

# A Symbolic Model-Checking Framework for Transient Fault Robustness Classification and Quantification

ANR-SESUR 2007 – FME3

S. Baarir    C. Braunstein    E. Encrenaz    J-M. Ilié    I. Mounier  
                  D. Poitrenaud    S. Younes (LIP6)

and L. Pierre    R. Leveugle (TIMA)

Dec. 13 2011



# Outline

- 1 Motivation
- 2 Fault and Reparation Models
- 3 Robustness classification and quantification
- 4 Experiments
- 5 Conclusion and Perspectives

# Motivation

- 1 Motivation
- 2 Fault and Reparation Models
- 3 Robustness classification and quantification
- 4 Experiments
- 5 Conclusion and Perspectives

## Context : Circuits' Robustness

### Robustness

Ability of a circuit to behave “correctly” even in presence of some perturbations.

### Possible causes of perturbation

- Fault :
  - Manufacturing defect, component aging (permanent / sporadic)
  - Particle strikes / cross-talk coupling (transient)
- Error : visible consequence of a fault,
- Failure : non conform behavior of the system.

### Several robustness criteria

- Error detection and external reset
- Conformance to a set of reference behaviors

## Contributions

### Needs

Evaluate the robustness level of a circuit for a given perturbation type and robustness criterion

- Multiple occurrence of faults
- Localization of areas to be protected
- Selection and Validation of protection mechanisms

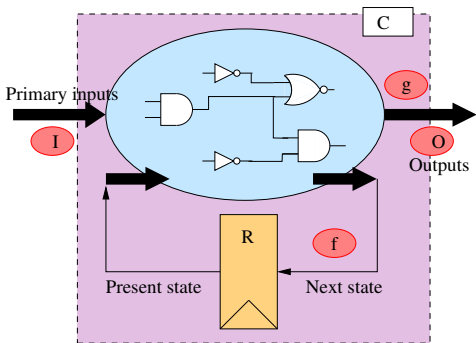
### Our Proposal

Context :

- Synchronous digital circuits (post-synthesis RTL description)
- Transient Faults (multiple occurrence in time and space)

Contributions :

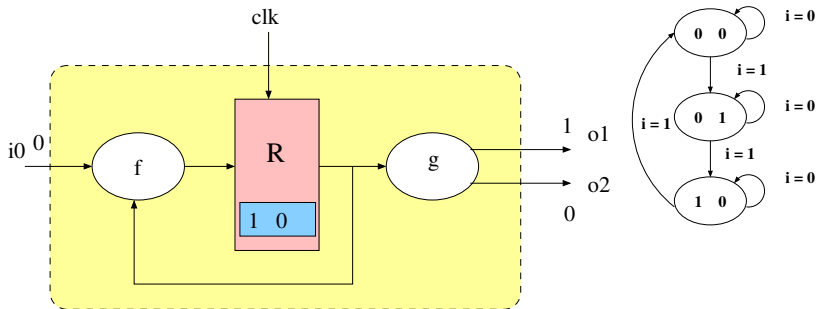
- Faults and Reparation Models
- Robustness classification and quantification



## Reachable States and Sequences

- $\mathbf{r} \in 2^R$ : a state of  $C$
- $\mathbf{R}_0$ : the set of initial state:
- $\mathbf{i}_1.\mathbf{i}_2 \dots \mathbf{i}_{n-1}$ : an input sequence
- $f(\mathbf{r}, \mathbf{i}_1.\mathbf{i}_2 \dots \mathbf{i}_{n-1})$ : a state sequence
- $g(\mathbf{r}, \mathbf{i}_1.\mathbf{i}_2 \dots \mathbf{i}_{n-1})$ : an output sequence
- $reach(C)$ : the set of reachable states of  $C$  from  $\mathbf{R}_0$

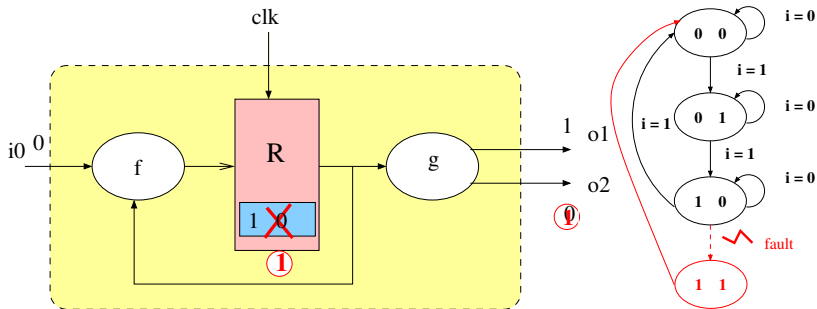
## A fault occurrence producing an error



## Faults

- Single or multiple bit flips in memory elements (registers) [SEU/MEU].
- Errors are observable on the output of the system

# A fault occurrence producing an error



## Faults

- Single or multiple bit flips in memory elements (registers) [SEU/MEU].
- Errors are observable on the output of the system



## Types of Robustness Analysis

### Experimental Outcomes (from [Leveugle 2010])

- Classification of soft errors wrt. a list of pre-defined effects
- Quantification of derating factors : percentage of errors propagating to outputs (block or system level)
- Identification of propagation paths
- Identification of critical locations (register grading)
- Proof of a given set of properties (detection/correction/tolerance mechanisms)

## Analysis Means

### Simulation-based

- Enumeration of fault injection / not exhaustive / statistical simulation + confidence margins [Leveugle 2009][Daveau 2010]
- Hardware emulation with ad-hoc architectures (failure rate, register grading) [Entrena 2007-2009]

### Formal Methods

- Model-checking
  - Equivalence checking of faulty wrt. golden [Leveugle 2005][Drechsler,Fey 2008-2010]
  - Preservation of initial specification [Seshia 2007-2009][Krautz 2007]
- Theorem-Proving
  - Correctness of detection + correction mechanisms (ACL2) [Pierre 2009]

## Our Concern : Self-stabilization evaluation

After a period of particles strikes, how to ensure the circuit's return into a *safe behavior*?

### Analyzing the self-healing capabilities of circuits

Concerns of our measures:

- ① Rates of reparation ability
  - Number of *potentially* and *eventually* repairable states
- ② Reparation Promptness
  - Bounds of the reparations sequences

This allows designers to

- Choose part of design to be hardened
- Choose between different implementations of the same functionality

# Fault and Reparation Models

1 Motivation

**2 Fault and Reparation Models**

3 Robustness classification and quantification

4 Experiments

5 Conclusion and Perspectives

## Fault Model

### Type of faults

- Errors appear as *bit-flips* on register elements.
- There exists a set of *protected* register elements  $P \subseteq R$  (this set may be empty).

### Fault occurrences

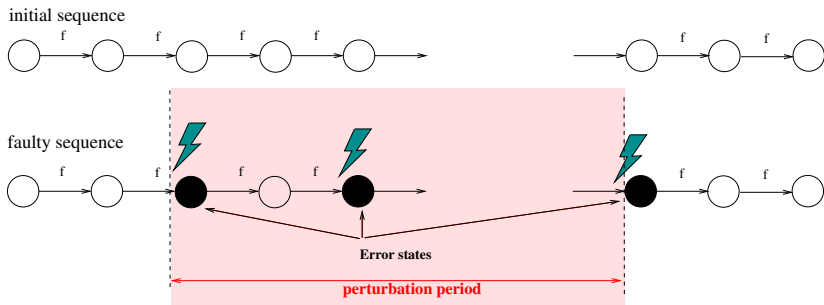
- Spatial multiplicity : faults may occur in multiple registers, except in protected registers.
- Temporal multiplicity : faults may occur at different time instants.

## Faulty sequence

A faulty sequence is characterized by the following properties:

- ① It begins in an initial state of the circuit,
- ② At least one effective fault occurred (at least one bit is affected),
- ③ After a while, the perturbation period ends.

The set of faulty sequences is denoted by  $FS$ .



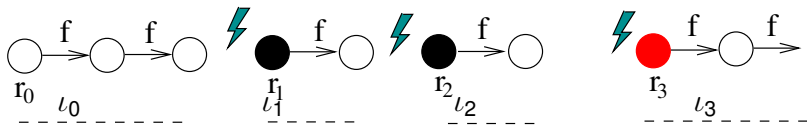
## Error and End of disturbance states

An *error* state follows any fault. The *end of disturbance* state is the state following the last fault occurring along a disturbance period.

### End of disturbance states (Eds)

A state  $\mathbf{r}$  is a *end of disturbance state* iff there exists a faulty sequence  $\sigma = (\mathbf{r}_0, \iota_0) \cdot (\mathbf{r}_1, \iota_1) \dots (\mathbf{r}_n, \iota_n)$  respecting  $f$  and such that  $\mathbf{r}_n = \mathbf{r}$ . Such a faulty sequence  $\sigma$  contains a unique end of disturbance state  $\mathbf{r}$ .

Along  $\sigma$ , states  $\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_n$  are also *Error* states.



## Reparation model

The Reparation Model  $T = \langle Ref, Eq \rangle$  is composed of

- a Reference function  $Ref : FS \mapsto 2^{2^R}$ ,
- an Equivalence relation  $Eq \in 2^R \times 2^R$ ,

### Reference Function

It defines, for each input sequence and a given initial state, a set of reference states, seen as the set of recovery states. This set may depend on the initial state and the input sequence.

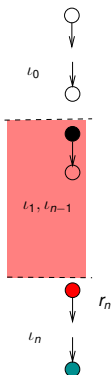
### Equivalence relation

Any equivalence relation between states after a perturbation period and those returned by the reference function (equality, observational equivalence,...)

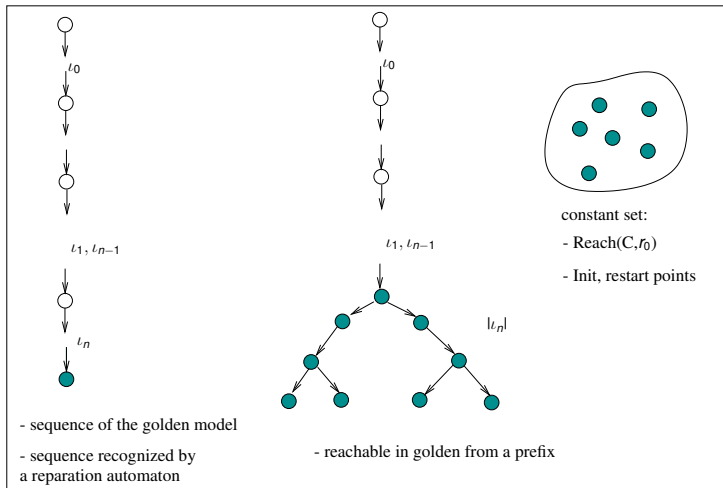


# Several Reference functions

faulted sequence



states returned by the reference function



## Repairing sequence

### Definition (Repairing Sequence)

A faulty sequence  $\sigma = (\mathbf{r}_0, \iota_0) \cdot (\mathbf{r}_1, \iota_1) \dots (\mathbf{r}_n, \iota_n)$  is a *repairing sequence* iff one of the two following cases holds:

- It contains a strict prefix which is a repairing sequence.
- The prolongation of  $\mathbf{r}_n$  by  $\iota_n$  leads to a state equivalent to a state given by the reference function (for the same input sequence). i.e.  $\exists \mathbf{r} \in \mathit{Ref}(\mathbf{r}_0, \iota_0, \dots, \iota_n)$  such that  $(f(\mathbf{r}_n, \iota_n), \mathbf{r}) \in \mathit{Eq}$

## Reparable and Non-Reparable states

### Definition (Reparable and Non-Reparable states)

An end of disturbance state  $\mathbf{r}$  is:

- a **reparable state** iff all faulty sequences having  $\mathbf{r}$  as end of disturbance state are repairing;
- a **potentially-reparable state** if  $\mathbf{r}$  some of the faulty sequences are repairing, but not all of them.
- a **non-reparable state** in the other cases.

## Robustness Measures

Reparable states measure :

$$\nu_{pot} = \frac{|PotentiallyReparable|}{|Eds|}$$

$$\nu_{ev} = \frac{|Reparable|}{|Eds|}$$

Length of reparation sequences:

For each  $k$  :

count non-looping elementary reparation sequences of length  $k$ .

# Robustness classification and quantification

- 1 Motivation
- 2 Fault and Reparation Models
- 3 Robustness classification and quantification**
- 4 Experiments
- 5 Conclusion and Perspectives

## Classification Criteria

Question : What is the reference, and how to compare with it ?

### Observed elements

- Internal registers + outputs : complete state of the circuit
- Outputs only : observational equivalence.  $\mathbf{r} \equiv \mathbf{r}'$  iff:

$$\forall \iota \in (2^I)^*, \forall o \in O, g(f(\mathbf{r}, \iota))[o] = g(f(\mathbf{r}', \iota))[o]$$

### Divergence acceptance

- No divergence : the environment is not able to support any non correct output at any time (communication protocols without recovery support)
- Finite divergence : the environment tolerates some non correct outputs for a while (video stream : corrupted images, commit-rollback execution)
- Infinite divergence : switch to a downgraded functioning

## Robustness Classification

- **Class 1:** no observable difference for any output from the golden model
  - [Lev05], [KPJ<sup>+</sup>06], [FD08, FSD09], itc-99
- **Class 2:** initial states satisfy the same set of (CTL) properties
  - [SLM07], spacewire ctrl
- **Class 3:** finite observable difference. will eventually become equivalent to the golden model for the same input sequence.
- **Class 4:** infinite observable difference. will eventually return into a set of Safe (Recovery) states
  - [BBC<sup>+</sup>09, BBE<sup>+</sup>11], gcd, multipliers, ATM, itc-99

Classes 1, 3 and 4 are expressible with our reparation model while class 2 is not.

## Robustness Classification

Let  $\sigma = (\mathbf{r}_0, \iota_0).(\mathbf{r}_1, \iota_1) \dots (\mathbf{r}_n, \iota_n)$  be a faulty sequence,

Robustness	<i>Ref</i>	<i>Eq</i>
Class 1	<i>Ref</i> <sub>1</sub>	$\equiv$
Class 2	not definable	
Class 3	<i>Ref</i> <sub>3</sub>	= or $\equiv$
Class 4	<i>Ref</i> <sub>4</sub>	= or $\equiv$

$$Ref_1(\sigma) = \begin{cases} \{f(\mathbf{r}_0, \iota_0 \dots \iota_{n-1})\} & \text{if } \iota_n = \epsilon \\ \emptyset & \text{otherwise} \end{cases}$$

$$Ref_3(\sigma) = \{f(\mathbf{r}_0, \iota_0 \dots \iota_n)\}$$

$$Ref_4(\sigma) = \begin{cases} Inv & \text{if } cd(\sigma) \\ \emptyset & \text{otherwise} \end{cases}$$



## Robustness Quantification

- State quantification :
  - Build an instrumented circuit embedding the reference function,
  - Compute  $|Eds|$  according to the fault model,
  - Compute  $|Potentially\ Repairable|$ ,  $|Repairable|$  by symbolic Model Checking.
  
- Repairing velocity :
  - Compute *min* and *max* length of elementary repairing sequences
  - Compute distribution of elementary repairing sequences between *min* and *max*

## Measure Computation for each robustness class



### Computation of EDS (multiple fault model)

- *Each faulty state is potentially an end of disturbance state.*
- Compute the set *Eds* :
  - 1  $S = \text{post}^*(r_0)$
  - 2  $S' = \text{Inject any fault in } S \text{ (according to } P)$
  - 3  $S'' = \text{post}^*(S')$
  - 4 if  $S'' \neq S : S \leftarrow S''$ , goto 2 else return  $S''$

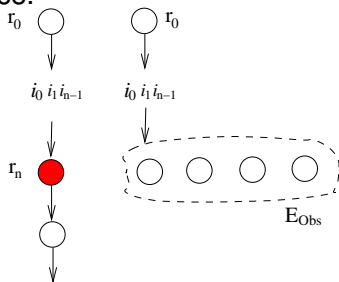
Fault injection is based on BDD manipulation of the symbolic state space. The effectiveness of the fault is ensured.

# Computation of repairing states, class 1

Reference function : no observable difference.

## Computation of repairing states

- 1 Build  $S = F \parallel G$
- 2 Compute  $Eds(S)$  (all registers of  $G$  are protected) : of the form  $(\mathbf{r}_f, \mathbf{r}_g)$
- 3 Repairable are states in  $Eds$  satisfying  $\mathbf{r}_f \equiv \mathbf{r}_g$ , i.e. satisfying  $AG(\mathbf{o}_f = \mathbf{o}_g)$

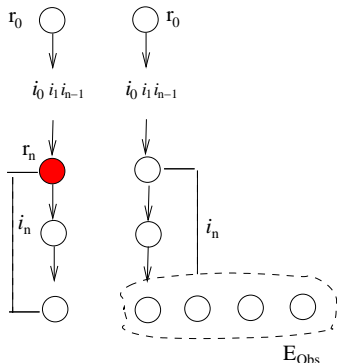


# Computation of repairing states, class 3

Reference function : finite observable difference.

## Computation of repairing states

- 1 Build  $S = F \parallel G$
- 2 Compute  $\text{Eds}(F \parallel G)$  (all registers of  $G$  are protected)
- 3 Compute
  - Repairable =  $\mathbf{r} \models \text{Eds} \wedge \text{AF}(\text{AG}(\mathbf{o}_f = \mathbf{o}_g))$
  - Potentially repairable =  $\mathbf{r} \models \text{Eds} \wedge \text{EF}(\text{AG}(\mathbf{o}_f = \mathbf{o}_g))$
  - Non repairable =  $\mathbf{r} \models \text{Eds} \wedge \text{EG}(\neg(\text{AG}(\mathbf{o}_f = \mathbf{o}_g)))$

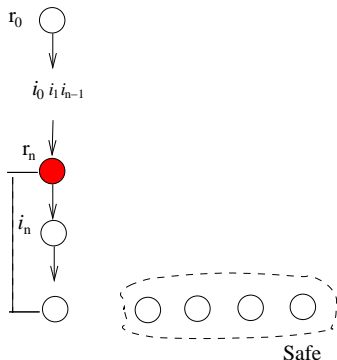


## Computation of repairing states, class 4

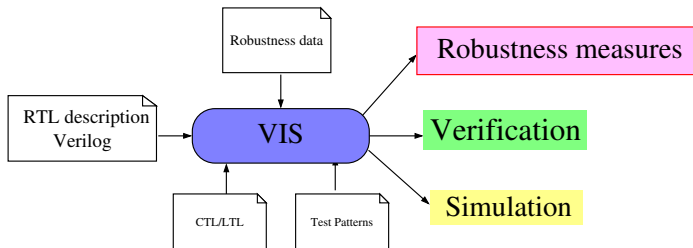
Reference function : will eventually return into a set of Safe (Recovery) states

### Computation of repairing states

- 1 Compute  $E ds(F)$
- 2 Compute
  - reparable =  $r \models E ds \wedge AF(Safe)$
  - potentially reparable =  $r \models E ds \wedge EF(Safe)$
  - non reparable =  $r \models E ds \wedge EG(\neg Safe)$



# Robustness Analysis framework

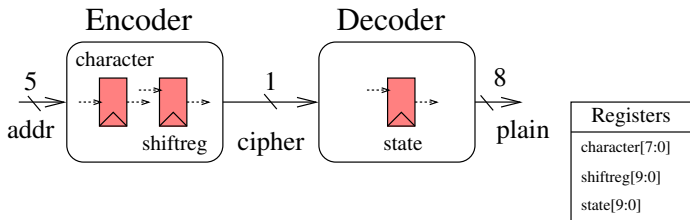


The VIS group. VIS : A System for Verification and Synthesis. In R. Alur and T. A. Henzinger, editors, *CAV'96: Proceedings of the 8th International Conference on Computer Aided Verification*, volume 1102 of *LNCS*. Springer-Verlag, 1996

# Experiments

- 1 Motivation
- 2 Fault and Reparation Models
- 3 Robustness classification and quantification
- 4 Experiments**
- 5 Conclusion and Perspectives

## Case study: a Huffman encoder-decoder [project of X. Wang, 2011] (1)



### Static encoding/decoding

- Encoder: Read a character and return the encoding form bit by bit
- Decoder: Explore the encoding tree and return the character.



## Huffman: First analysis

### Robustness analysis without any protection

Rob	Fault	$ Err $	$\nu_{ev}$	$\nu_{pot}$	time
Rob4	SEU	61654	48%	99.5%	0
	MEU	2.68e+08	7%	99.8%	0
Rob3	SEU	1.43e+06	45%	99.6%	1min
	MEU	2.36e+11	0.6%	99.8%	1min
Rob1	SEU	1.43e+06	30.5%	47%	-
	MEU	2.36e+11	0%	0%	-

Originally reachable states : 878

# Huffman: Register grading

## Which registers are critical ?

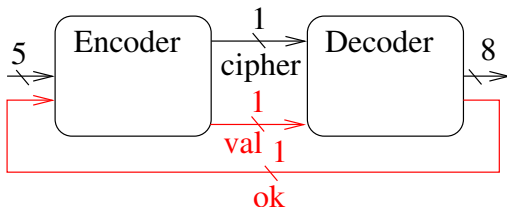
Protected registers	Fault type	Rob 4		Rob 3		Rob 1
		$\nu_{ev}$	$\nu_{pot}$	$\nu_{ev}$	$\nu_{pot}$	$\nu_{ev} = \nu_{pot}$
character	SEU	25.17%	99.3%	22.63%	99.3%	2.72%
	MEU	7.07%	99.8%	0.59%	99.8%	0%
state	SEU	58.45%	99.3%	55.13%	100%	44.61%
	MEU	24.73%	99.8%	2.22%	100%	0.10%
shiftreg	SEU	62.80%	100%	58.90%	100%	47.30%
	MEU	10.92%	100%	13.86%	100%	0.62%
character/ state	SEU	25.45%	99.8%	19.23%	99.8%	0.30%
	MEU	24.89%	99.8%	2.12%	100%	0%
character/ shiftreg	SEU	24.62%	100%	26.02%	100%	5.14%
	MEU	7.39%	100%	13.79%	100%	0.53%
state/ shiftreg	SEU	100%	100%	100%	100%	100%
	MEU	100%	100%	100%	100%	100%

## Huffman: Strengthening robustness

### Two approaches

- Triplicate register: replace register by TMR
- Periodic reset mechanism

## Huffman: Periodic reset implementation



### Encoder

- Re-initialize every 5 characters
- Send a signal (val) to the decoder
- Wait the acknowledgment from the decoder

### Decoder

- Receive a reset signal from the encoder
- Re-initialize
- Send the acknowledgment to the encoder

## Huffman: Reset robustness

### Robustness analysis with "reset" protection

Class	Fault	$\nu_{ev}$	$\nu_{pot}$
Rob4	SEU	100%	100%
	MEU	100%	100%
Rob3	SEU	78.8%	100%
	MEU	4.12%	100%
Rob1	SEU	33.3%	-
	MEU	0%	-

## Case study: a Huffman encoder-decoder (8)

Protection mechanisms :

### TMR

- adapted for single spatial / multiple temporal faults
- immediate correction -> class 1 robustness
- costly : add of  $2 * 20$  FF for a initial circuit of 28 FF + vote glue logic

### periodic reset

- adapted for multiple spatial / multiple temporal faults
- delayed reparation -> class 4 robustness (but not class 3)
- cheap : add a 3 bit counter.

# Conclusion and Perspectives

- 1 Motivation
- 2 Fault and Reparation Models
- 3 Robustness classification and quantification
- 4 Experiments
- 5 Conclusion and Perspectives**

## Conclusion

### New metrics

- Identification of several robustness criteria wrt. self-healing capabilities
- Metrics to determine
  - the minimal set of registers to be protected for a given architecture
  - select a more robust architecture between several possibilities
  - evaluate several reparation strategies

### A Robustness Analysis Framework

- Single or Multiple transient faults by symbolic management
- Early in a design flow
- First implementation within a classical model checker (VIS)



## Perspectives (1)

### Framework improvements

- Robustness classification : something in-between Class 3 and 4
- Calculus :
  - Compositional reasoning
  - Use of abstractions

### Evaluation and enhancement of the robustness level

- Automatic insertion of reparation patterns
  - Localization + correction with ad-hoc structures (TMR, ECC)
- Insertion of several functioning modes and switch
  - Synthesis of external controller driving execution modes

## Perspectives (2)

### Other levels of modeling

Asynchronous concurrent tasks, SystemC-TLM

Links between high-level and RTL robustness measures

### Robustness Debug

How to combine robustness measures and properties to identify causes of non-robustness ?

### Other robustness qualification

How to avoid the pure state-based model ? Combination of identified state-space structures ?