

One Bit is (Not) Enough: An Empirical Study of the Impact of Single and Multiple Bit-Flip Errors



Behrooz Sangchoolie, Chalmers

Karthik Pattabiraman, UBC

Johan Karlsson, Chalmers



<http://blogs.ubc.ca/karthik/>

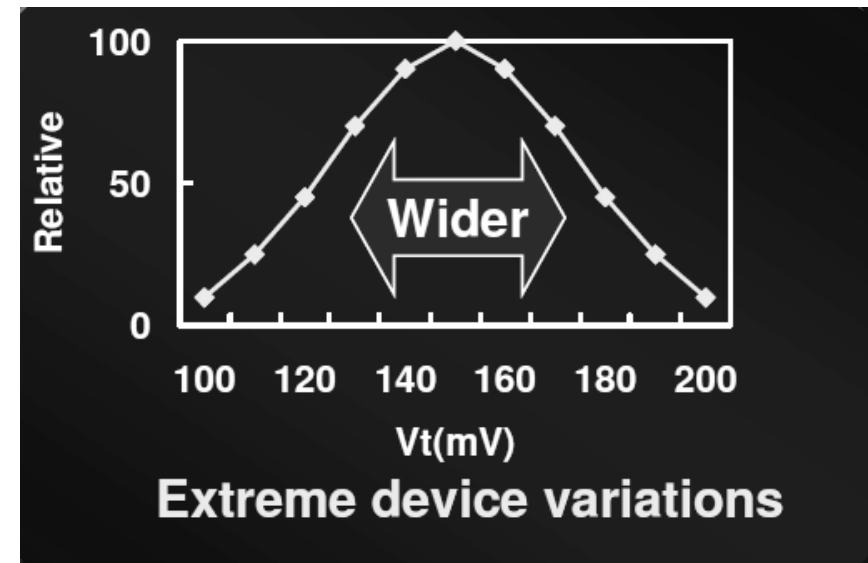
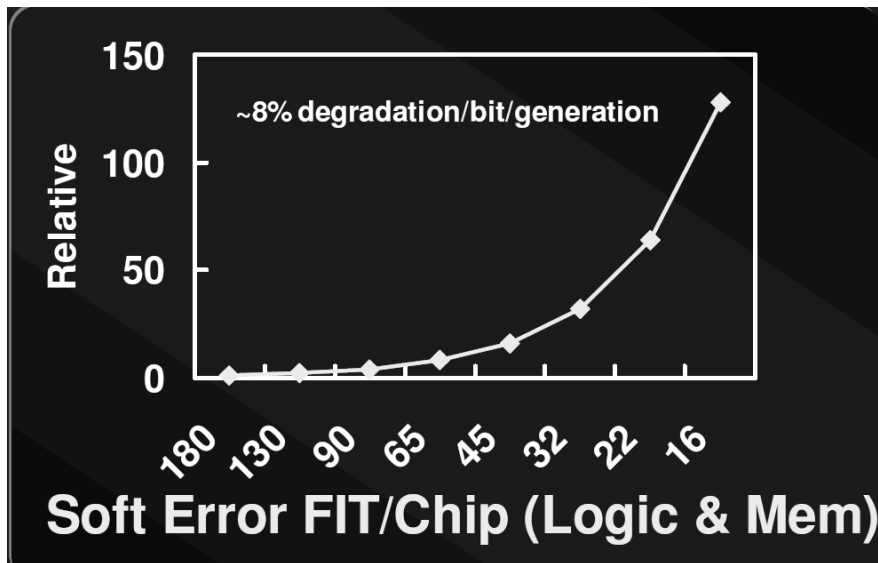


THE UNIVERSITY
OF BRITISH COLUMBIA

CHALMERS

Soft Error Problem

- Soft errors are increasing in computer systems



Source: Shekar Borkar (Intel) - Stanford talk in 2005

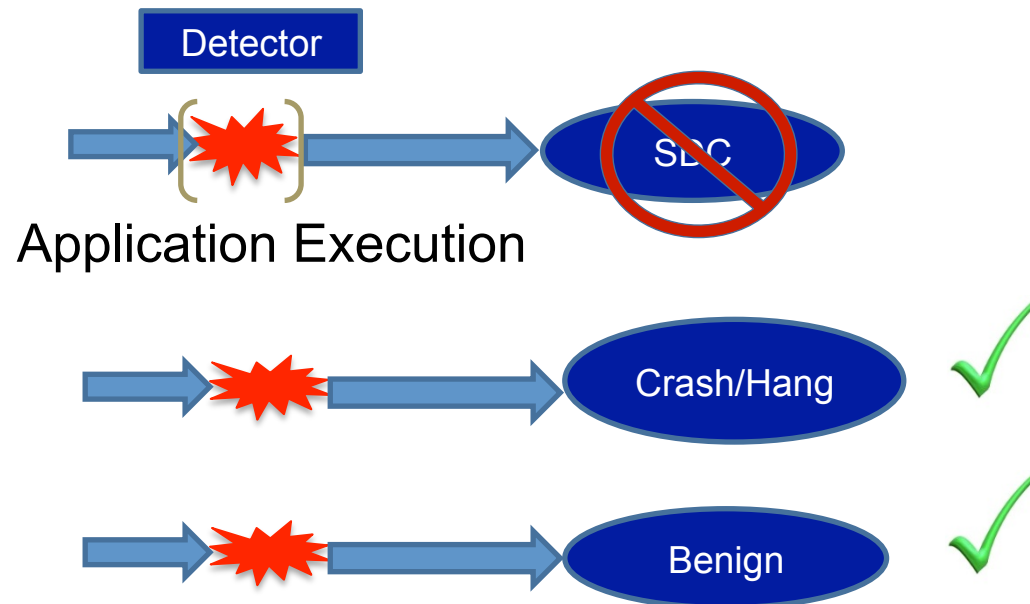
Error Resilience

- Ability of a program to **NOT** produce an **SDC** (Silent Data Corruption) upon a hardware fault
- **SDC**: Deviation of output from golden output



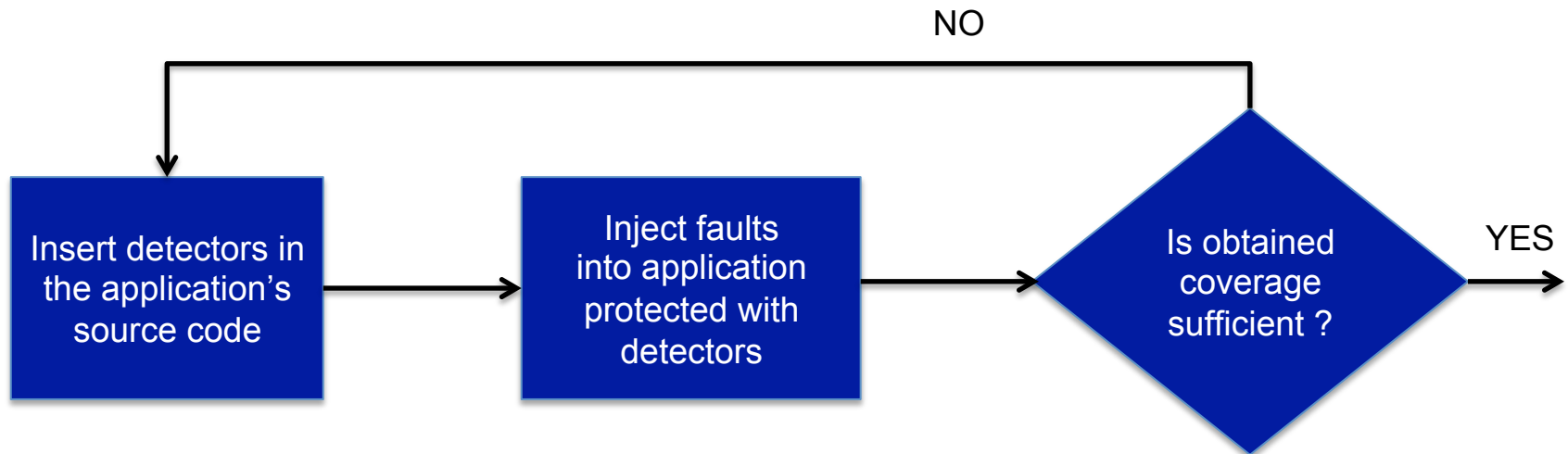
Our Groups' Research: Application-level Selective Fault-Tolerance

- Add error detectors to applications to detect SDCs
 - Much more efficient than “all-or-nothing” techniques

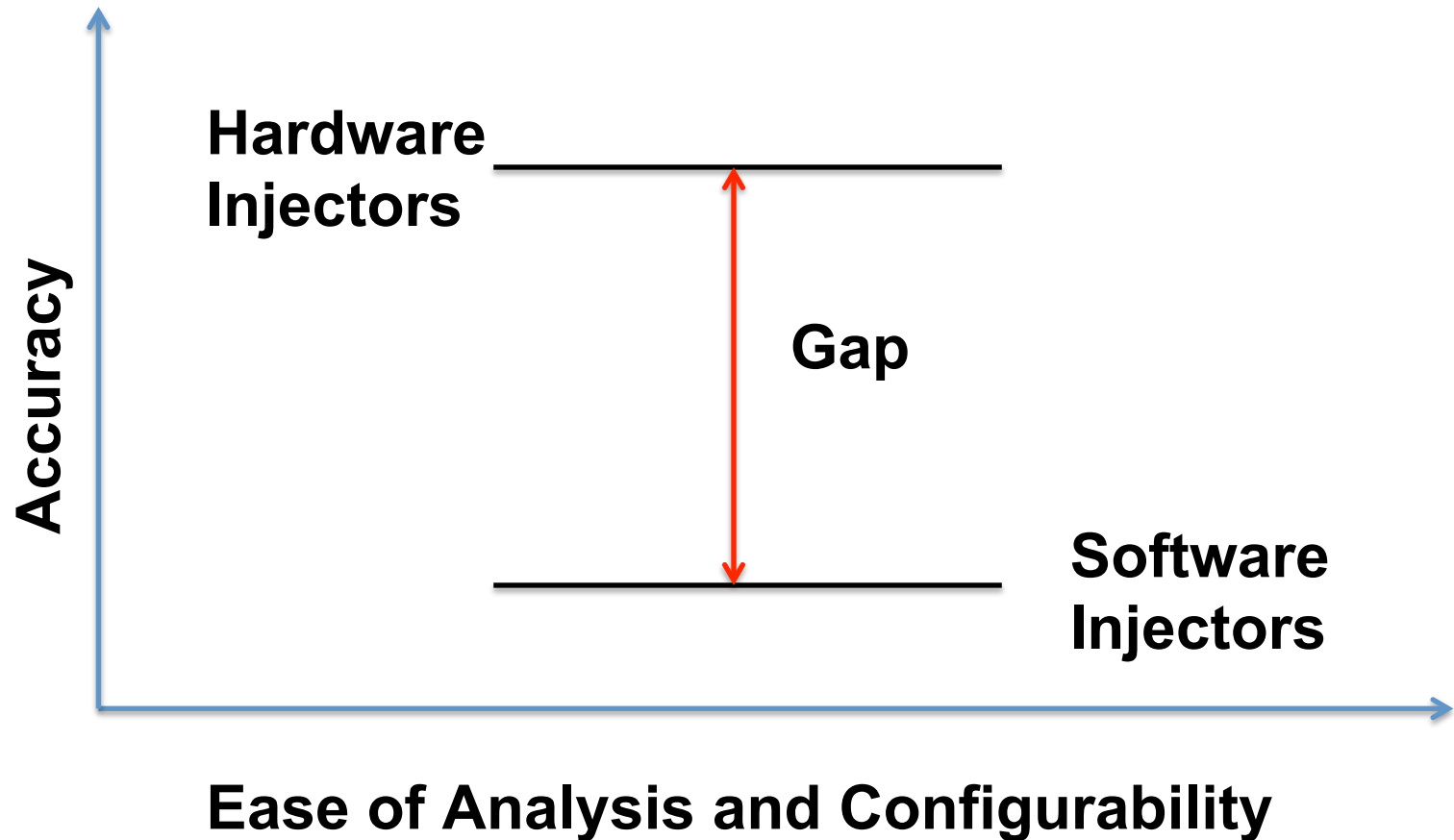


Software Fault Injection

- Inject faults to iteratively improve coverage
 - Find which errors result in SDCs
 - Find errors that are missed by detectors



Hardware Vs. Software Injectors



Main Problem

- Software Fault Injectors use the **single bit-flip** fault model to abstract the effect of soft errors
- But a single soft error is likely to manifest as **multiple-bit errors** at the **application level**

DAC 2013

**Quantitative Evaluation of Soft Error Injection Techniques for
Robust System Design**

Hyungmin Cho¹, Shahrzad Mirkhani³, Chen-Yong Cher⁴, Jacob A. Abraham³, Subhasish Mitra^{1, 2}

¹Department of EE and ²Department of CS ³Computer Engineering Research Center ⁴IBM T. J. Watson Research Center,
Stanford University, Stanford, CA, USA The University of Texas at Austin, Austin, TX, USA Yorktown Heights, NY, USA

Fault Model

- **Faults in the processor**

- Register file
- Computational elements



- **Faults in memory**

- Assume memory is ECC protected



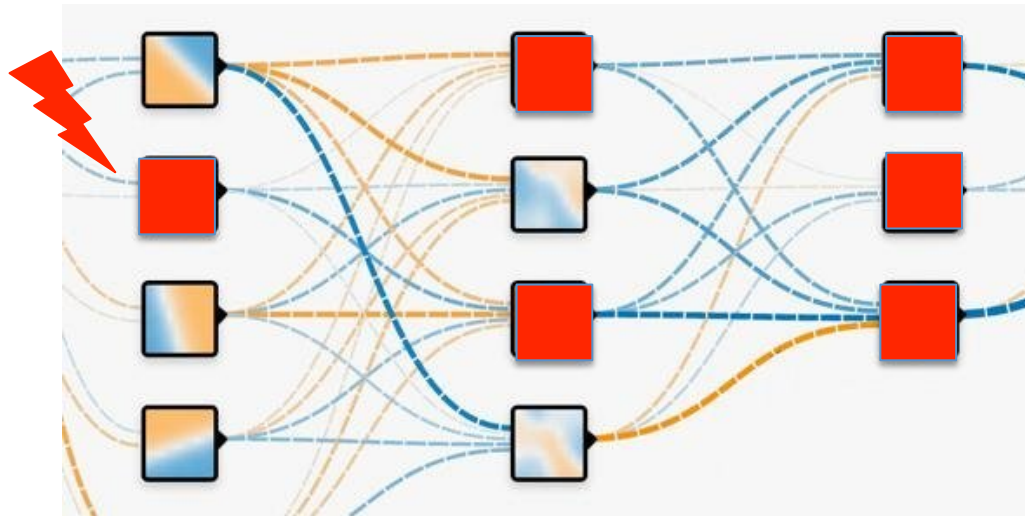
- **Faults in control logic**

- Assumed to be protected by other means



Multiple Bit Flip Errors

- Single Soft error → **Multiple bit-flips** in software
- Error propagation in the micro-architectural level



Source: Chen-Yong Chen, IBM Research

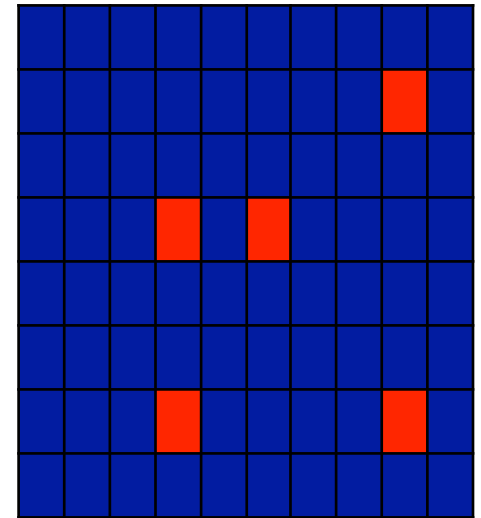
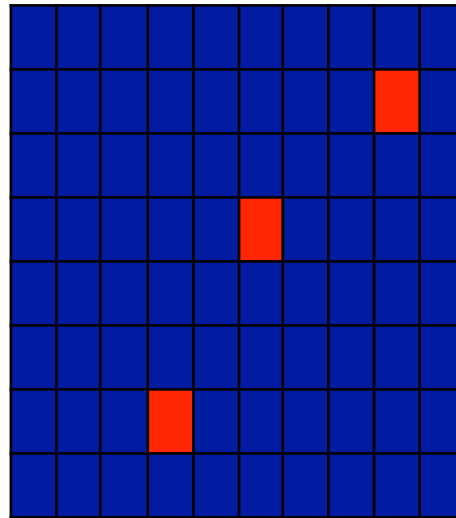
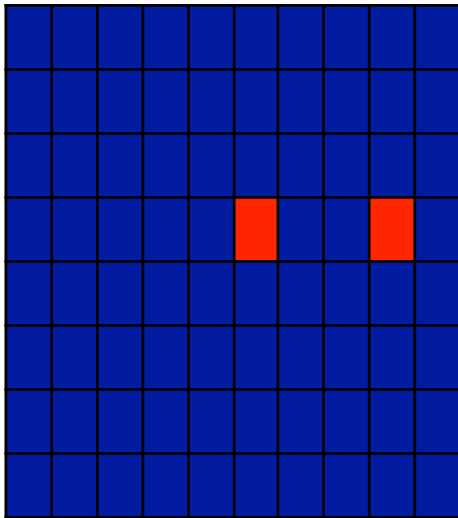
21bc:	mr	r2, r1
21c0:	or	r3, r2 , r1
21c4:	neg	r5, r4
21c8:	and	r7, r5 , r6
21cc:	and	r8, r3 , r7

This Paper: Main Question

- Does the *multiple bit-flip* model result in significantly different **error resilience results** compared with the *single bit-flip* model?
 - For different kinds of injection techniques
 - For different kinds of fault distributions

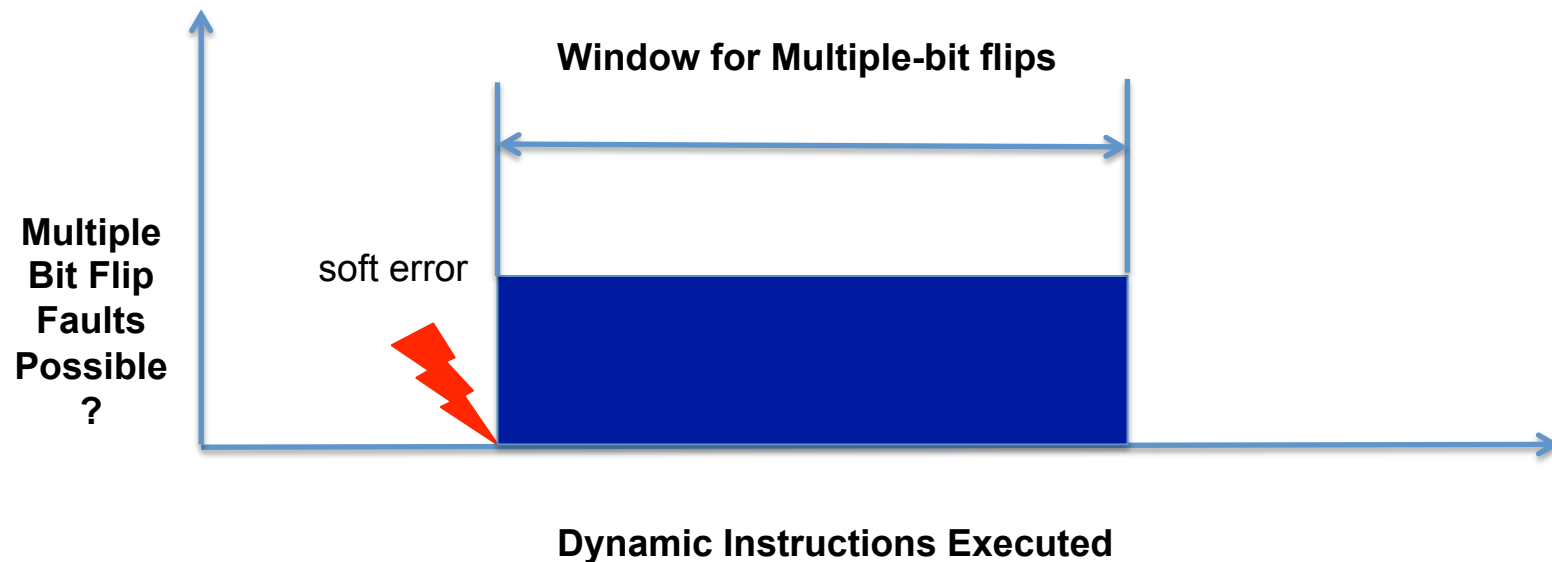
Challenge

- **Multiple-bit injection space is extremely large**
 - Multiple bit-flips in a single register
 - Multiple bit-flips in multiple registers
 - Any combination of the above



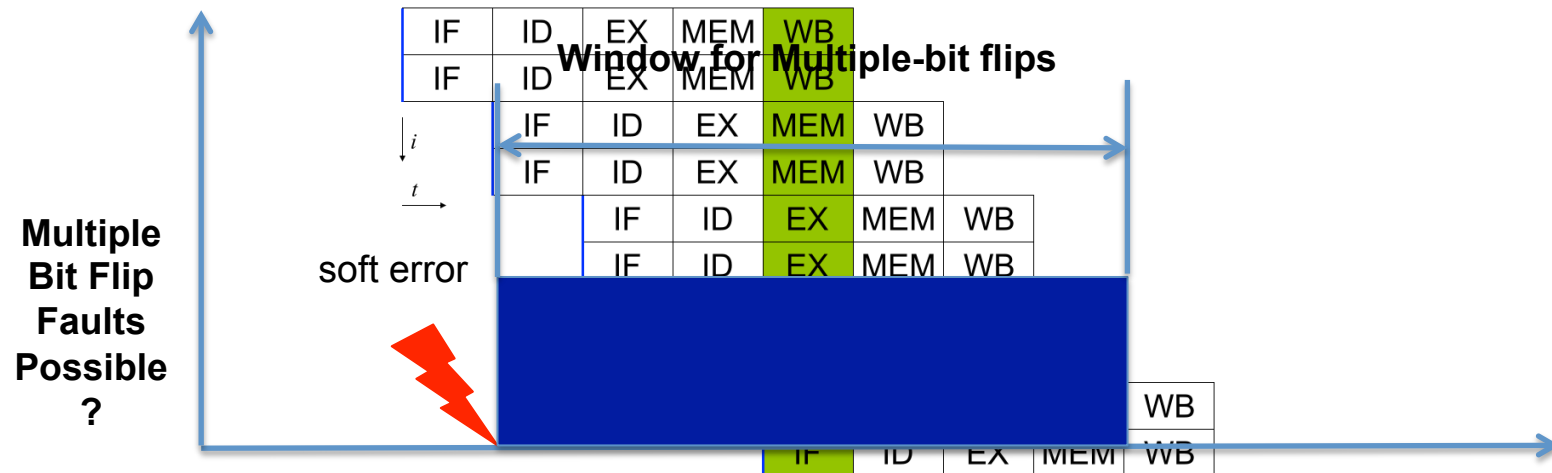
Main Insight

- Effect of multiple-bit faults are confined within a (small) dynamic instruction window from fault
- Sufficient to consider multiple-bit faults within the window for injecting faults into application



Why does this hold in practice ?

- Soft errors manifest as multiple bit-flips in the program through propagation in the hardware
- Hardware error propagation is confined to the instruction window in superscalar processors



Dynamic Instructions Executed

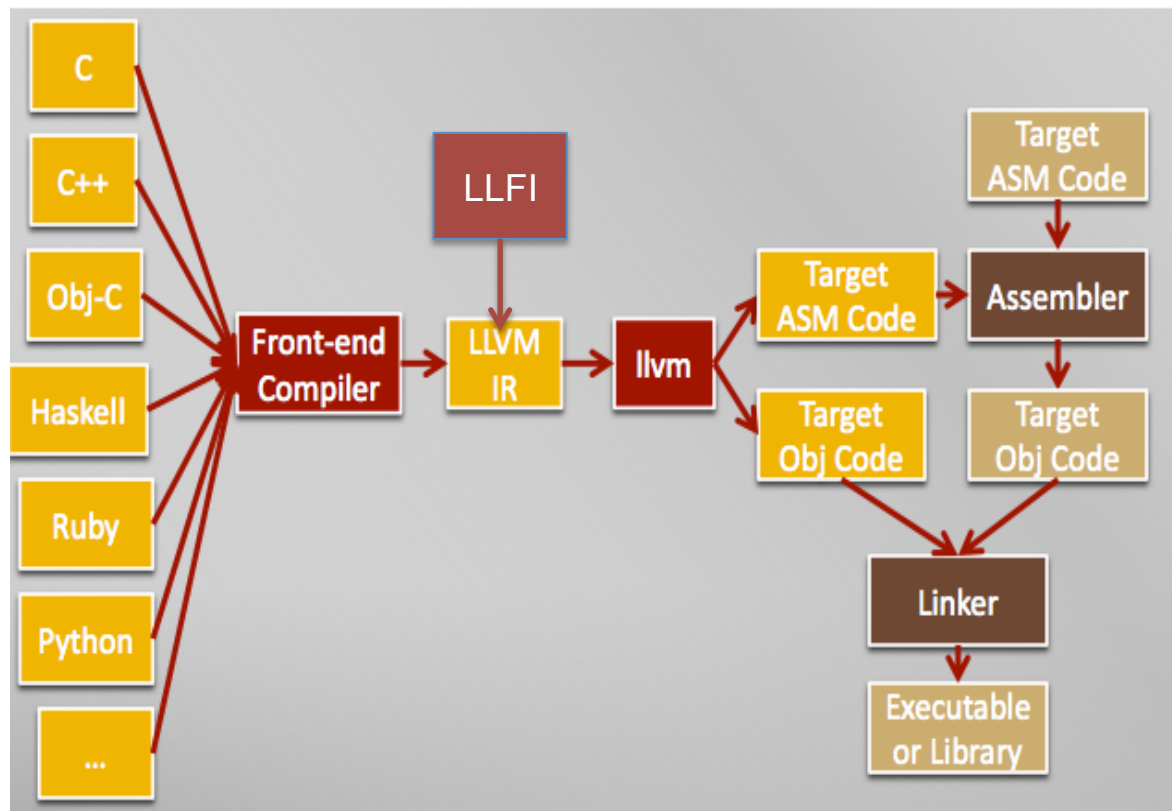
By Amit6, original version (File:Superscalarpipeline.png) by User:Poil (Own work) [CC BY-SA 3.0 (<http://creativecommons.org/licenses/by-sa/3.0/>)], via Wikimedia Commons

Outline

- Motivation and Goal
- **Experimental Setup**
- Results
- Conclusion

Experimental Setup: LLFI tool

Works at LLVM compiler's intermediate (IR) [Wei14]



Experimental Setup: FI techniques

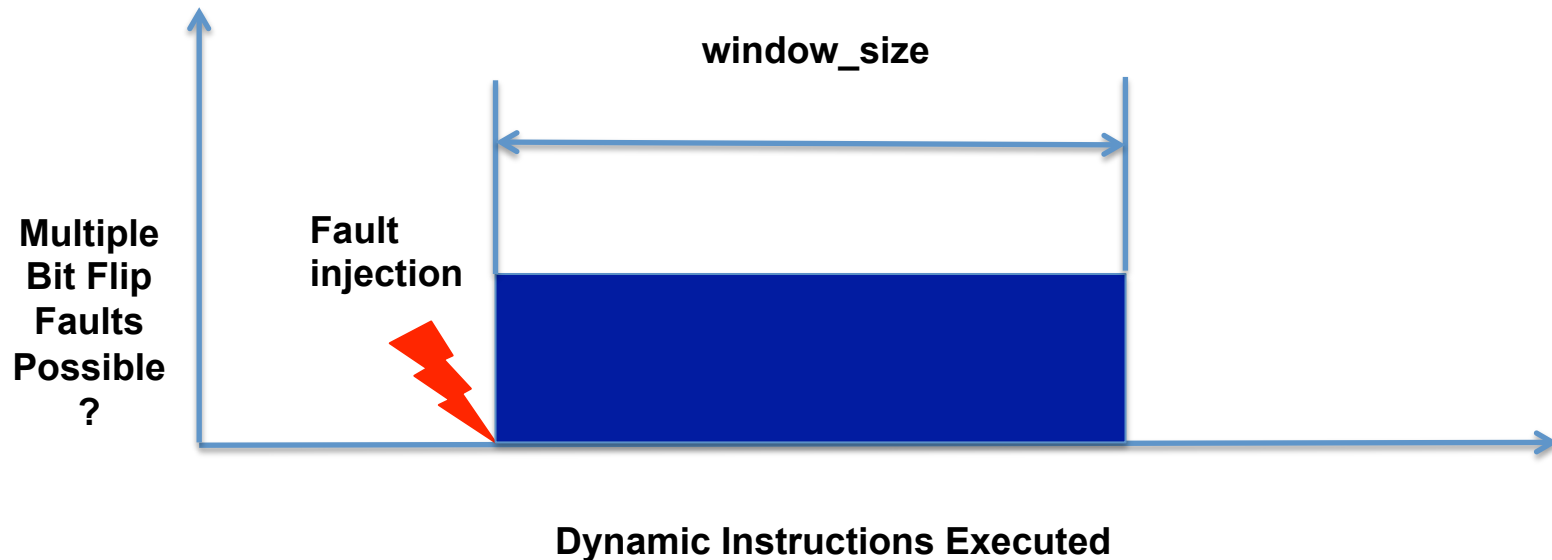
- **Inject on Read:** Inject fault **before** reading a **source** register
- Models faults that occur in the **register file**
- **Inject on Write:** Inject fault **after** writing to a **destination** register
- Models faults that occur during the **computation**

21bc:	mr	r2, r1
21c0:	or	r3, r2 , r1
21c4:	neg	r5, r4
21c8:	and	r7, r5 , r6
21cc:	and	r8, r3 , r7

21bc:	mr	r2 , r1
21c0:	or	r3 , r2, r1
21c4:	neg	r5 , r4
21c8:	and	r7 , r5, r6
21cc:	and	r8 , r3, r7

Experimental Setup: Parameters

- **window_size**: max distance between faults
 - Varies from 1 to 1000 (random and fixed values)
- **number_of_bits**: max number of faults/run
 - Varies from 1 to 30 (1-10 and 30 as an extreme)



Experimental Setup: Benchmarks

- 11 **MiBench** programs – embedded systems
- 4 **Parboil** programs – parallel computing

Automotive

Network

Telecom

Security

Office

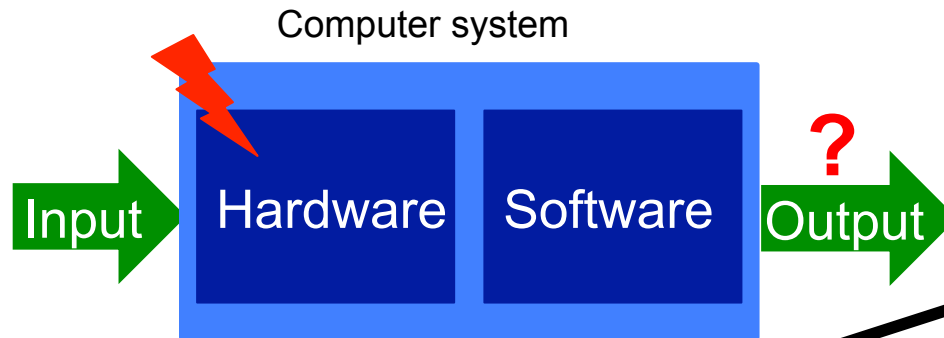
HPC

Experimental Setup: Approach

We perform over **27 Million fault injection** experiments for each combination of the benchmark, parameters, and FI technique !

$$\begin{aligned} & \mathbf{15 \text{ benchmarks} * 91 \text{ parameter values}} \\ & \quad * \mathbf{2 \text{ techniques} * 10,000 \text{ fault}} \\ & \quad \mathbf{injections/combination} \\ & \quad = \mathbf{27,300,0000} \end{aligned}$$

Experimental Setup: Outcome Classification



Benign

No output

Detected by hardware exceptions

Hang

Incorrect Output
a.k.a.
Silent data corruption (SDC)

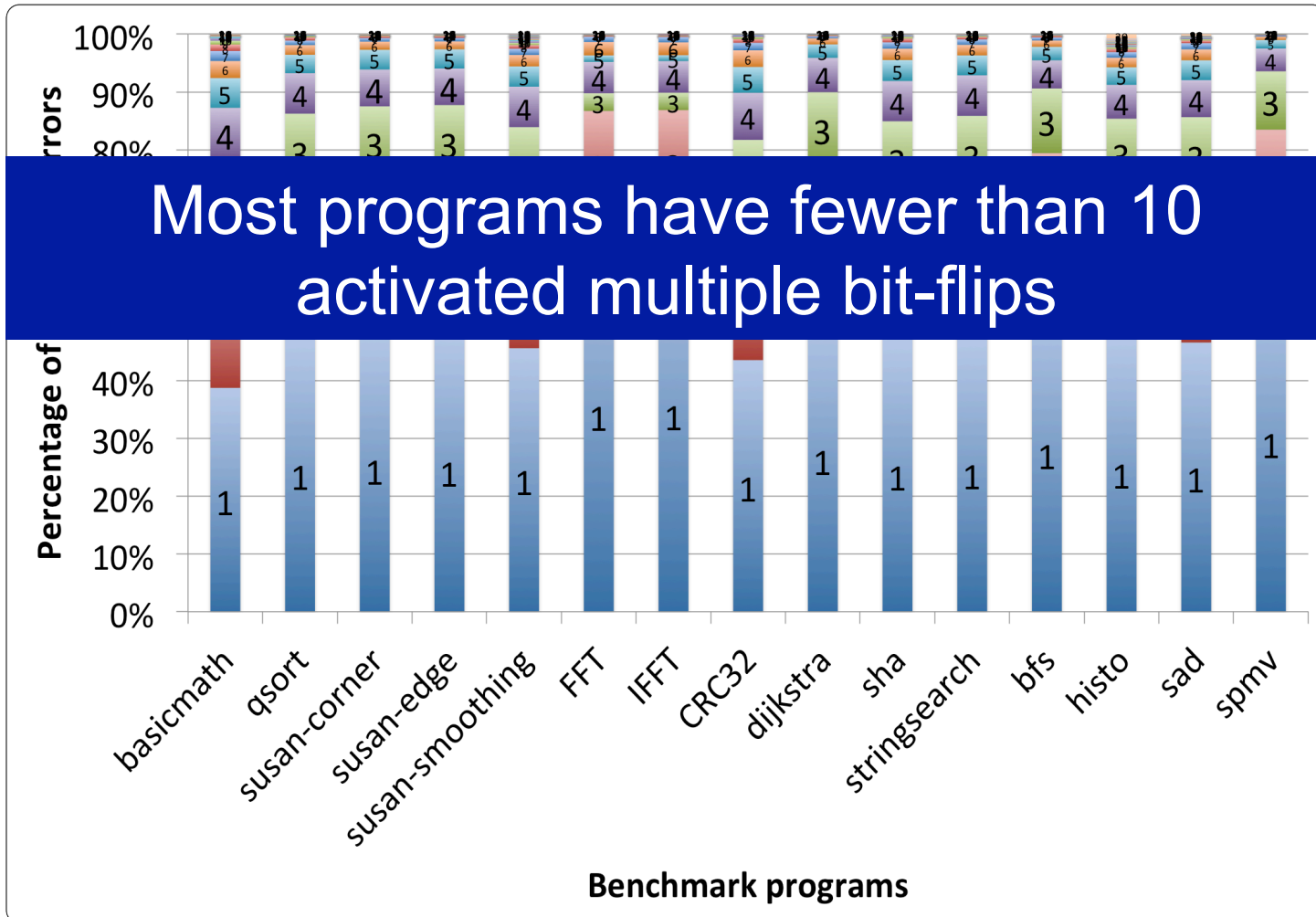
Outline

- Motivation and Goal
- Experimental Setup
- **Results**
- Conclusion

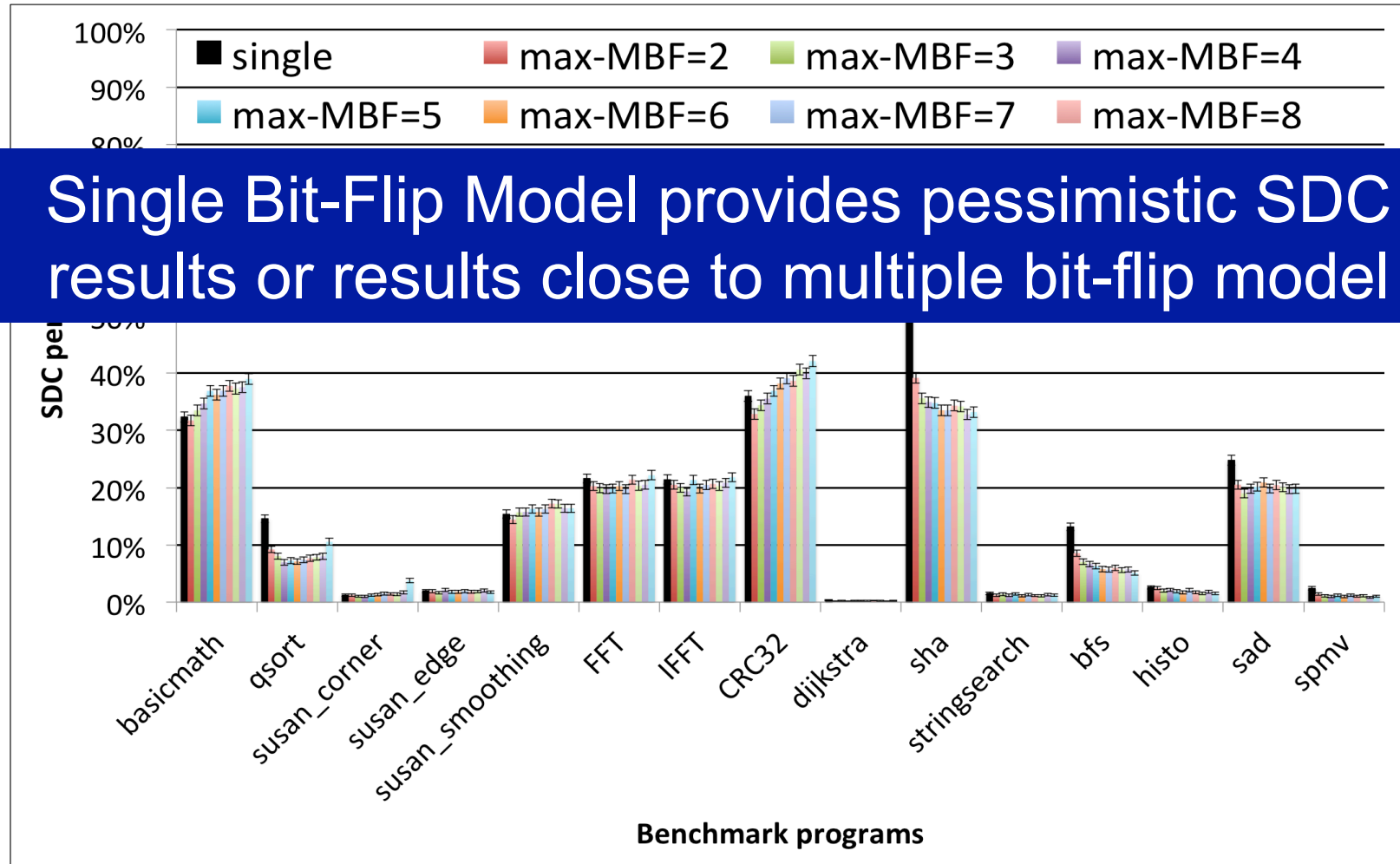
Research Questions (RQs)

- **RQ1:** How many multi-bit errors are **activated** in a program?
- **RQ2:** Does the single bit-flip error model result in **pessimistic** percentage of **SDCs** compared with multiple bit-flip error model?
- **RQ3:** Is there an **upper bound** to the maximum **number of multiple bit-flips** needed to cause pessimistic percentage of SDCs?
- **RQ4:** Is there a **maximum dynamic window size** that causes pessimistic percentage of SDCs?
- **RQ5:** Can we use **single bit-flip** results to prune the multiple bit flip fault injection space?

RQ1: Activation of Multiple Bit Flips



RQ2: Single Bit-flip Vs. Multiple Bit Flips



RQ3: Upper bound on multiple bit flips

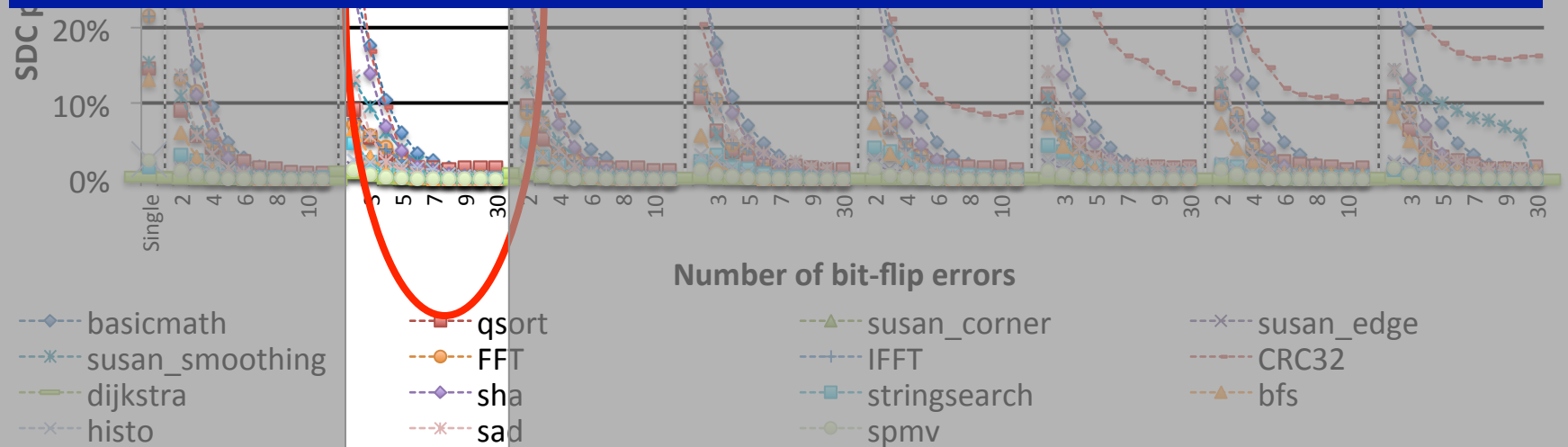
Program	inject-on-read		inject-on-write	
	max-MBF	win-size	max-MBF	win-size

At most 3 bit flips are sufficient to get pessimistic SDC results in most applications (in the few cases where single bit-flip model is not sufficient)

FFT	single bit-flip		single bit-flip	
IFFT	single bit-flip		single bit-flip	
CRC32	2	100	2	100
dijkstra	single bit-flip		3	4
sha	single bit-flip		single bit-flip	
stringsearch	2	RND(2-10)	2	4
bfs	single bit-flip		single bit-flip	
histo	single bit-flip		6	1
sad	single bit-flip		single bit-flip	
spmv	single bit-flip		single bit-flip	

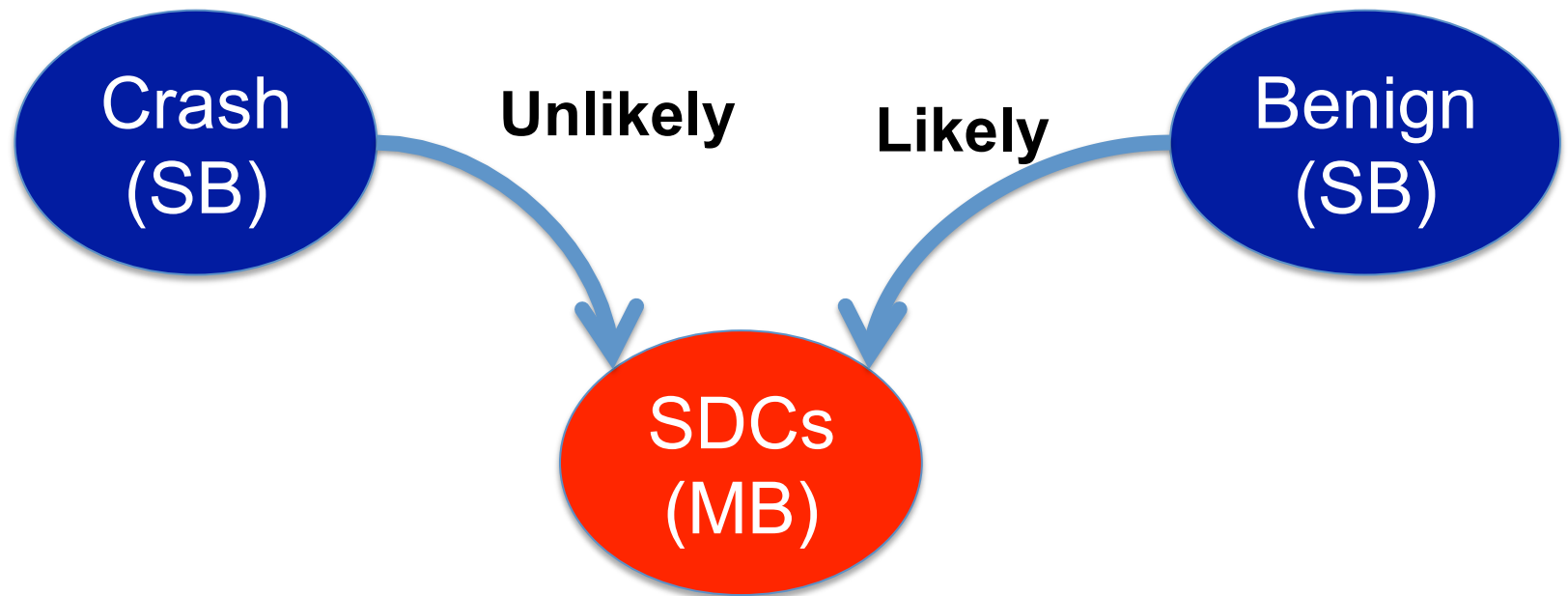
RQ4: Effect of window_size

Window sizes do not have a significant effect on SDC rates. However, smaller window sizes (<5) result in higher percentages of SDCs



RQ5: Multiple bit-flip (MB) Error Space Pruning from Single Bit-Flip (SB)

Sufficient to inject multiple bit-flips into locations where single bit-flips result in benign outcomes to get pessimistic SDC results



Takeaways

- In most cases, single bit flip fault model yields **comparable resilience** to multiple bit fault model
- To get pessimistic estimates of resilience, we need atmost **3 multiple bit flips** across applications
- **Smaller window sizes** result in (slightly) higher percentages of SDCs for a given no. of bit flips
- Sufficient to **inject** multiple-bit errors into locations where single bit-flips result in **Benign** outcomes

Outline

- Motivation and Goal
- Experimental Setup
- Results
- Conclusion

Conclusion

Does the *multiple bit-flip* model result in significantly different **error resilience results** compared with the *single bit-flip* model?

- No, in most cases
- Yes, in a few cases

Based on a total of **27 million** fault injection experiments

Bottom line: One bit is Often Enough !

LLFI: <http://github.com/DependableSystemsLab/llfi>