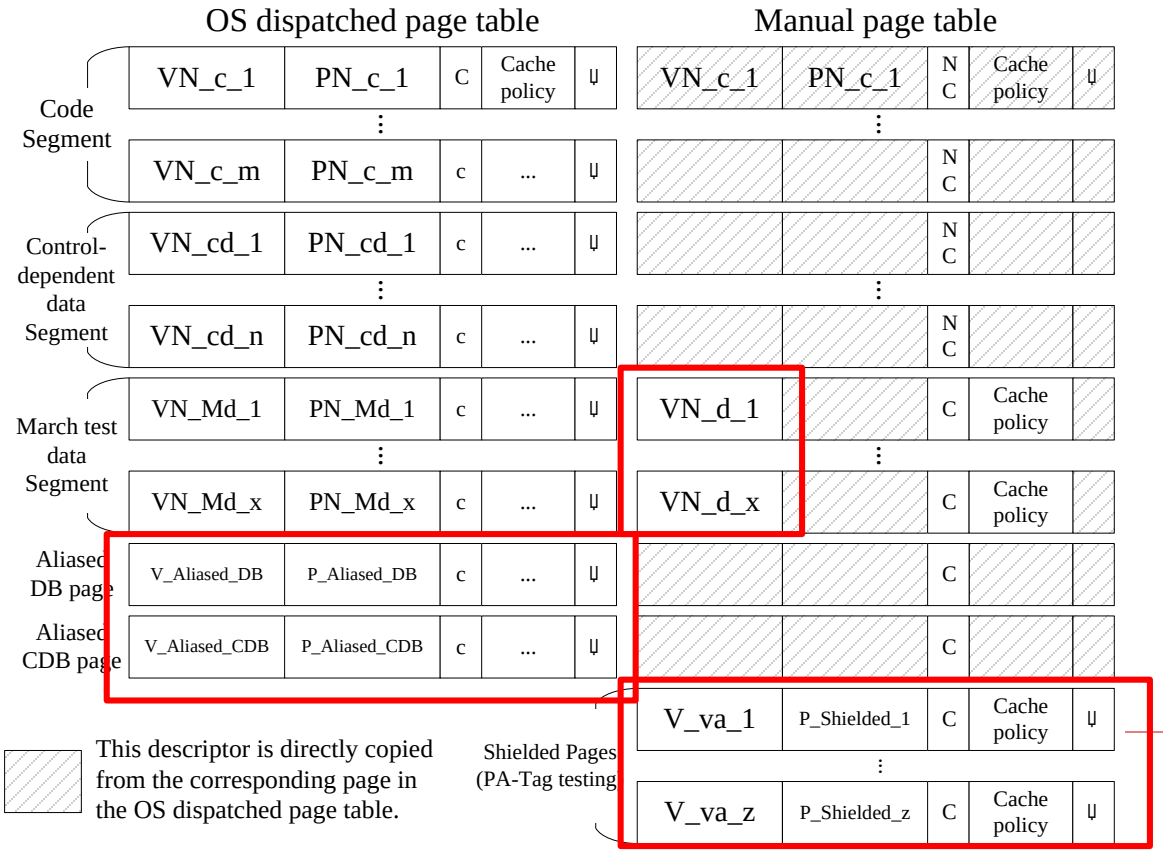
Software Framework

- System call implementation
- Testing environment initialization
  - Request free memory pages from OS
  - Prepare manual page table for testing
  - Back up process context
- Cache SBST body function
  - RAM modules
  - Control logic
- Process context recovery

System Call Execution Flow

Manual Page Table

- Get memory access right
- Resolve misalignment problem

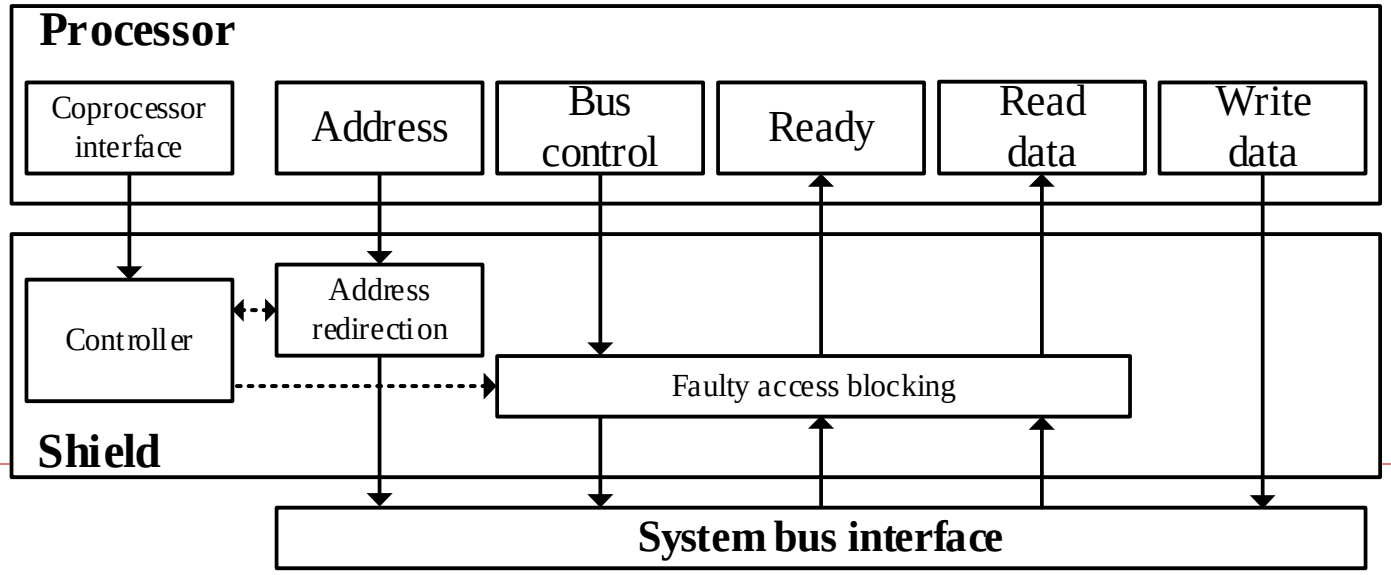


Physical Tag RAM Testing

- Write-back and write-allocate cache
- March write
  - Write a special datum to target cache line
- March read
  - Clean the target cache line
  - Read the same address in the next level memory without accessing cache
- Shielded address redirection
  - Aliased DB/CDB page
  - DFT hardware

DFT Hardware Design

- System bus wrapper
  - Redirect shielded page to aliased DB/CDB page
  - Block faulty access
- Coprocessor design / bus slave design

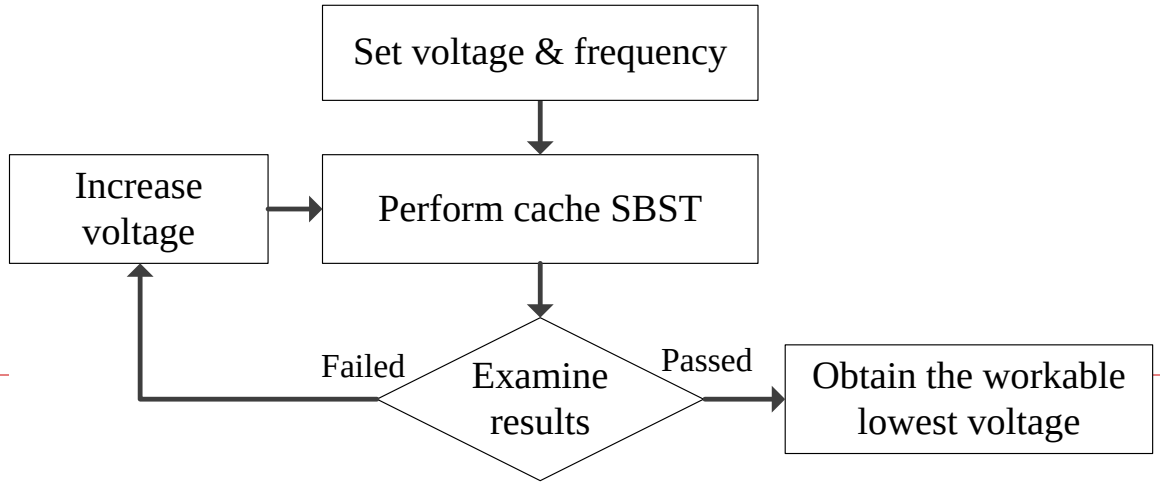


Outline

- Introduction
- Challenge for on-line cache SBST
- Processor shield design
- Guardband calibration
- Case study : ARMv5 processor
- Conclusion

Iterative Cache SBST Flow

- Obtain the workable lowest voltage for a specified frequency
- Cooperate with DVFS system
- Calibrate the required guardband for RAM module and control logic



Outline

- Introduction
- Challenge for on-line cache SBST
- Processor shield design
- Guardband calibration
- Case study : ARMv5 processor
- Conclusion

Experimental Environment

- ARMv5-compatible processor
  - 16KB direct-mapped data cache
  - Virtual tag/virtual index
    (a physical tag associated with each line)
  - Linux Kernel 2.6.33
- Cache control functions
  - Enable/disable, clear, write-back, write-through, write-allocate, write-around

Simulation and Result

- Control logic stuck-at fault coverage
  - Syntest Turboscan
  - 98.99% of stuck-at fault
- RAM module fault coverage
  - RAMSES simulator
  - March C- algorithm
  - 100% (virtual TAG, physical TAG and data)

Results (2)

- Hardware overhead
  - TSMC 40nm technology library, 1GHz
  - Additional latency: 0.06ns

	Core	Cache controller	Processor Shield DFT
Area (um^2)	70,880	2,731	1,488

- SBST process statistics

Target	Code size (KB)	Memory usage (KB)		Execution time (CPU cycle)
Data RAM	0.69	0.99	32 KB March test pages	327,432
Tag RAM	2.77	3.21		132,543
Phy. Tag RAM	5.24	6.17	8 KB Aliased Pages	157,331
Logic device	24.11	25.13		631,521
Total	32.81	75.5		1,248,737

Outline

- Introduction
- Challenge for on-line cache SBST
- Processor shield design
- Guardband calibration
- Case study : ARMv5 processor
- Conclusion

Conclusion

- On-line cache SBST issues
  - System memory mapping
  - Alteration to current system
  - Faulty effect isolation
- Processor shield design
  - Seamless process switch between SBST and OS kernel
- Required guardband calibration
  - Cooperation with DFVS system

Q&A