IEEE International Symposium on Circuits and Systems, ISCAS 2012, COEX, Seoul, Korea, 20-23, May, 2012

An LDPC Decoding Method for Fault-Tolerant Digital Logic

Yangyang Tang[1,2], Chris Winstead[2], Emmanuel Boutillon[1], Christophe Jégo[2] and Michel Jézéquel[3]

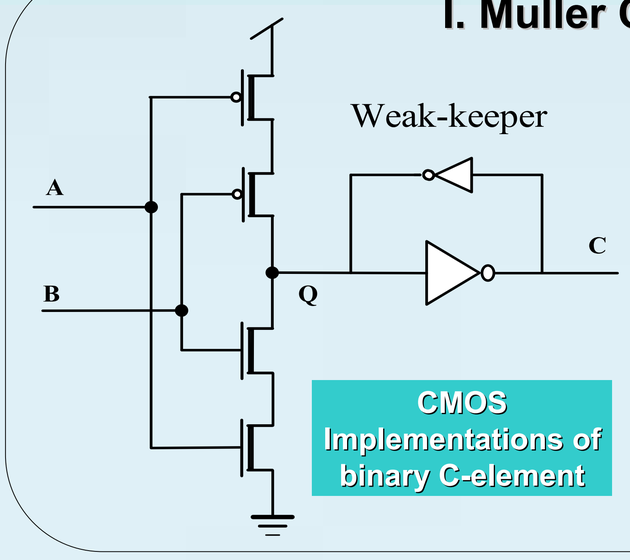
[1]Université de Bretagne Sud, UMR CNRS 6285 Lab-STICC, Lorient, France
[2]Dept. of Electrical and Computer Engineering, Utah State University, Logan, Utah 84322
[3]Institut Polytechnique Bordeaux, UMR CNRS 5218 Lab-IMS, Bordeaux, France
[4]Institut TELECOM/TELECOM Bretagne, UMR CNRS 6285 Lab-STICC, Brest, France
yangyang.tang@univ-ubs.fr

INTRODUCTION

- **Unreliable Electronics & New Challenge:**
  - ✓ In the emerging nano-scale era, electronic devices are increasing susceptible to logic errors that can degrade the reliability.

  Fault-Tolerant solution needs to be tolerant of internal errors.
- **Proposed embedded fault-masking solution:**
  - ✓ LDPC code
  - ✓ Referred to as LDPC Stochastic Decoding (LSD) Method
  - ✓ Resilient against internal transient gate errors
  - ✓ Asynchronous Implementation of the iterative decoder

I. Muller C-element [1]



CMOS Implementations of binary C-element

A	B	C
0	0	0
0	1	C_{n-1}
1	0	C_{n-1}
1	1	1

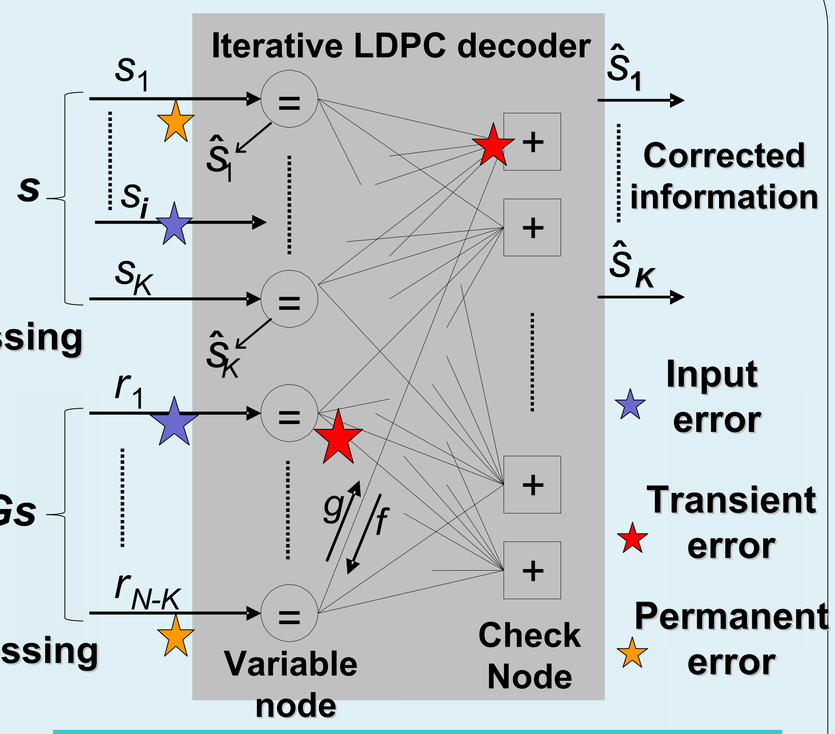
C-element

Truth-Table of Binary C-element

- C_{n-1} denotes the state maintained via weak feedback
- C-element is increasingly used for fault-masking design

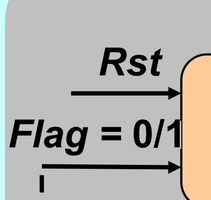
II. Iterative LDPC Decoder [2]

(*s*, *r* = *Gs*) LDPC codeword coming from memory, bus or computation

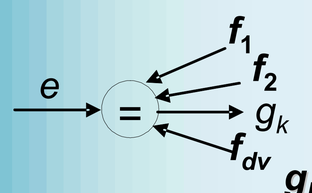


Check Node Processing

$f_i = \oplus g_k,\ k \neq i$



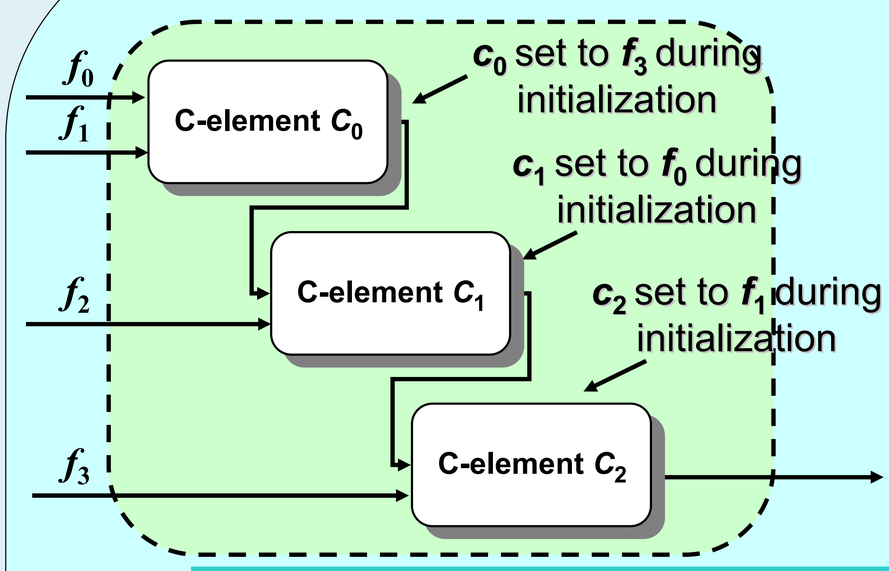
Variable Node Processing



g_k= e if (f_i, $i \neq k$) disagree, f_1 elsewise

Conventional Iterative LDPC decoder: Massages Exchanging between variable node and Check node.

III. Basic Error-Correction: Cascaded C-elements



Cascaded C-elements

- Cascaded C-element is able to correct a single error during one iteration
- It can correct many multi-error patterns as well when more iterations run.

A cascaded C-elements set used in LSD decoding method

					Initialization			In 1 Iteration		
	f_0	f_1	f_2	f_3	c_0	c_1	c_2	c_0	c_1	c_2
Error-Free	0	0	0	0	0	0	0	0	0	0
Error in f_0	1	0	0	0	0	1	0	0	0	0
Error in f_1	0	1	0	0	0	0	1	0	0	0
Error in f_2	0	0	1	0	0	0	0	0	0	0
Error in f_3	0	0	0	1	1	0	0	0	0	0

A Table that illustrates the behaviors of the error-correction

- LSD is a message-passing decoding method [2], where Variable nodes (Cascaded C-elements) and Check nodes (XOR Arrays) perform the message processing for the received codeword.

IV. Asynchronous Architecture of the LSD

Asynchronous Handshaking Signal Flow of the LSD Architecture

- **LSD's Error-Correction Steps:**

Initialization Phase:

1: Modulo-2 summation in XORs for the received codewords.
2: Cascaded C-elements initialize the memories with the incoming messages and perform a basic error-correction.

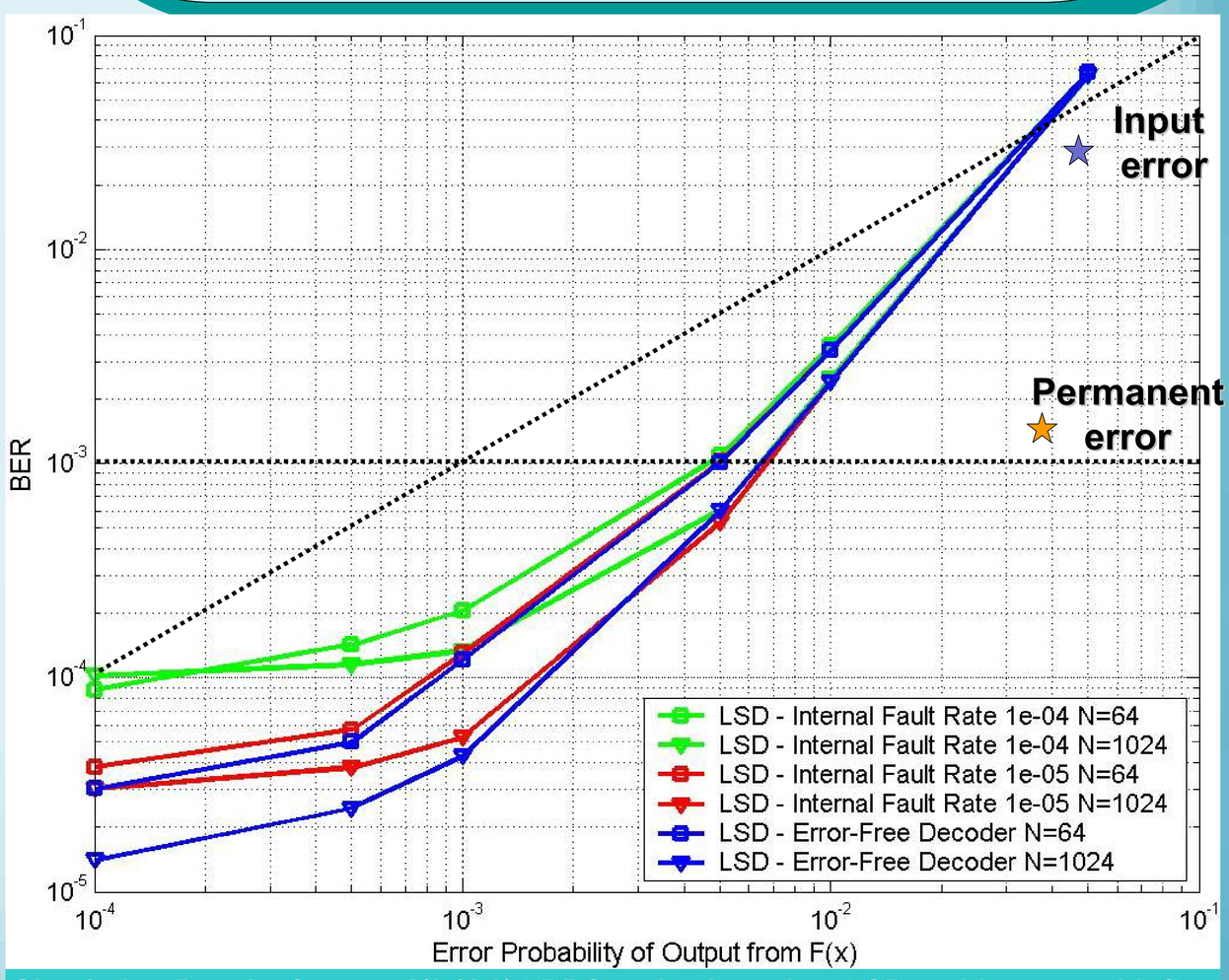
Error-Correction Phase:

3: Modulo-2 summation in XORs for the massages from C-elements.
4: C-elements perform the error-correction for incoming massages.
5: Iterate step 3 and step 4 during a number of iterations.
6: Output decoded codewords.

Rst: Indicates the decoding phase, *Switch*: to switch the phase, *flag*: signals to control the handshaking protocol.

V. Conclusion

- The originality of LSD method is to implement variable node's computations by C-elements that can resilient against internal faults.
- The LSD approach is able to reduce the error probability at the output of a combinational function by multiple orders of magnitude.



Simulation Results for rate-1/2 (4,8) LDPC codes based on LSD architecture with five iterations. The hard-fault rate for the output from F(x) and E•F(x) is 0.001.

References:
[1] D.E. Muller and W.S. Bertky, "A theory of asynchronous circuits," in Proc. International Symposium on the Theory of Switching, 1959.
[2] T.J. Richardson and R.L. Urbanke, "The capacity of low-density parity-check codes under massage-passing decoding," IEEE Transactions on Information Theory, 2001.