Computation on Unreliable Architecture

Yangyang TANG

29 January 2013

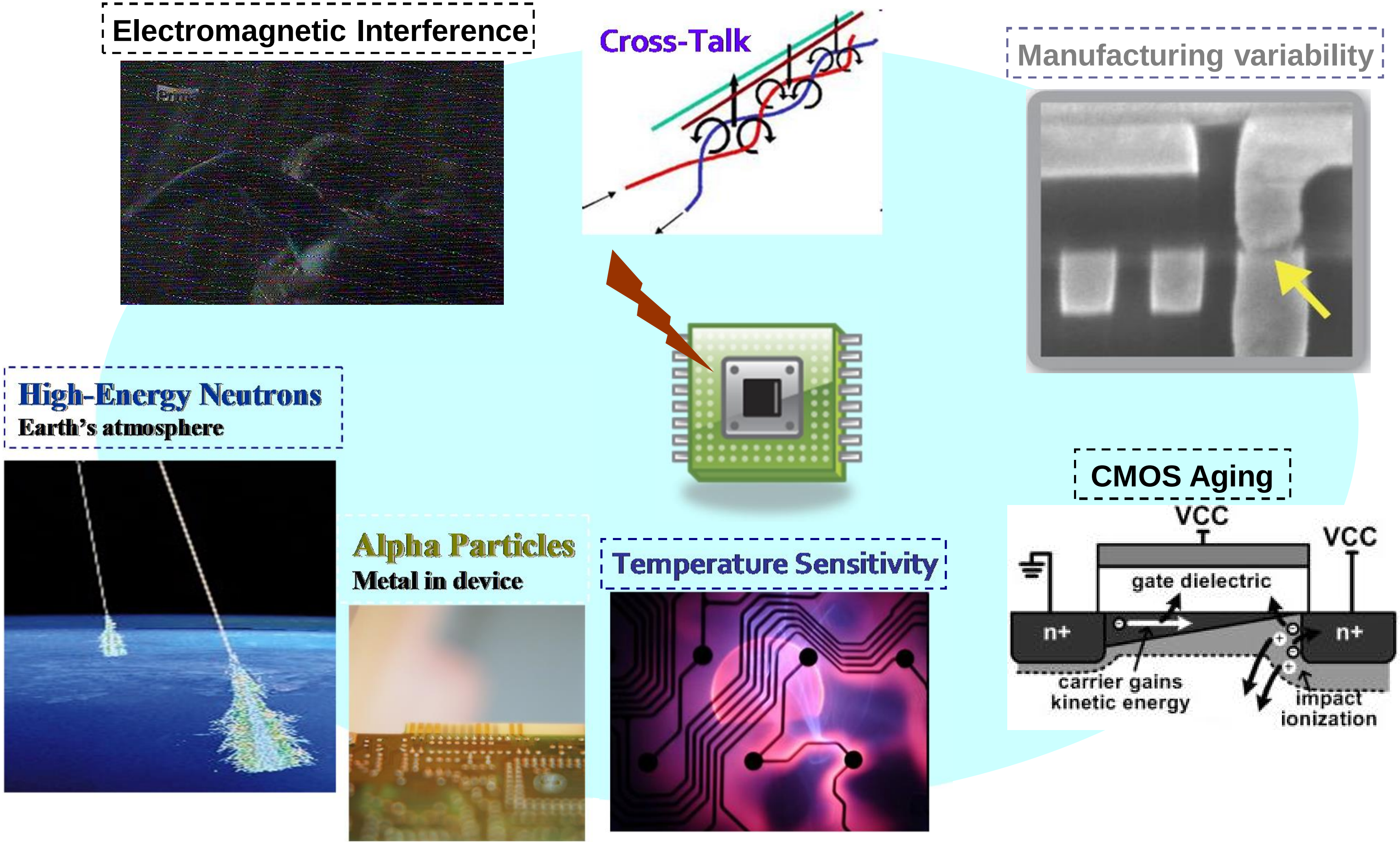
Supervised by *Pr. Emmanuel Boutillon*, Université de Bretagne Sud

Joint supervisions of *Pr. Christophe Jégo* Institut Polytechnique de Bordeaux

Pr. Michel Jézéquel Télécom Bretagne.

Electronic Devices

Major Concerns: Permanent Faults and Transient Faults

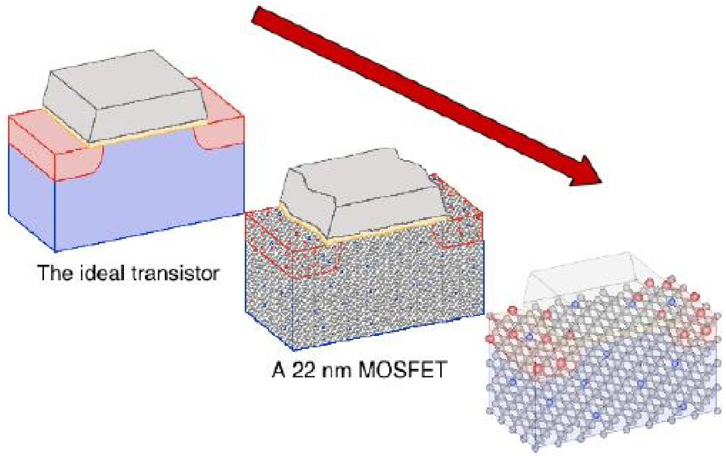


Size Shrinking in CMOS

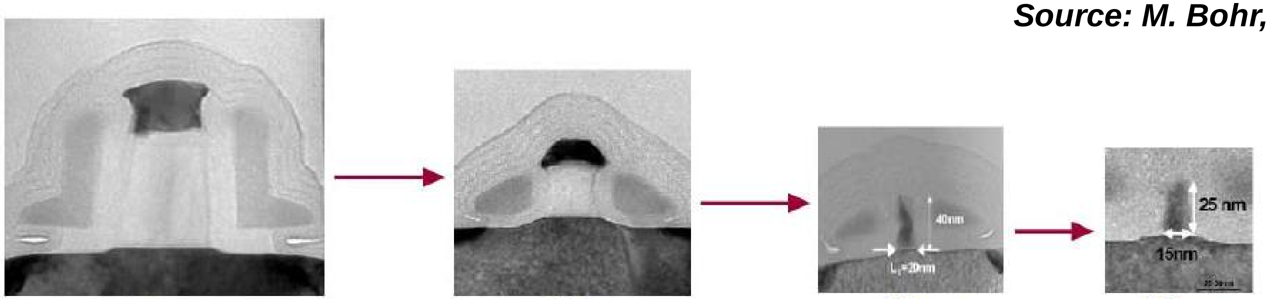
Scaling to atomistic scale devices…

Nanometer CMOS scaling problems:

- Noise problems (signal integrity)
- Leakage (digital)
- Channel length modulation
- Velocity saturation
- Mobility degradation
- Drain induced barrier lowering (DIBL)
- Parasitic effects
- **IC reliability**
- …



Source: M. Bohr, Intel, IRPS 2003



Roadmap 2011 Edition

International Technology Roadmap for Semiconductors
2011 Edition

ITRS

TABLE OF CONTENTS

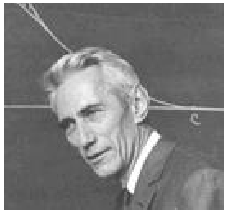
- **The Reliability Issue becomes major concern to semiconductors.**
- **In fact, with deep-micro or nanoscale:**
  - **Error free component becomes more and more expensive**
  - **More and more Transient Faults**

Outline

- Introduction
- Formal Models of Embedded ECC
- Architectural Approaches
- Experimental Results
- Conclusion & Future Perspectives

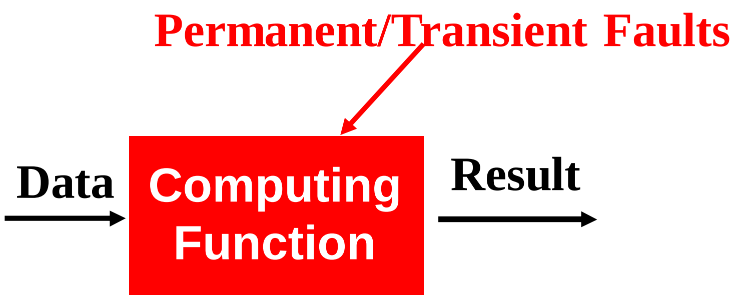
Question on Reliable Computation...

- Can we make reliable communication?



- YES! According to Shannonõs theory (with a rate limit)

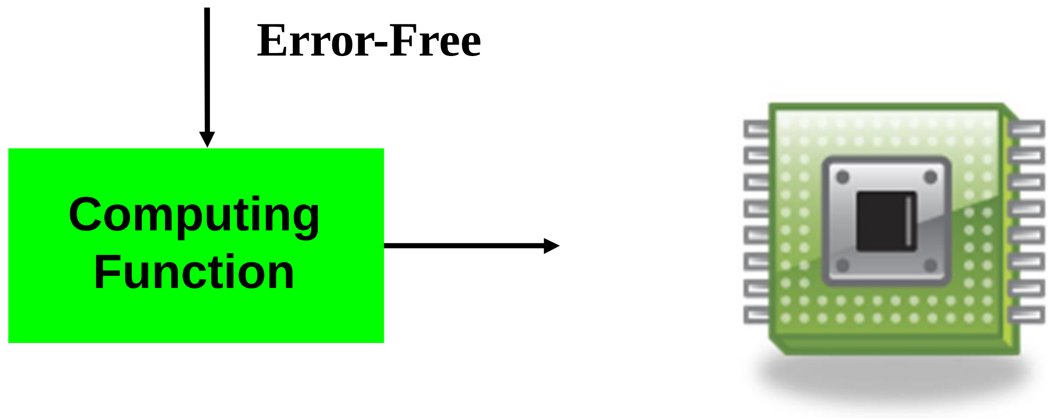
Ref: Mathematical theory of communication, C. E. Shannon, 1948.



- Can we make reliable computation?
- Uh, still an open questioné é

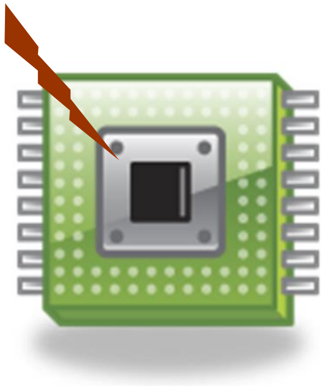
Phases of Computation on Circuit

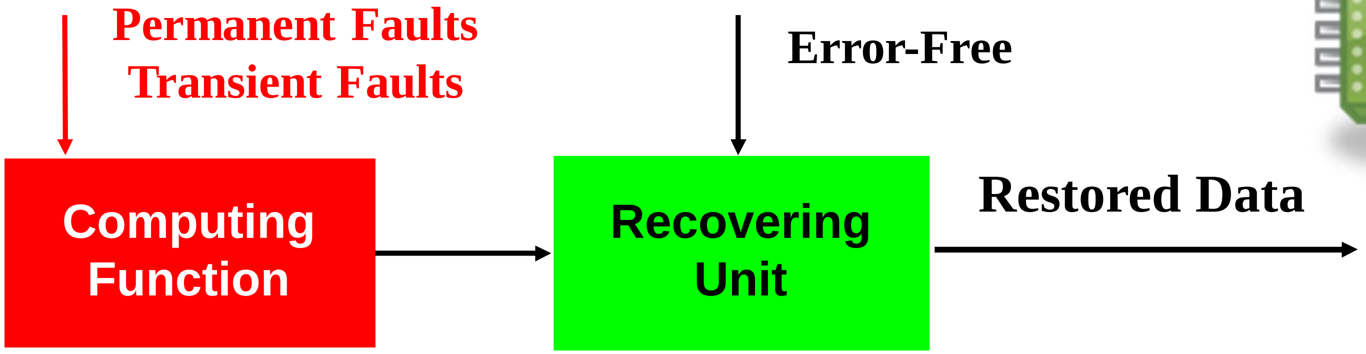
- **Phase 1: Reliable, Computation is Error-Free.**



Phases of Computation on Circuit

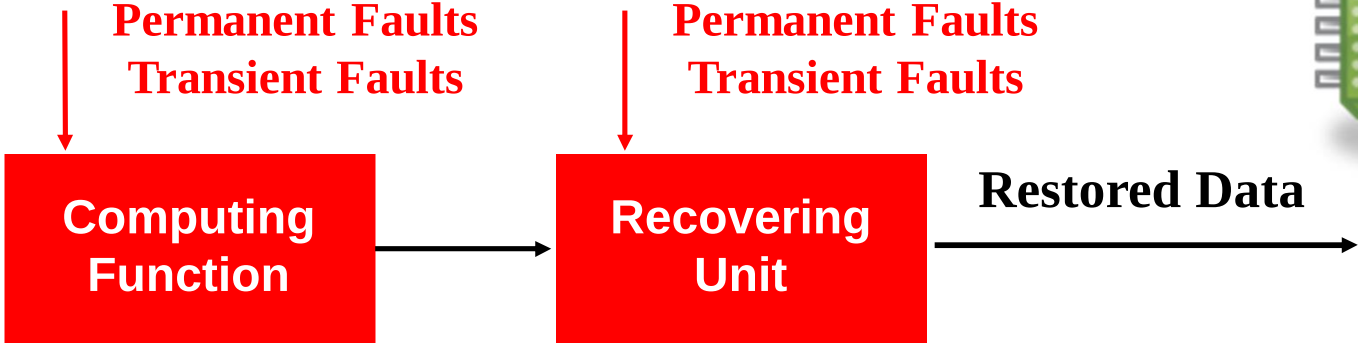
- **Phase 2: Computation is Erroneous, recovering is Error-Free.**

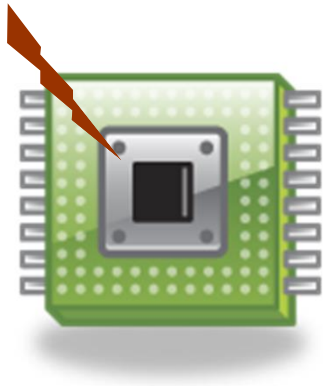




Phases of Computation on Circuit

- **Phase 3: Computation is Erroneous, recovering is Erroneous as well.**





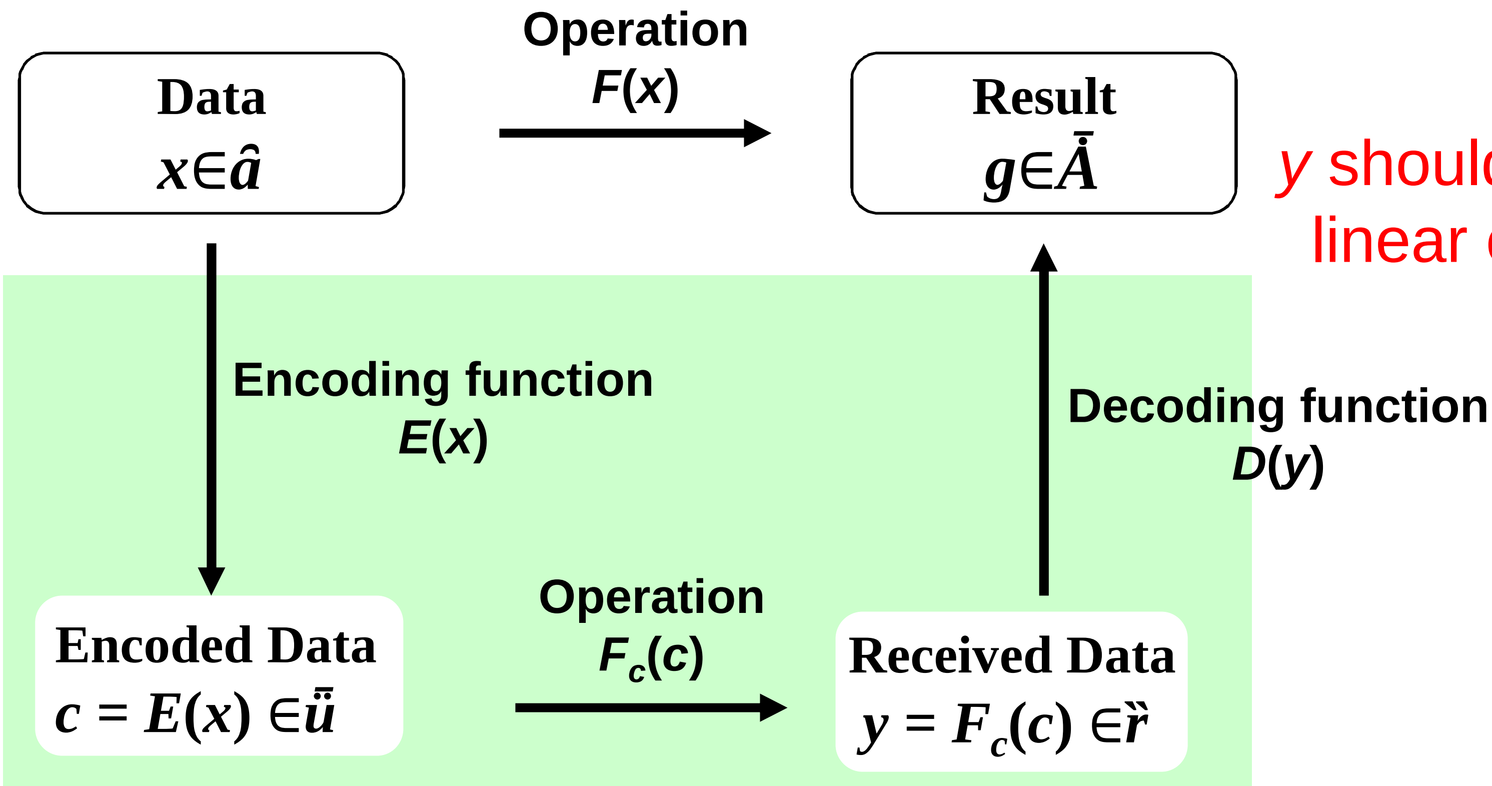
Objective

- **Approaches for reliable computation on unreliable circuit**
- **To design an efficient error-resilient architecture by ECC**
- **For contemporary logics and future nanoelectronic systems**

Outline

- Introduction
- Formal Models of Embedded ECC
- Architectural Approaches
- Experimental Results
- Conclusion & Future Perspectives

Formal Model for the application of ECC



y should be a linear code

Solution 1: Group Homomorphism From *E*(*x*) and $F_c(c)$

➤ **Whatôs a group Homomorphism?**

Let (*â*, ★ star**) and (*ū*,⁎** asterial**) be two groups with an internal composition law.**

**A group homomorphism from (*â*, ★) and (*ū*,⁎) is a morphism E: *â Ÿ ū* such that ∀ (x_1 , x_2) ∈ *â*², **

$$E(x_1 \star x_2) = E(x_1) * E(x_2)$$

Solution 1: Group Homomorphism From $E(x)$ and $F_c(c)$

- **A group Homomorphism**

$$E(x_1 \star x_2) = E(x_1) * E(x_2)$$

$F_c(c)$ = Group (â, ★)

(x_1, x_2)∈â^2 —— **Operation ★** ——→ $(x_1 \star x_2)$∈â^2

***E* as a morphism**
***E*: âŸ ǖ** (↓)

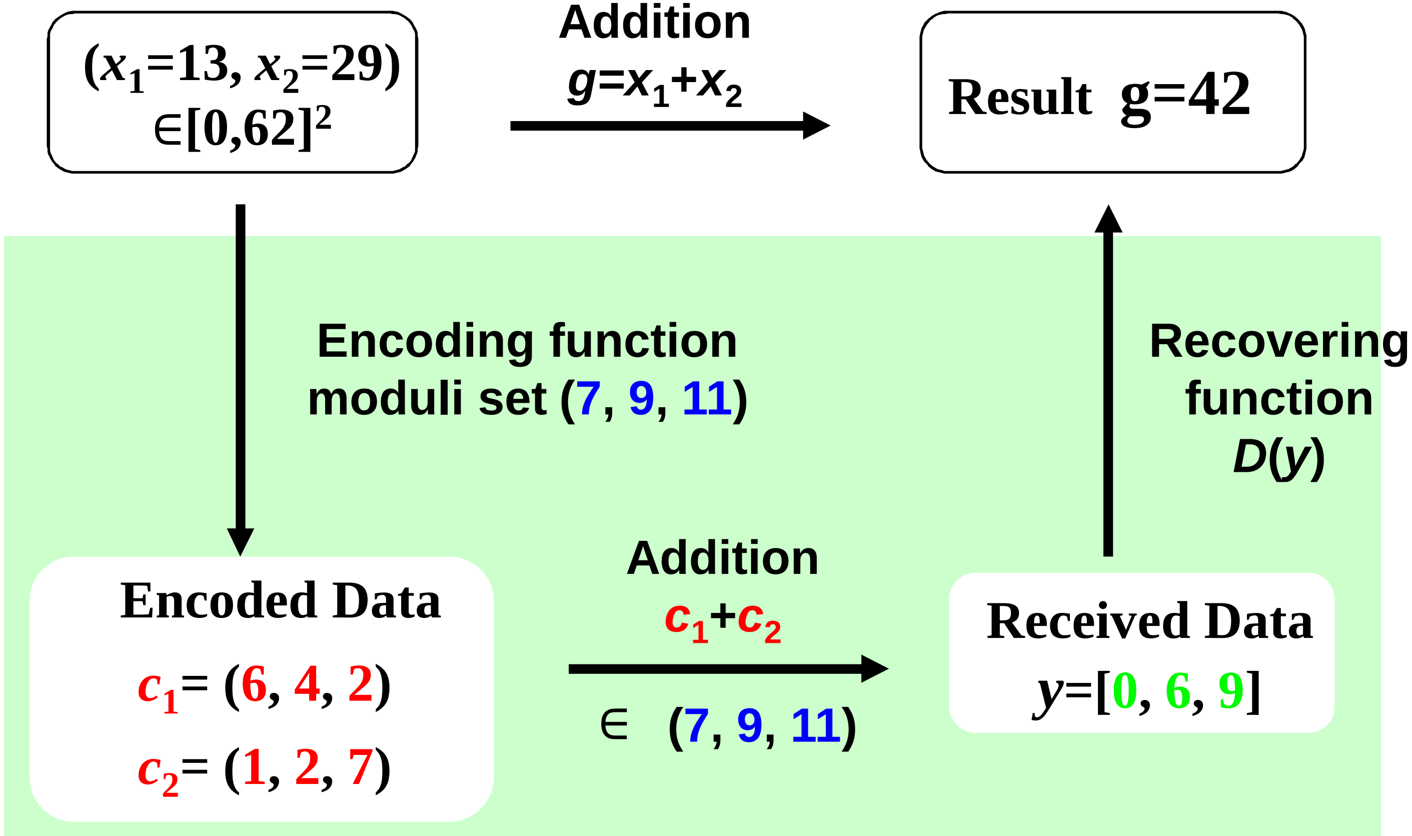
According to the property of a group homomorphism (↕)

Group (ǖ, ⁎)

$(E(x_1), E(x_2))$∈ǖ^{2} —— **Operation ⁎** ——→ $(E(x_1) \ast E(x_2))$∈ǖ^{2}

Solution 1: Example – Residue Number System (RNS)

$42=(42 \bmod 7 = 0,\ 42 \bmod 9 = 6,\ 42 \bmod 11 = 9)$



$c_1=(13 \bmod 7 = 6,\ 13 \bmod 9 = 4,\ 13 \bmod 11 = 2)$

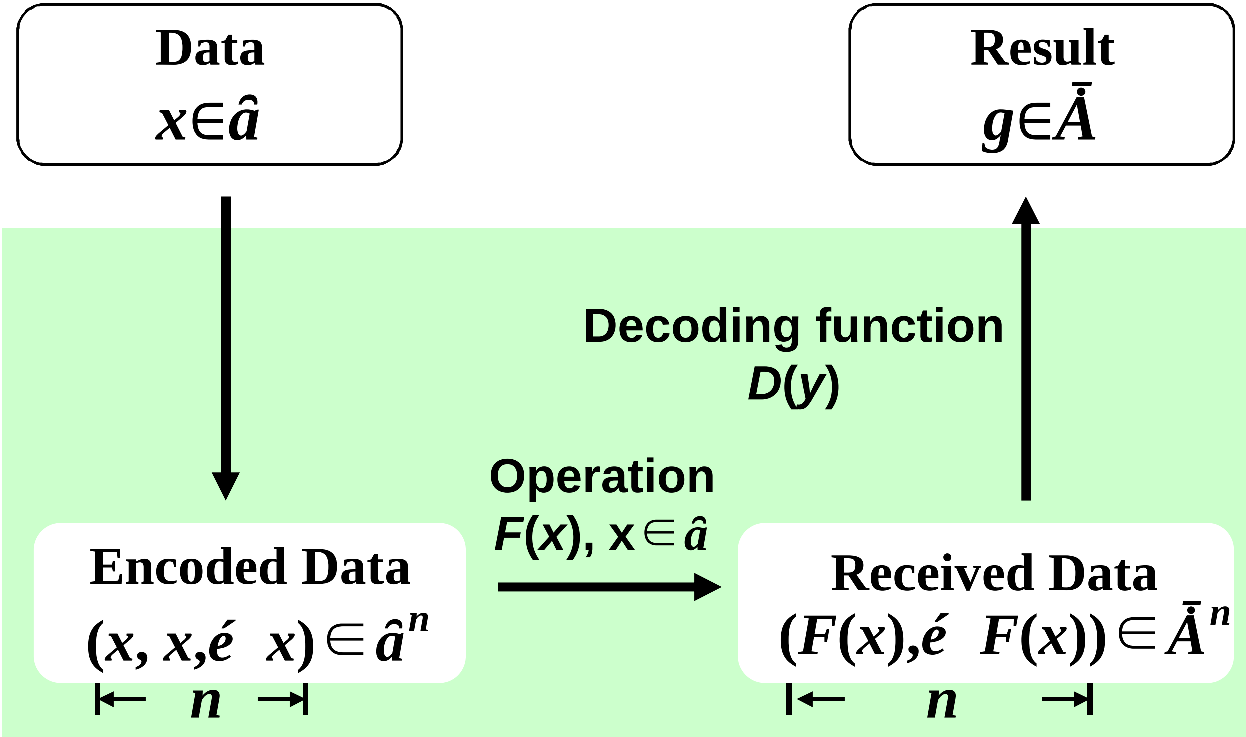
$c_1+c_2 =(6+1 \bmod 7 = 0,\ 4+2 \bmod 9 = 6,\ 2+7 \bmod 11 = 9)$

Solution 1:Contribution-RNS based Fault-Tolerant method

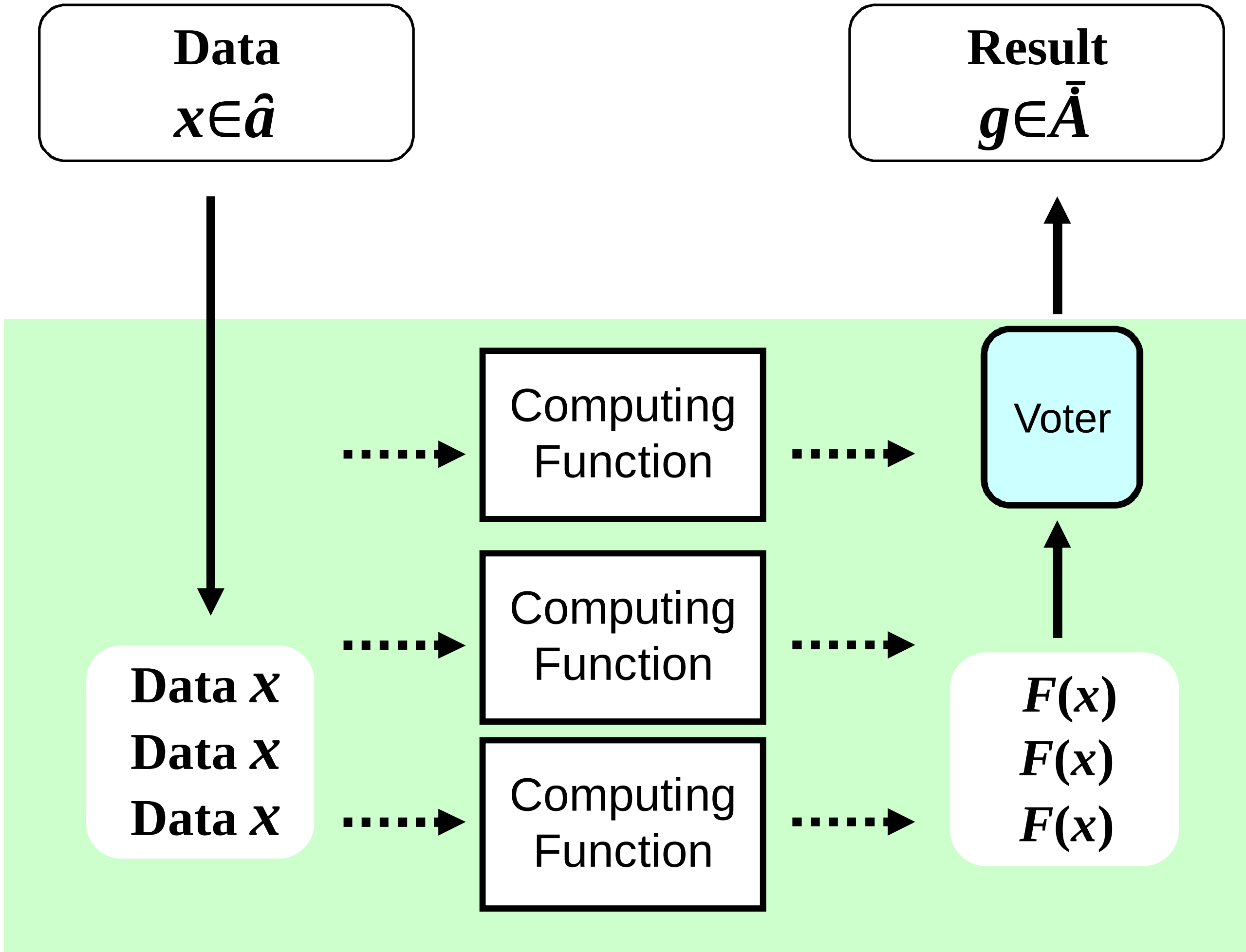
- A contribution of this thesis work:
- An arithmetic fault-tolerant method
- Redundant Residue Number System, called BRRNS
- Fast computation & Error-resilience

- ***Yangyang Tang***, *Emmanuel Boutillon, Christophe Jégo, and Michel Jézéquel* ñA new single-error correction scheme based on self-diagnosis residue number arithmetic,ò Design and Architectures for Signal and Image Processing (DASIP), 2010.

Solution 2: Repetitions of Computation

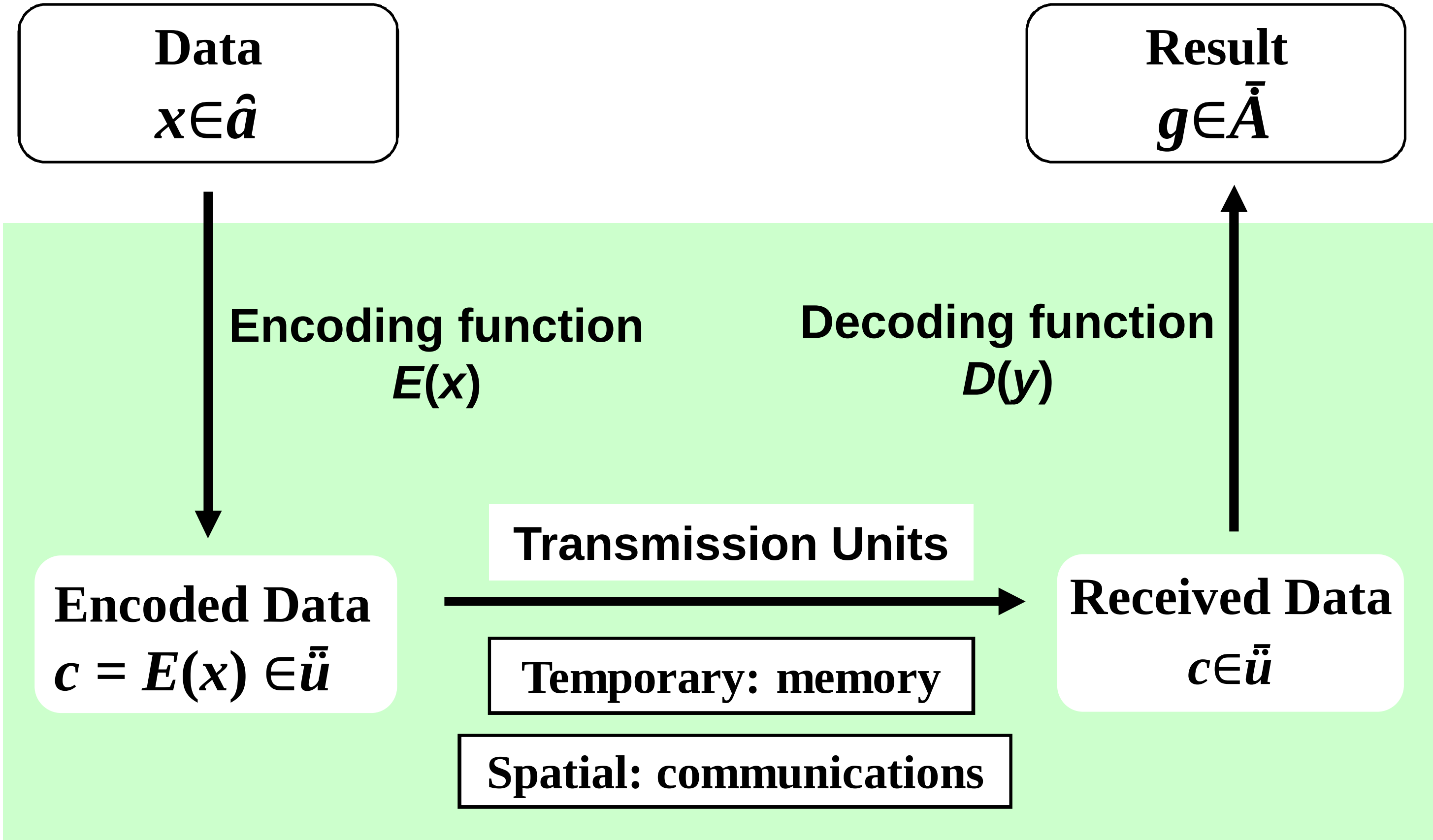


Solution 2: Example – Triple Modular Redundancy (TMR)

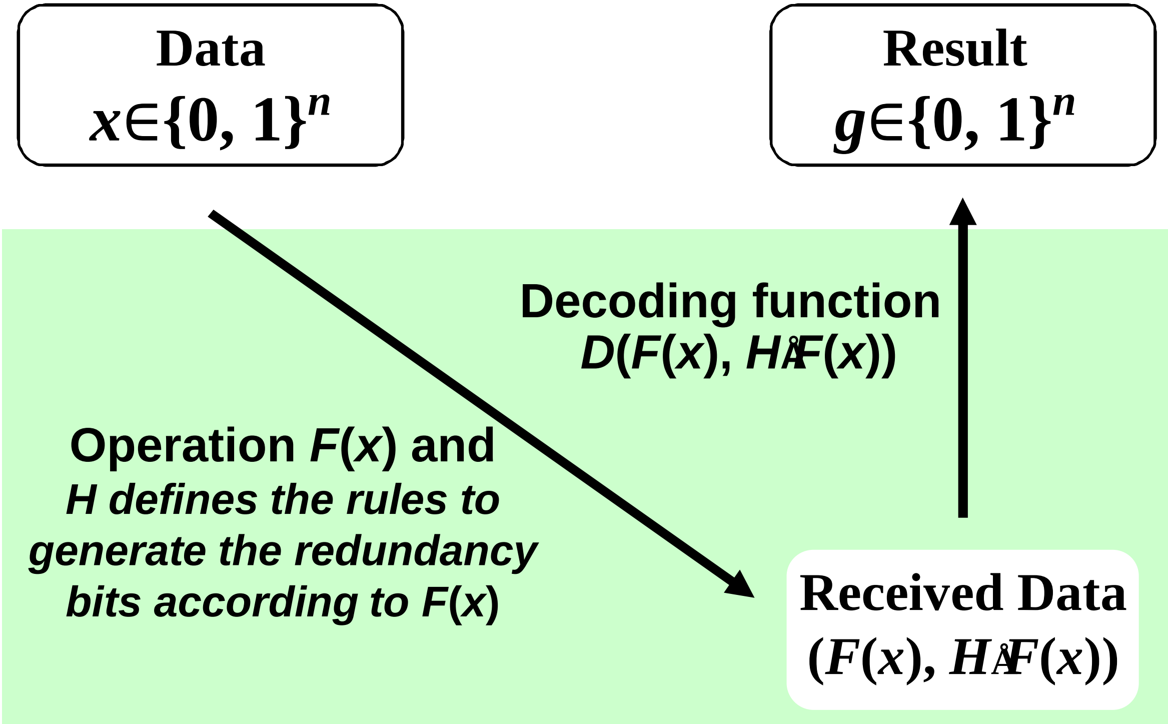


Ref: R. Lyons et al., IBM Journal, 1962

Solution 3: Applying ECC Directly

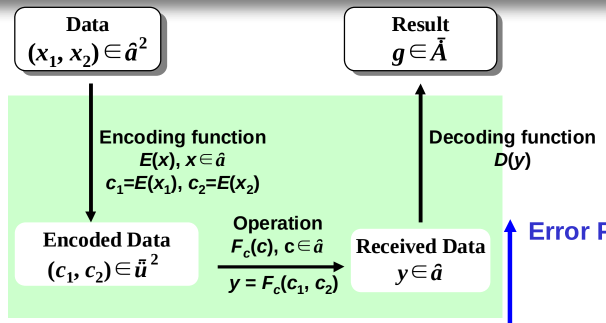


Solution 4: Function & Parity-Mapped Function & ECC

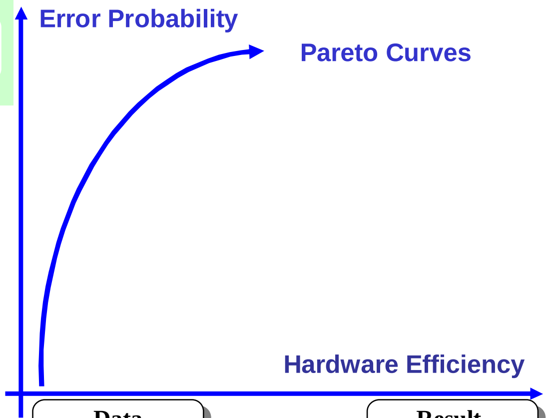


Contributions

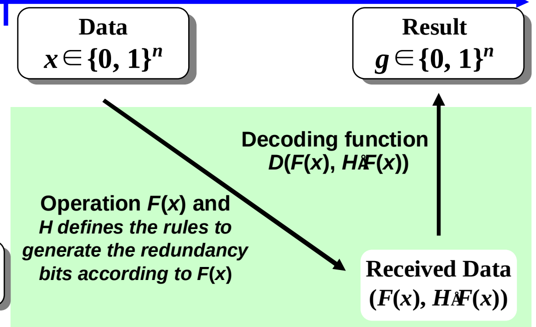
- Contribution 1:
- A RNS based fault-tolerant method
- Fast computation & Error-resilience



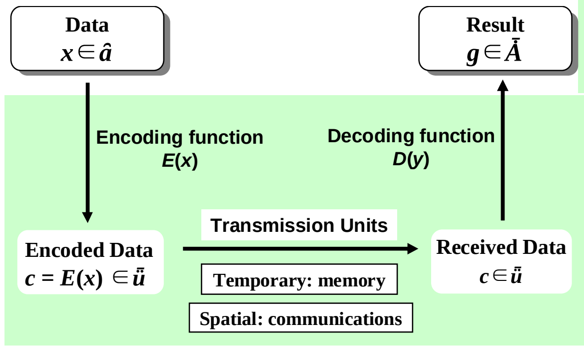
- Contribution 2:
- Reliability-Efficiency Criteria
- Estimations on Reliability and Efficiency



- Contribution 3:
- A fault-tolerant system for computation
- Function & Parity-Mapped Function & ECC



- Contribution 4:
- A decoder against internal faults
- Efficient Error-correction in the presence of high error rate noise



Outline

- Introduction
- Formal Models of Embedded ECC
- Architectural Approaches
  - Reliability-Efficiency Criteria (RE-Criteria)
  - A Fault-Tolerant System for Computation (cDMR)
  - A Decoder Against Internal Faults (MCD)
- Experimental Results
- Conclusion & Future Perspectives

Contribution 2: Reliability-Efficiency Criteria (RE-Criteria)

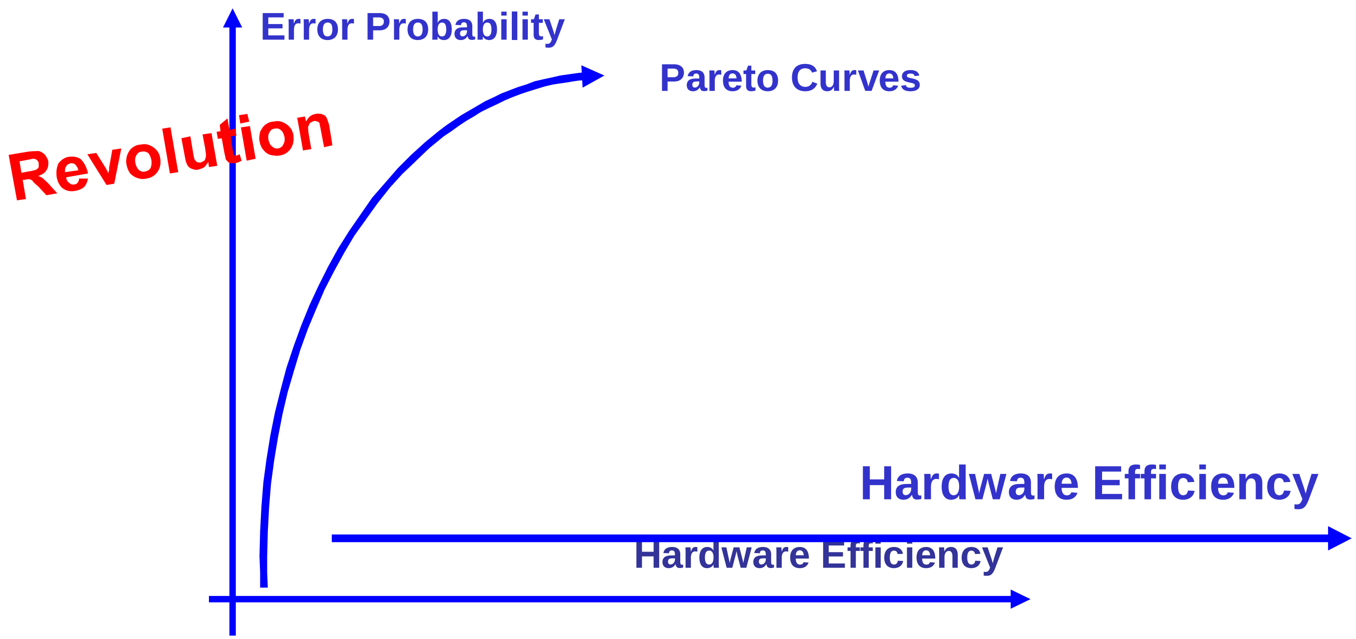
Ref: Y. Tang et al., SiPS, 2011

- **Reliable Electronics:**
  - ✓ **Designer objective in signal processing applications**

    Efficiency = Number of operation / (Area unit x Time unit)

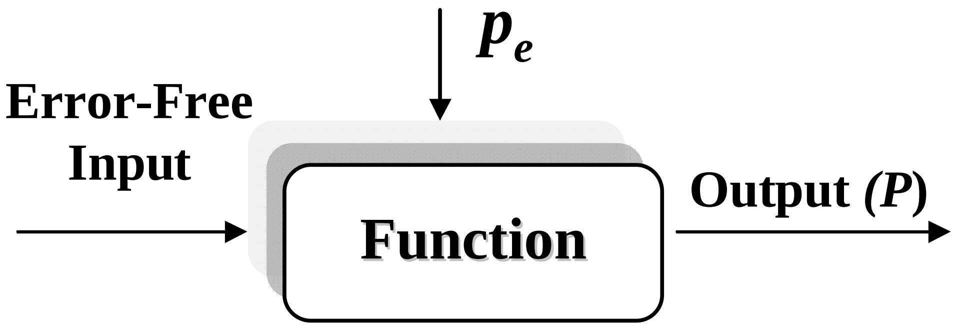
- **Unreliable Electronics: (Occurrence of transient errors)**
  - ✓ **New designer objective**

    Compromise of Efficiency to Reliability (Error Probability)



Contribution 2: Reliability Criterion

- **Model of Error (Worst-Case)**



Hypothesis 1 – Isotropy: an overall ñunitaryò error probability p_e as constant (in an unity area during one clock cycle)

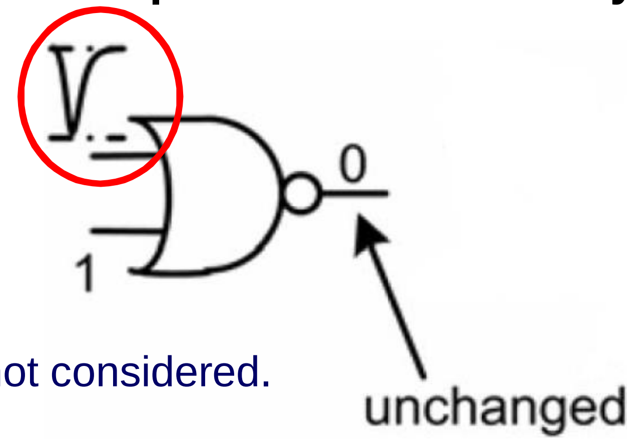
Hypothesis 2 ï Contamination: All errors lead to an output error.

Hypothesis 3 ï Irreversibility: Two successive errors cannot lead to a correct result.

Contribution 2: Worst-Case hypothesis

Hypothesis 2 ï Contamination:

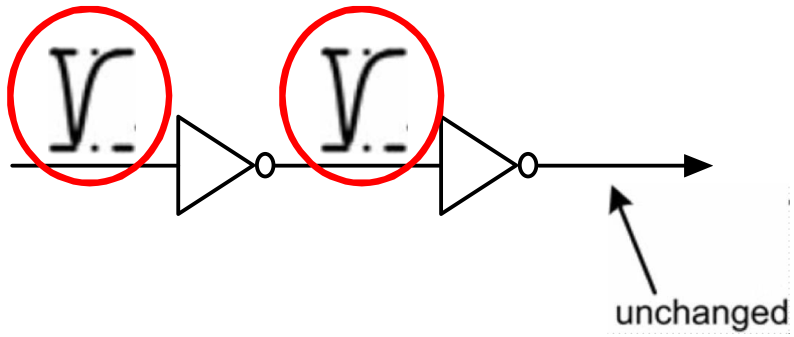
All errors lead to an output error is not always true



e.g., logical mask is not considered.

Hypothesis 2 ï irreversibility:

Two successive errors cannot lead to a correct result.

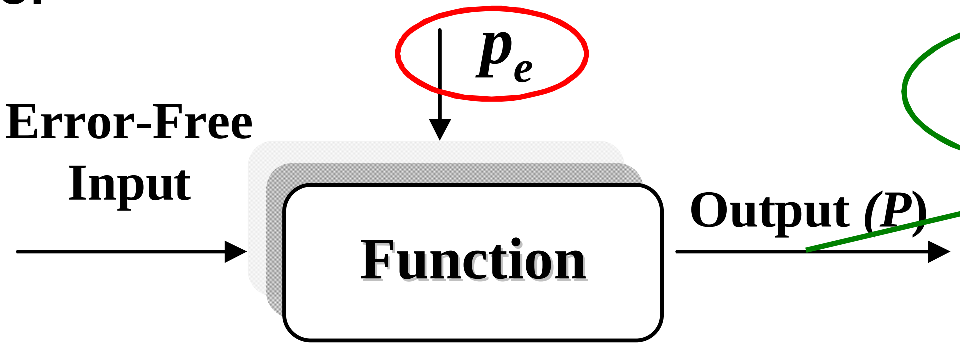


Contribution 2: Reliability Criterion

- **Model of Error**

Å**Probability that Output is correct:**

$$(1 - p_e)^{a \cdot t}$$



Å**ñunitaryò error probability p_e (in an unity area during one clock cycle)**

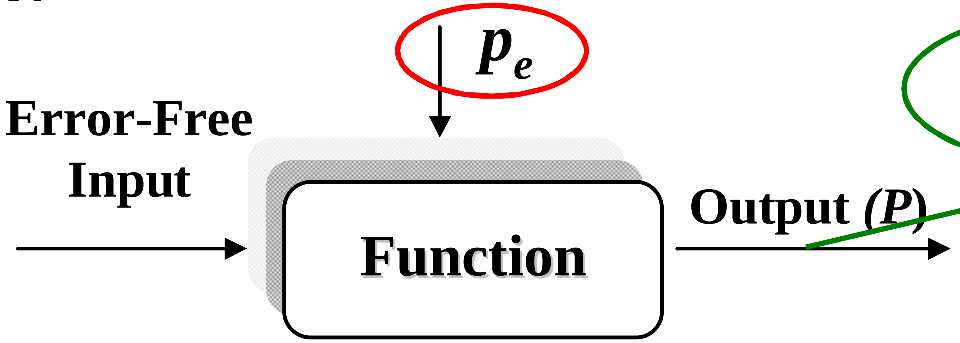
Å**A function of *a* unity area requiring *t* clock cycles to generate an output**

Contribution 2: Reliability Criterion

- **Model of Error**

ÅProbability that Output is correct:

$$(1-p_e)^{a \cdot t}$$



Åñunitaryò error probability p_e (in an unity area during one clock cycle)

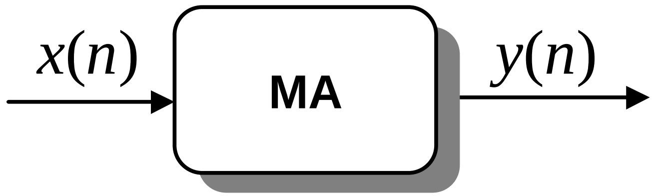
ÅA function of *a* unity area requiring *t* clock cycles to generate an output

ÅResulting error probability

$$P_{(a,\,t)} = 1-(1-p_e)^{a \cdot t}$$

Contribution 2: Example of Formal Derivation of $P_{(a,t)}$

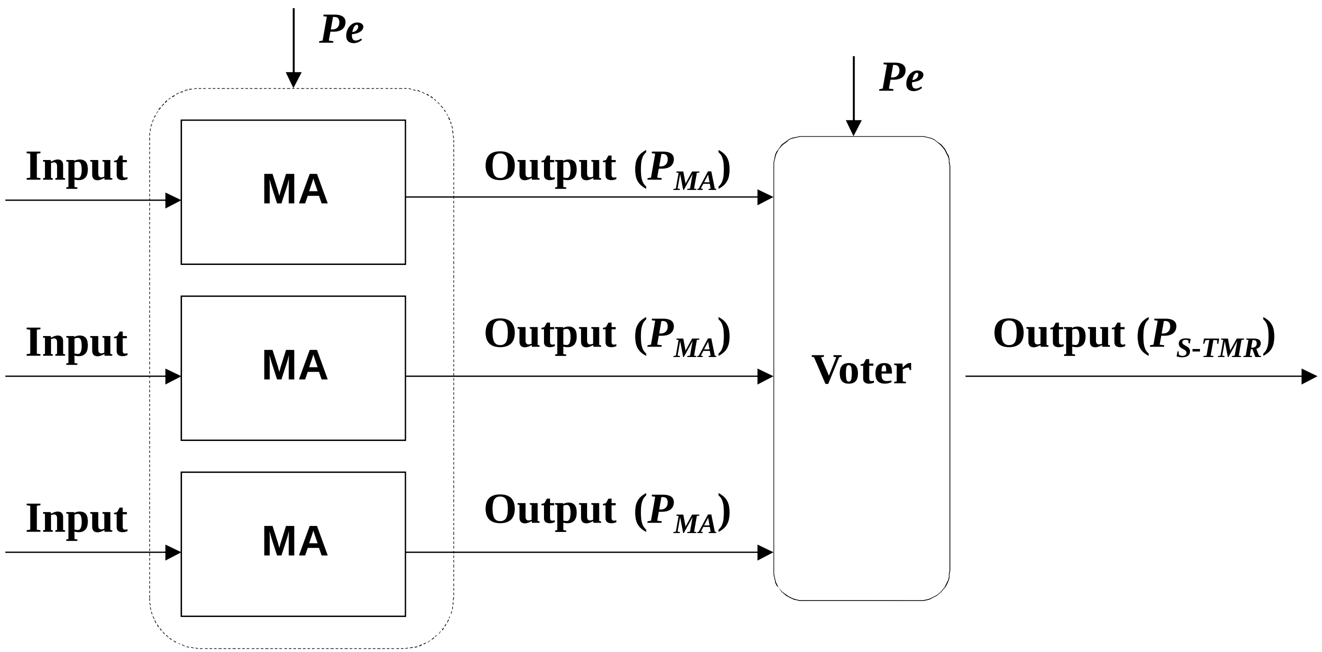
- **A pipeline Moving Average (MA) filter is considered as a case study.**



$$y(n) = \sum_{i=0}^{N} x(n-i)$$

Contribution 2: Example of Formal Derivation of $P_{(a,t)}$

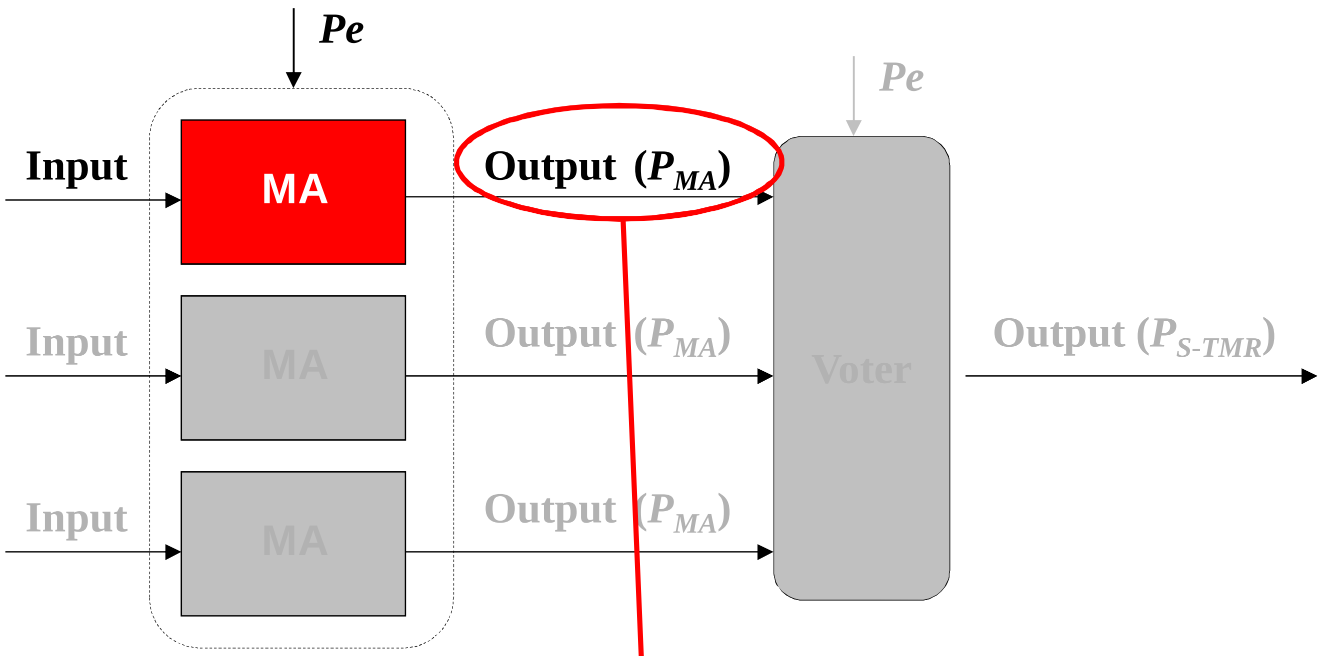
- A pipeline Moving Average (MA) filter is considered as a case study.



Spatial-TMR based MA filter structure

Contribution 2: Example of Formal Derivation of $P_{(a,t)}$

ÅLet P_{MA} be the error probability in a single module of size ***a*** that performs a computation in ***t*** cycles.

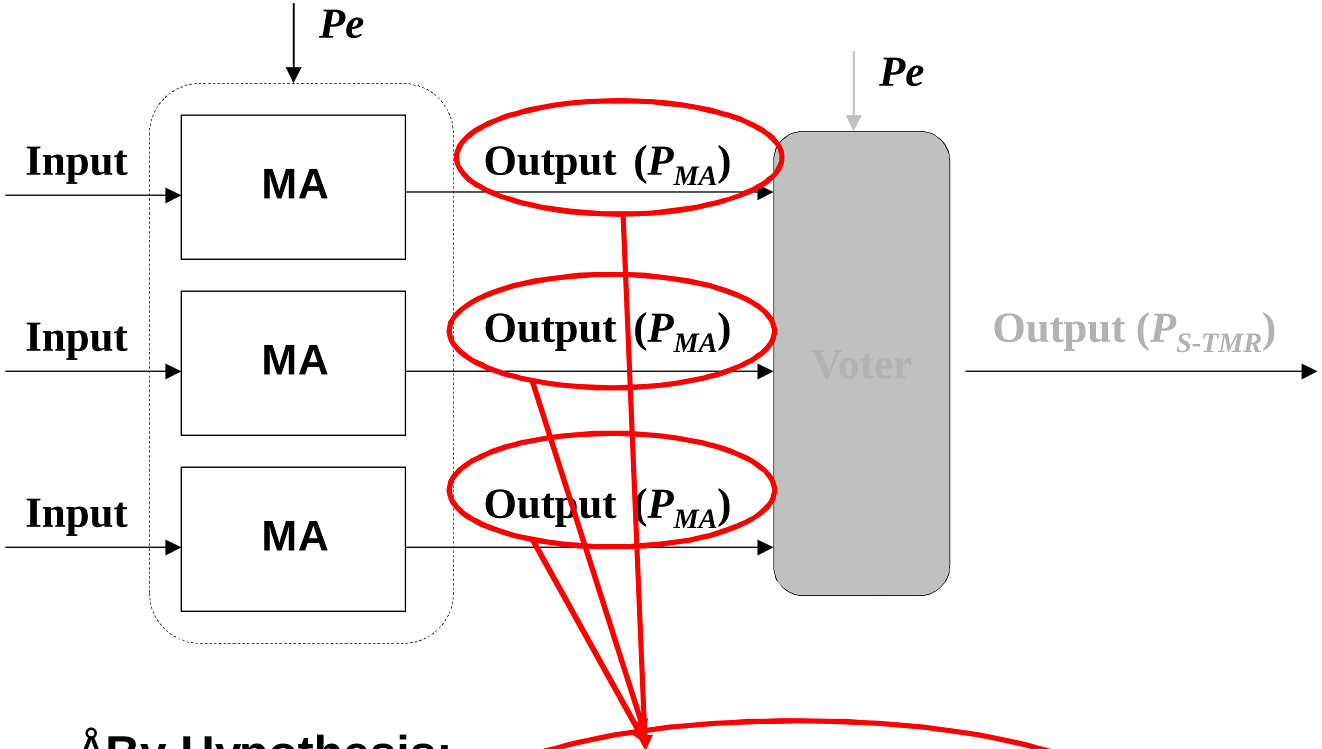


ÅBy Hypothesis:

$$P_{MA} = 1 - (1 - p_e)^{a \cdot t}$$

Contribution 2: Example of Formal Derivation of $P_{(a,t)}$

- Let P_{MA} be the error probability in a single module of size a that performs a computation in t cycles.

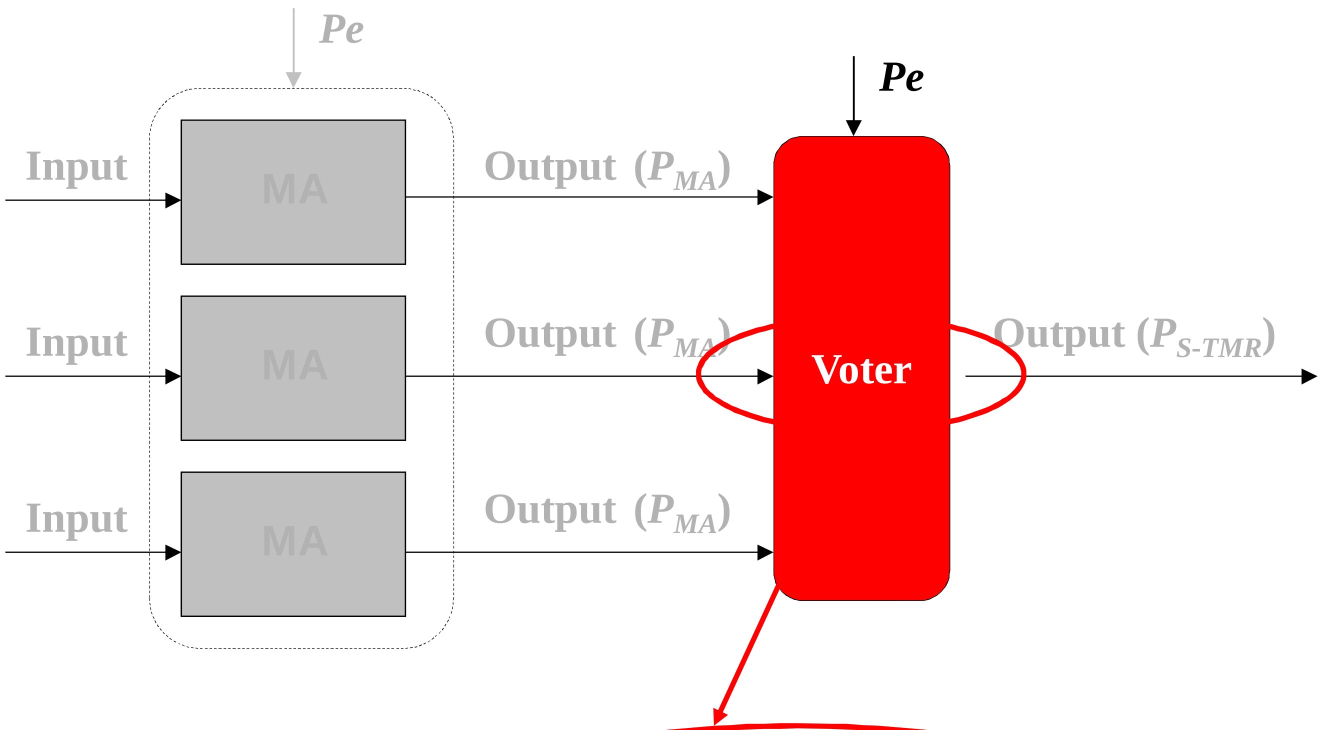


- By Hypothesis:

$$P_{MA} = 1-(1-p_e)^{a \cdot t}$$

Contribution 2: Example of Formal Derivation of $P_{(a,t)}$

- **Event of probability *V*: voter is faulty and a_v as the area unit of voter.**

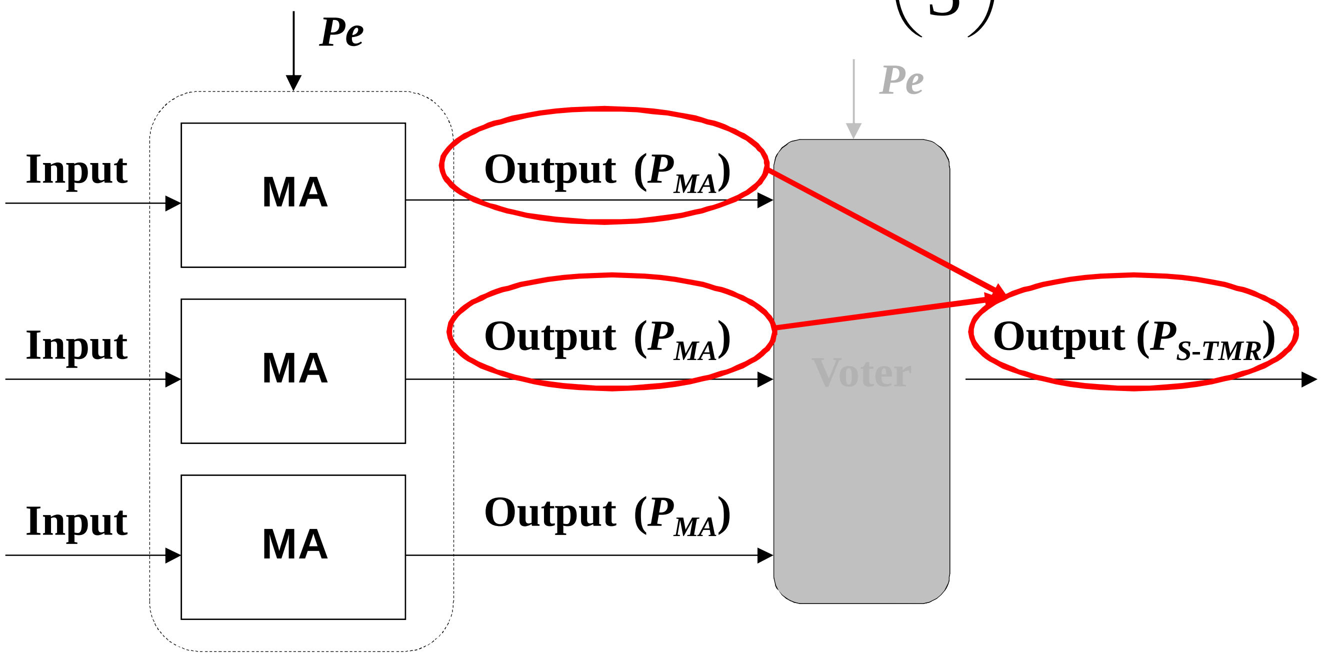


- **By Hypothesis:** $V = 1\text{-}(1 - p_e)^{a_v}.$

Contribution 2: Example of Formal Derivation of $P_{(a,t)}$

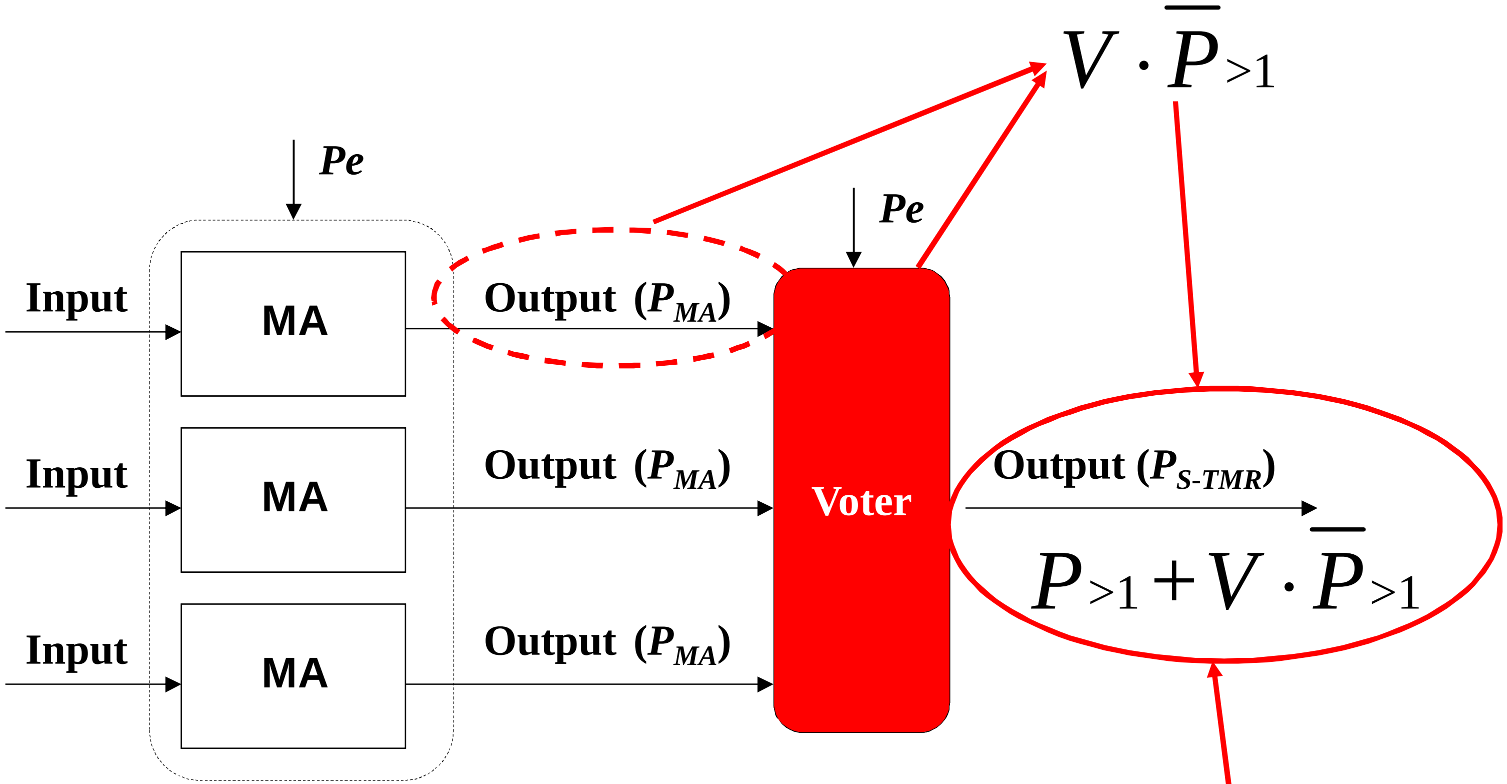
- Event of probability $P_{>1}$: at least two faulty modules

$$P_{>1} = \binom{2}{3} \cdot P_{MA}^2 \cdot \overline{P}_{MA} + P_{MA}^3$$



Contribution 2: Example of Formal Derivation of $P_{(a,t)}$

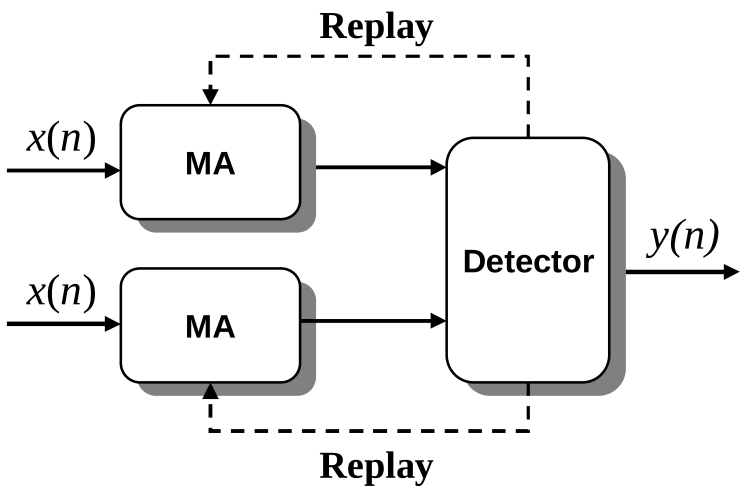
- Event of probability: only one faulty module and voter is faulty



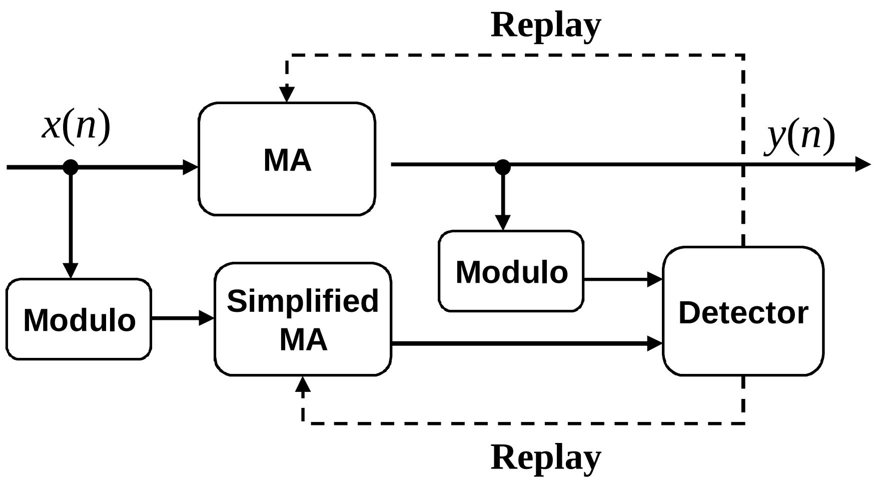
- Event of probability $P_{>1}$: at least two faulty modules

$$P_{>1} = \binom{2}{3} \cdot P_{MA}^2 \cdot \overline{P}_{MA} + P_{MA}^3$$

Contribution 2: Strategy of Error-Correction for MA Filter



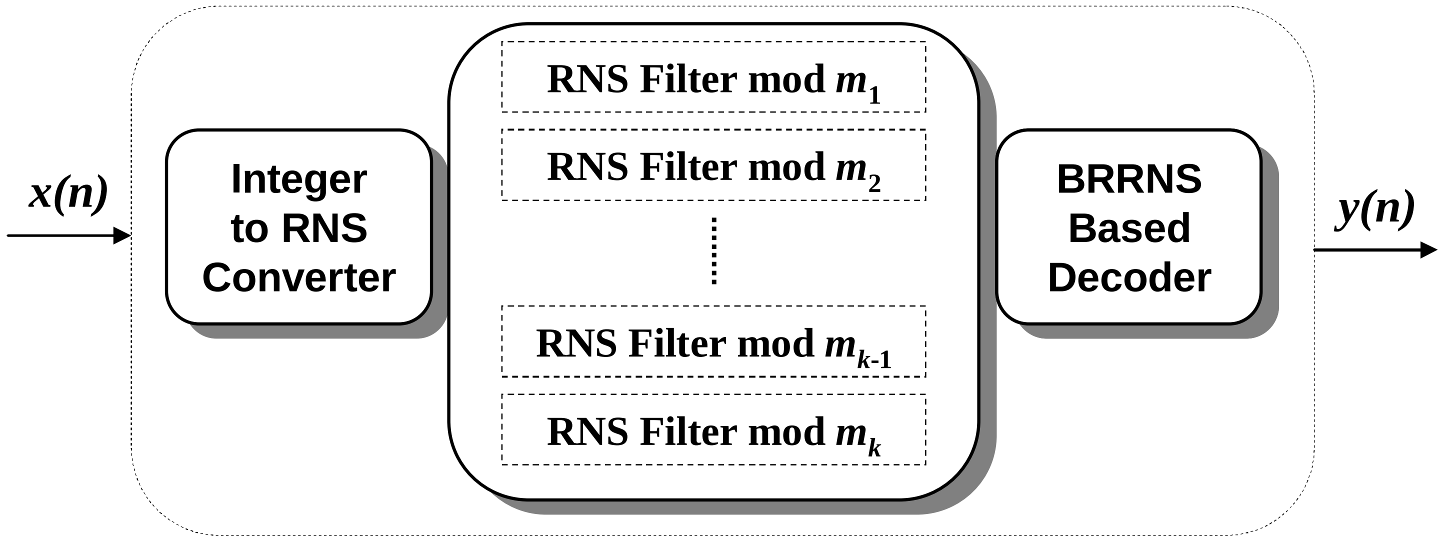
Dual Modular Redundancy (DMR) based ARQ



Modulo operation detecting based ARQ

Ref: S. LIn et al., IEEE Mag., 1984

Contribution 2: Strategy of Error-Correction for MA Filter



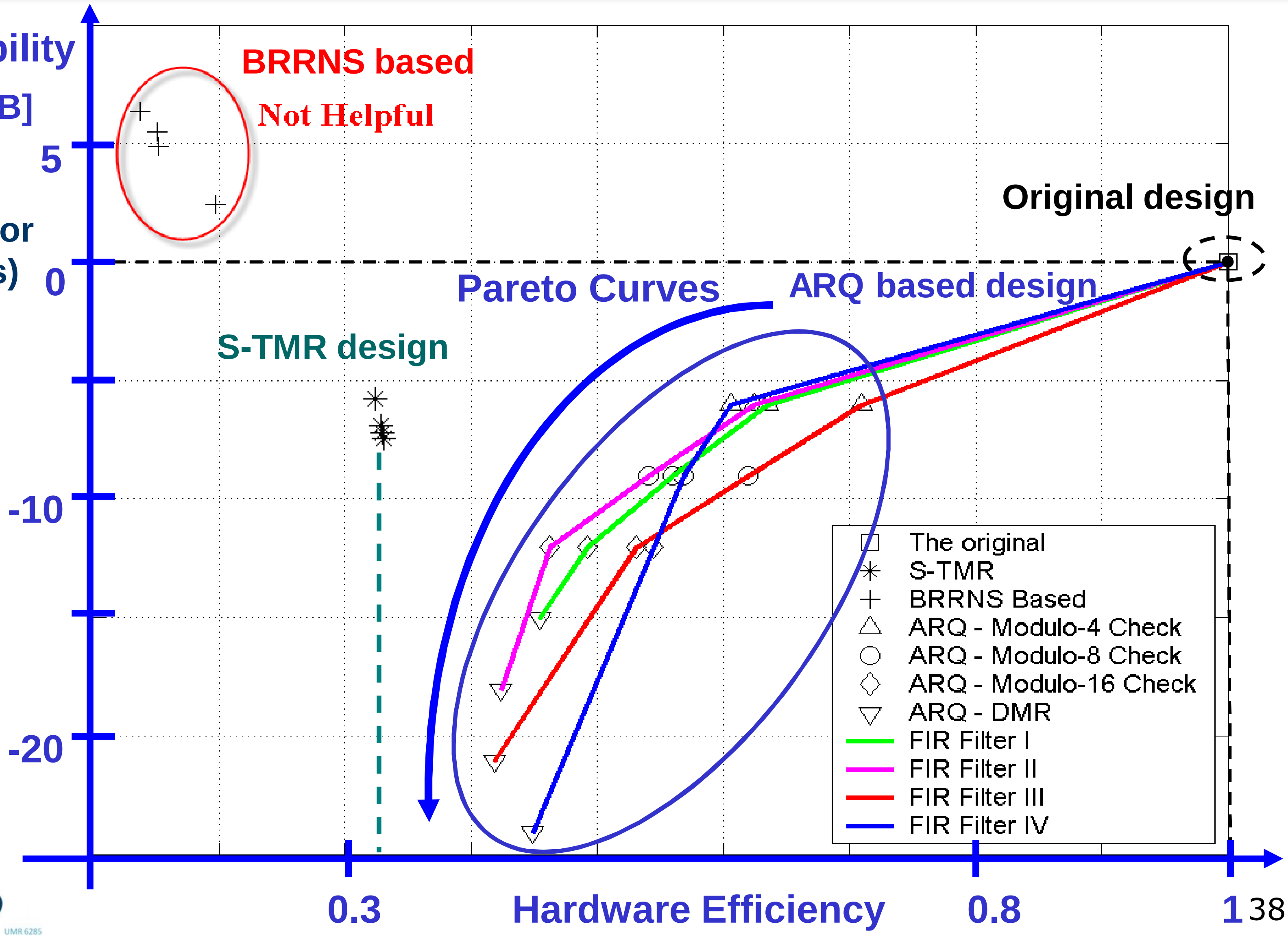
Filter implemented in RNS arithmetic with a BRRNS based decoder

Ref: Y. Tang et al., DASIP, 2010

Ref: Y. Tang et al., SiPS, 2011

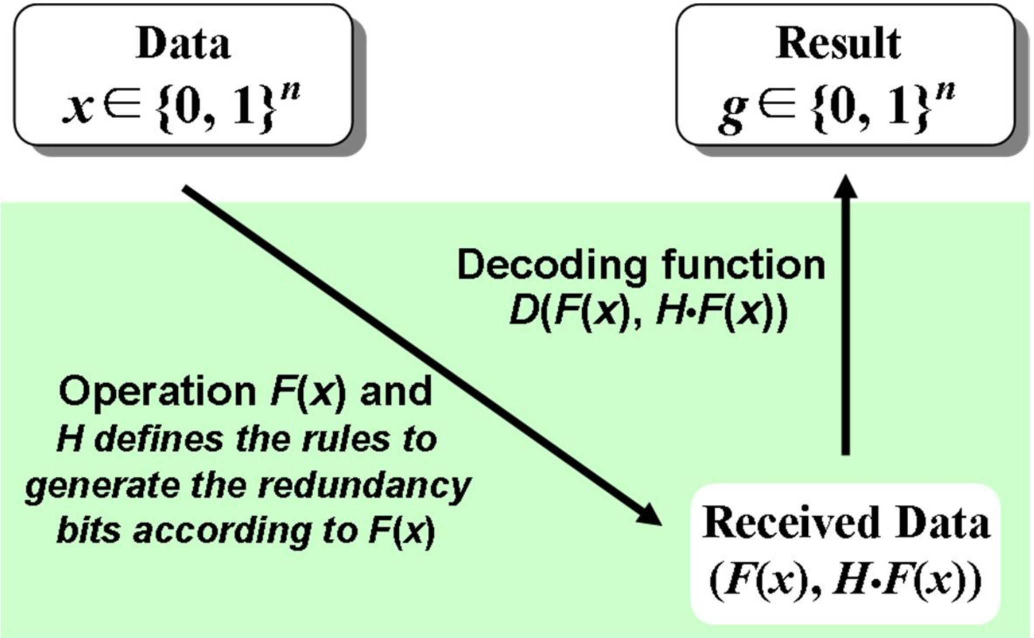
Contribution 2: Experimental Results – Pareto Curves

- Model of Error (3 hypotheses)
- Estimations on Efficiency and Error Probability
- Pareto Curves
- Trade-off Efficiency Vs. Reliability

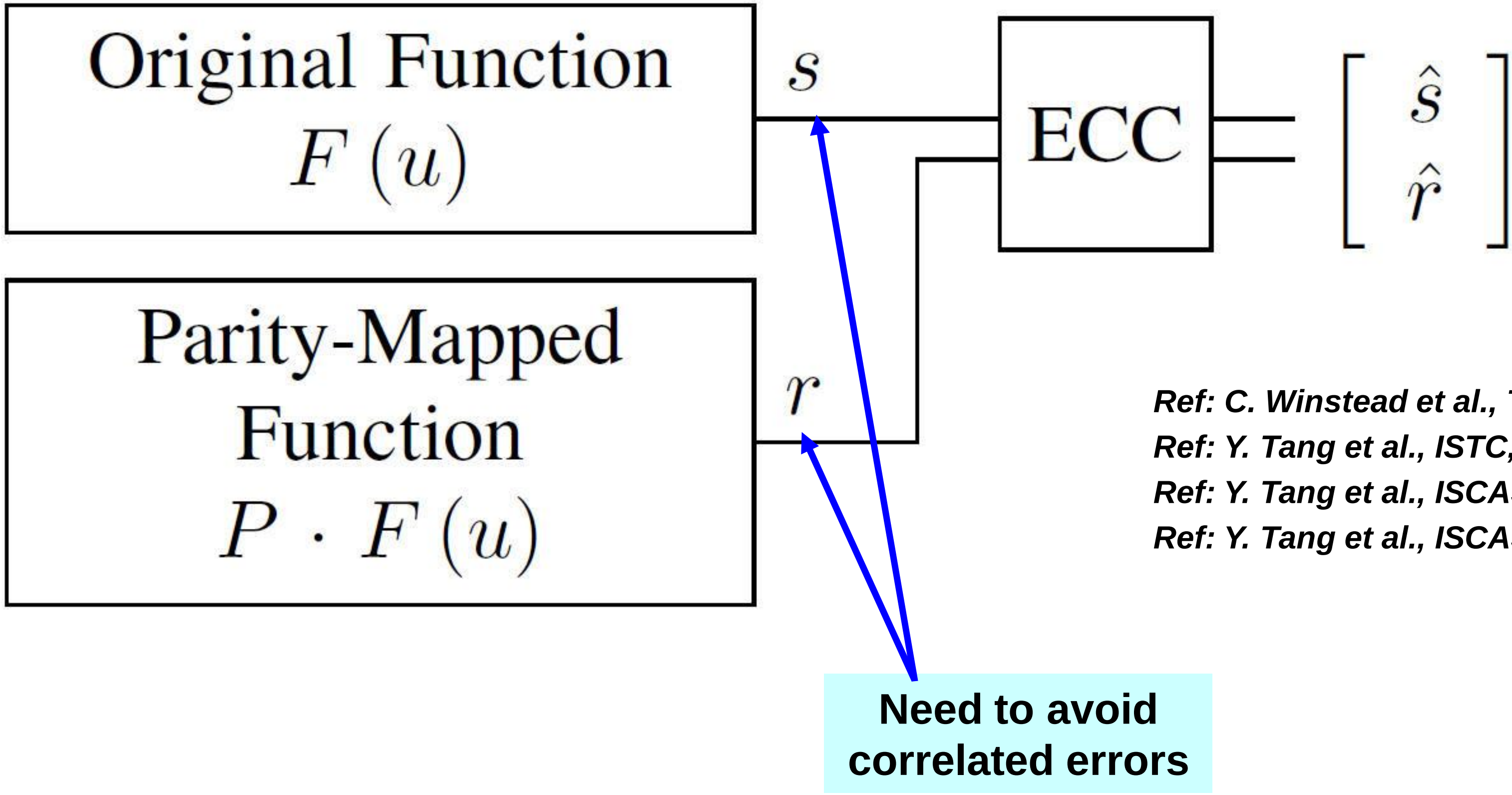


Proposed Work

- Contribution 3:
- A fault-tolerant system for computation
- Function & Parity-Mapped Function & ECC



Contribution 3: Coded Dual Modular Redundancy (cDMR)



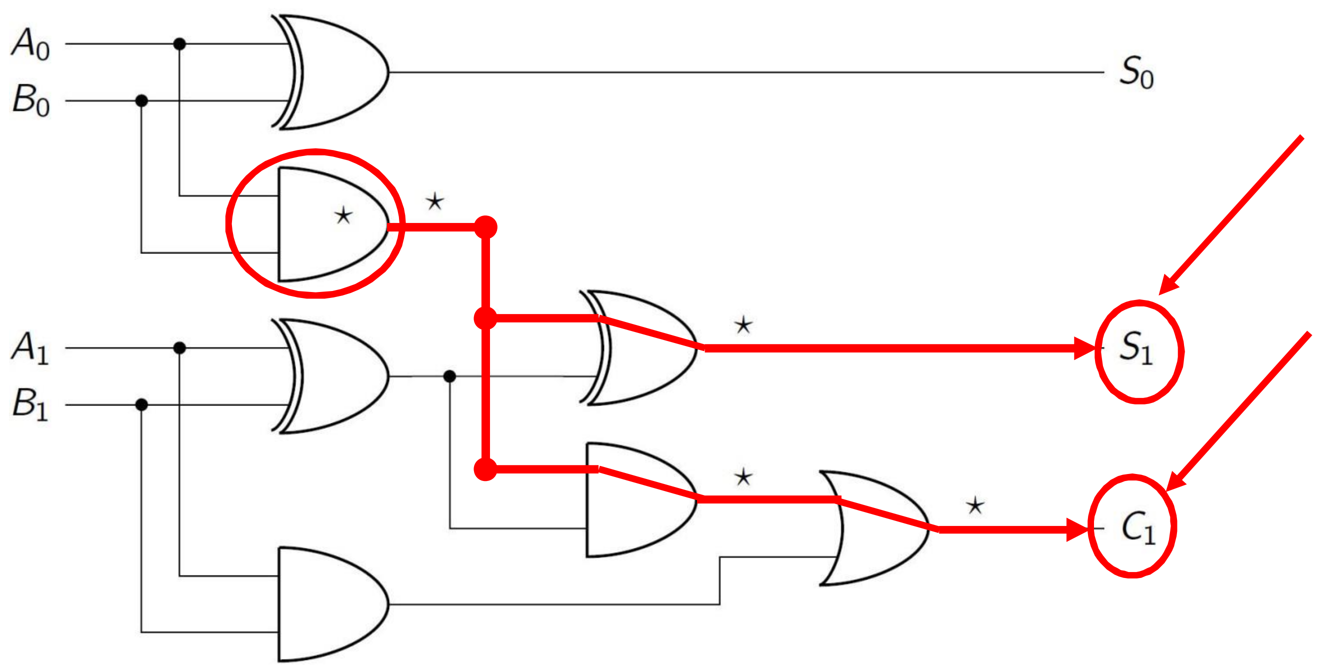
Ref: C. Winstead et al., TCAS-II, 2009
Ref: Y. Tang et al., ISTC, 2012
Ref: Y. Tang et al., ISCAS, 2012
Ref: Y. Tang et al., ISCAS, 2013

ÅThe original logic function *F*(u) is composed with a block ECC code to create the parity-mapped function *P* Å*F*(u).

Contribution 3: Traditional Design

Original Function
$F(u)$

= Two-bit Adder



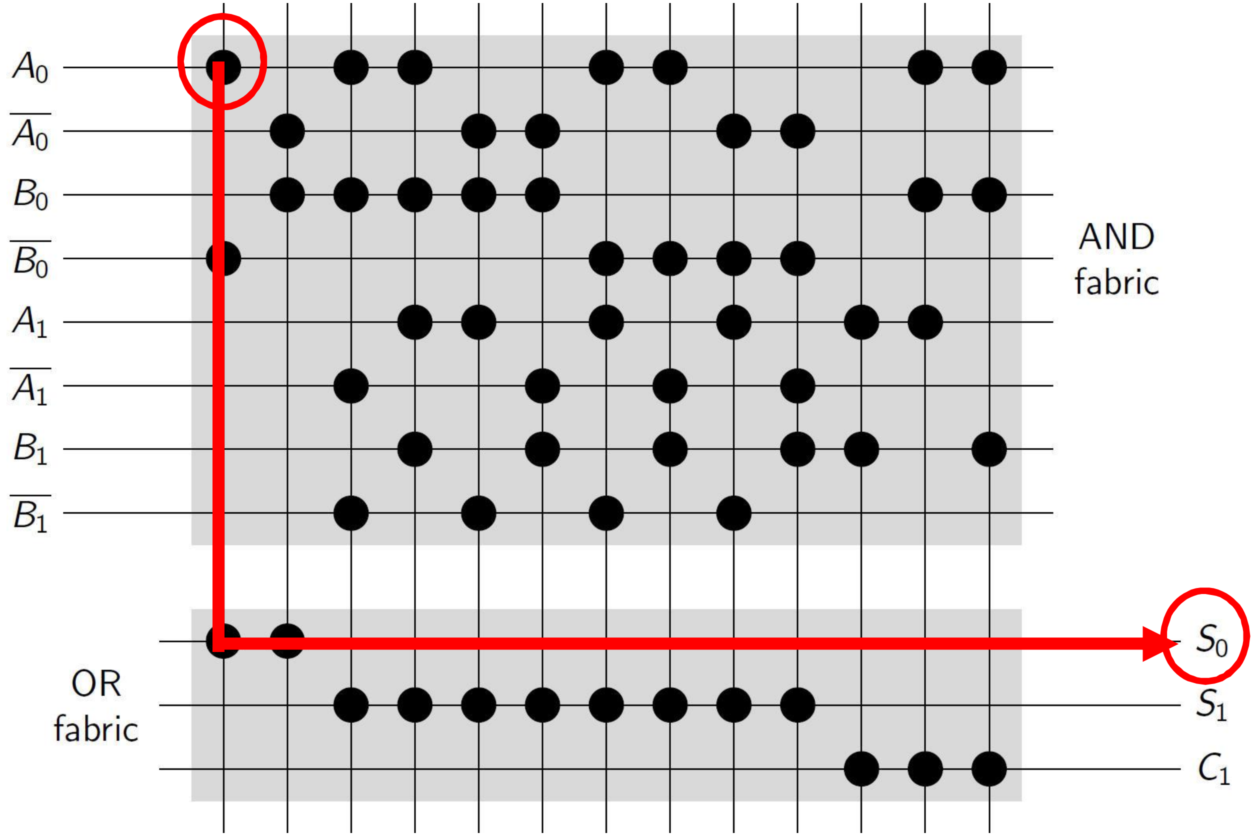
Traditional design, ǂôsymbol indicates the occurrence of an error.

Contribution 3: Cross-Bar Technique

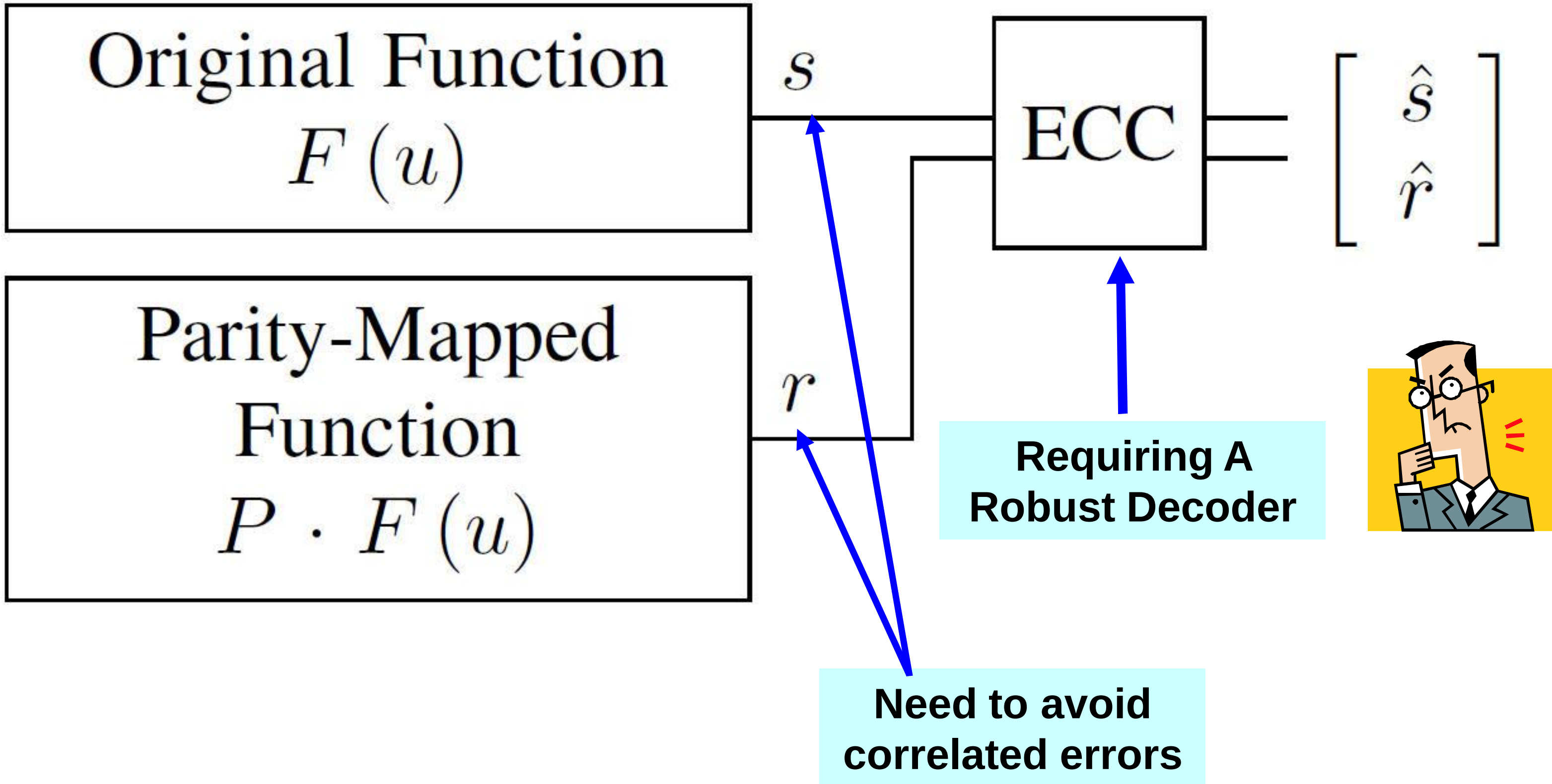
Original Function
$F(u)$

= Two-bit Adder

Crossbar design suitable for some nanoelectronic device families.

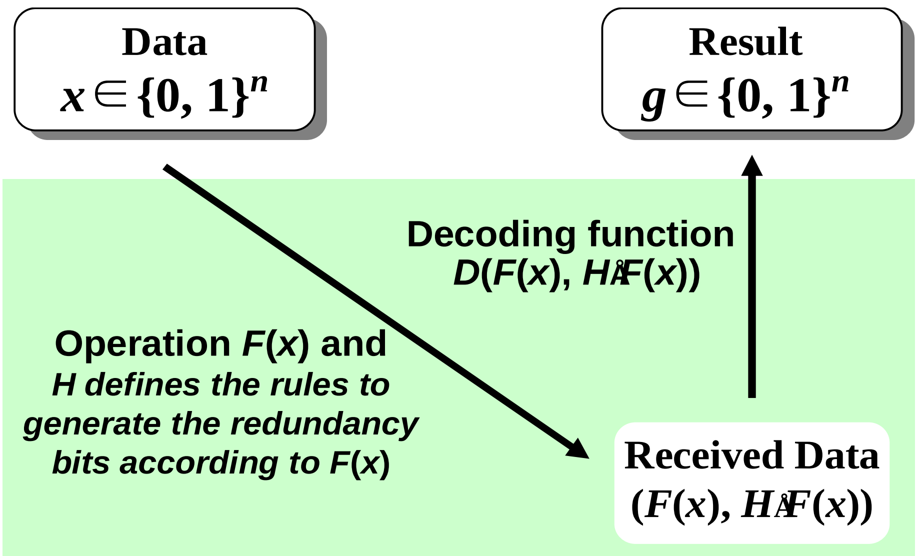


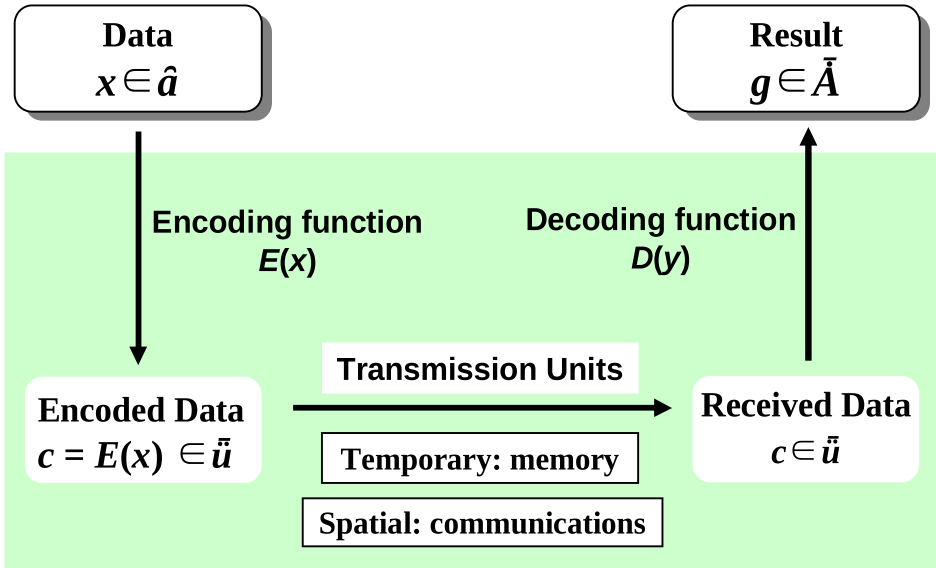
Contribution 3: Coded Dual Modular Redundancy (cDMR)



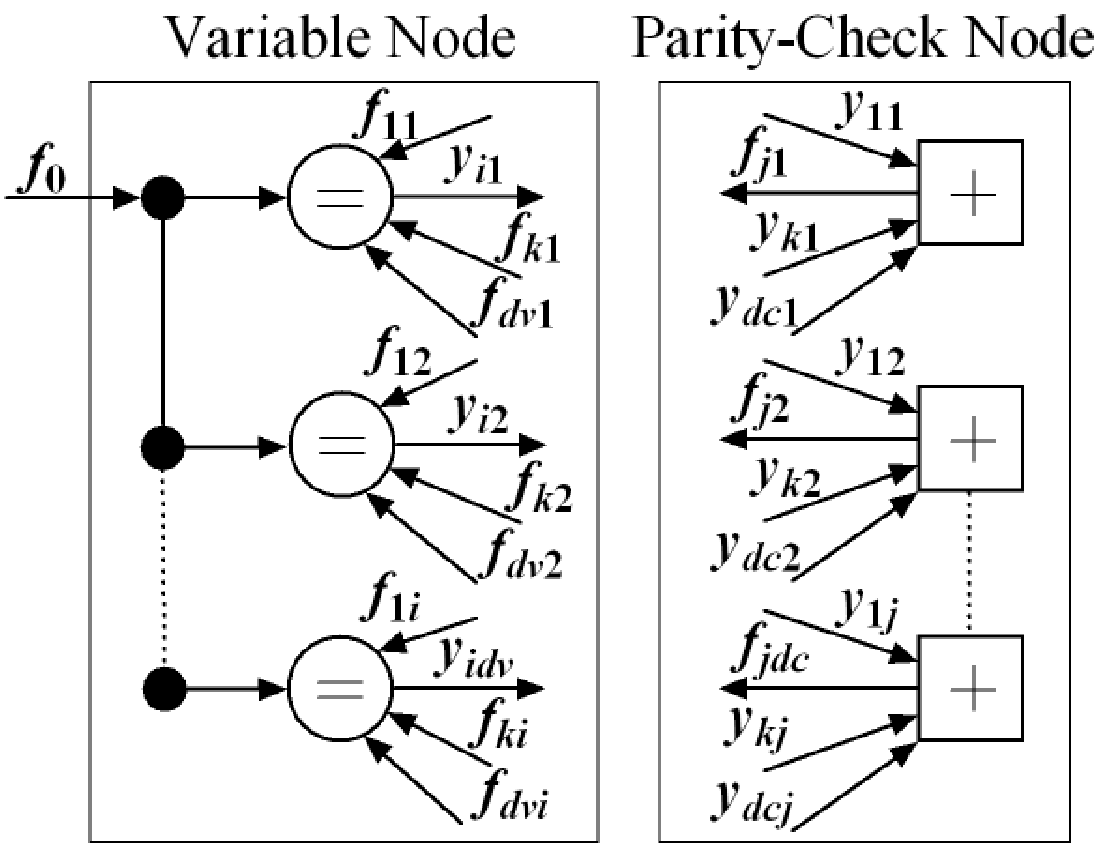
Contribution 4: Muller C-element based Decoder (MCD)

- Contribution 4:
- A decoder against internal faults
- Efficient Error-correction in the presence of high error rate noise





Related Work - Message Passing Decoding Method

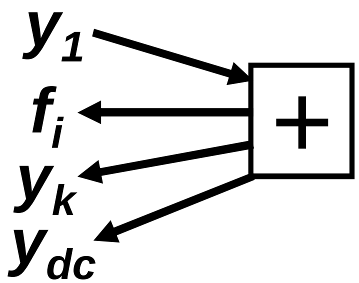


Ref: message-passing decoding, T. Richardson and R. Urbanke, 2001.

Gallager-A Decoder

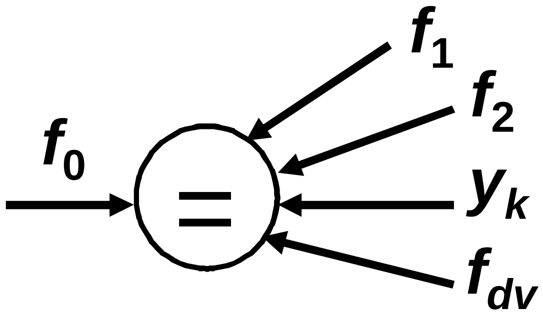
Check Node Processing

$f_i = \oplus\, y_k$, *kÍ i*



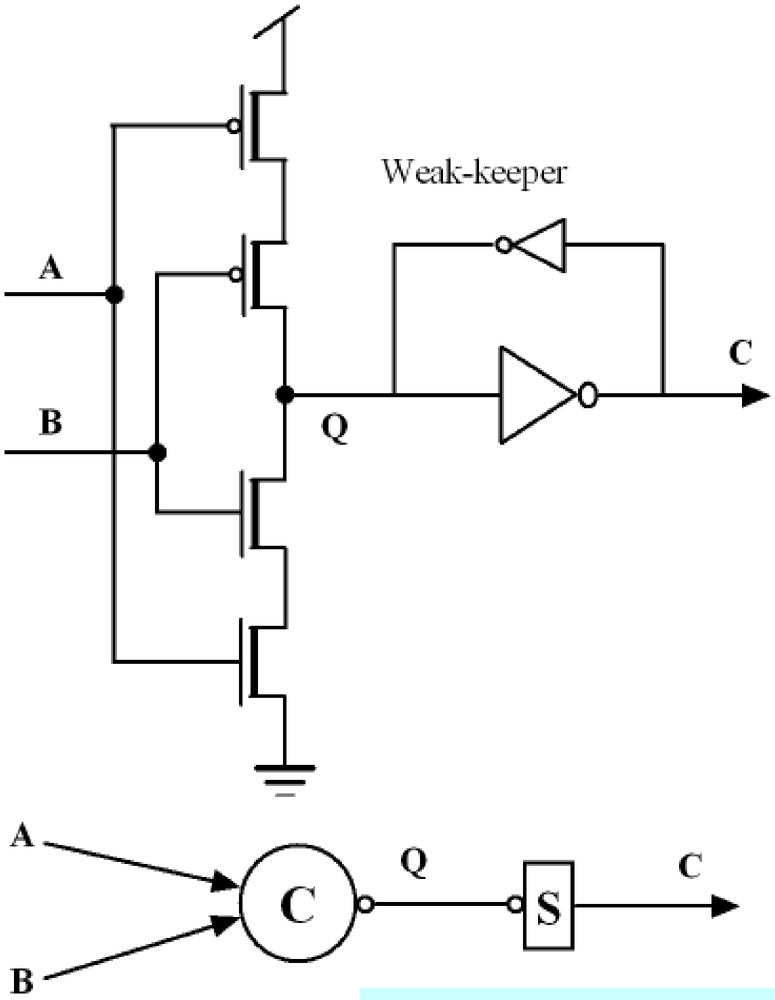
Variable Node Processing

$y_k = f_0$ if (f_i, *iÍ k*) disagree,
f_1 elsewise



Contribution 4: Muller C-element based Decoder (MCD)

Ref : D. E. Muller and W. S. Bertky, 1959



CMOS Implementations of binary C-element

C-element

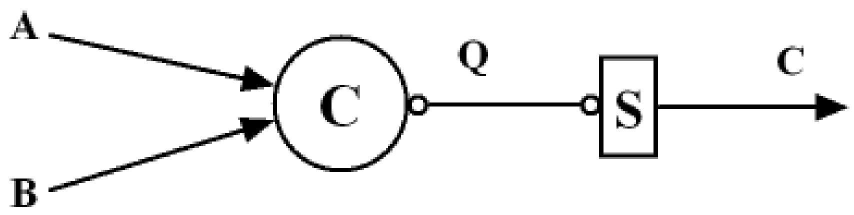
Truth-Table of Binary C-element

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

ÅC_{n-1} denotes the state maintained via weak feedback.

Contribution 4: C-element – Error-Resilience in Nature

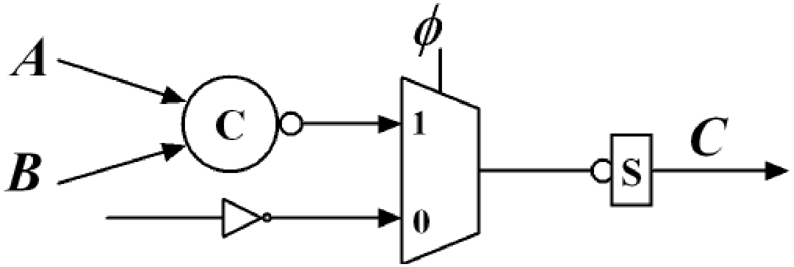
C-element



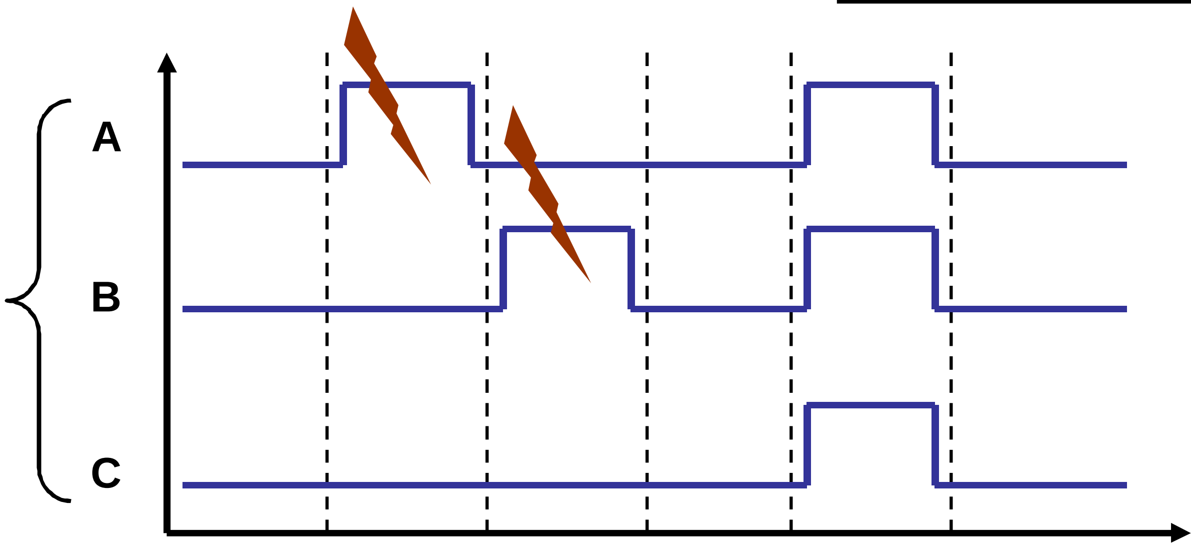
Truth-Table of Binary C-element

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

Å Initializing the state memory.



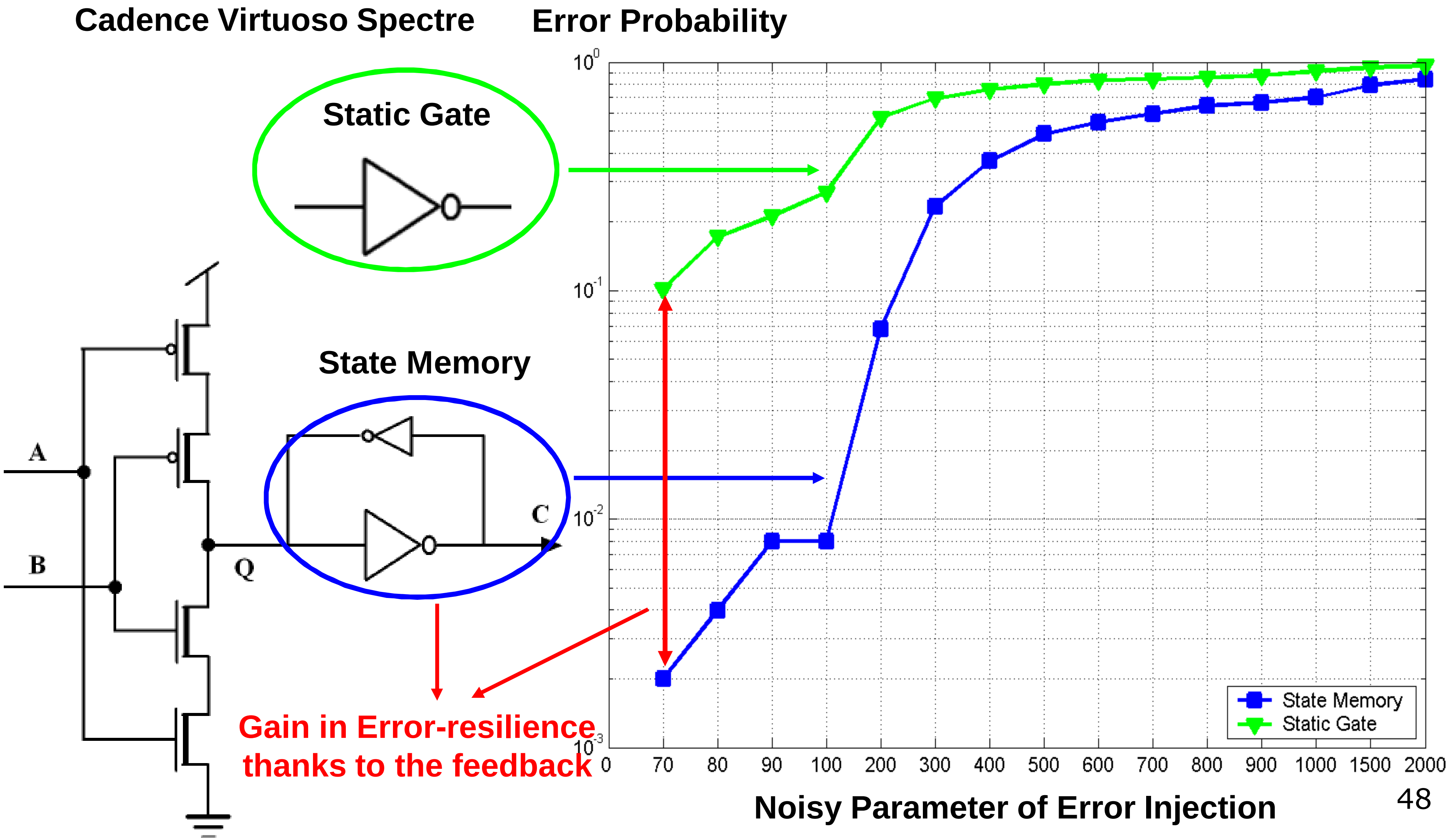
Error-Resilience in Nature



Contribution 4: C-element – Error-Resilience in Nature

- Monte Carlo Simulation in Cadence Virtuoso Spectre

Ref : Y. Tang et al., ISCAS, 2013.



Contribution 4: Muller C-element based Decoder (MCD)

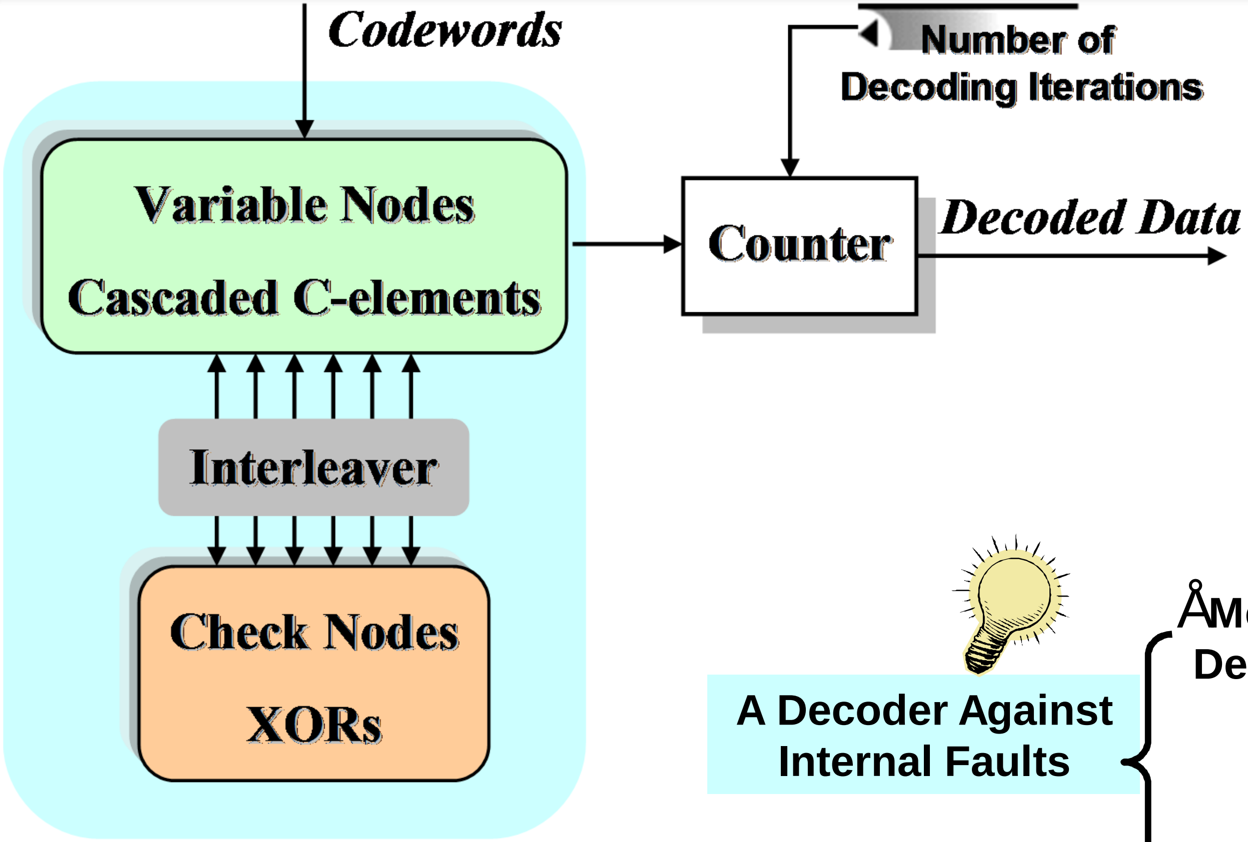
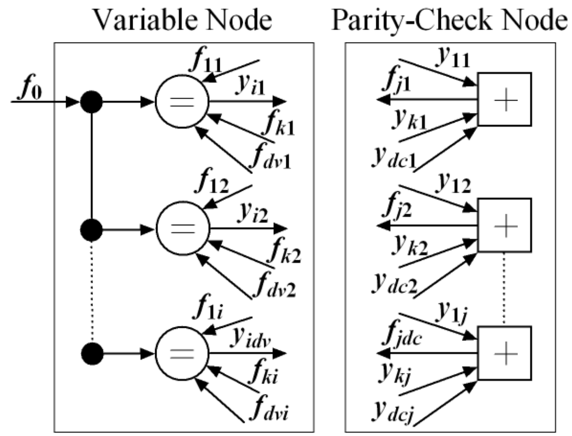


Diagram of process for message-passing decoding method

Ref : Y. Tang et al., ISTC, 2012.
Ref: Y. Tang et al., ISCAS, 2012

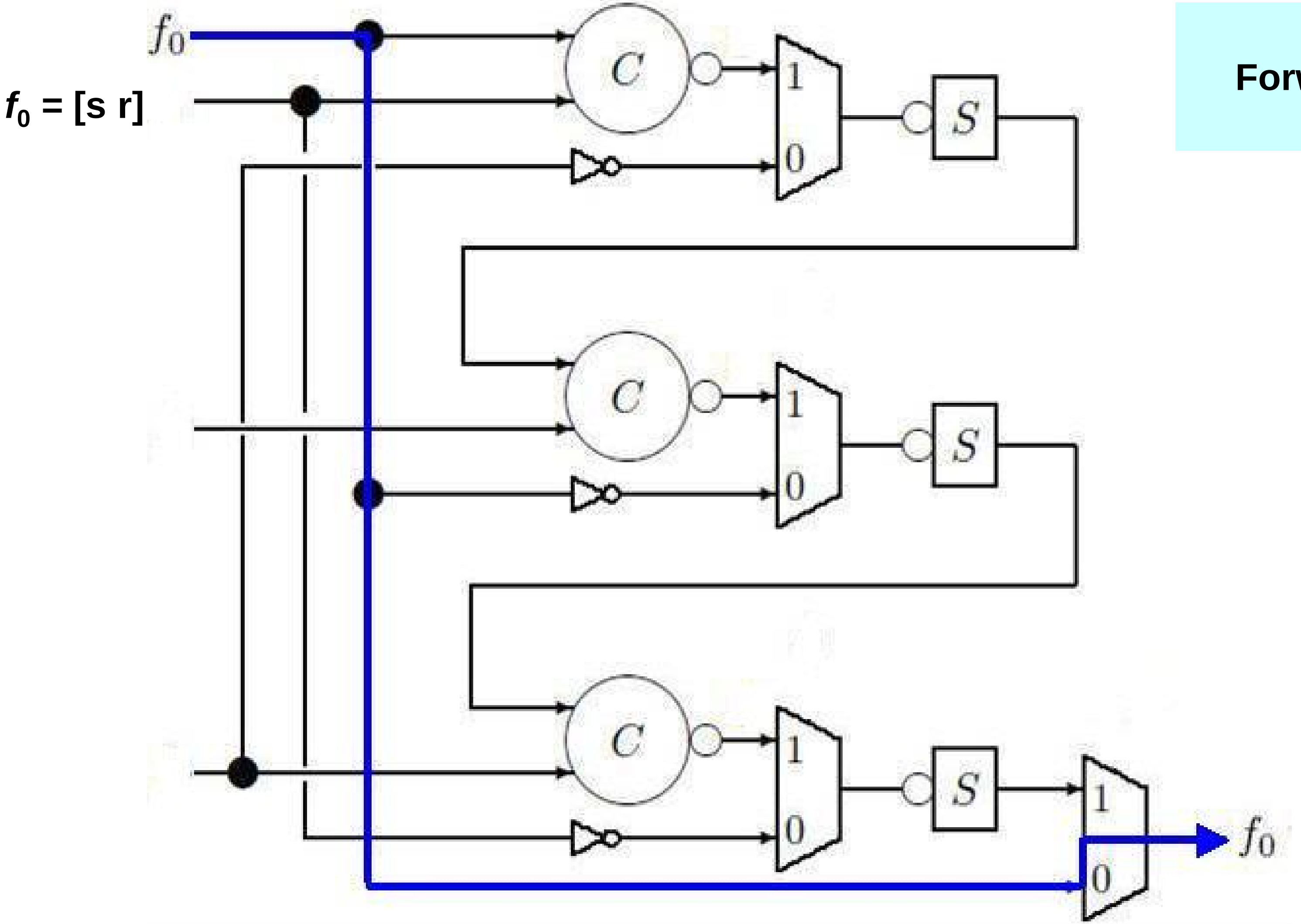


A Decoder Against Internal Faults

- ÅMessage-Passing Decoding Method
- ÅInherent error-resilience C-element

C-element

Contribution 4: Variable Node in MCD Phase 0



Phase 0
Forward Channel Message

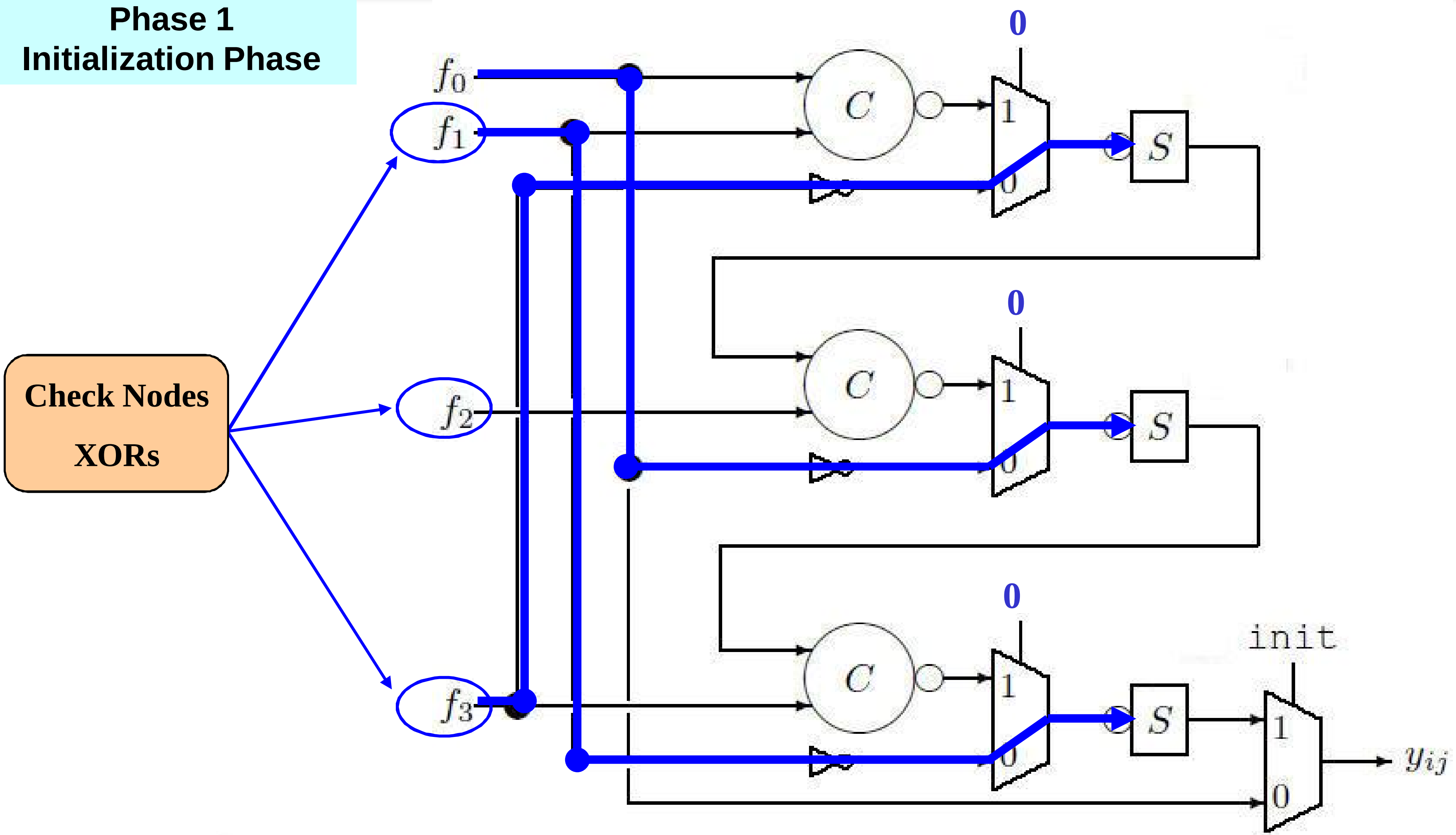
Check Nodes

XORs

Basic Error-Correction Unit Cascaded C-elements, d_v=4

Contribution 4: Variable Node in MCD Phase 1

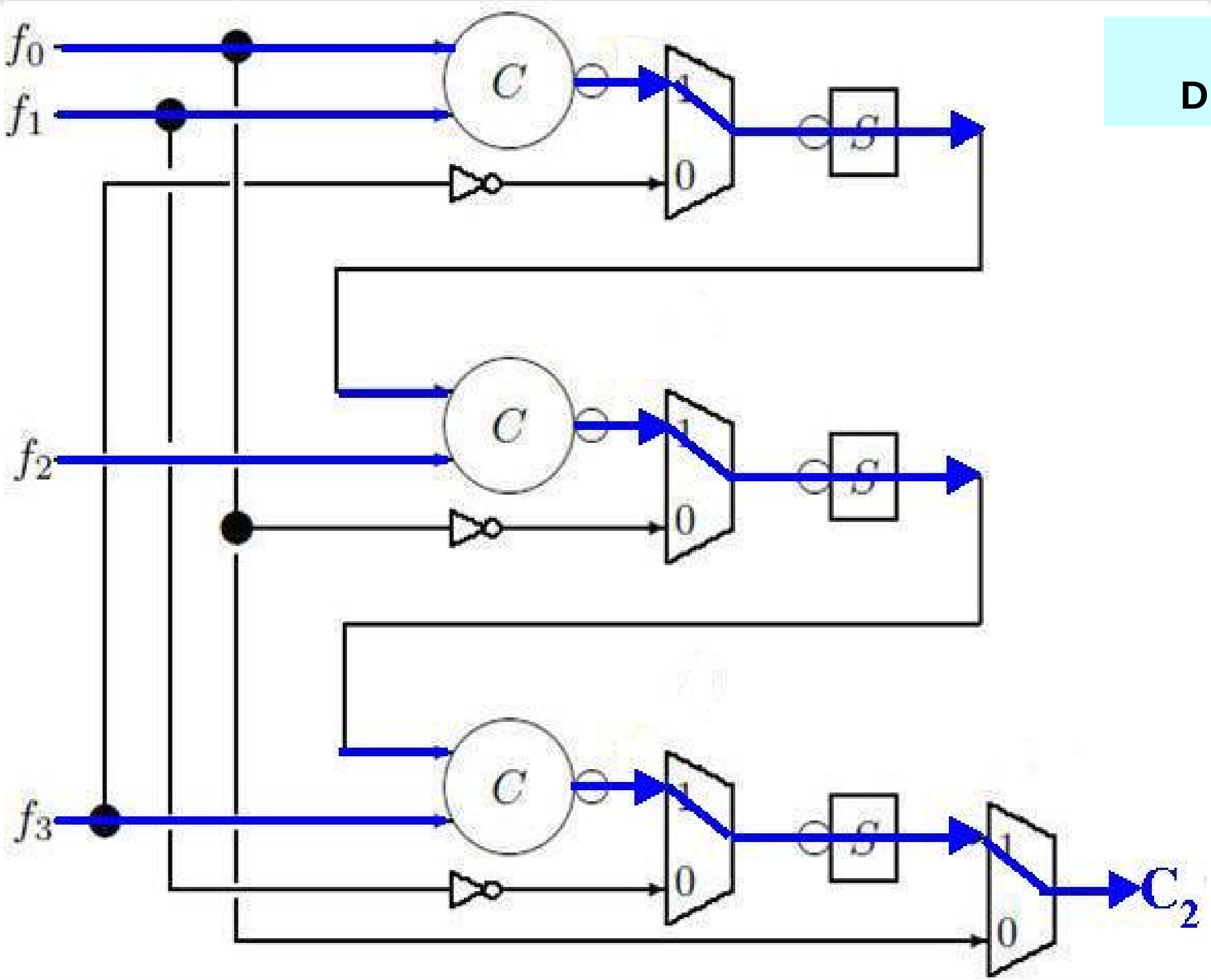
Phase 1
Initialization Phase



Basic Error-Correction Unit Cascaded C-elements, d_v=4

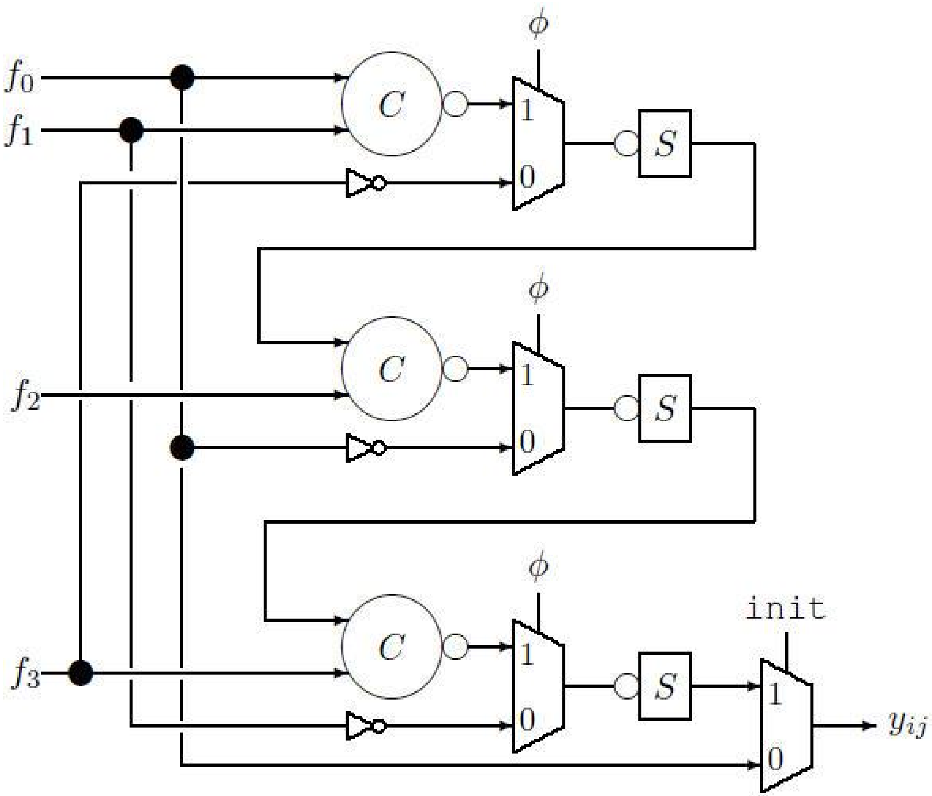
Contribution 4: Variable Node in MCD Phase 2

Iterative Phase 2
Driven by C-element

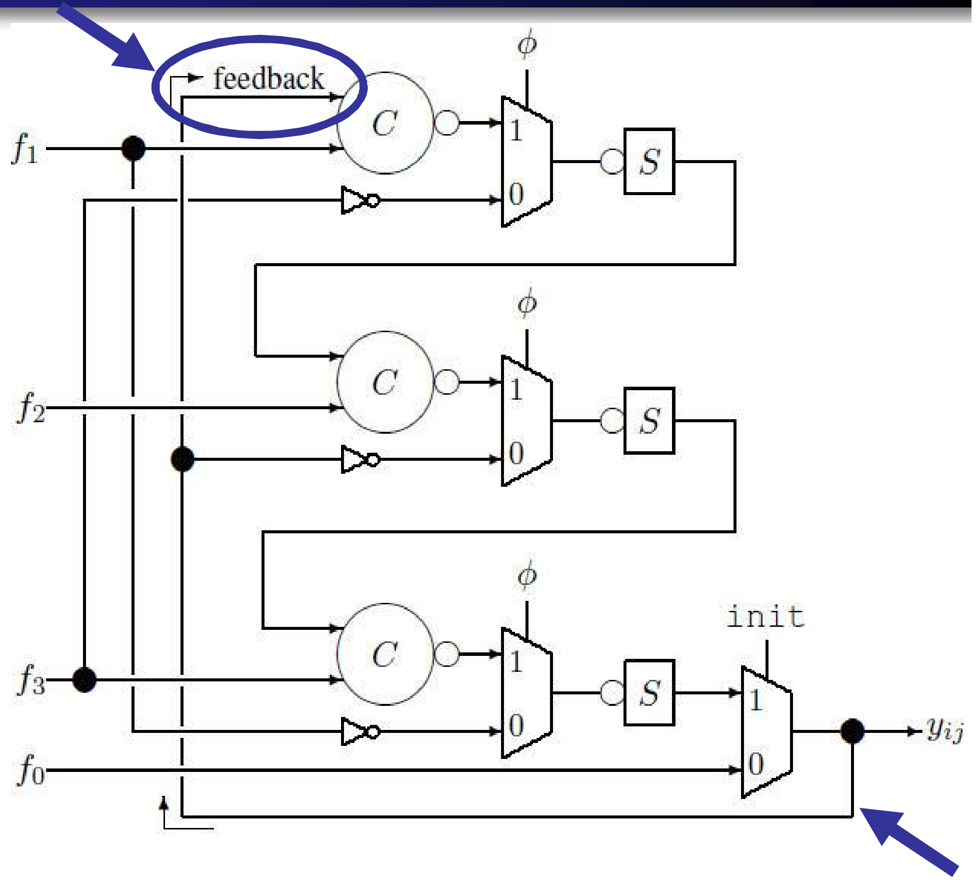


Basic Error-Correction Unit Cascaded C-elements, d_v=4

Contribution 4: MCDfb – feedback mechanism



MCDôs variable-node architecture, d_v=4

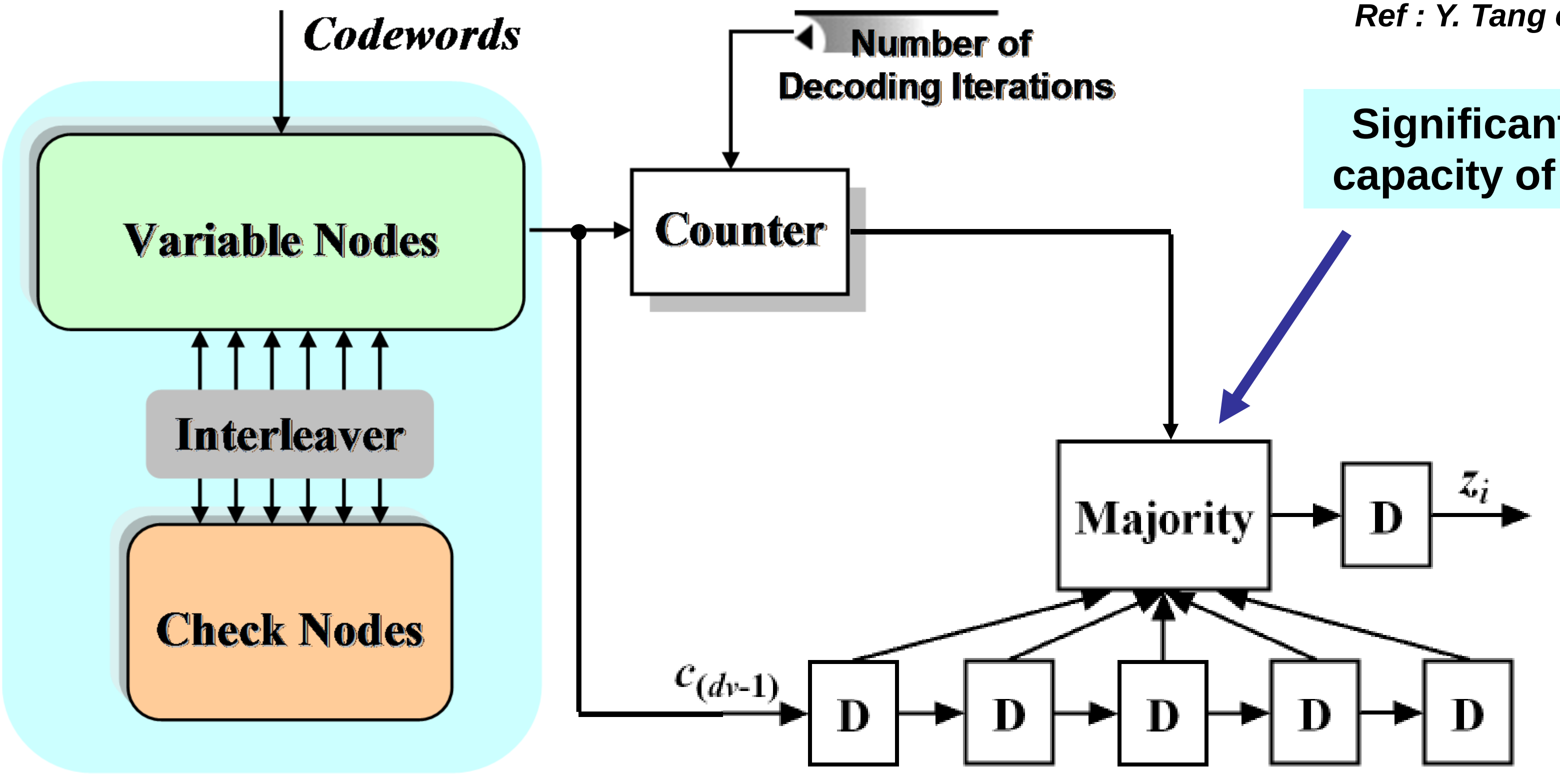


MCDfbôs variable-node architecture, d_v=4

A feedback mechanism employed to suppress internal fault event.

Contribution 4: Space-Time Technique – Improving BER

Ref : Y. Tang et al., ISTC, 2012.



Significantly Improve the capacity of error-resilience

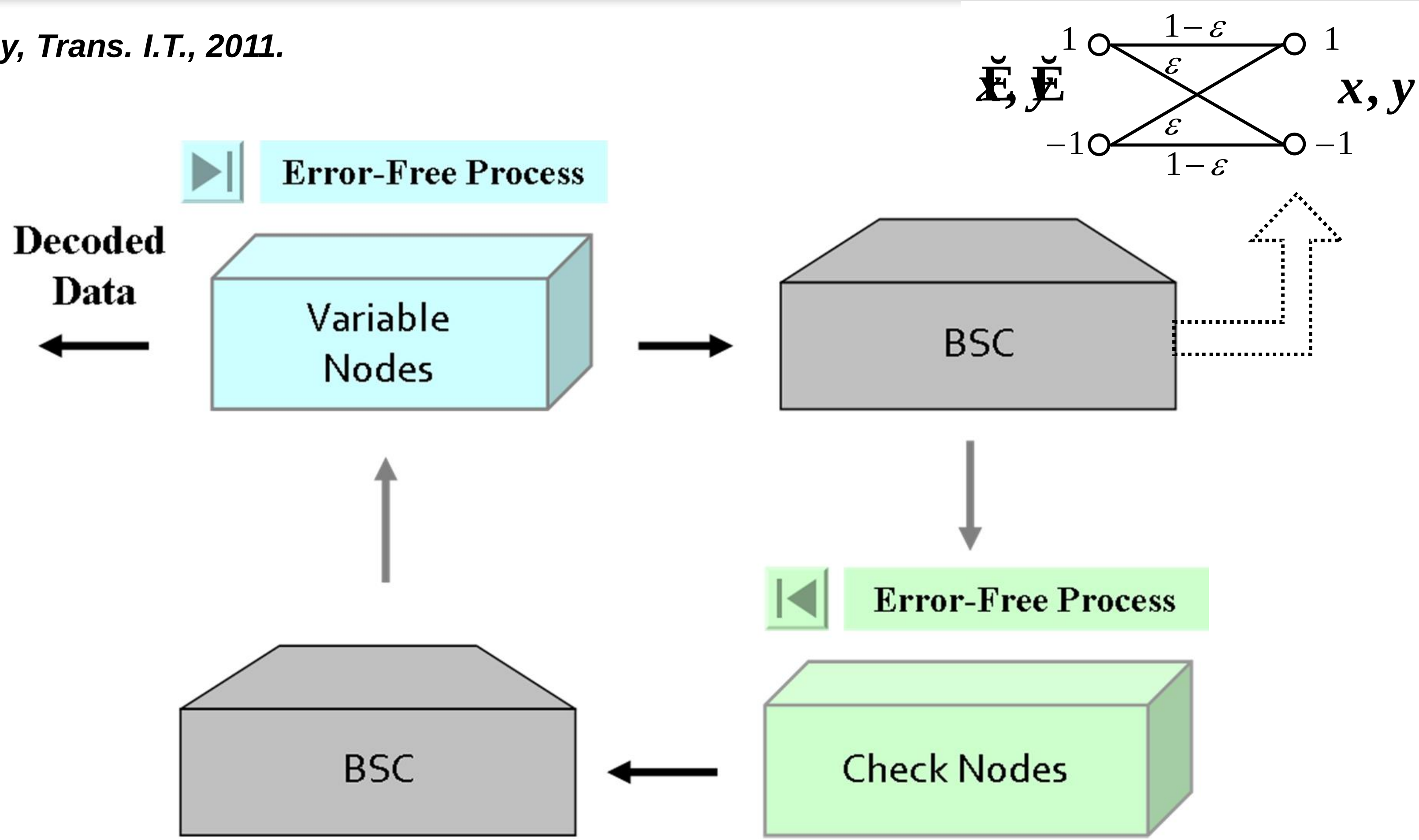
Message-Passing Decoder with Space-Time Redundancy Technique. For instance, the Majority unit can be a 3-of-5 voter.

Outline

- Introduction
- Formal Models of Embedded ECC
- Proposed Approaches
- Experimental Results
  - MCDôs BER Performance Under a Faulty Process
  - Comparisons Between MCD and GBF
  - The Improvement of Decoding Performance by Space-Time Technique
- Conclusion & Future Perspectives

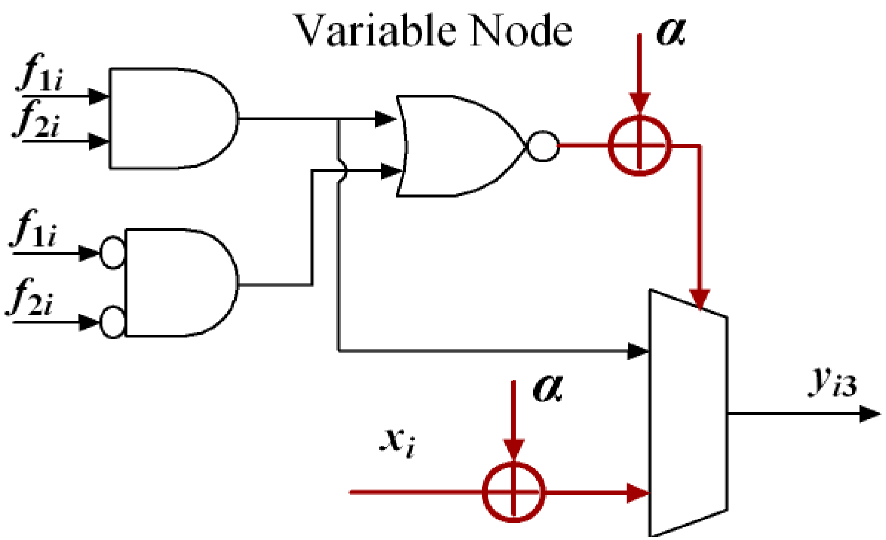
Error Model

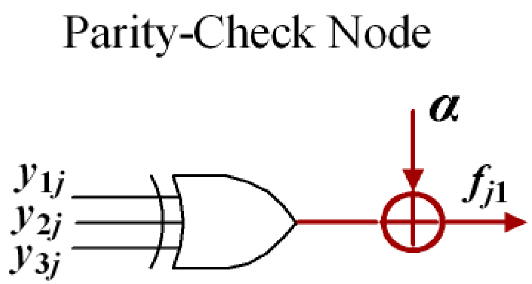
Ref : L. Varshney, Trans. I.T., 2011.



MCD's BER Performance Under a Faulty Process

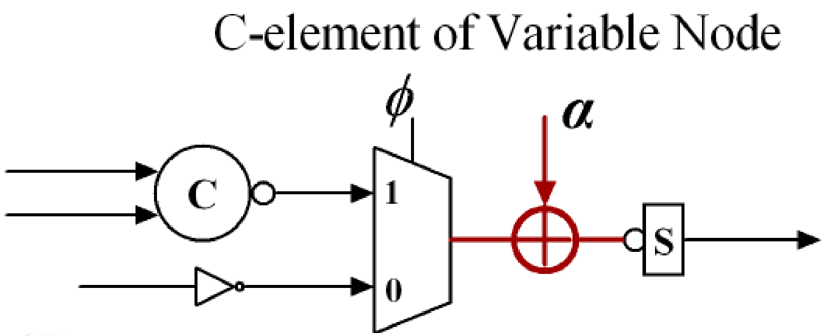
- Gallager Bit-Flipping Decoding Method (GBF)

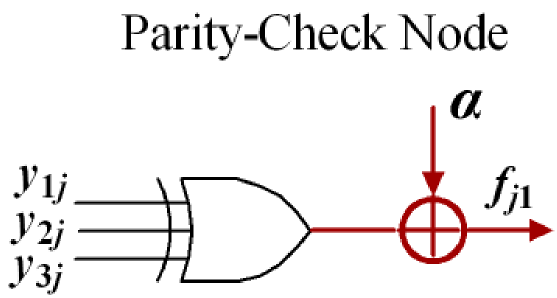




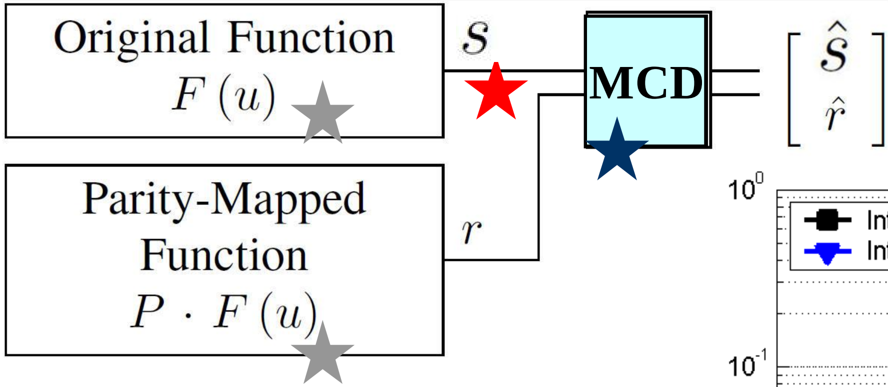
- **Injected into critical path in RTL**

- MCD





MCD's BER Performance Under a Faulty Process

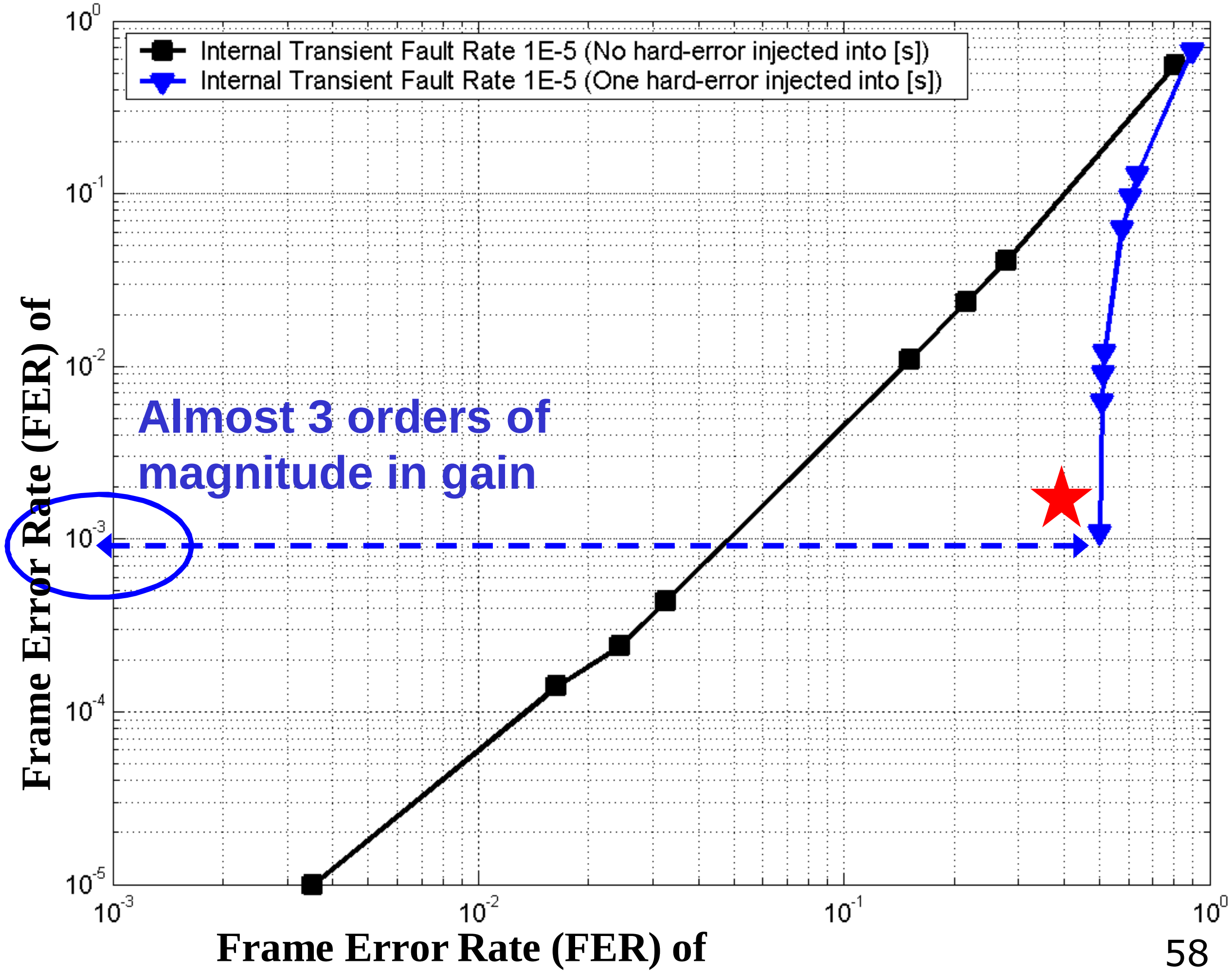


Simulation Results for rate (3,6) LDPC codes based on MCD architecture, 2 iterations.

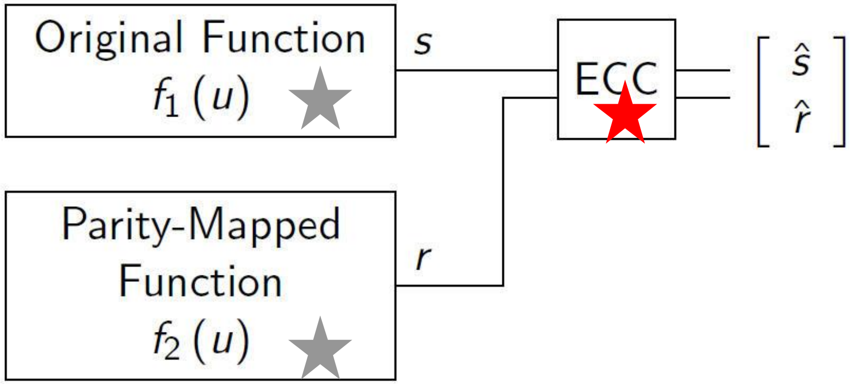
Error occurred in Function

Single Hard-Error Injection

Internal Fault Injection (Error Rate 10^{-5})



Comparisons Between MCD and GBF, (3,6) LDPC codes *N*=64

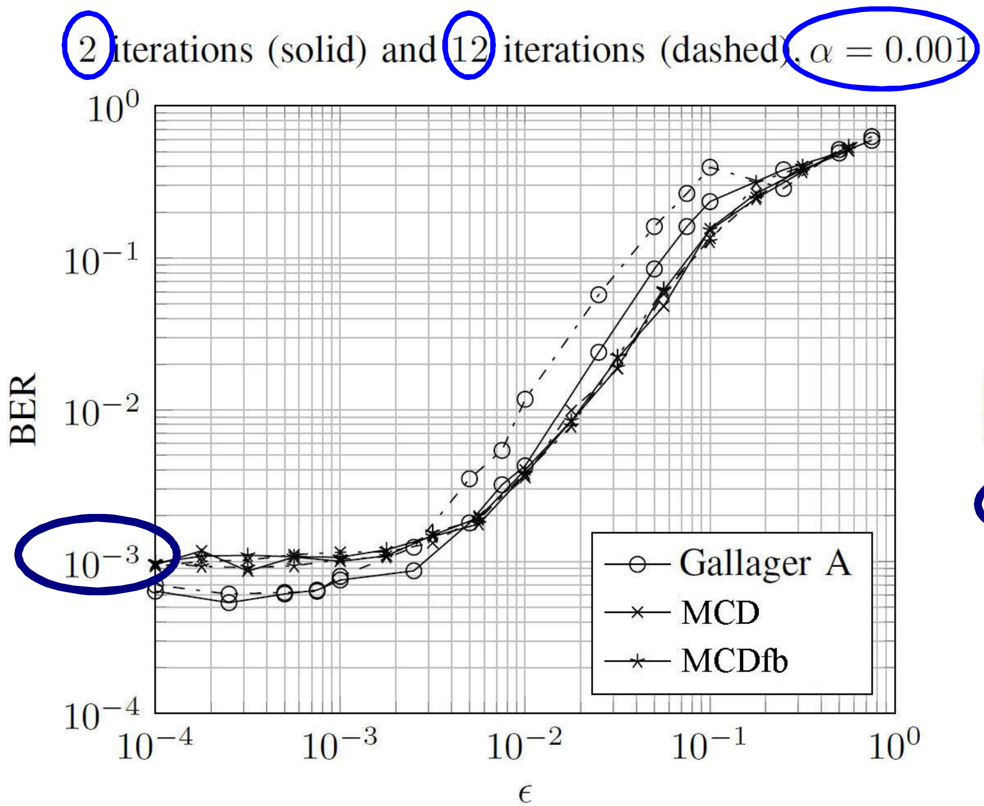


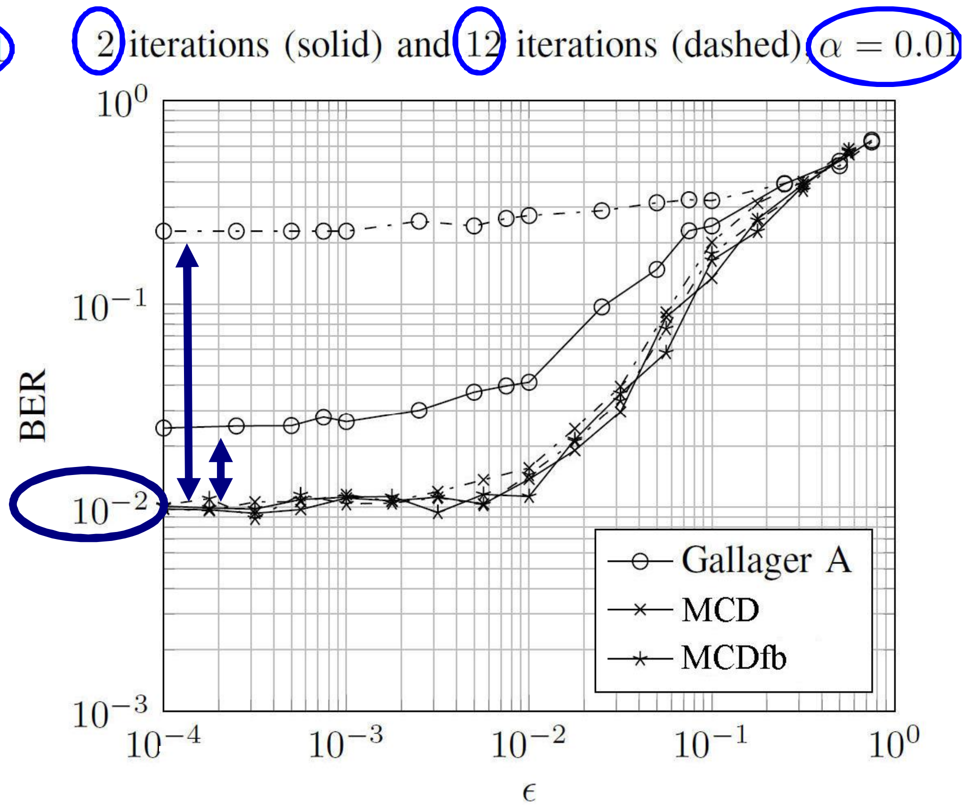
Ref : Y. Tang et al., ISTC, 2012.

★ **Error occurred in Function: *Epsilon***

★ **Internal Fault Injection: *Alpha***

Comparisons Between MCD and GBF, (3,6) LDPC codes *N*=64





Å Gallager-A performance worsens with increased iterations when internal faults rate is high.

Å But MCD does not exhibit this degradation.

Decoding Performance Improvement by Space-Time Technique

WITH Space-Time Redundancy

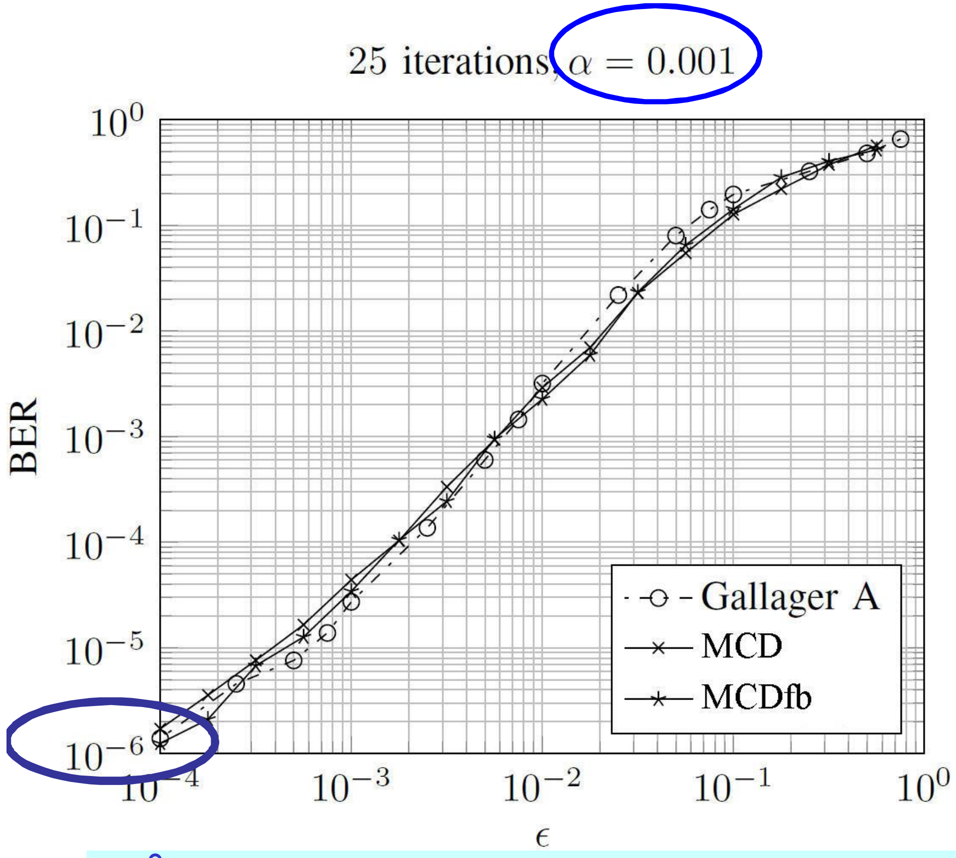
- **Internal Fault Injection: *Alpha***
- **(3,6) LDPC code of length 64 over a BSC parameter of *Epsilon***

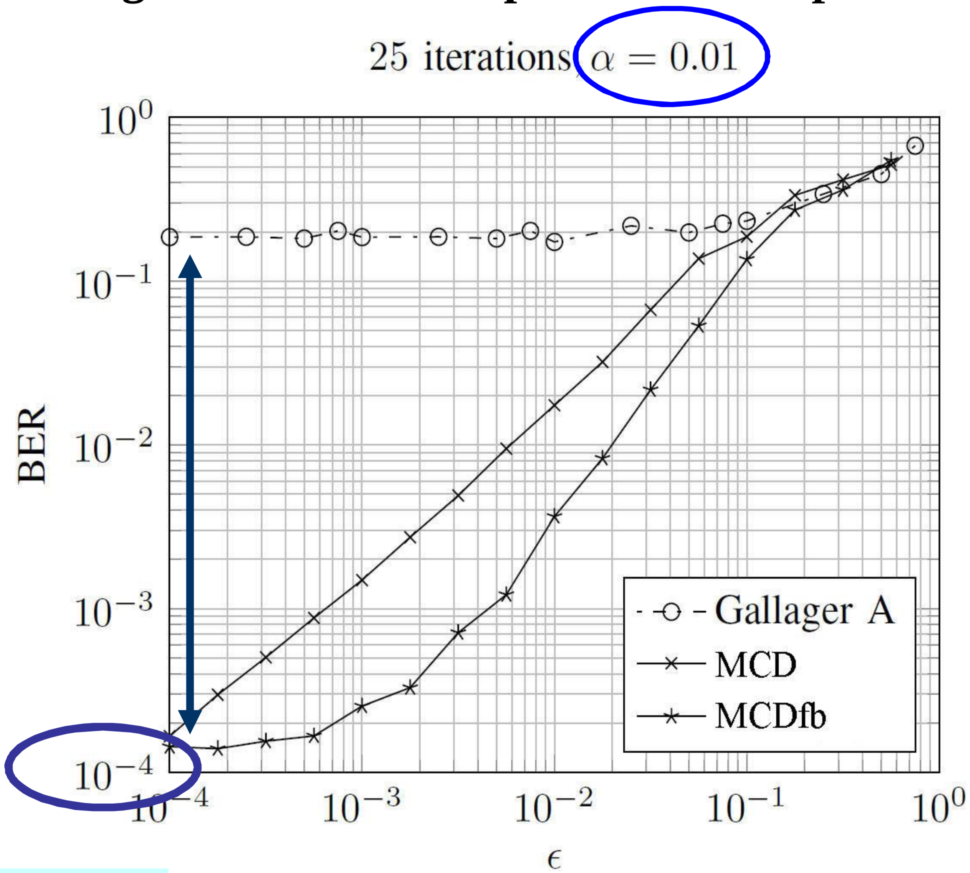
Ref : Y. Tang et al., ISTC, 2012.

Decoding Performance Improvement by Space-Time Technique

WITH Space-Time Redundancy

- **Internal Fault Injection: *Alpha***
- **(3,6) LDPC code of length 64 over a BSC parameter of *Epsilon***





- **The majority voter significantly improves the decoderô BER performance.**
- **Feedback-Mechanism is helpful as well.**

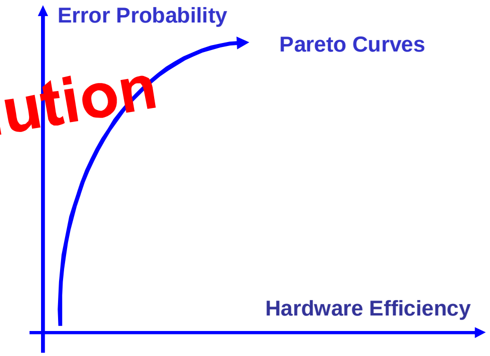
Outline

- Introduction
- Formal Models of Embedded ECC
- Architectural Approaches
- Experimental Results
- Conclusion & Future Perspectives

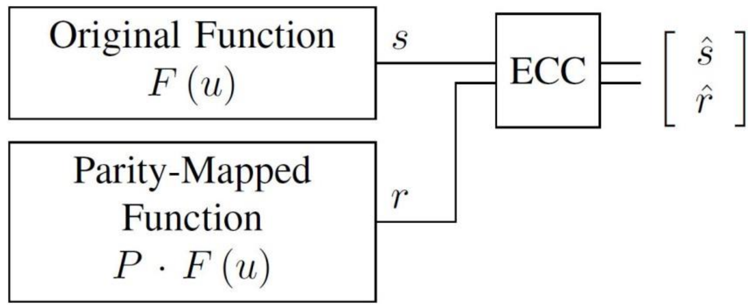
Conclusion

- Objective : to design an efficient error-resilient architecture by ECC
- Contribution :
- An arithmetic method (Fast computation & Error-resilience)
- Reliability-Efficiency Criteria

Revolution



- A fault-tolerant system for computation



- A decoder against internal faults

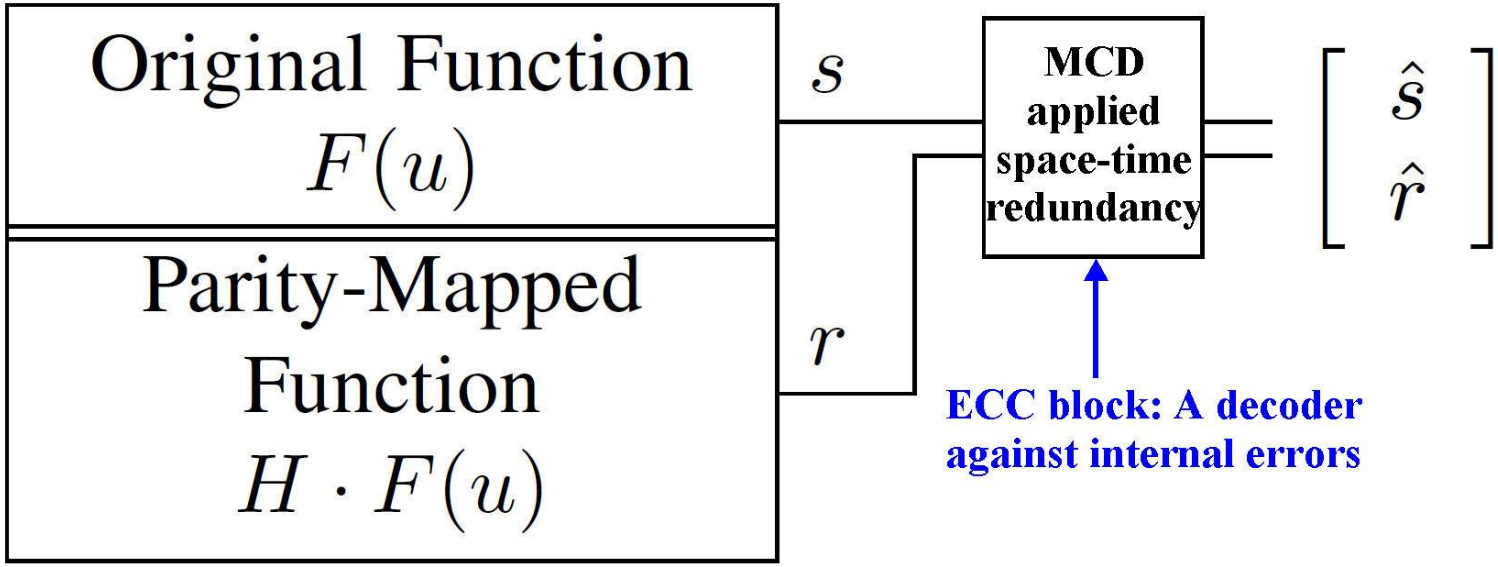
A Decoder Against Internal Faults

ÅMessage-Passing Decoding Method

ÅInherent error-resilience C-element

Future Perspectives

- For MCD, high code rate, such like a (3,4) code, would be worth to try.
- Moreover, simulating a 3-of-5 voter based space-time technique.
- Approaching the proposed reliability criterion as a precise error model.
- Implementation of the cDMR & MCD & Space-Time Technique.



Future Directions

- **Collaboration** (2011. June - 2012) with LEFT-Lab directed by Dr. Chris Winstead, Utah State University, Logan, Utah, USA.

- **Continued Work** ï Application of an ANR Project on this subject
  - Collaboration with, LEFT-Lab, *Dr. Winstead*
  - ï Ongoing thesis work, *Pr. Jégo*

ï Asynchronous Decoder to decrease power consumption

List of Publications

- ***Yangyang Tang***, *Emmanuel Boutillon, Chris Winstead, Christophe Jégo, and Michel Jézéquel*
  ñMuller C-element based Decoder (MCD): A decoder against transient faults,ò
  In Circuits and Systems (ISCAS), 2013 IEEE International Symposium on, May 2013.
- *Chris Winstead*, ***Yangyang Tang***, *Emmanuel Boutillon, Christophe Jégo, and Michel Jézéquel*
  ñA space-time redundancy technique for embedded stochastic error correction,ò
  Turbo Codes and Related Topics (ISTC), 2012 7th International Symposium on, August 2012.
- ***Yangyang Tang***, *Chris Winstead, Emmanuel Boutillon, Christophe Jégo, and Michel Jézéquel*
  ñAn ldpc decoding method for fault-tolerant digital logic,ò
  In Circuits and Systems (ISCAS), 2012 IEEE International Symposium on, May 2012, pp. 3025-3028.
- ***Yangyang Tang***, *Gopalakrishnan Sundararajan, Chris Winstead, Emmanuel Boutillon, Christophe Jégo, and Michel Jézéquel*
  ñTechniques and prospects for fault-tolerance in post-CMOS ULSI,ò
  21st International Workshop on Post-Binary ULSI Systems (ULSIWS), May 2012.
- ***Yangyang Tang***, *Emmanuel Boutillon, Christophe Jégo, and Michel Jézéquel*
  ñHardware efficiency versus error probability in unreliable computation,ò
  Signal Processing Systems (SiPS), 2011 IEEE workshop on, October, 2011
- ***Yangyang Tang***, *Emmanuel Boutillon, Christophe Jégo, and Michel Jézéquel*
  ñA new single-error correction scheme based on self-diagnosis residue number arithmetic,ò
  Design and Architectures for Signal and Image Processing (DASIP), 2010.
- *Emmanuel Boutillon*, ***Yangyang Tang***, *Cédric Marchand, and Pierre Bomel*
  ñHardware Discrete Channel Emulator,ò
  High Performance Computing and Simulation (HPCS), 2010.

2013

2012

2011

2010

Acknowledgement

Å(2010. Jan - 2013. Jan) This work was supported by the funding from East China Institute of Technology and Lab-STICC, Lorient.

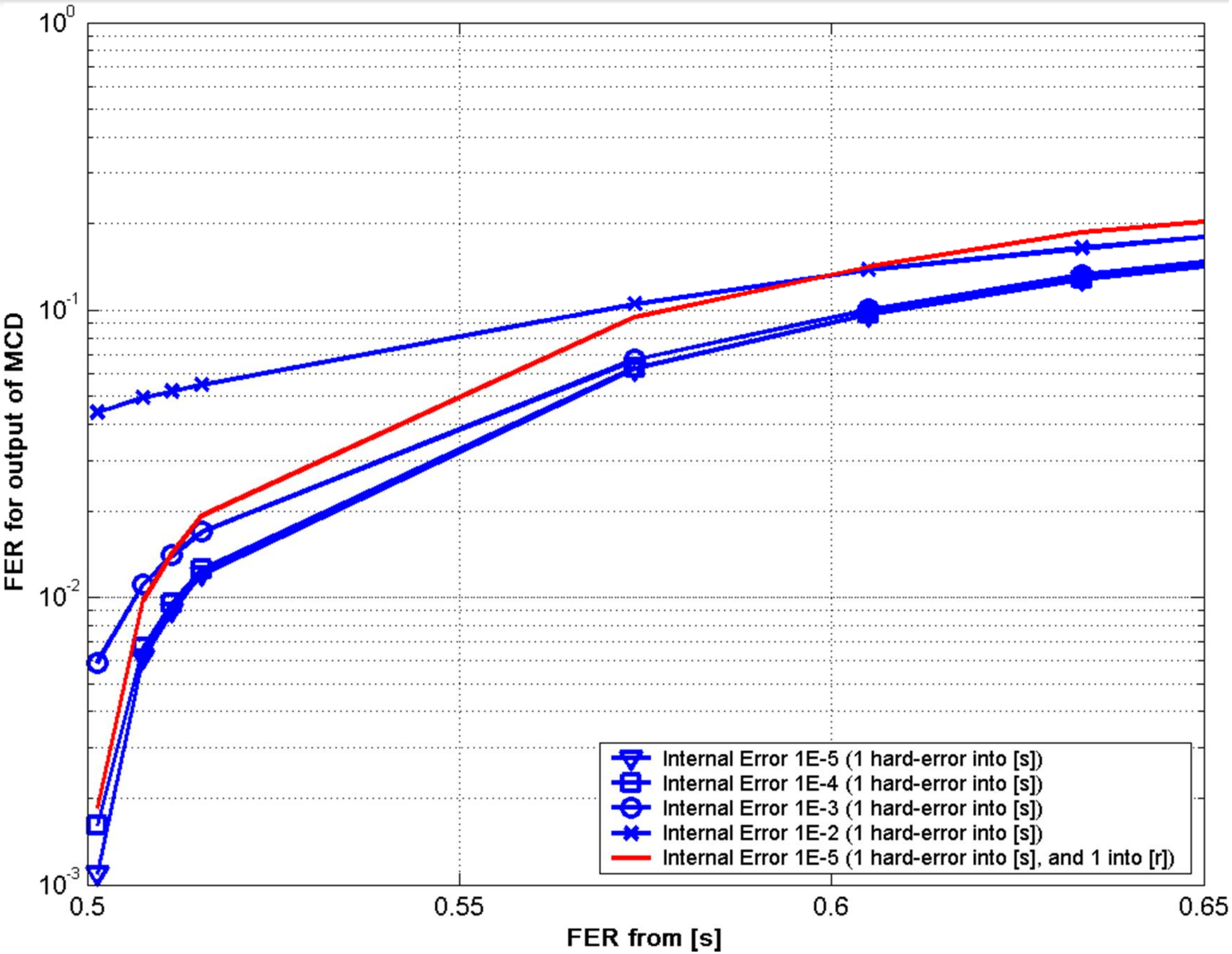
Å(2011. June - 2012. June) This work was also supported by the US National Science Foundation under award ECCS-0954747 and CCF-0916105.

Questions

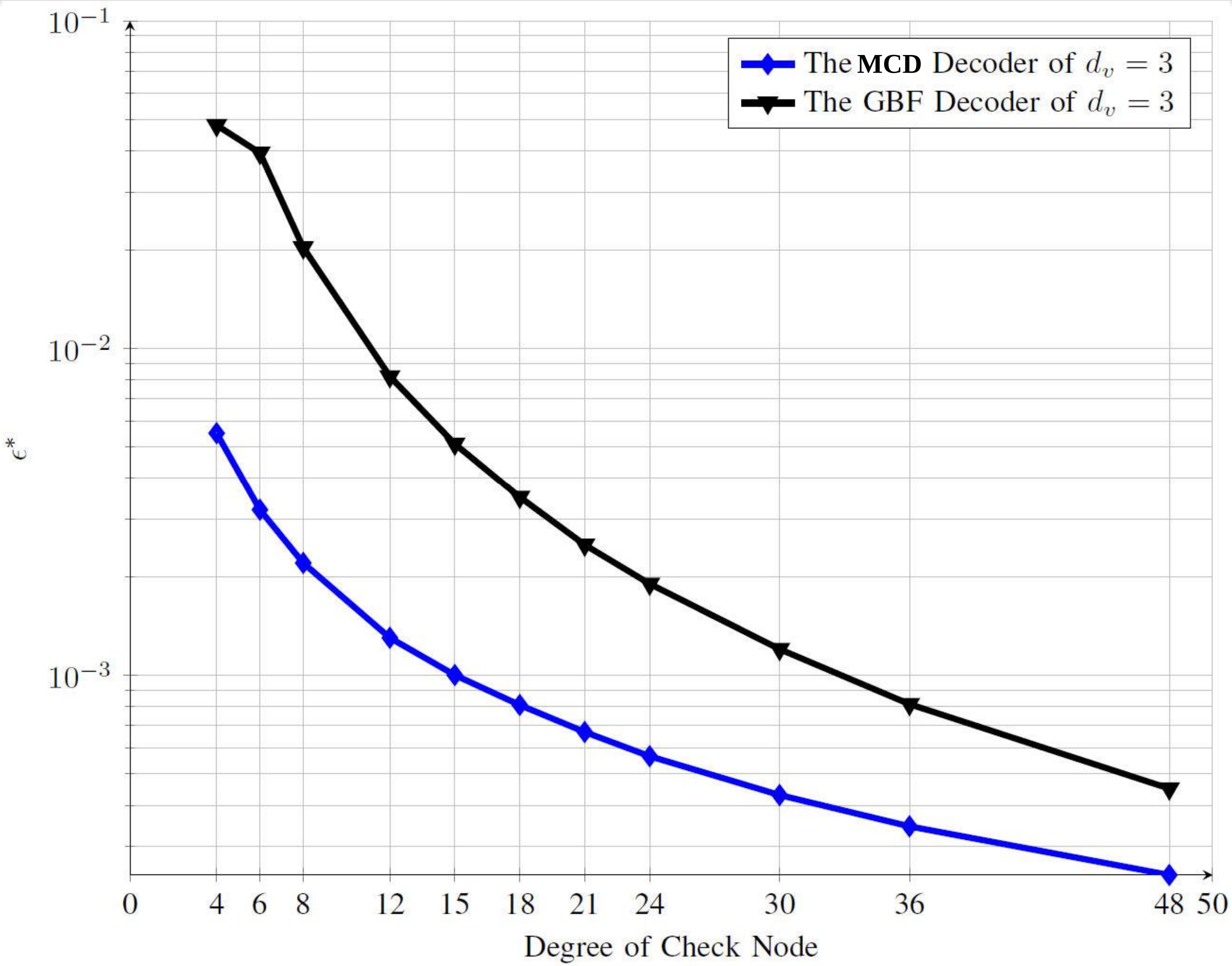
Thank You For Your Attention!

Back-Ups: FERs by Applying cDMR

(3,6) LDPC code of length 64 under 2 iterations



Back-Ups: Threshold Determinations



Ů: BSC Channel Parameter

Threshold of channel parameter over BSC for GBF (Gallagerô s Bit-Flipping method) and MCD, if the decoder process is error-free.

Back-Ups: Threshold Determinations

	GBF	MCD			
	(d_v, d_c)	(d_v, d_c)	(d_v, d_c)	(d_v, d_c)	(d_v, d_c)
	(3,6)	(3,6)	(3,12)	(3,24)	(3,48)
α^*	0.0082	0.0303	0.0096	0.0032	0.0011
ϵ^*	0.0126	0.0690	0.0199	0.0059	0.0011

Ŭ: Decoderĝs Internal Transient Error Rate

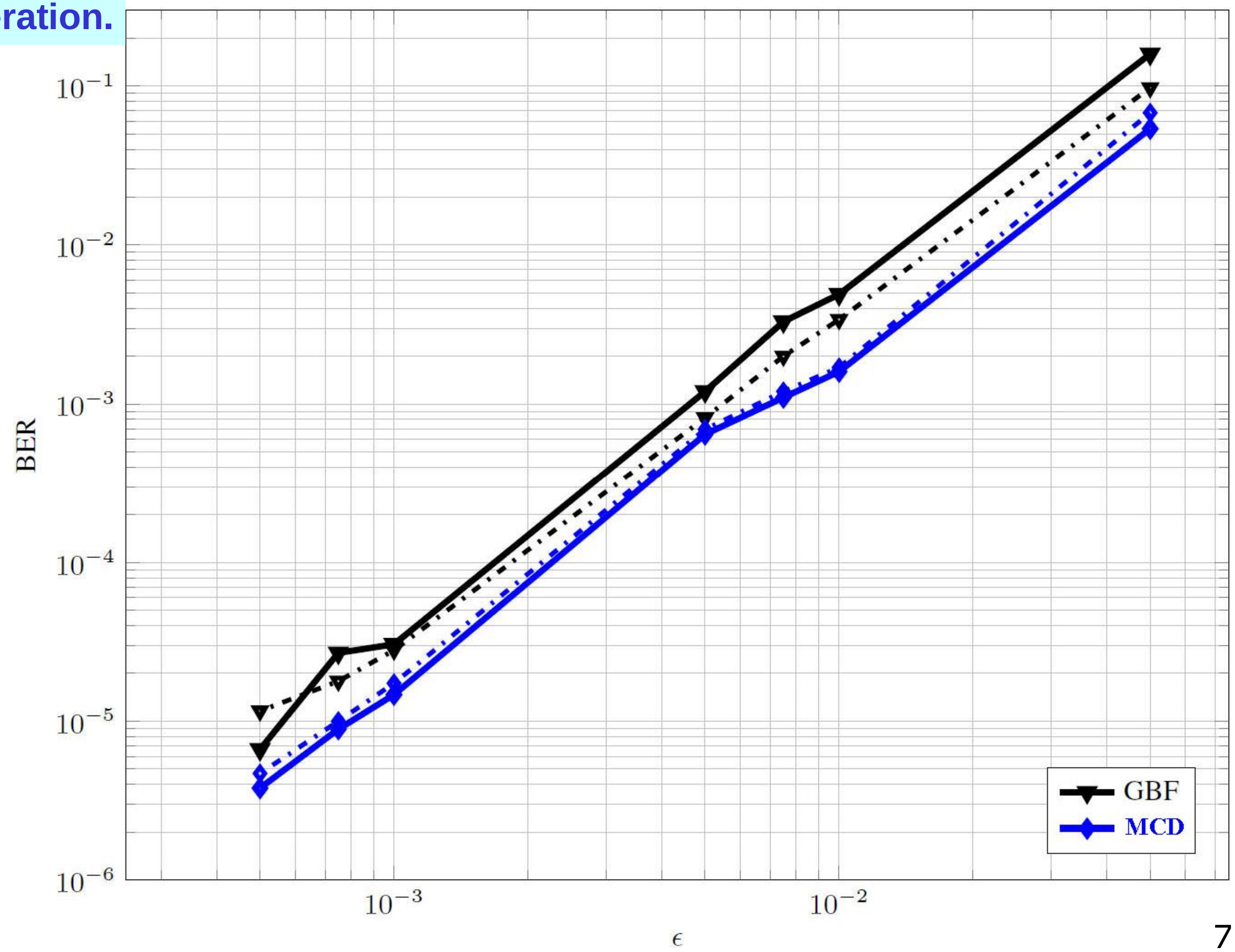
Ů: BSC Channel Parameter

Tab. 3 Maximal Parameter Ŭ* and Maximal Parameter Ů* are determined when it is beneficial to use the decoder under a faulty decoder process.

Good Candidate for the ECC of cDMR: Error-Free Process

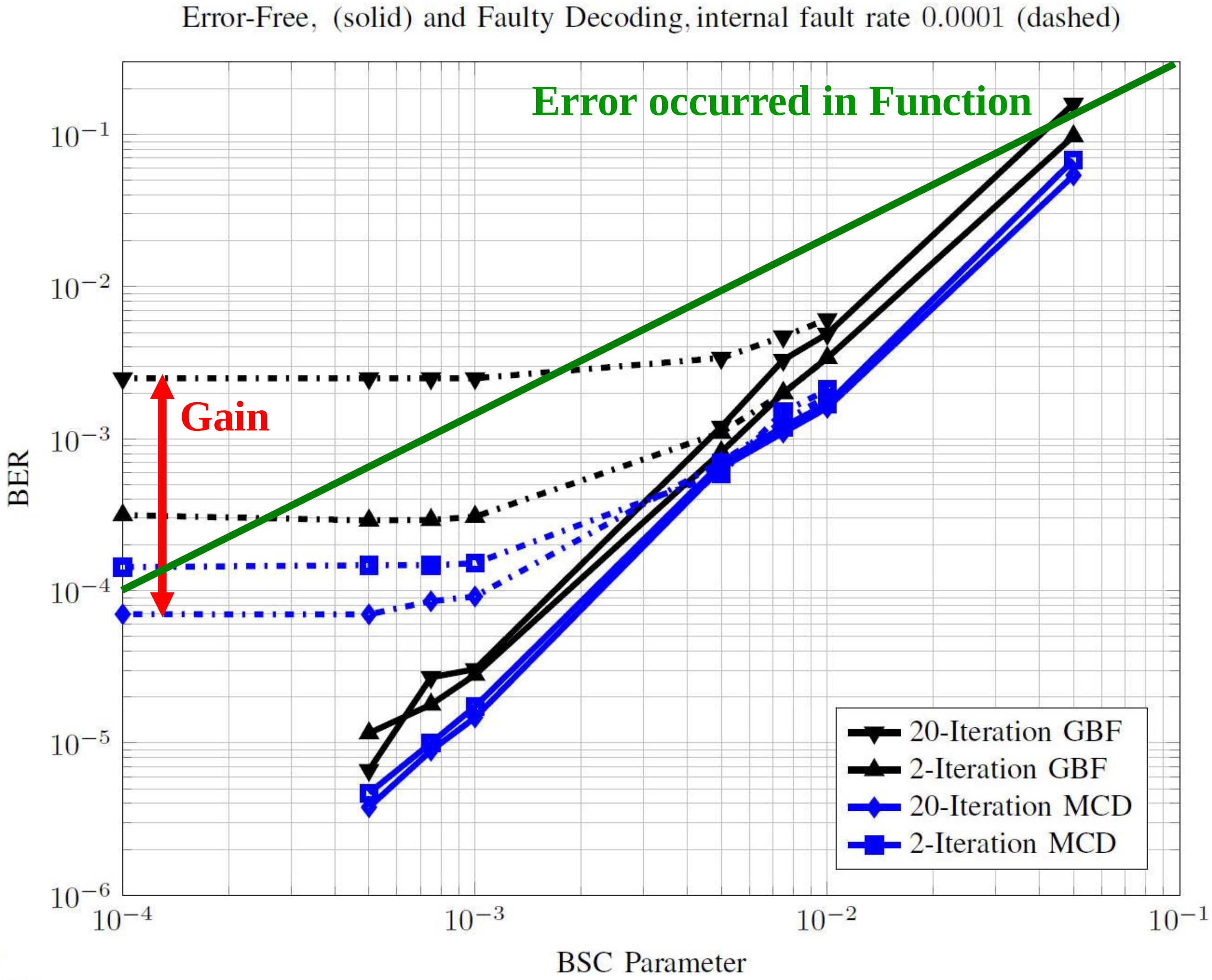
**dash lines for 2-iteration,
solid lines for 20-iteration.**

Ref : Y. Tang et al., ISCAS, 2013.

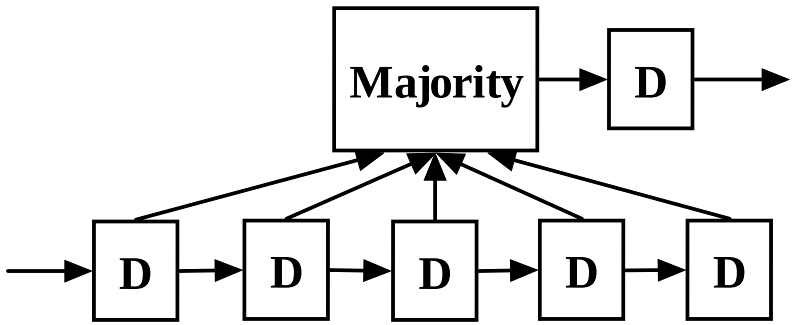


Good Candidate for the ECC of cDMR: Erroneous Process

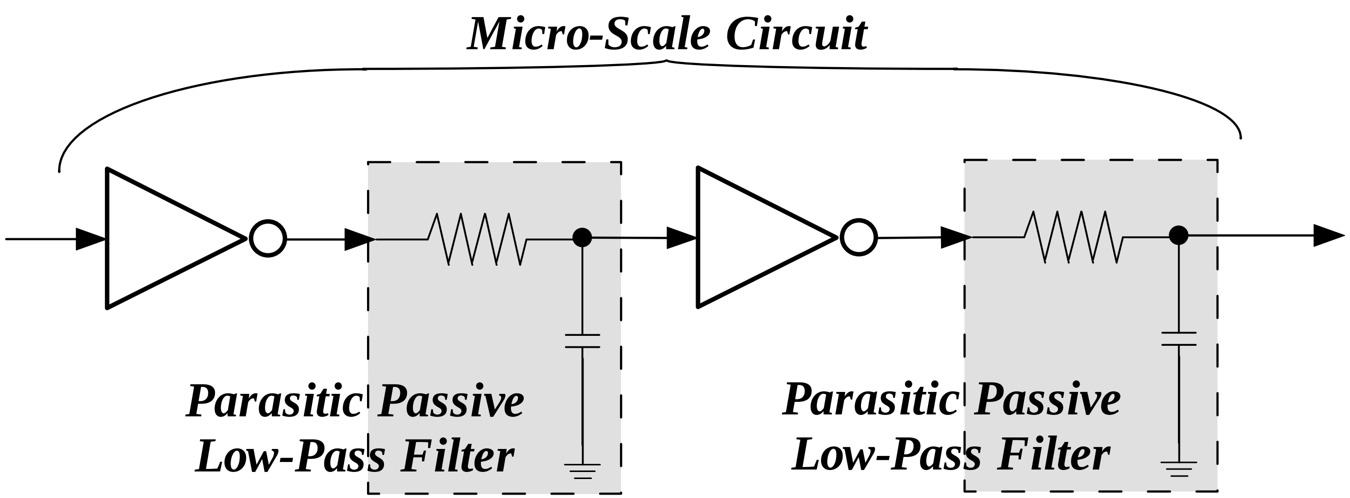
Ref : Y. Tang et al., ISCAS, 2013.



Solution 4: A Reliable Way to Implement S-T Technique

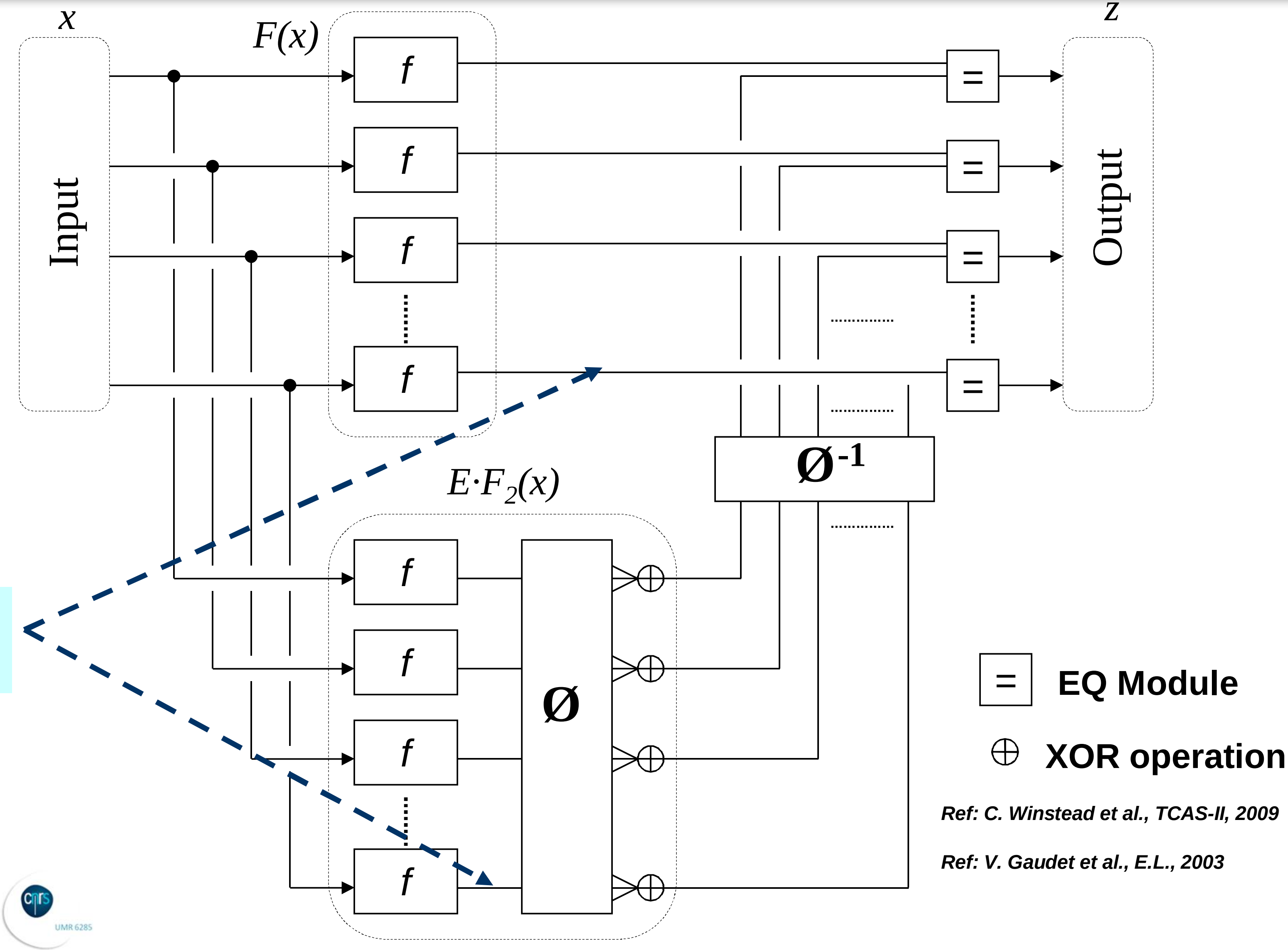


Easy Implementation for Space-Time Redundancy Unit



This approach may be implemented, for instance, with the native R-C parasitics in a large-size output buffer.

Solution 4: Example – LDPC-coded Fault Compensation Technique (LFCT)



Correlated Errors

Ref: C. Winstead et al., TCAS-II, 2009

Ref: V. Gaudet et al., E.L., 2003

Contribution 2: Experimental Results – Pareto Curves

Synthesis Results from XILINX Virtex 5 in Terms of [Slice(#)/Clock Freq.(mhz)].
(*N*: order of filter, *l*: bit length of input)

MA Filter Sizes (N, l)	MA-I (16, 5)	MA-II (32, 6)	MA-III (64, 7)	MA-IV (128, 8)
Strategies	Slice/Fre.	Slice/Fre.	Slice/Fre.	Slice/Fre.
The Original	[94, 424]	[128, 417]	[171, 410]	[232, 403]
S-TMR	[249, 405]	[383, 398]	[508, 300]	[681, 309]
BRRNS Based	[477, 353]	[843, 263]	[1133,232]	[1686,227]
ARQ-Modulo-4	[148, 349]	[205, 312]	[241, 300]	[383, 298]
ARQ-Modulo-8	[168, 349]	[237, 312]	[276, 300]	[408, 298]
ARQ-Modulo-16	[191, 349]	[277, 312]	[322, 300]	[426, 298]
ARQ-DMR	[207, 349]	[302, 312]	[409, 300]	[517, 298]

Åa unity area

Åt clock cycle = 1 (due to the pipeline design)

Solution 1: RRNS – Single Error-Correction

- Redundant RNS (RRNS)
  ⇔ RNS + Redundancy

| Error-Correction and Carry-Free properties |

21 + 17

Redundant Moduli

4, 5, 7, 9, 11

Redundant Residues

1, 1, 0, 3, 10 + 1, 2, 3, 8, 6

2, 3, 3, 2, 5

1, 3, 3, 2, 5

- Consistent-Checking (Detecting)

a_1, a_2, a_3, a_4

~~1,~~ 3, 0, 1, 0 $(a_4=0?)$

3 + 0*5 + 1*5*7 = 38

Error-Free

Solution 1: RRNS – High Latency

- Redundant RNS (RRNS) ⇔ RNS + Redundancy

| Error-Correction and Carry-Free properties |

21 + 17

Redundant Moduli

4, 5, 7, 9, 11

Redundant Residues

1, 1, 0, 3, 10 + 1, 2, 3, 8, 6

2, 3, 3, 2, 5

Soft-Error

2, 3, 3, 2, 6

- Consistent-Checking (Detecting)

a_1, a_2, a_3, a_4

1, 3, 0, 1, ~~a_4~~ (a_4=0?)

Five Iterations at worst is needed to locate the error

Solution 1: Proposed Work – Bidirectional RRNS (BRRNS)

Å(RRNS)

Integer X: 21 + 17

Moduli Set m_i: 4, 5, 7, 11, 13 — Å(RRNS): 4, 5, 7, 9, 11

Residue Vector r_i: 1, 1, 0, 10, 8 + 1, 2, 3, 6, 4

2, 3, 3, 5, 12

Soft-Error → 1, 3, 3, 5, 12

Four of them for each checking

a_1, a_2, a_3, a_4, a_5

Consistent-Checkings

(a_4=0 ?) — a_1, a_2, a_3, a_4: 1, 3, 3, 1

a_1, a_2, a_3, a_5: 1, 3, 3, 3 — (a_5=0 ?)

Recover Integer X from ñInformationò or ñRedundancyò — Bi-Directional

a_1, a_2, a_3: 1 + 3*4 + 3*4*5 = 73 (crossed out)

a_4, a_5: 5 + 3*11 = 38