flexMEDiC: flexible Memory Error Detection by Combined data encoding and duplication

Norman A. Rink and Jeronimo Castrillon

Technische Universität Dresden

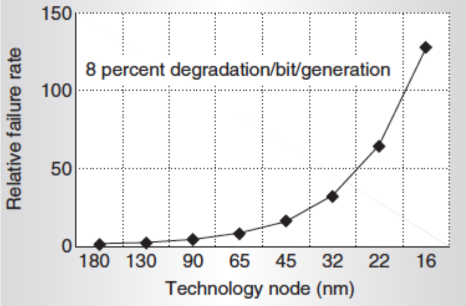
2nd International Workshop on Resiliency in Embedded Electronic Systems

31 March 2017

Lausanne, Switzerland

Motivation

- Frequency of transient HW faults (aka. *soft errors*) is increasing.
  - Traditional cause of faults: cosmic rays.
  - Vulnerability is increasing due to smaller feature sizes and lower operating voltages.
  - Dark/dim silicon in memory modules:
    - Extended refresh cycles for DRAM.
    - Lower supply voltage for SRAM.



S. Borkar, "Designing reliable systems from unreliable components: …," IEEE Micro, vol. 25, no. 6, 2005.

- Limitations of ECC memory modules.
  - Typically SEC-DED codes (*single error correction, double error detection*).
  - Large fractions of memory errors cannot be handled by SEC-DED codes (Hwang et al., ASPLOS 2012).

→ Software-implemented error detection can be flexibly adjusted to detect complex (multi-bit) error patterns as well.

Norman Rink, norman.rink@tu-dresden.de

Motivation – cont'd

- The simple **AN encoding** scheme is capable of detecting multi-bit errors:
  - Fix an integer constant A.
  - Encode integer values by multiplying by A: $n_{enc} = n * A$
  - Decode by dividing by A: $n = n_{enc} / A$
  - Check for errors: $n_{enc} \mathit{mod} A = 0$
- Error-detecting capability varies with the constant A.
  - Powers of 2 are ill-suited to error detection.
  - A = 58659 is known to have good properties; can detect up to 5 bit flips, Hoffmann et al., 2015.
- AN encoding introduces large overheads if used to protect operations: several **10*x*-100*x***.

Norman Rink, norman.rink@tu-dresden.de

flexMEDiC

- Detection of multiple bit errors in memory, including caches, load-store queues.
- Apply AN encoding only to values stored to memory → low overhead due to AN encoding.

encode before storing:

```
%0 = mul i64 %0,  A
store i64 %0, i64* %p
```

check and decode after loading:

```
%1 = load i64* %p
%2 = srem i64 %1, A
...
%3 = sdiv i64 %2, A
```

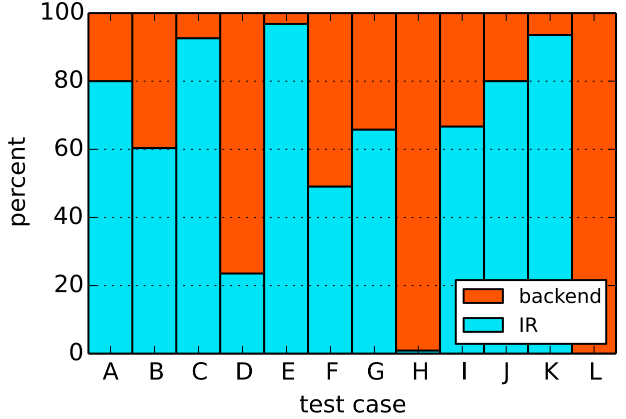
- AN encoding is applied at the LLVM IR level.
  - Common approach in software-implemented fault tolerance schemes.

→ Error detection at the IR level misses memory accesses that are inserted by the compiler backend.

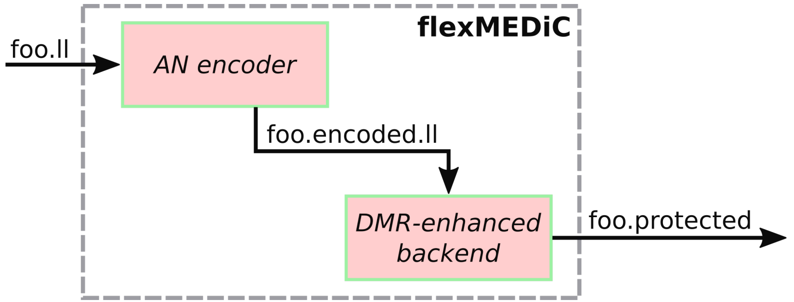
Norman Rink, norman.rink@tu-dresden.de

flexMEDiC – cont'd

- Frequency of unprotected memory accesses (at the level of LLVM IR):



Apply DMR to backend-inserted memory accesses:



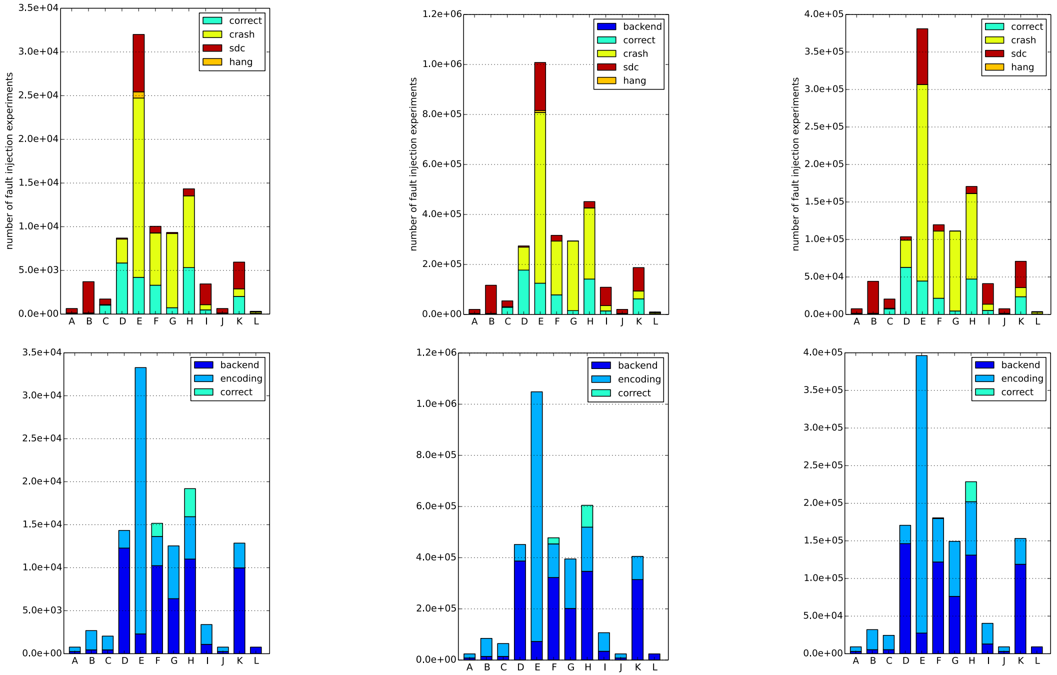
- Two major sources of backend-inserted memory accesses: **register spills, function calls.**
- Using DMR keeps function calls efficient.
- DMR can detect **arbitrarily many bit flips** within the same data word.

Norman Rink, norman.rink@tu-dresden.de

Full memory error detection

letter	test case
A	array reduction
B	bubblesort
C	CRC-32
D	DES encryption
E	Dijkstra (shortest path)
F	expression evaluation
G	token lexer
H	expression parser
I	matrix multiplication
J	array copy
K	quicksort
L	switch

single bit flip **double bit flip** **5-bit flip***

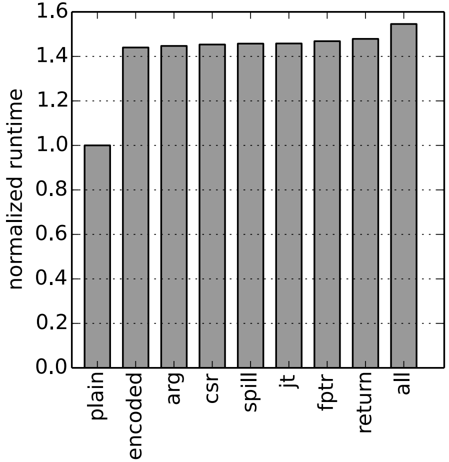


*) sampled 0.01% of all possible error patterns within a single data word.

Norman Rink, norman.rink@tu-dresden.de

Runtime overhead

Geometric means across test programs*:



Overhead due to DMR only*:

DMR	overhead
frame pointer	1.020
callee-saved register	1.009
jump tables	1.013
return address	1.027
function arguments	1.005
register spills	1.012
all	1.073

*) runtime measurements performed on 64-bit Intel Core i7, at 3.6GHz

- Total runtime overhead due to **flexMEDiC** is **1.55x.**
- **AN encoding** introduces an overhead of only about **1.4**x.
- **DMR measures** inserted by the **compiler backend** introduce an overhead of only **1.07**x.
- Memory overhead of AN encoding is **log(A)** bits per data word.
- Memory overhead of DMR is 100%, but used sparingly.

Norman Rink, norman.rink@tu-dresden.de

Summary and outlook

- flexMEDiC can successfully **detect multi-bit errors** in memory.
- The number of detectable bits can be adjusted flexibly by varying the encoding constant A.
  - Finding appropriate A is a fundamental problem in AN encoding.
- flexMEDiC relies on DMR only for local memory accesses → applicable to multi-threaded applications with shared memory.
- What hardware support can one think of to efficiently support the memory overhead of AN encoding of log(A)?
- Have observed similar distributions of program responses to varying numbers of bit flips.
  - Is there a fundamental reason for this (e.g. due to the nature of certain programs)?
  - How can one capitalize on this for the purpose of software-implemented fault tolerance?

Norman Rink, norman.rink@tu-dresden.de

flexMEDiC: flexible Memory Error Detection by Combined data encoding and duplication

Norman A. Rink and Jeronimo Castrillon

Technische Universität Dresden

Thank you.