# Resilient Microprocessor Design for Dynamic Variation Tolerance

## Keith A. Bowman
## Circuit Research Lab, Intel

keith.a.bowman@intel.com

**June 8, 2010**

**UPC Seminar**

1

# Problem Statement:

➢ **Variability is one of the primary challenges in the semiconductor industry**

➢ **Adversely impacts performance, power, yield, reliability, & time-to-market**

# Focus Area:

**1) Resilient design for dynamic variation tolerance**

# Outline

- **Review of Static Variations**

- **Resilient Microprocessor Core**

- **Error-Detection & Recovery Circuits**

- **Measurement Results**

- **Conclusion**

- **Future Research**

# Static Variations

- **Technology trends amplify microprocessor performance & power variability**

- **Static Variations:**

  - **Within-die impacts $F_{MAX}$ mean & leakage median**

  - **Die-to-die impacts $F_{MAX}$ & leakage variances**

- **Adaptive circuits mitigate the impact of static variations on performance & power**
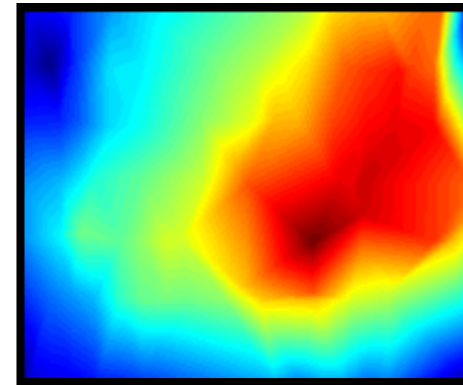
# Outline

- Review of Static Variations

- **Resilient Microprocessor Core**

- Error-Detection & Recovery Circuits

- Measurement Results

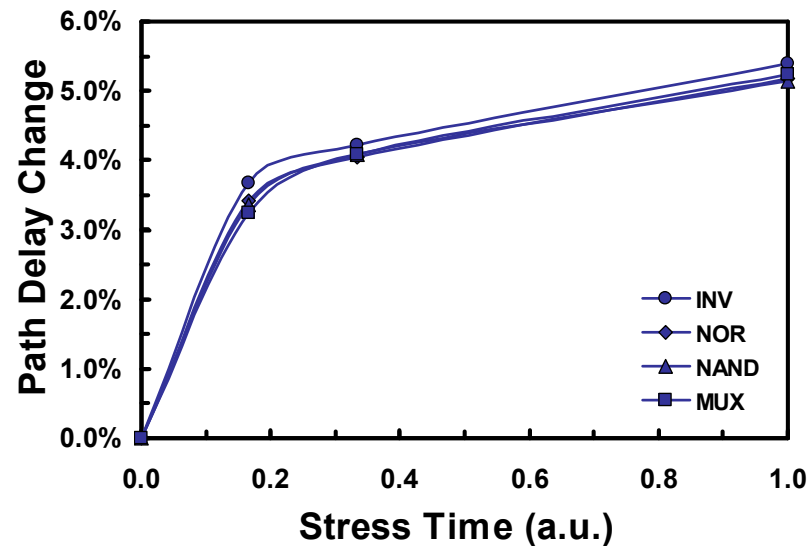- Conclusion

- Future Research

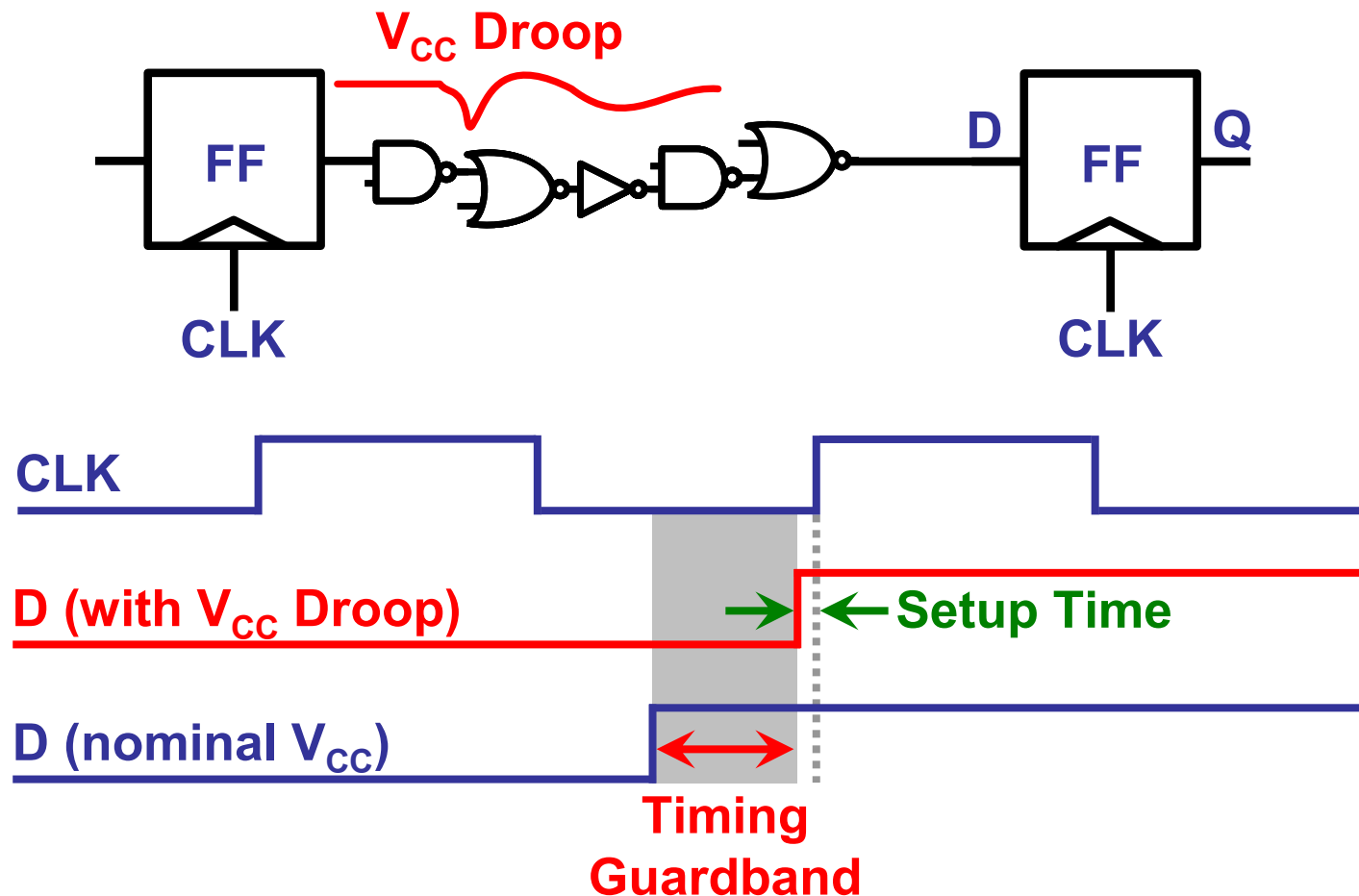# Dynamic Variations

### $V_{CC}$ Droop
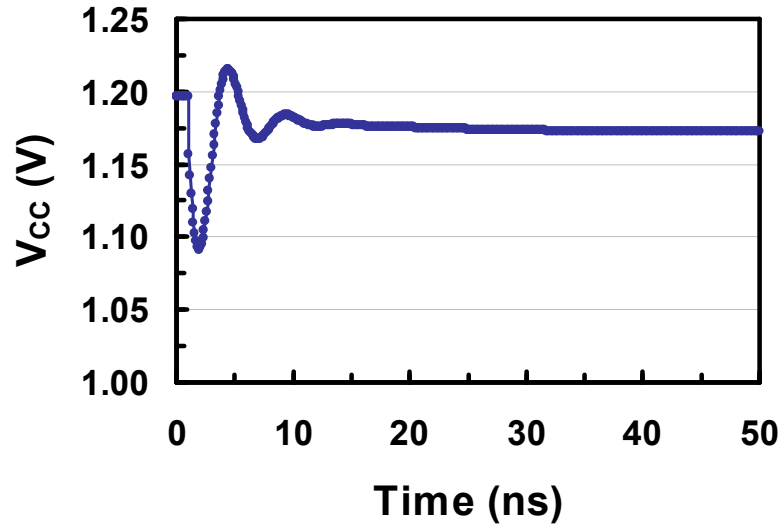
### Temperature

### Aging

# Impact of Dynamic Variations on Conventional Design



- Guardbands required to ensure correct operation within the presence of dynamic variations

# Resilient Design

- **Operate clock frequency ($F_{CLK}$) based on nominal conditions**

- **Resilient circuits detect and correct timing errors due to infrequent dynamic variations**

- **Throughput and energy benefits result from mitigating guardbands**

# Microprocessor Features

- **32-Bit Synthesized Core**

  - ➢ **Open-source RISC-style design**

  - ➢ **7-stage in-order pipeline**

  - ➢ **Modified with resilient and adaptive circuits**

- **16KB Instruction & Data Caches**
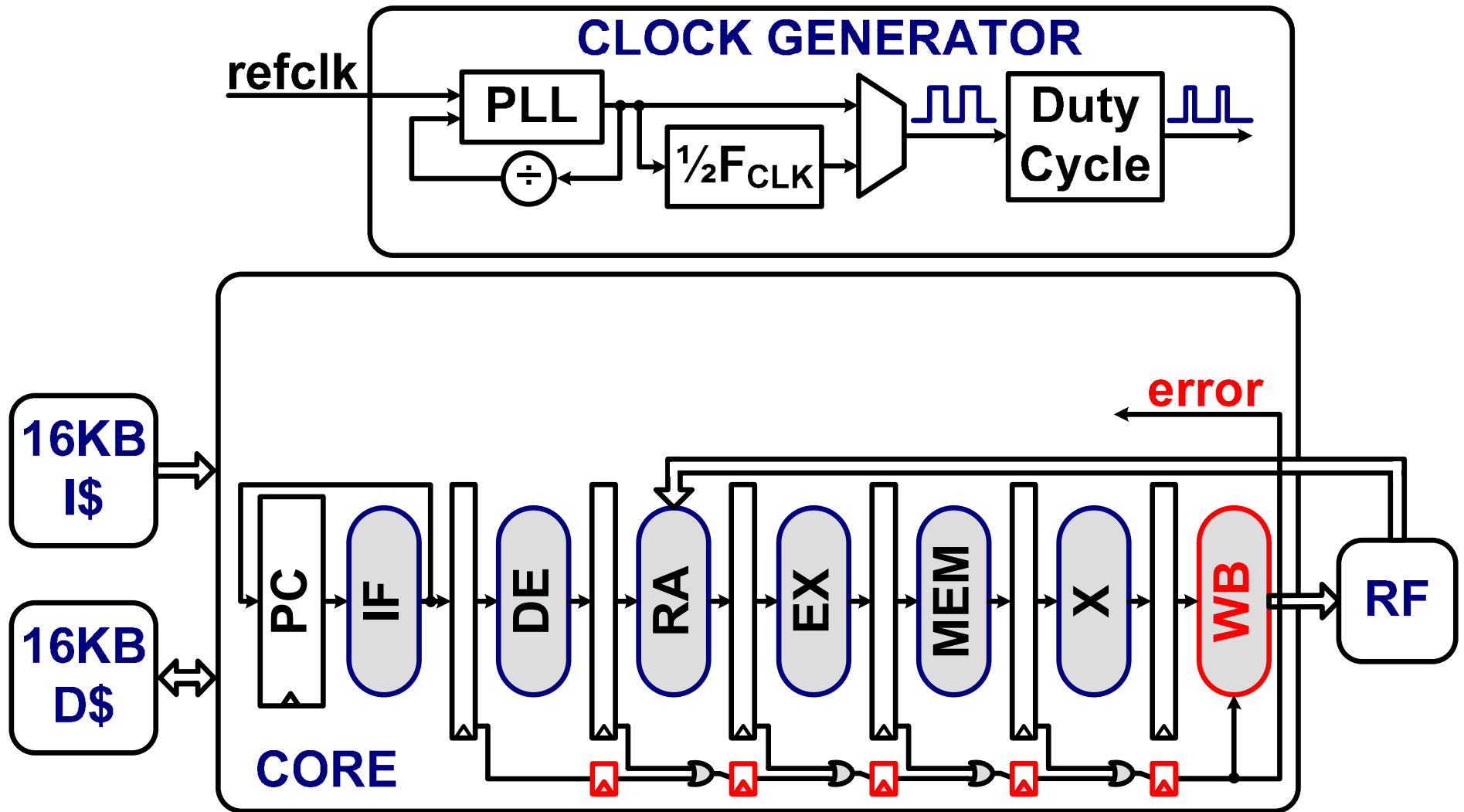
- **PLL-Based Clock Generator**

# Resilient & Adaptive Circuits

- **Error Detection:**

  **1)** Error-detection sequential (EDS)

  **2)** Tunable replica circuit (TRC)

- **Error Control Unit (ECU) for Recovery:**

  **1)** Instruction replay at $\frac{1}{2}F_{CLK}$

  **2)** Multiple issue instruction replay at $F_{CLK}$

- **Adaptive Clock Controller:**

  ➤ Adjust $F_{CLK}$ based on recovery cycles to maximize performance during persistent variations
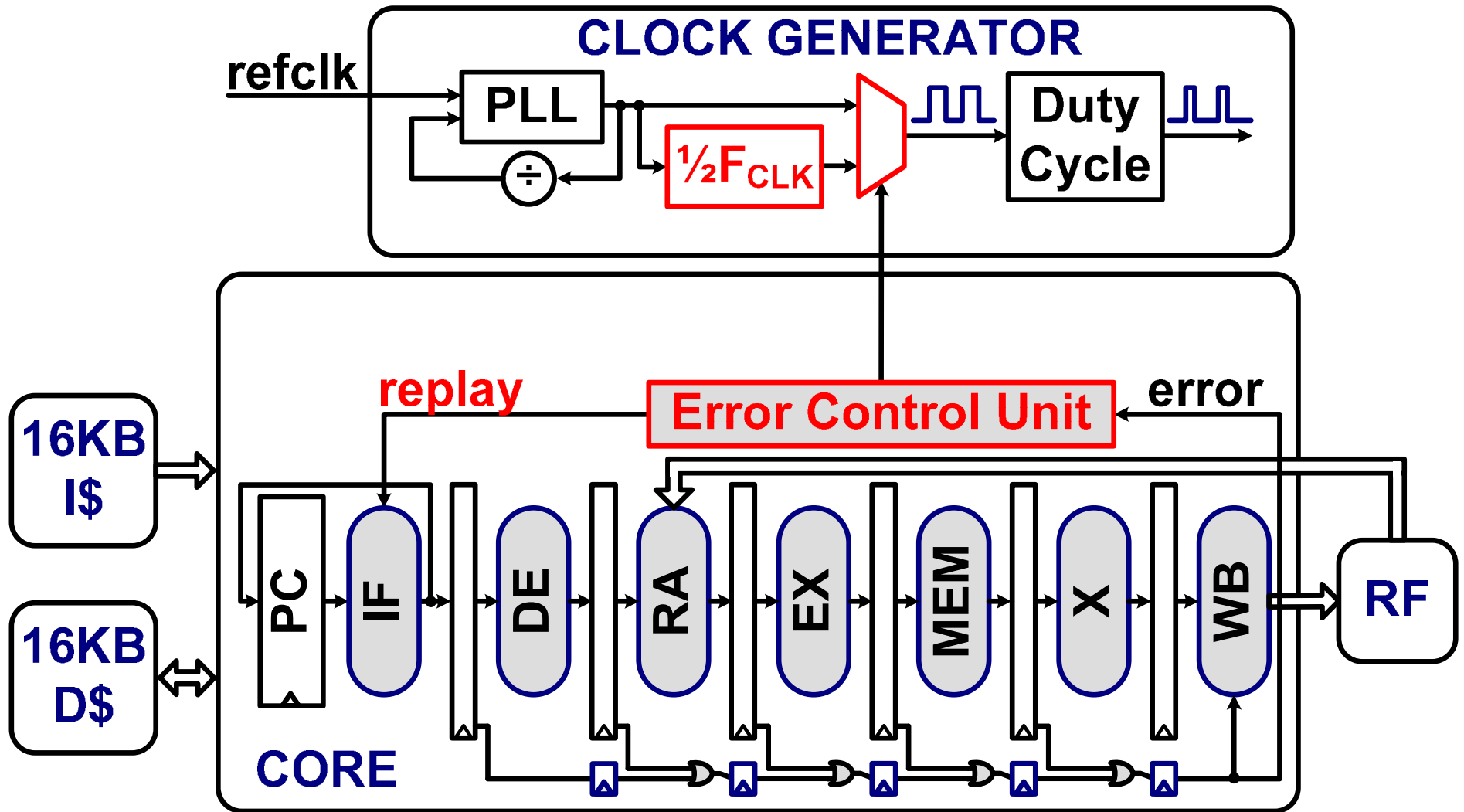
# Microprocessor Core Overview
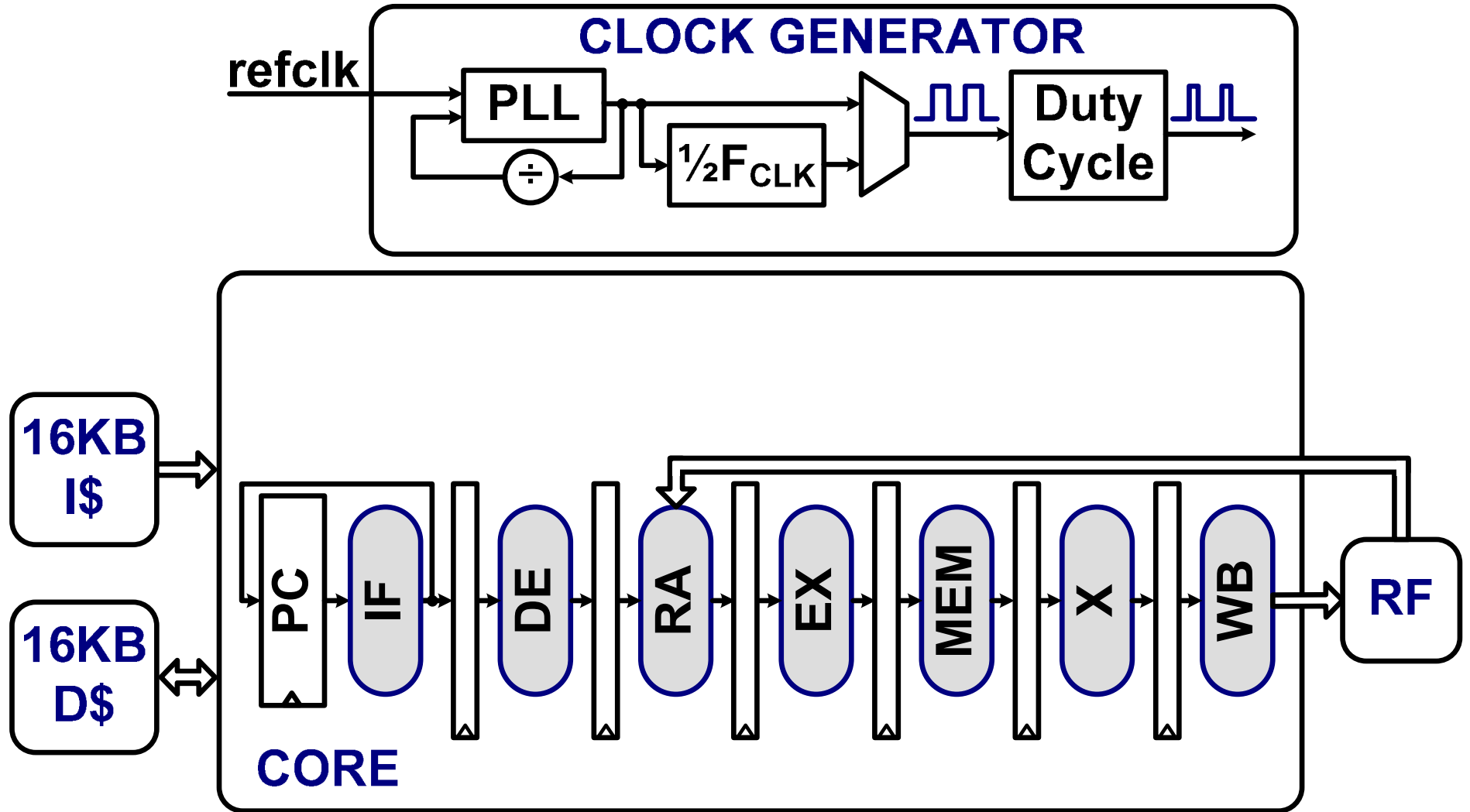
# Microprocessor Core Overview



> Errors are pipelined to write-back (WB) stage to invalidate erroneous instructions

12

# Microprocessor Core Overview



> **Error-control unit (ECU) enables recovery**

# Microprocessor Core Overview
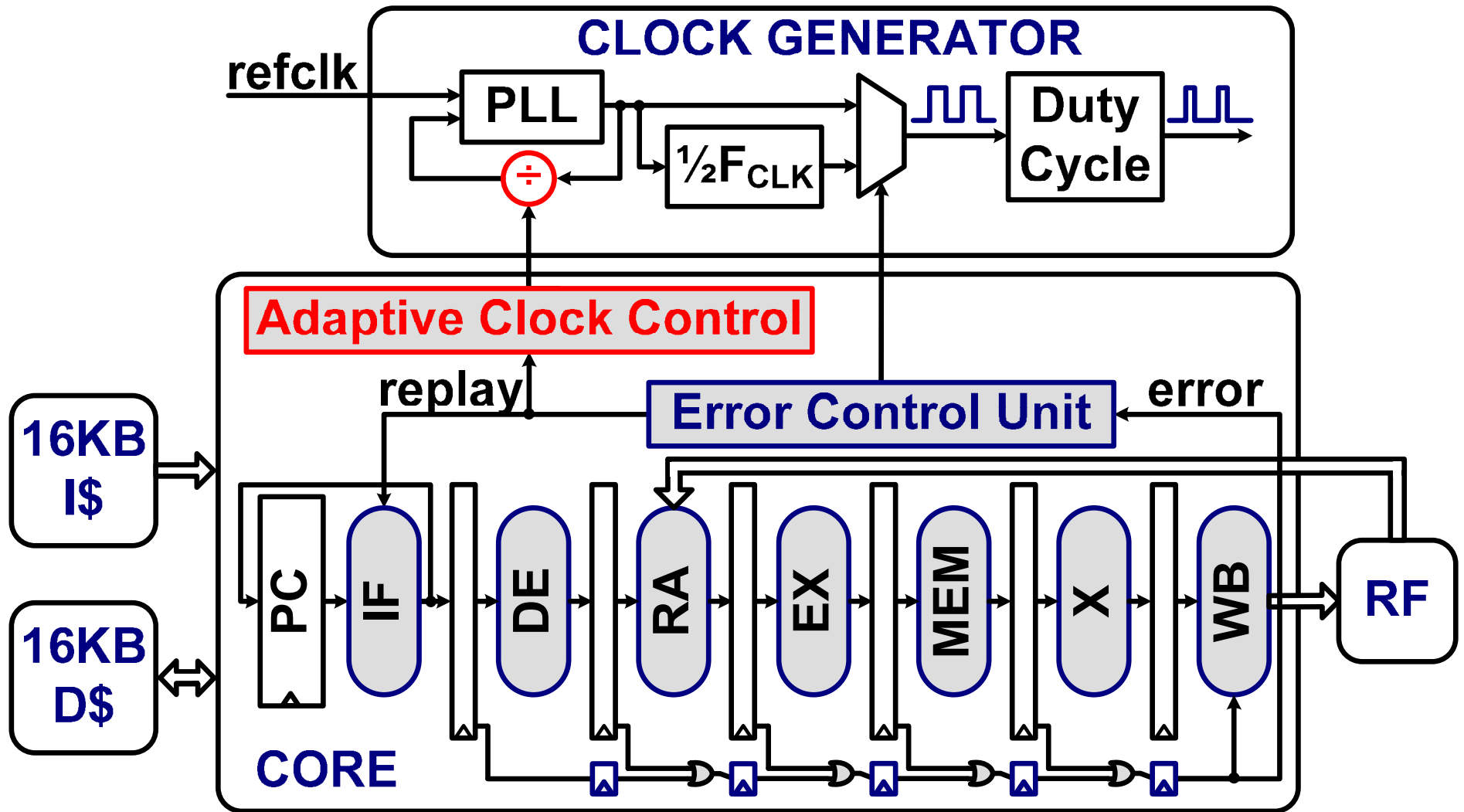


➢ **Adaptive clock control enables dynamic $F_{CLK}$ change**

# Outline

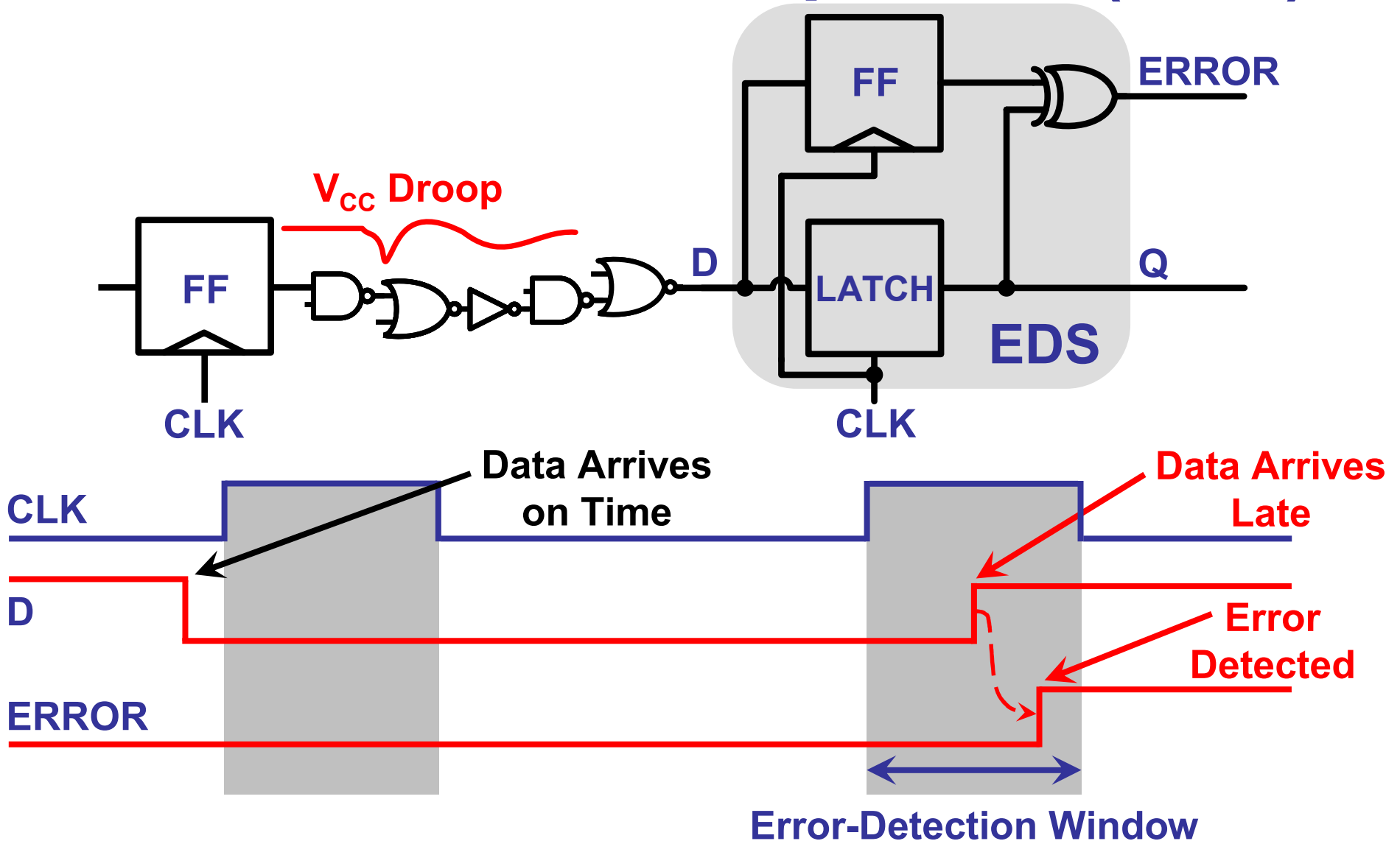- Review of Static Variations

- Resilient Microprocessor Core

- **Error-Detection & Recovery Circuits**

- Measurement Results

- Conclusion

- Future Research

# Error-Detection Circuits

**1) Error-Detection Sequential (EDS)**

**2) Tunable Replica Circuit (TRC)**

# Error-Detection Sequential (EDS)



> **Detect timing error within error-detection window**

K. Bowman, et al., *JSSC*, 2009.

# Error-Detection Sequential (EDS)

## Trade-off: Max-Delay ($t_{MAX}$) vs Min-Delay ($t_{MIN}$)
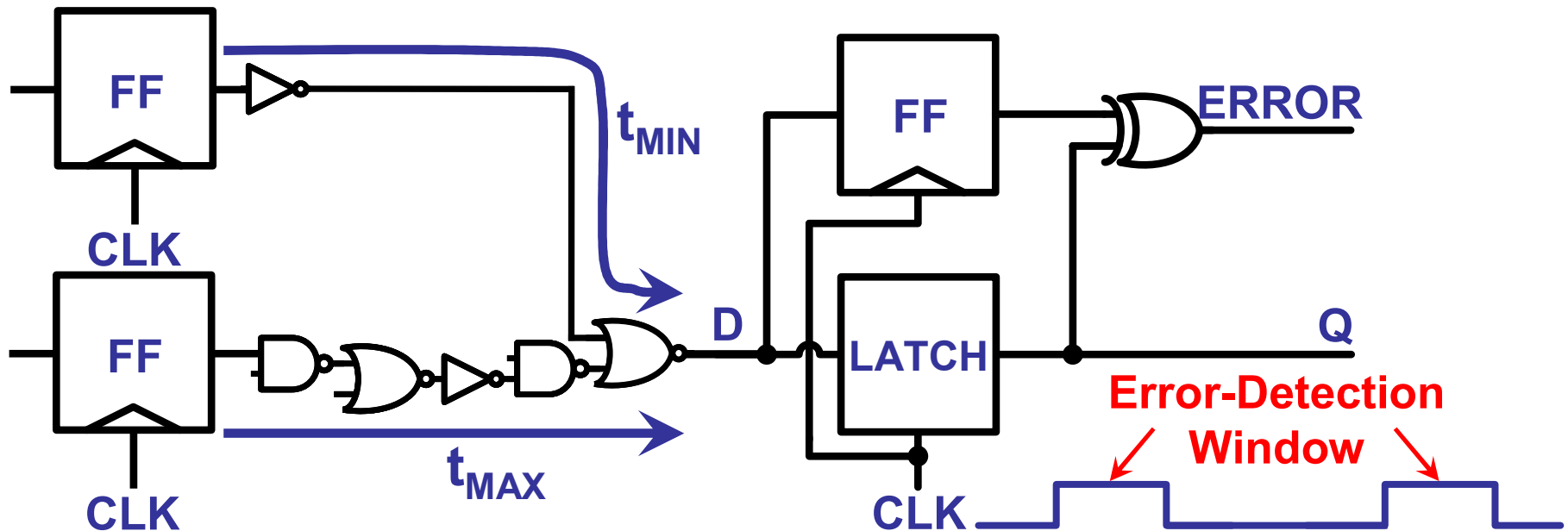


- **Min-delay penalty increases by error-detection window**

- **Clock duty-cycle control required to maintain constant high-phase delay during low and high $F_{CLK}$**

# EDS Circuits

## Double Sampling (Razor I) [1]-[3]

LATCH
ERROR
D
FF
Q
CLK

## Transition Detector with Time Borrowing (TDTB) [5]

ERROR
D
LATCH
Q
CLK

## Razor II [4]

CLK
D
Q
N
Detection Clock (DC) generator
Transition Detector (TD)
ERROR

DC-TG $V_{dd}$
CLK
DC

ERROR reset
ERROR
N
$d_0$  $d_1$
$d_0$  $d_1$
$d_2$  $d_3$
$d_2$  $d_3$
DC
TD-TG $V_{dd}$

## Double Sampling with Time Borrowing (DSTB) [5]

FF
ERROR
D
LATCH
Q
CLK
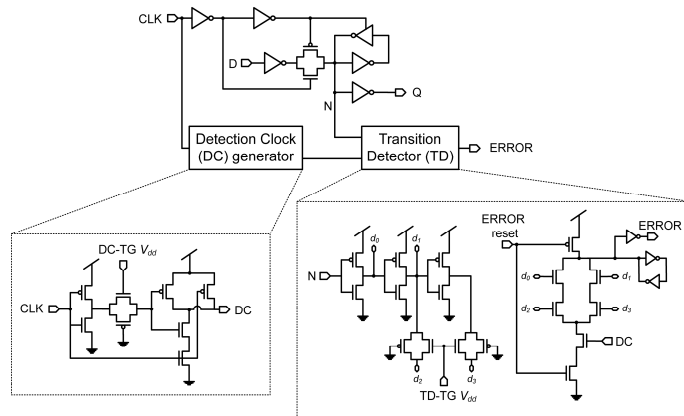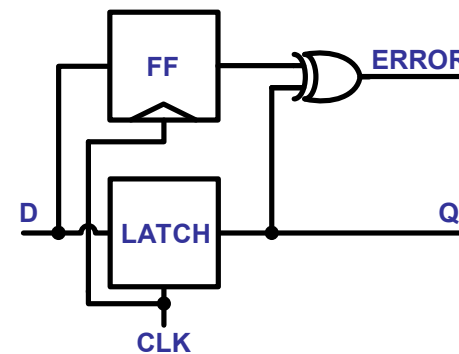
[1] P. Franco, et al., *VLSI Test Symp.*, 1994.
[2] M. Nicolaidis, *VLSI Test Symp.*, 1999.
[3] D. Ernst, et al., *MICRO*, 2003.
[4] S. Das, et al., *JSCC*, 2009.
[5] K. Bowman, et al., *JSSC*, 2009.

# Error-Detection Sequential (EDS)
## Implementation
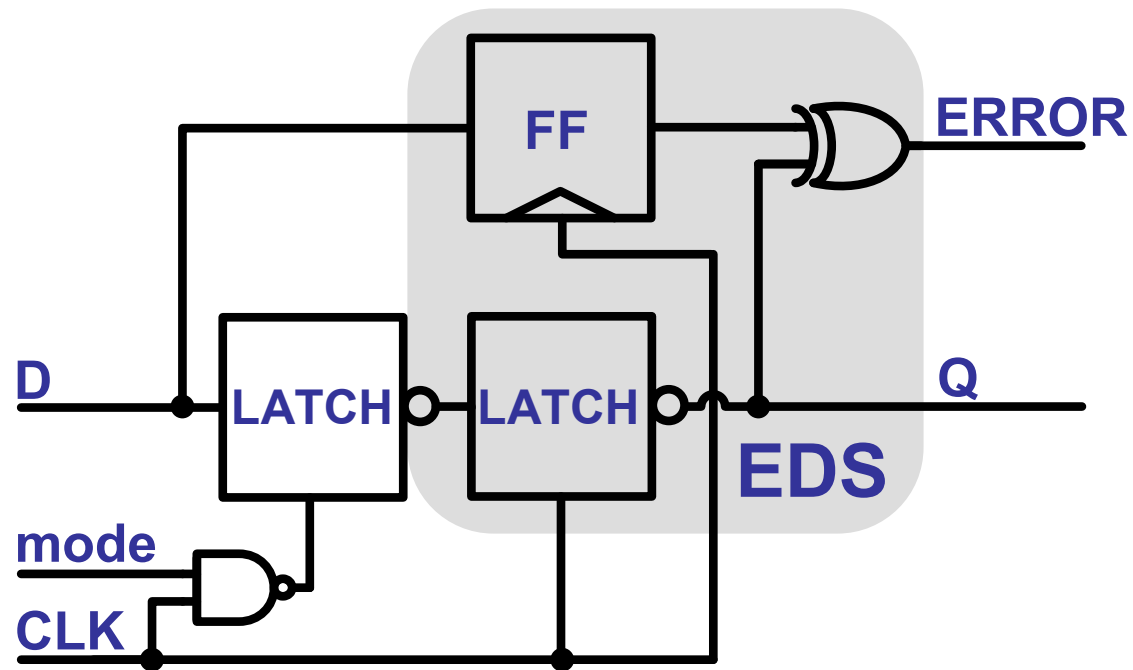


- **Contains additional scan-enabled latch for testing**
  - ➤ **mode=0: EDS**
  - ➤ **mode=1: FF**

# Error-Detection Sequential (EDS)
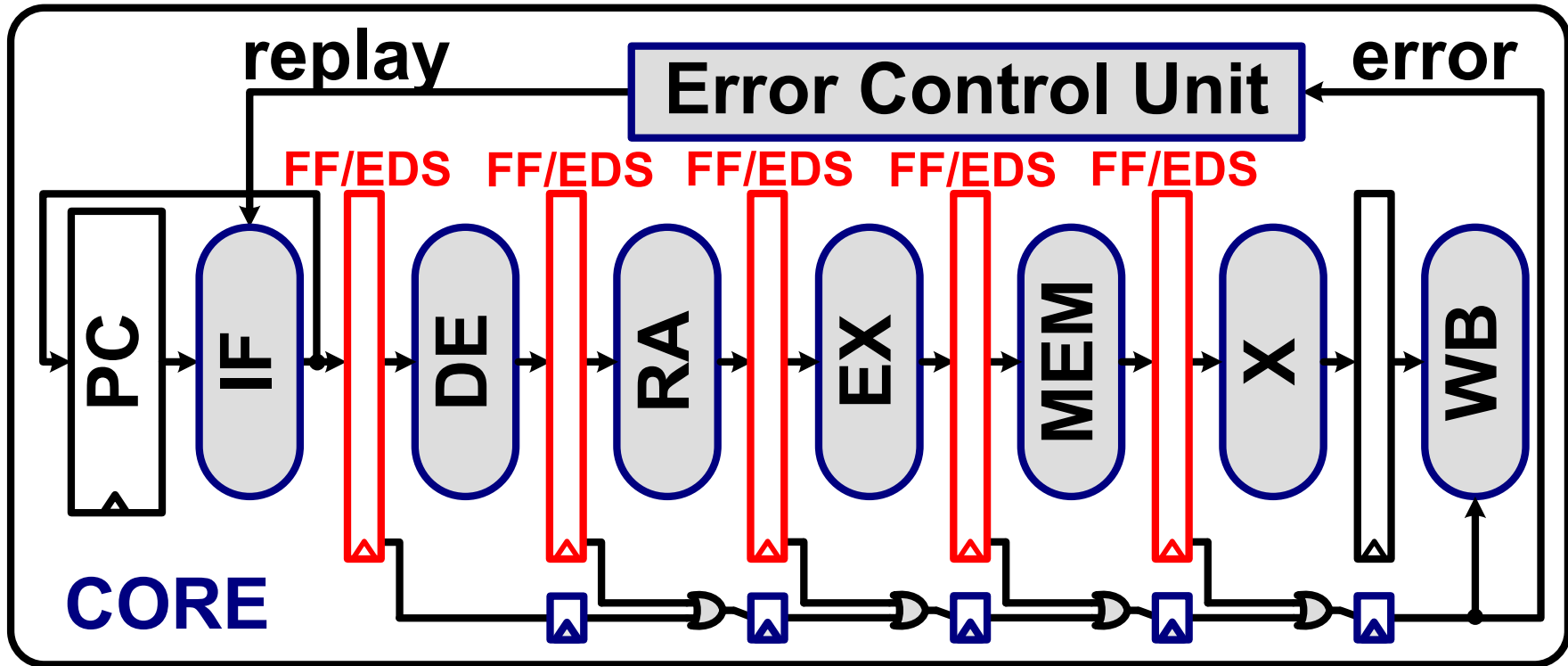


> EDS assigned during synthesis convergence

> EDS embedded in critical paths

> EDS inserted in 12% of core sequentials

# Error-Detection Circuits

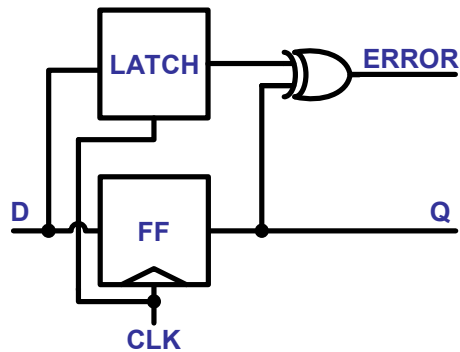1) Error-Detection Sequential (EDS)

2) **Tunable Replica Circuit (TRC)**

# Tunable Replica Circuit (TRC)
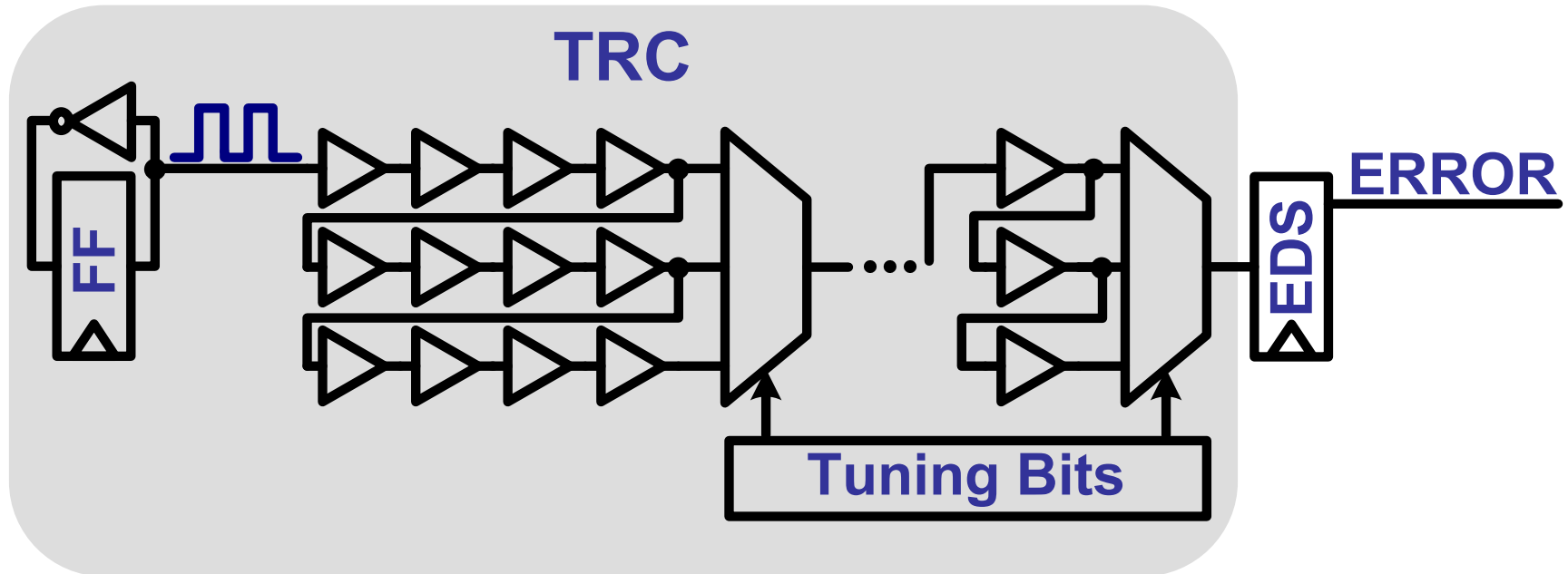


- ➤ **TRC monitors critical path delays**

- ➤ **Non-intrusive design**

J. Tschanz, et al., *Symp. VLSI Circuits*, 2009.

# Tunable Replica Circuit (TRC)
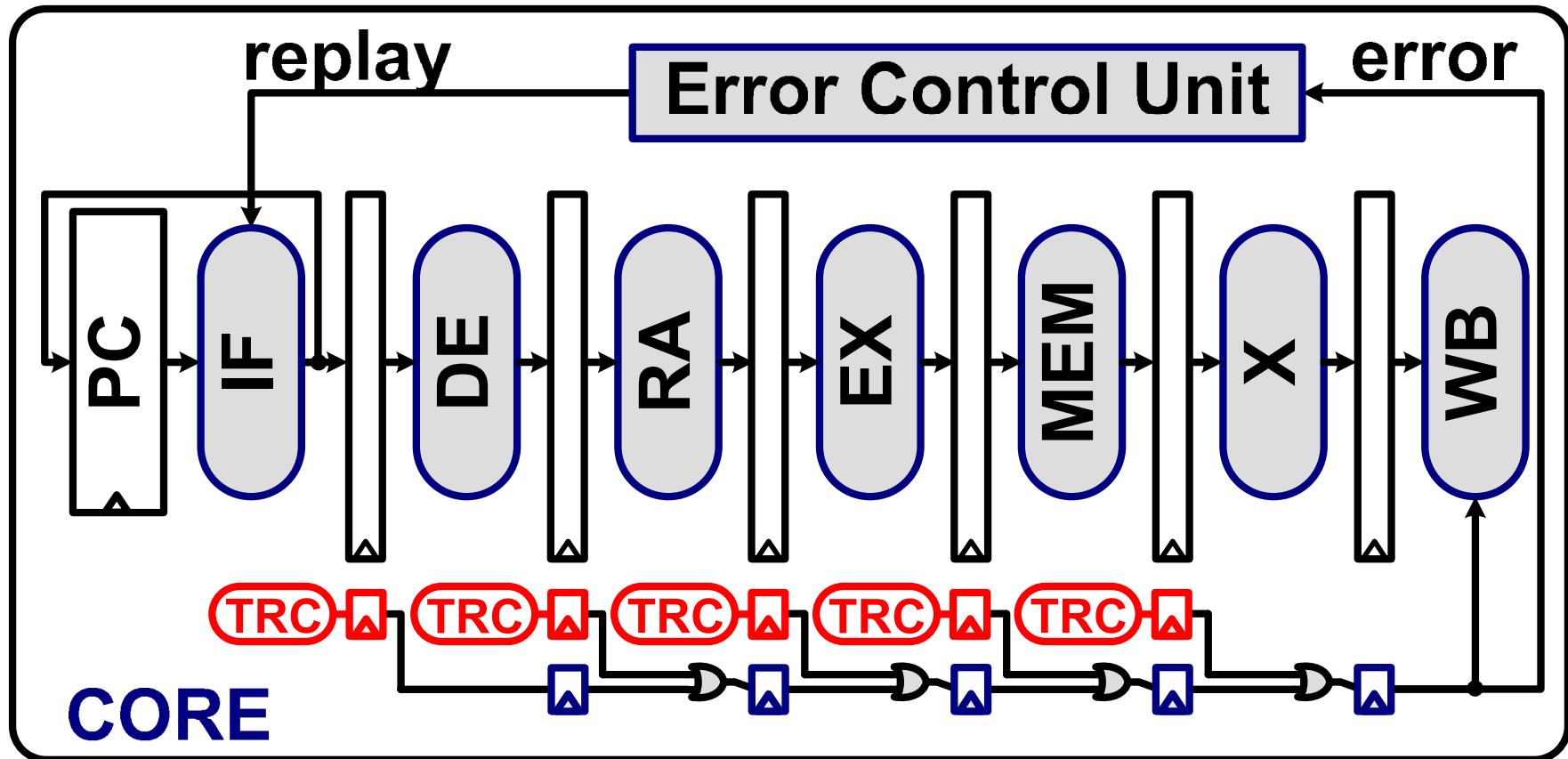


- ➢ TRC tuned to track critical paths per pipeline stage

- ➢ TRC must always fail if any critical path fails

- ➢ TRC error initiates pipeline error recovery

24

# EDS & TRC Overheads

| Circuit Blocks | EDS | TRC |
| --- | --- | --- |
| Error Detection & Accumulation Area Overhead | 2.2% | 0.8% |
| ECU & Clock Control Area Overhead | 1.4% | 1.4% |
| Min-Delay Buffer Insertion Area Overhead | 0.2% | – |
| Total Area Overhead | 3.8% | 2.2% |
| Total Power Overhead (iso-$F_{CLK}$, iso-$V_{CC}$) | 0.9% | 0.6% |

# Error-Recovery Circuits

**1) Instruction Replay at $\frac{1}{2}F_{CLK}$**

➢ **Clock divider generates $\frac{1}{2}F_{CLK}$ without PLL re-lock**

➢ **Clock high-phase delay remains unchanged**

**2) Multiple Issue Instruction Replay at $F_{CLK}$**

➢ **Does not require clock control**

➢ **Issue _replica instructions_ to setup pipeline registers**

➢ **Last issue is a valid instruction**

# Multiple Issue (MI) Replay

## Clock Cycle

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IF | I1 | I2 | I3 | I4 | I5 | | | | | | | |
| DE | | I1 | I2 | I3 | I4 | | | | | | | |
| RA | | | I1 | I2 | I3 | | | | | | | |
| EX | | | | I1 | I2 | | | | | | | |
| MEM | | | | | I1 | | | | | | | |
| X | | | | | | | | | | | | |
| WB | | | | | | | | | | | | |

**Pipeline Stage**

**I2 Error in EX Stage**

**1)** Error occurs on instruction I2 in EX pipeline stage

27

# Multiple Issue (MI) Replay

## Clock Cycle

| Pipeline Stage | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IF | I1 | I2 | I3 | I4 | I5 | I6 | I7 | I8 | | | | |
| DE | | I1 | I2 | I3 | I4 | I5 | I6 | I7 | I8 | | | |
| RA | | | I1 | I2 | I3 | I4 | I5 | I6 | I7 | I8 | | |
| EX | | | | I1 | I2 | I3 | I4 | I5 | I6 | I7 | I8 | |
| MEM | | | | | I1 | I2 | I3 | I4 | I5 | I6 | I7 | I8 |
| X | | | | | | I1 | I2 | I3 | I4 | I5 | I6 | I7 |
| WB | | | | | | | I1 | I2 | I3 | I4 | I5 | I6 |

**I2 Error in EX Stage**

**Invalid: Does Not Affect Architecture State**
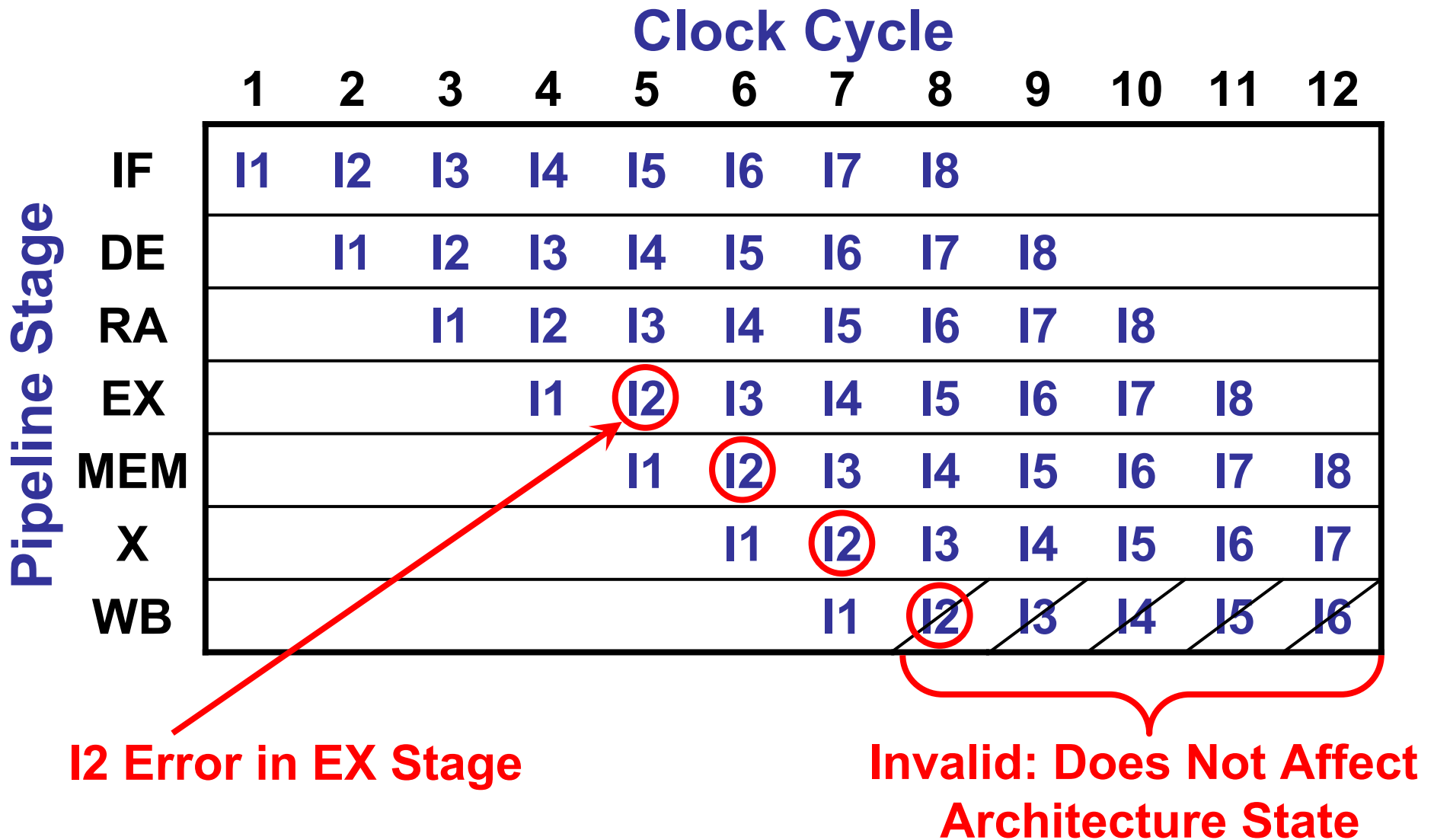
**2) Invalidate errant instruction and subsequent instructions** 28

# Multiple Issue (MI) Replay

## Clock Cycle

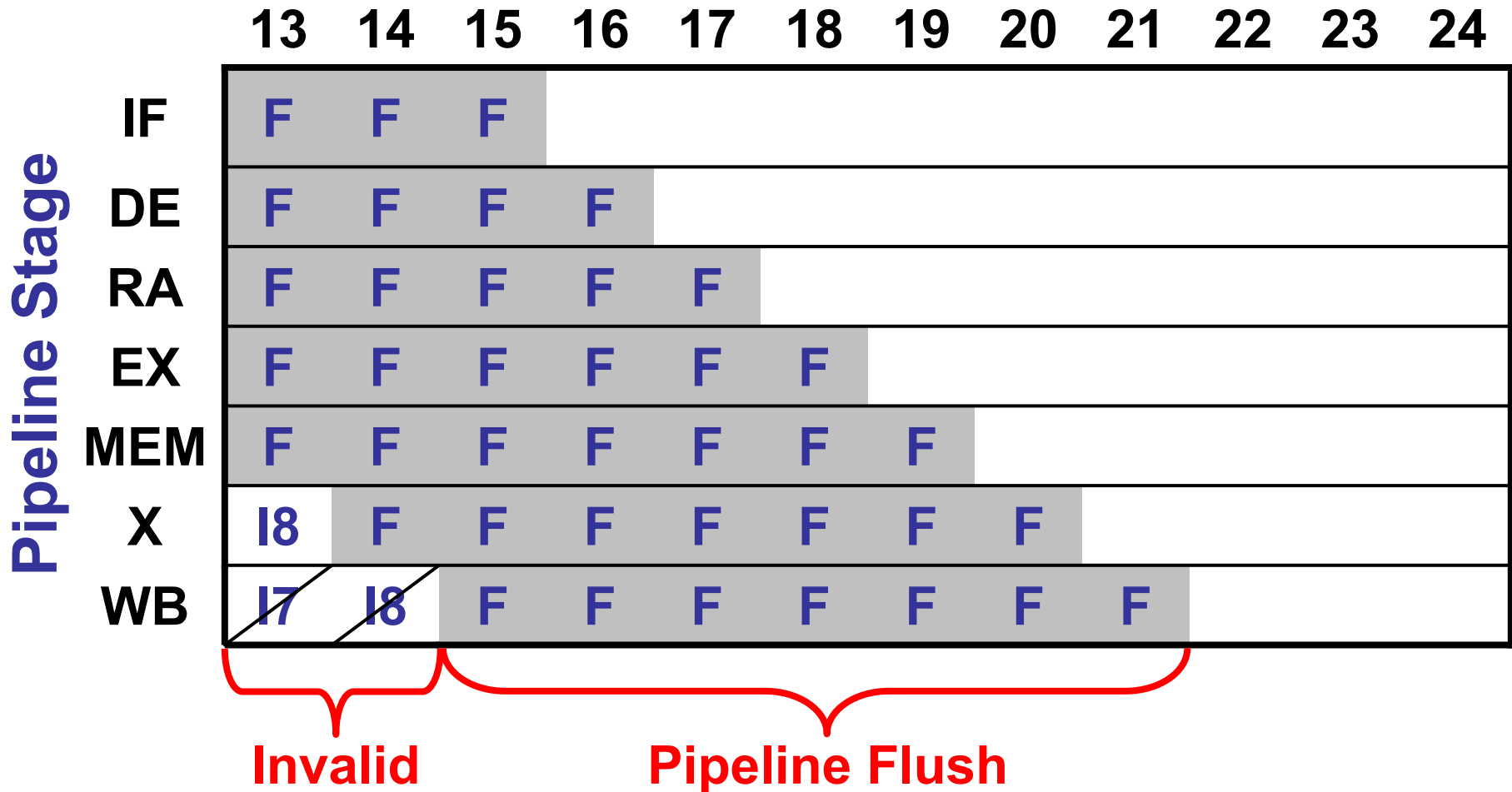| Pipeline Stage | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IF | I1 | I2 | I3 | I4 | I5 | I6 | I7 | I8 | F | F | F | F |
| DE | | I1 | I2 | I3 | I4 | I5 | I6 | I7 | I8 | F | F | F |
| RA | | | I1 | I2 | I3 | I4 | I5 | I6 | I7 | I8 | F | F |
| EX | | | | I1 | I2 | I3 | I4 | I5 | I6 | I7 | I8 | F |
| MEM | | | | | I1 | I2 | I3 | I4 | I5 | I6 | I7 | I8 |
| X | | | | | | I1 | I2 | I3 | I4 | I5 | I6 | I7 |
| WB | | | | | | | I1 | I2 | I3 | I4 | I5 | I6 |

I2 Error in EX Stage

Invalid: Does Not Affect Architecture State

3) Flush pipeline
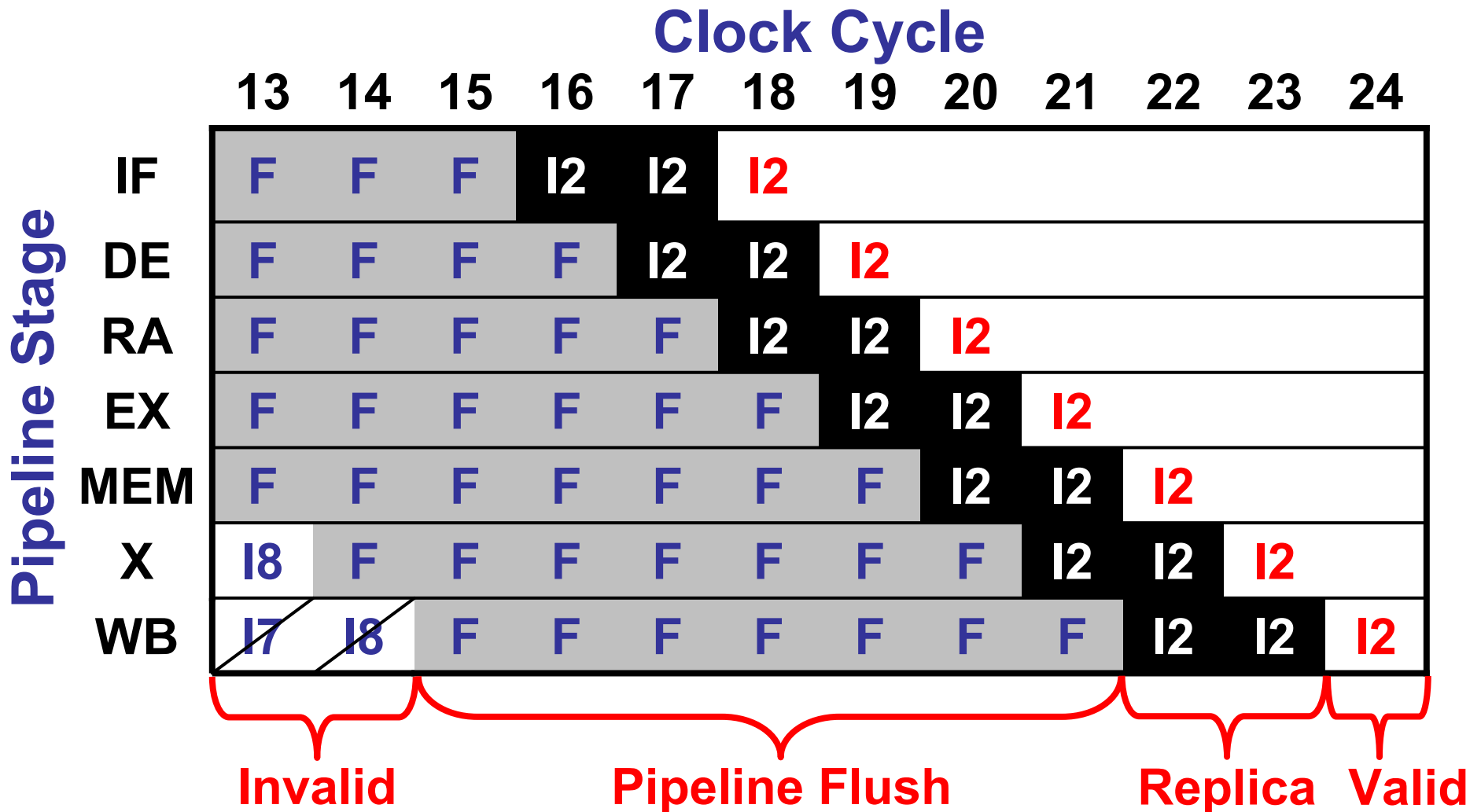
29

# Multiple Issue (MI) Replay

# Multiple Issue (MI) Replay

## Clock Cycle



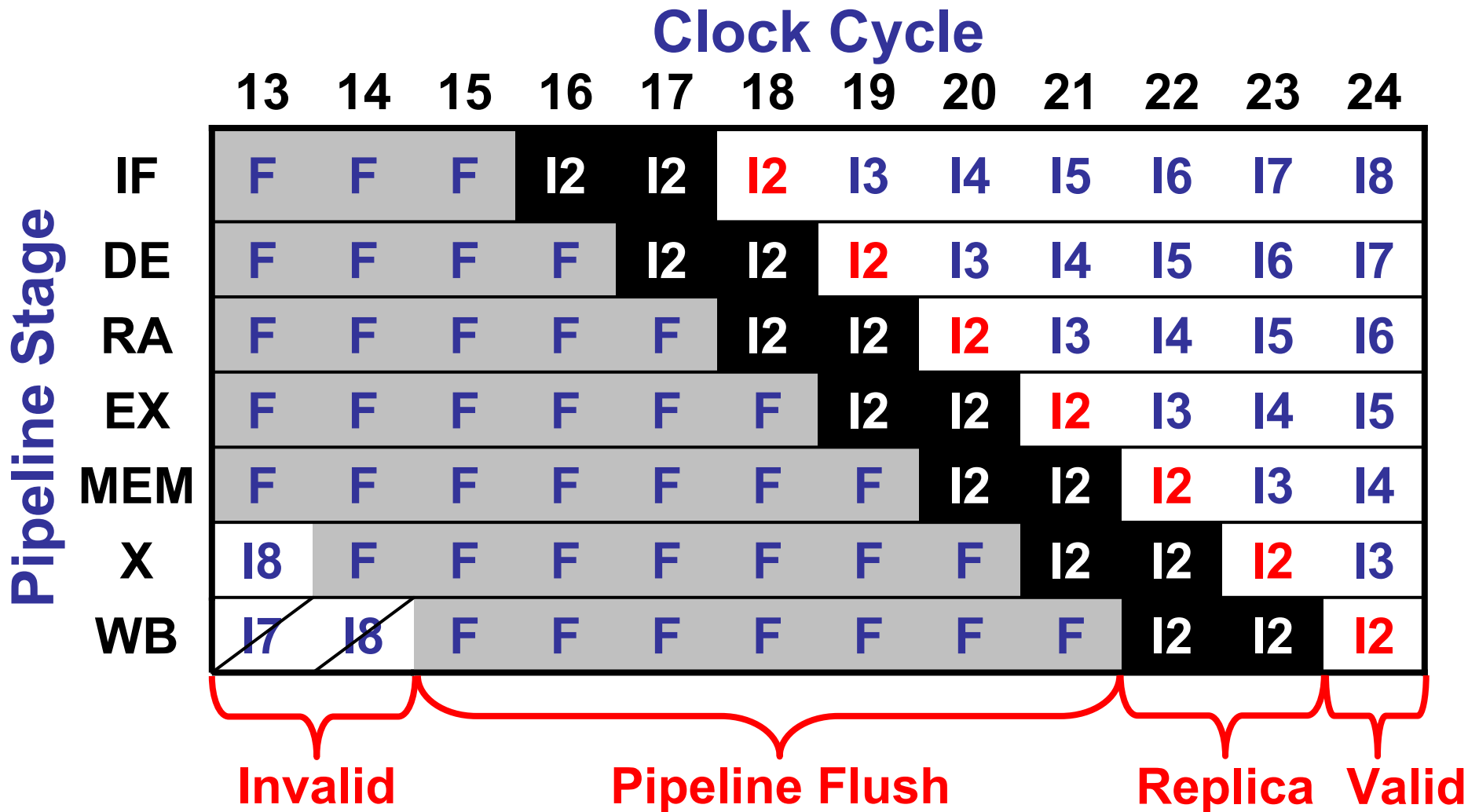| Pipeline Stage | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IF | F | F | F | I2 | I2 | I2 | | | | | | |
| DE | F | F | F | F | I2 | I2 | I2 | | | | | |
| RA | F | F | F | F | F | I2 | I2 | I2 | | | | |
| EX | F | F | F | F | F | F | I2 | I2 | I2 | | | |
| MEM | F | F | F | F | F | F | F | I2 | I2 | I2 | | |
| X | I8 | F | F | F | F | F | F | F | I2 | I2 | I2 | |
| WB | I7 | I8 | F | F | F | F | F | F | F | I2 | I2 | I2 |

**Invalid**  **Pipeline Flush**  **Replica**  **Valid**

**4)** Issue errant instruction N times: N-1 issues setup pipeline register values; Nth issue may change architecture state  31

# Multiple Issue (MI) Replay

## Clock Cycle

| Pipeline Stage | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IF | F | F | F | I2 | I2 | I2 | I3 | I4 | I5 | I6 | I7 | I8 |
| DE | F | F | F | F | I2 | I2 | I2 | I3 | I4 | I5 | I6 | I7 |
| RA | F | F | F | F | F | I2 | I2 | I2 | I3 | I4 | I5 | I6 |
| EX | F | F | F | F | F | F | I2 | I2 | I2 | I3 | I4 | I5 |
| MEM | F | F | F | F | F | F | F | I2 | I2 | I2 | I3 | I4 |
| X | I8 | F | F | F | F | F | F | F | I2 | I2 | I2 | I3 |
| WB | I7 | I8 | F | F | F | F | F | F | F | I2 | I2 | I2 |

Invalid     Pipeline Flush     Replica    Valid

**4) Issue errant instruction N times: N-1 issues setup pipeline register values; Nth issue may change architecture state** 32

# Outline

- Review of Static Variations

- Resilient Microprocessor Core

- Error-Detection & Recovery Circuits

- **Measurement Results**

- Conclusion

- Future Research
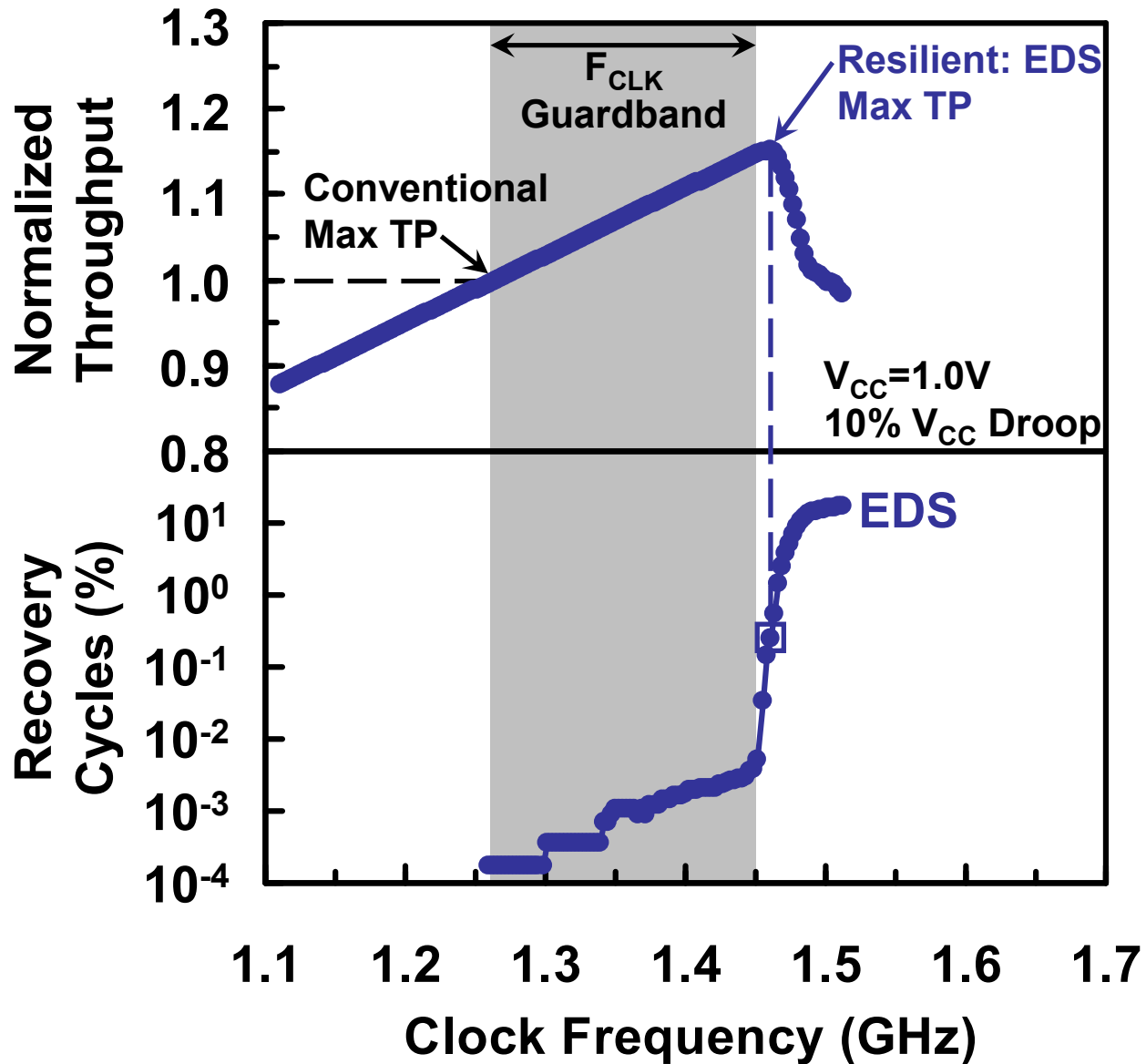
# Characteristics & Measurements

| Technology | 45nm CMOS |
|---|---|
| Die Area | 13.64 mm$^2$ |
| Core Area | 0.39 mm$^2$ |
| Core F$_{MAX}$ | 1.45GHz at 1.0V |
| Core Power | 135mW at 1.0V |
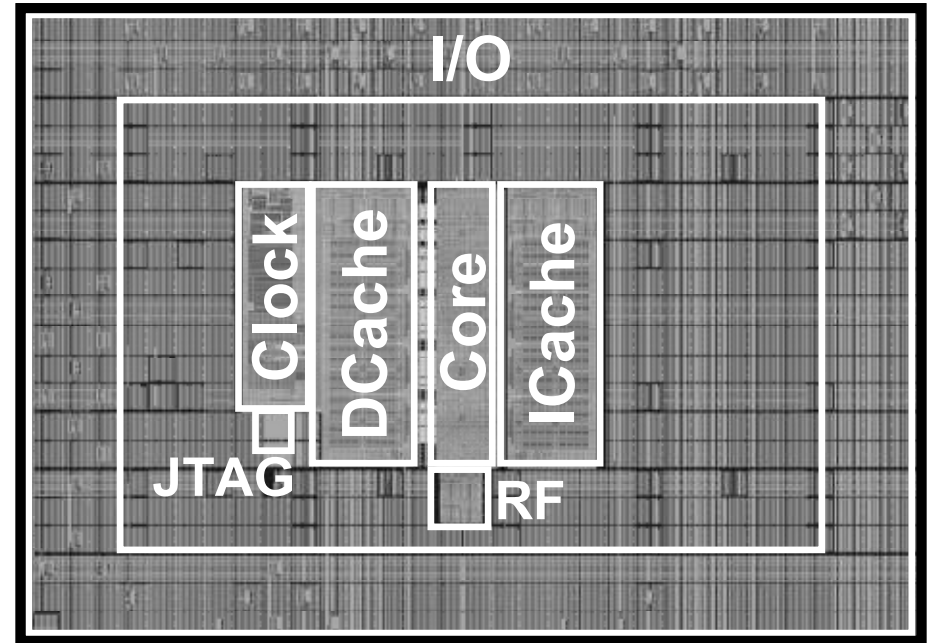


- ➤ **Programs compiled from C code**
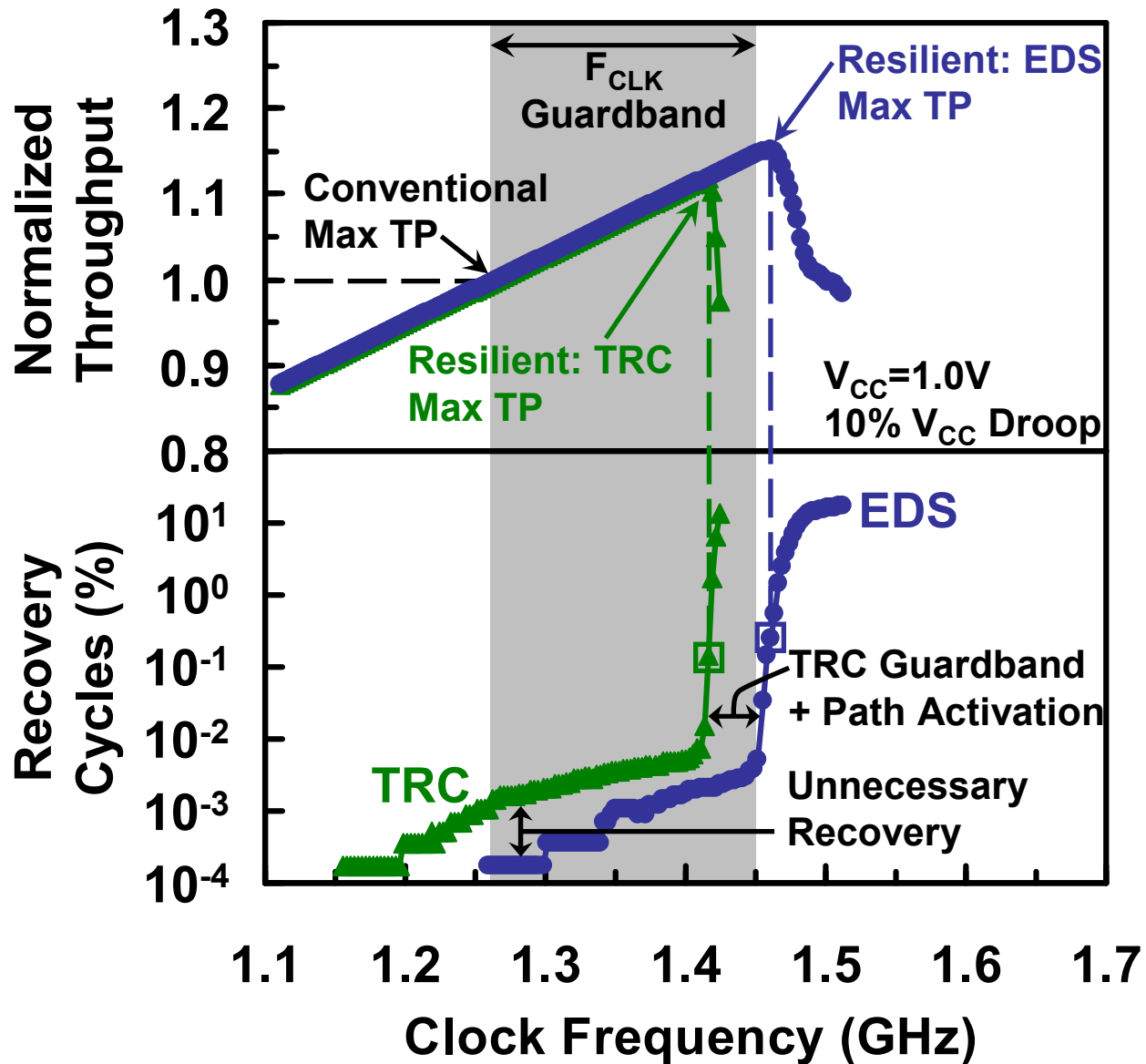
- ➤ **Caches and settings loaded via JTAG scan**

# Measured Throughput (TP) vs $F_{CLK}$



➤ **16% TP gain with EDS**

# Measured Throughput (TP) vs $F_{CLK}$



➤ **16% TP gain with EDS**        ➤ **12% TP gain with TRC**   36

# Measured Throughput Gain vs Application
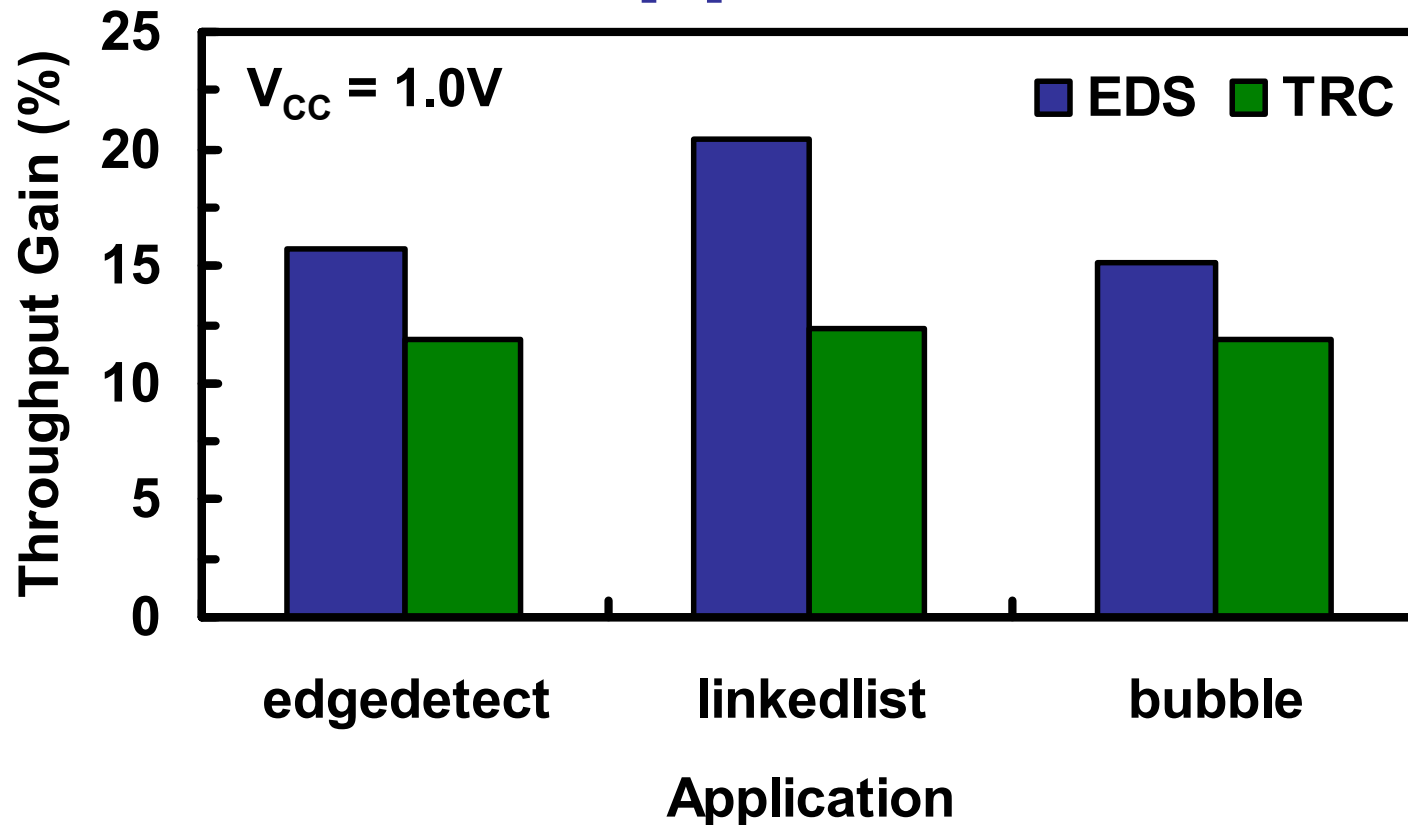


- EDS exploits path-activation differences across programs
- EDS throughput benefits range from 15% to 20%
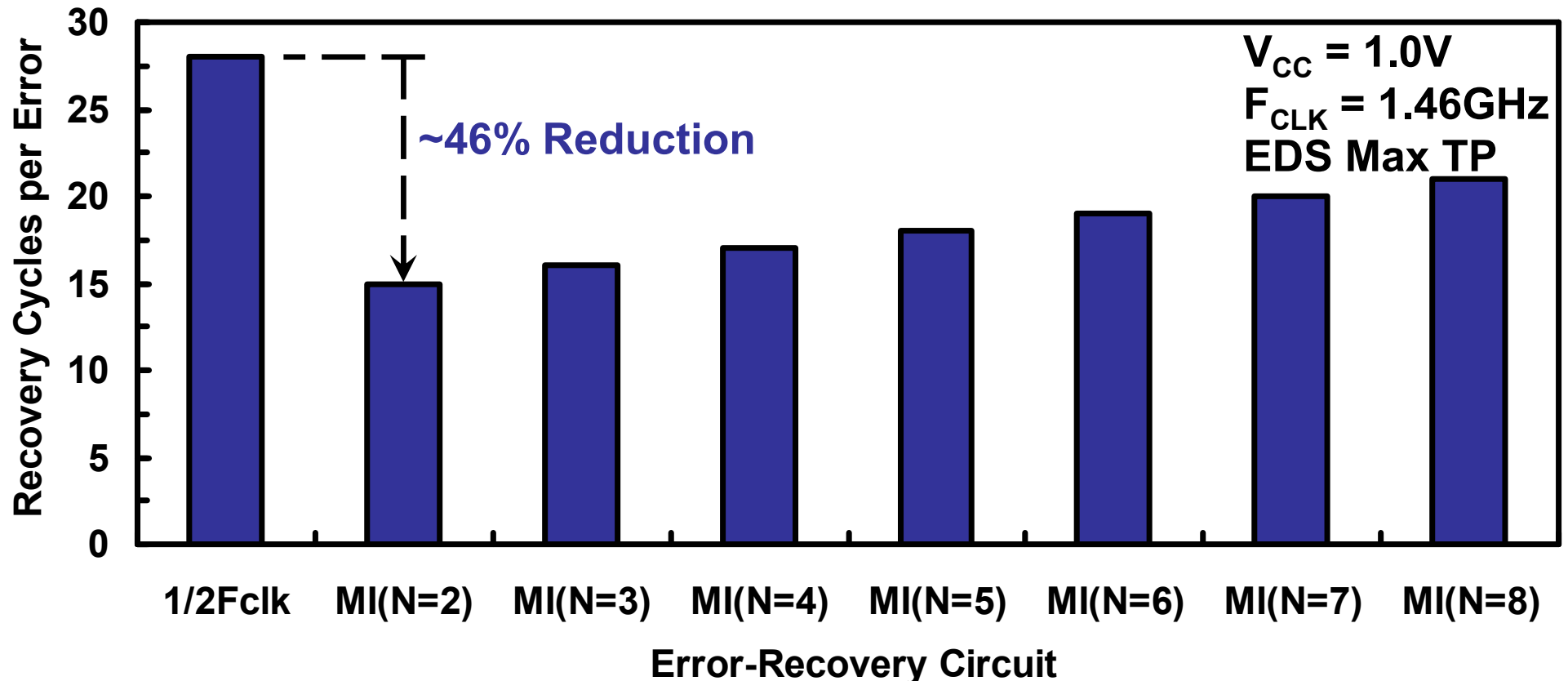- TRC throughput benefits remain at 12%

# Measured Throughput Gain vs $V_{CC}$
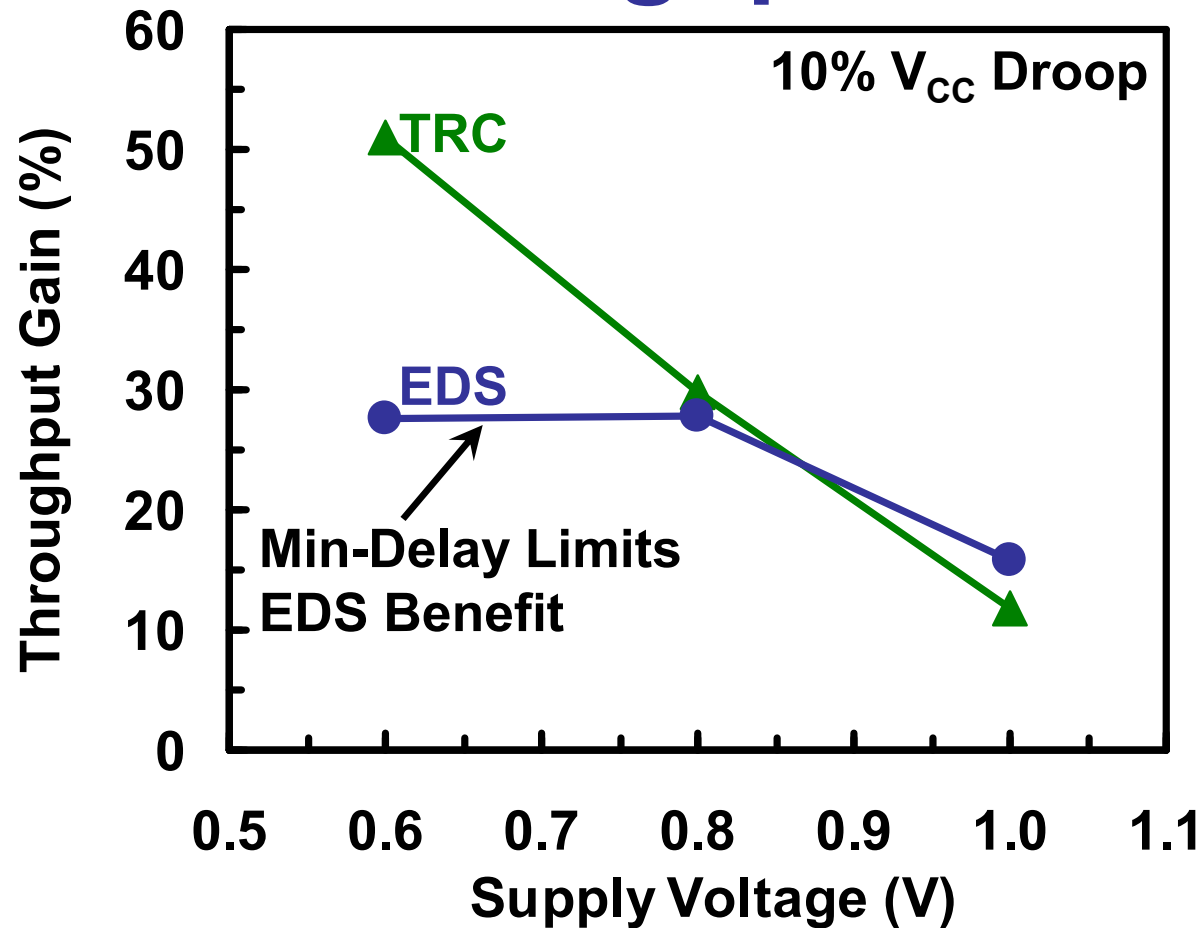


- ➤ TRC TP gains exceed EDS TP gains at low $V_{CC}$

- ➤ Error-detection window determines EDS & TRC TP benefits
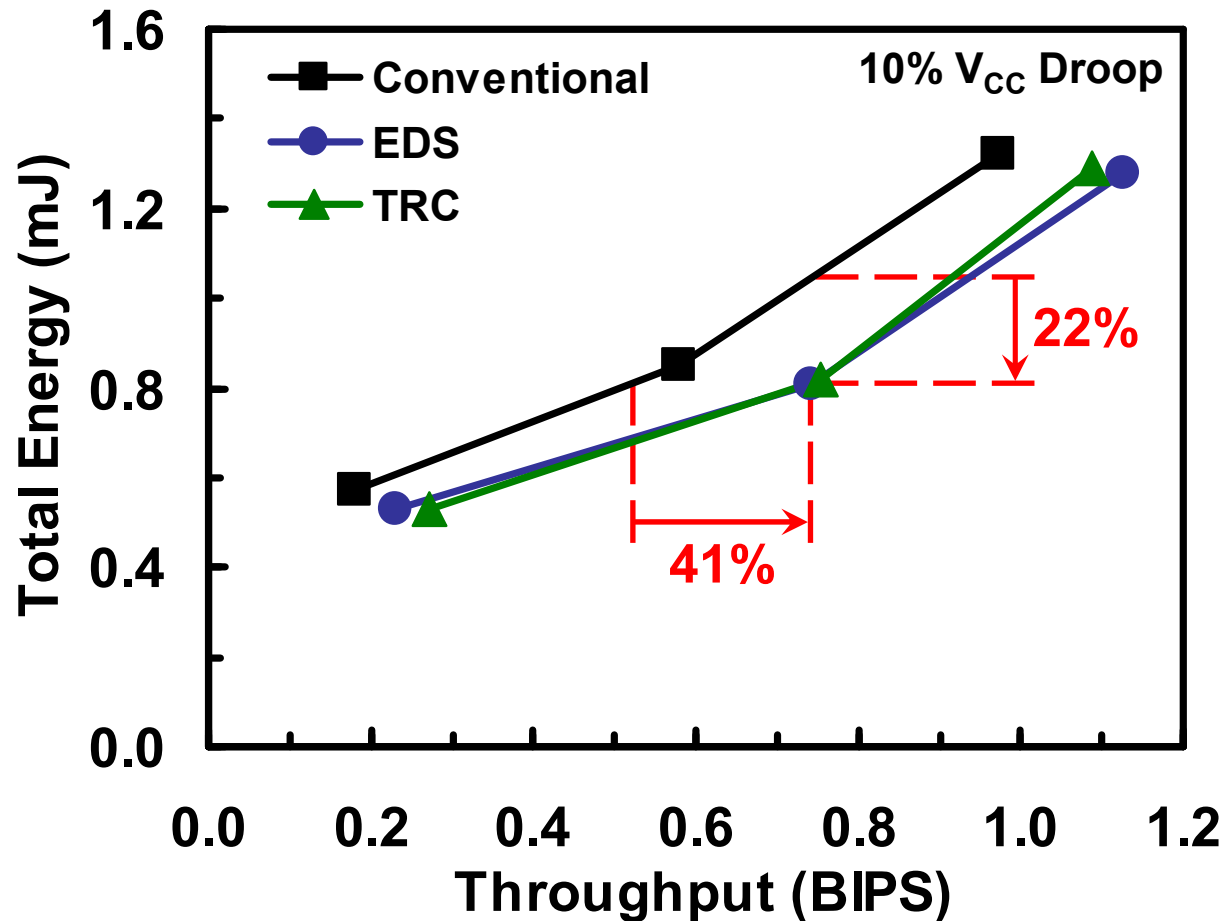
- ➤ Min-delay limits EDS error-detection window

38

# Measured Average Recovery Cycles

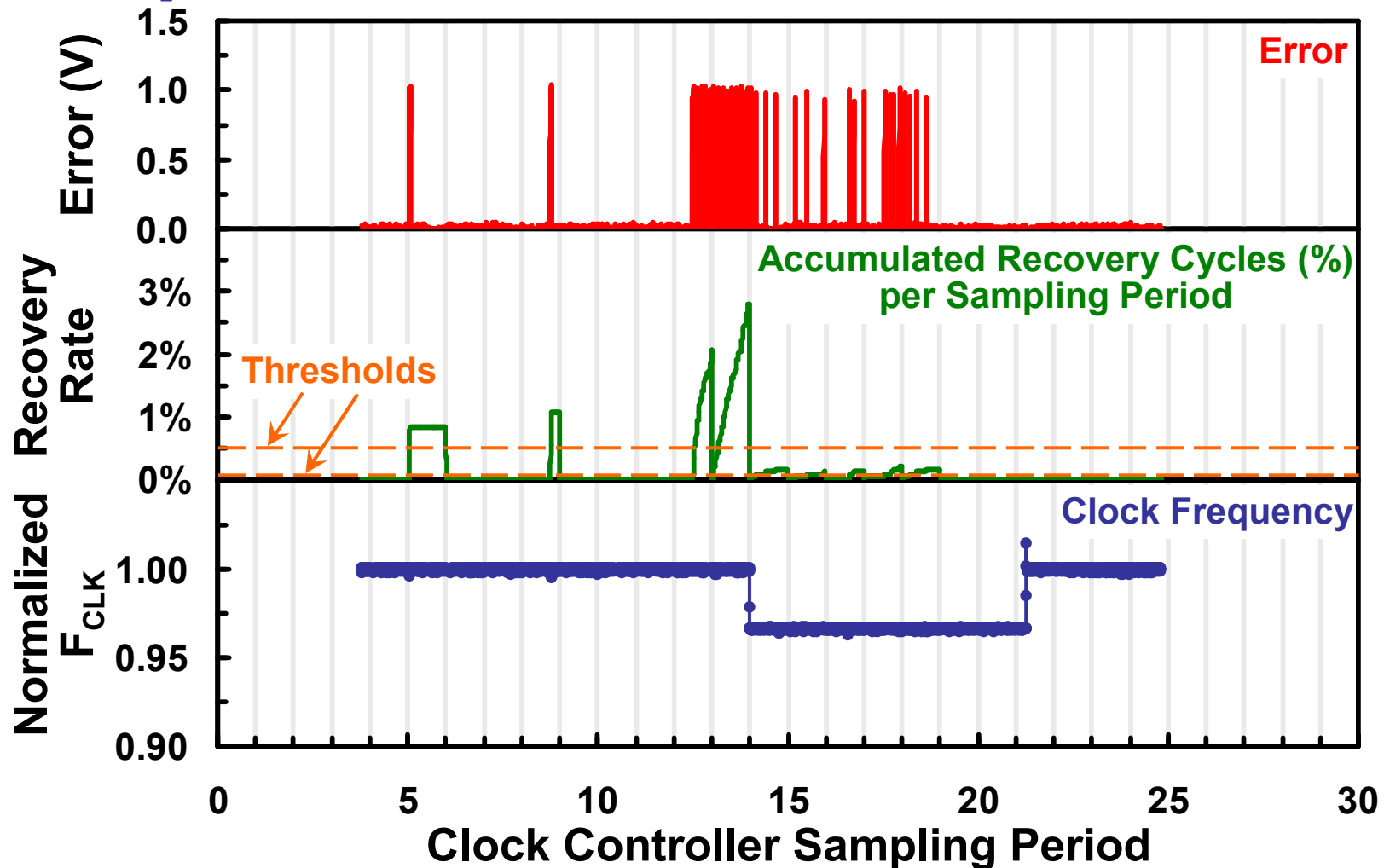## Replay at ½$F_{CLK}$ & Multiple Issue (MI) Replay at $F_{CLK}$



- **Multiple issue replay:**

  ➢ **~46% reduction in average recovery cycles**

  ➢ **Does not require clock control**

# Measured Energy vs Throughput



- **TRC & EDS resilient circuits enable:**

  - ➤ **41% throughput gain at equal energy**

  - ➤ **22% energy reduction at equal throughput**

# Adaptive Clock Control Measurement



➤ Adaptive $F_{CLK}$ compensates for persistent variations
➤ Maintains optimum recovery rate for maximum throughput
➤ Core operates through PLL lock – Jitter errors corrected  41

# Outline

- Review of Static Variations

- Resilient Microprocessor Core

- Error-Detection & Recovery Circuits

- Measurement Results

- **Conclusion**

- Future Research

# Conclusion

- **Microprocessor core employs resilient circuits to mitigate dynamic variation guardbands**

- **Error-detection circuits:**
  - ➢ **Error-detection sequential (EDS)**
  - ➢ **Tunable replica circuit (TRC)**

- **Error-recovery circuits:**
  - ➢ **Instruction replay at $\frac{1}{2}F_{CLK}$**
  - ➢ **Multiple issue instruction replay at $F_{CLK}$**

- **Silicon measurements indicate:**
  - ➢ **41% throughput gain at iso-energy**
  - ➢ **22% energy reduction at iso-throughput**

- **Resilient & adaptive circuits enable the microprocessor to adjust to operating variations for maximum efficiency**

# References

[1] A. Muhtaroglu, G. Taylor, and T. R. Arabi, "On-Die Droop Detector for Analog Sensing of Power Supply Noise," *IEEE J. Solid-State Circuits*, pp. 651-660, Apr. 2004.

[2] P. Franco and E. J. McCluskey, "Delay Testing of Digital Circuits by Output Waveform Analysis," in *Proc. IEEE Intl. Test Conf.*, Oct. 1991, pp. 798-807.

[3] P. Franco and E. J. McCluskey, "On-Line Testing of Digital Circuits," in *Proc. IEEE VLSI Test Symp*, Apr. 1994, pp. 167-173.

[4] M. Nicolaidis, "Time Redundancy Based Soft-Error Tolerance to Rescue Nanometer Technologies," in *Proc. IEEE VLSI Test Symp.*, Apr. 1999, pp. 86-94.

[5] D. Ernst, et al., "Razor: A Low-Power Pipeline Based on Circuit-Level Timing Speculation," in *Proc. IEEE/ACM Intl. Symp. Microarchitecture (MICRO-36)*, Dec. 2003, pp. 7-18.

[6] S. Das, et al., "A Self-Tuning DVS Processor Using Delay-Error Detection and Correction," *IEEE J. Solid-State Circuits*, pp. 792-804, Apr. 2006.

[7] S. Das, et al., "Razor II: In Situ Error Detection and Correction for PVT and SER Tolerance," *IEEE J. Solid-State Circuits*, pp. 32-48, Jan. 2009.

[8] K. A. Bowman, et al., "Energy-Efficient and Metastability-Immune Resilient Circuits for Dynamic Variation Tolerance," *IEEE J. Solid-State Circuits*, pp. 49-63, Jan. 2009.

[9] J. Tschanz, et al., "Tunable Replica Circuits and Adaptive Voltage-Frequency Techniques for Dynamic Voltage, Temperature, and Aging Variation Tolerance," in *IEEE Symp. VLSI Circuits Dig. Tech. Papers*, June 2009, pp.112-113.

[10] K. Bowman, et al., "Circuit Techniques for Dynamic Variation Tolerance," in *Proc. 46th ACM/IEEE DAC*, July 2009, pp. 4-7.

[11] J. Tschanz, et al., "A 45nm Resilient and Adaptive Microprocessor Core for Dynamic Variation Tolerance," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2010, pp. 282-283.
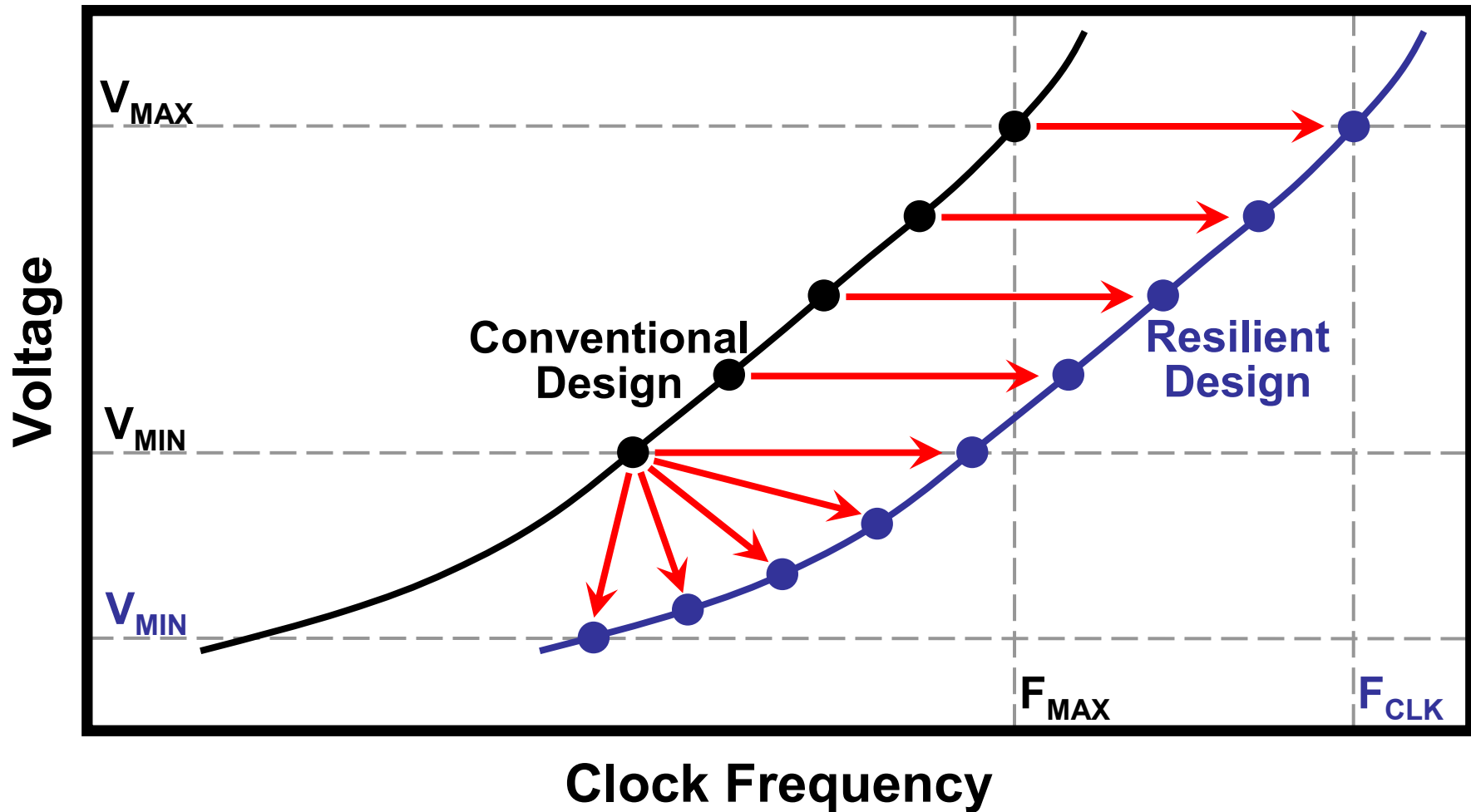
# Outline

- Review of Static Variations

- Resilient Microprocessor Core

- Error-Detection & Recovery Circuits

- Measurement Results

- Conclusion

- **Future Research**

# Wide Dynamic Operation Range
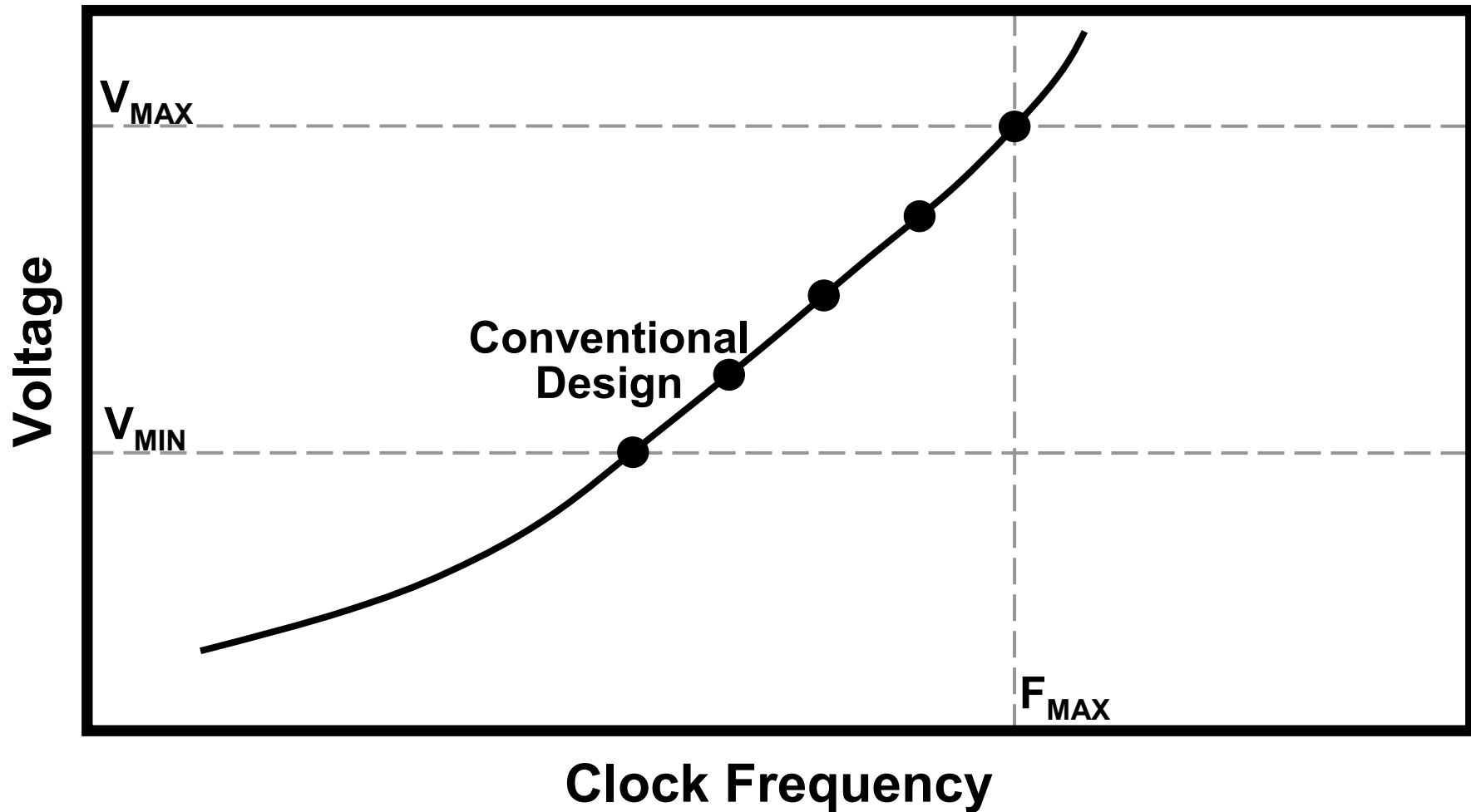


➤ $V_{MAX}$: Limited by reliability or power constraints

➤ $V_{MIN}$: Limited by circuit failures

46

# Wide Dynamic Operation Range



➢ **Resilient design expands the operating range**

# Wide Range of Platform Segments

| Platform | | Power | Perf. | Cores | Thermal | Ambient | RAS |
|---|---|---|---|---|---|---|---|
| Server |  | High | Very High |  | Active | Controlled | Very High |
| Desktop |  | Med | High |  | Fan | Controlled | High |
| Mobile |  | Low | Med |  | Fan or Fan-less | Uncontrolled | Med |
| MID |  | Very Low | Low | SOC | Fan-less | Uncontrolled | Low |

➢ **Few designs must support many segments**

➢ **Resilient design to satisfy various platform targets**

48

# Future Research

- **Resilient Design for Wide Operation Range:**

  - ➢ Explore error-detection & recovery capabilities throughout system hierarchy

  - ➢ Optimize resilient features at the system level across unique platform segments

  - ➢ Opportunities & challenges for validation & test

  - ➢ Opportunities for CAD

# Q&A