Fault Tolerant Linear State Machine Design Approach for Safety Critical Systems Implemented on FPGA

Sunanda, Fathima Rehana, Abdulla Gubbi
VTU University, P.A. College of Engineering Nadupadav Manglore India

Abstract: In this paper, a new method for the design of fault tolerant linear state machines with initial state 0 and one dimensional input and one-dimensional output is proposed. It is shown that the LFSR-implementation of the transfer function of a linear automaton can be utilized to correct transient errors in the memory elements. Since the state vector of a linear automaton is uniquely determined by the last n inputs and outputs, a transient error in a memory element can be corrected within n clock cycles by use of the corrected output symbols, where n is the number of components of the state vector.

Keywords: TMR (triple modular redundancy), Voter logic.

I. INTRODUCTION

New method for the design of fault tolerant linear state machines with initial state 0 and one dimensional input and one-dimensional output is proposed. It is shown that the LFSR-implementation of the transfer function of a linear automaton can be utilized to correct transient errors in the memory elements. Since the state vector of a linear automaton is uniquely determined by the last n inputs and outputs, a transient error in a memory element can be corrected within n clock cycles by use of the corrected output symbols, where n is the number of components of the state vector.

Linear state machines are of importance in different fields of application such as encoding and decoding of cyclic codes, especially BCH codes generation of pseudorandom test patterns and test response compaction by multiple input signature analyzers, the implementation of stream ciphers and others. In modern technologies the number of errors in memory elements caused by transient faults is increasing and for many applications fault-tolerant designs are now of growing interest. The standard approach for fault tolerance is triple modular redundancy (TMR).

A system is triplicated into three (functionally) identical systems and the outputs of the triplicated Systems are connected to a three-input voter which determines the output of the fault-tolerant system as the majority of the outputs of the triplicated systems. The main advantages of TMR are that errors due to an arbitrary fault in only one of the triplicated systems are tolerated, and that no specific error model is required.

II. OBJECTIVE

1. Study the state machine theory.
2. Study the fault-tolerant application in VLSI systems.
3. Develop a RTL fault-tolerant state-machine model and verify it using a RTL test bench.
4. Develop an application for fault-tolerant state machine and implement it on FPGA to observe the performance.

III. PROJECT BACKGROUND

If the linear state machine is having faults due to the transient errors those errors will be corrected by TMR technique and also by the voter logic which will be implemented on a FPGA. The corrected outputs of the three voters are besides x(t) the input of the corresponding triplicate linear state machines. The linear state machines are duplicated. Two parity bits are implemented for error detection. The linear state machines are duplicated. A single parity bit is used for error detection.

IV. PROPOSED SYSTEM

A linear state machine which is also called a LDA, linear automaton over a field K is defined by the next-state function

\[ z(t+1) = A \cdot z(t) \oplus B \cdot x(t) \]

And the output function

\[ y(t) = C \cdot z(t) \oplus D \cdot x(t) \]

Where x(t), y(t) and z(t) are the 1-dimensional input vector, the m-dimensional output vector and the n-dimensional state vector at time t with its components from the field K. The general principle of the proposed fault-tolerant design for linear state machines with initial state z(0) = 0 is to implement the output y(t) as a fault-tolerant signal y^\text{corr}(t) and to use this fault-tolerant signal for state correction. Considered two methods they are:

- Triplication with State Correction
- Duplication, Parity Checking and State Correction

The following design proposals are made:

- The linear state machine is triplicated and the outputs are voted by a single voter. The corrected
output $y^{cor}(t)$ is besides $x(t)$ the input of all the triplicates linear state machines.

- The linear state machine is triplicates and the outputs are voted by three voters. Put a right outputs of the three voters are besides $x(t)$ the input of the corresponding triplicates linear state machines.

- The linear state machines are replicated. For error detection two parity bits are implemented.

- The linear state machines are repeated. A single parity bit is used for error detection.

**V. EXPERIMENTAL RESULTS**

Since in all proposed designs single transient errors in the state components and in the combinational circuit parts are corrected (at least after n clock cycles), they are compared with known solutions, which also correct state errors and errors in the combinational parts of the circuit. Linear state machines of different dimensions (from 8 to 256 bits)

**VI. RESULTS WITHOUT TMR**

**VII. RESULT WITH CORRECTION**
VIII. CONCLUSION
A new method for the fault tolerant linear state machine design approach for safety critical systems implemented on FPGA, in this linear state machines with initial state 0 and one-dimensional input and one-dimensional output was proposed. The usage of state machine in order to efficiently control mechanism within design has become very popular. When the n number of faults occurs in the microprocessor core that will be corrected by using the technique that is TMR (triple modulation redundancy) that is using the replication block triplicates the all the flip flops within the n cycles and also by the voter logic it chooses the best result. Where n is the number of components of state vector.

Without the TMR getting the synthesis result using the minimum period that is 13614ns and the maximum frequency is 73.876MHZ, and also area will be more that is 73,876 and speed also increasing up to 1238 slices.

With TMR getting the synthesis result using the minimum period that is 13536ns and the maximum frequency is 73.452MHZ, and also area will be more that is 73,452 and speed also increasing up to 1239 slices.

Experimental results have shown that the lowest area overhead can be obtained if the linear state machine is duplicated and a single parity bit is used to distinguish which of the duplicated machines is correct.

Special directives must be used in order to drive the synthesis tools when implementing fault tolerant redundant logic because the tools are generally focused on area and speed optimization. Thus, once the gates are produced, the designer should check that no functionality has been “optimized” away and that the appropriate state machine has been realized.

IX. FUTURE ENHANCEMENT
For future work we are planning to extend the evaluation test bed to study the application performance with a larger application and also by considering developing a richer with additional features. Special directives must be used in order to drive the synthesis tools when implementing fault tolerant redundant logic because the tools are generally focused on area and speed optimization. And also for the timing optimization thus, once the gates are produced, the designer should check that no functionality has been “optimized” away and that the appropriate state machine has been realized. Using the state machines in the microprocessor avoiding the n number of faults. By another technique that is online error correction reducing the n number of faults in the microprocessor. Especially using the 5th modular redundancy technique correcting the one bit errors in each flip-flop with high speed.

X. APPLICATION
Aerospace
Data Communication
Ships

REFERENCES
[1] Fault Tolerant Linear State Machines Stefan Weidling and Michael Goessel, Department of Computer Science(2014) University of Potsdam, Potsdam, Germany.