DESIGN APPROACH FOR FAULT RECOVERABLE ALU WITH IMPROVED FAULT TOLERANCE

Ankit K V[1], S Murali Narasimham[2] and Dr. Viajya Prakash A M[3]

[1]PG Student, Department of E & C, B.I.T, Bangalore, KA, India
[2]Associate Professor, Department of E & C, B.I.T, Bangalore, KA, India
[3]Professor, Department of E & C, B.I.T, Bangalore, KA, India

ABSTRACT

A new design for fault tolerant and fault recoverable ALU System has been proposed in this paper. Reliability is one of the most critical factors that have to be considered during the designing phase of any IC. In critical applications like Medical equipment & Military applications this reliability factor plays a very critical role in determining the acceptance of product. Insertion of special modules in the main design for reliability enhancement will give considerable amount of area & power penalty. So, a novel approach to this problem is to find ways for reusing the already available components in digital system in efficient way to implement recoverable methodologies. Triple Modular Redundancy (TMR) has traditionally used for protecting digital logic from the SEUs (single event upset) by triplicating the critical components of the system to give fault tolerance to system. ScTMR- Scan chain-based error recovery TMR technique provides recovery for all internal faults. ScTMR uses a roll-forward approach and employs the scan chain implemented in the circuits for testability purposes to recover the system to fault-free state. The proposed design will incorporate a ScTMR controller over TMR system of ALU and will make the system fault tolerant and fault recoverable. Hence, proposed design will be more efficient & reliable to use in critical applications, than any other design present till today.

KEYWORDS

Fault recovery, tolerance, ALU, ScTMR, ALU designing, critical applications, scan chains & ScTMR.

1. INTRODUCTION

Today is the era of technological advancement in number of features contain by any system. The technological trends are moving towards incorporation of more and more functionality inside small integrated chips. The exponential growth in amount of SOC design coming out in market further supports this trend. But the primary characteristic property of any device still counts with Reliability. Accumulation and increment of features cannot overrun the surety requirement a device have to show to last in the challenging competitive market. A lot of research is going nowadays in VLSI design domain to make designs more reliable and error resistant.

For all the critical applications, we need to develop techniques which will make a design more reliable and error resistant. Just testing and verification is not enough to increase the reliability of the system but design should have inbuilt property to tolerate the fault, if any occurs while the device is in operation. Moreover, just fault tolerance is not sufficient without incorporating the fault recovery techniques also inside the design, for making it completely reliable and less error prone.

Microprocessors are the most critical, important and necessary part of any modern digital systems and the heart of microprocessor is ALU [1]. If the ALU itself got faulty, the entire device will

come to halt. Hence, it is a necessity to make ALU to be fault tolerable in all the critical application where reliability is most critical factor.

In this paper a new design for ALU system is purposed which is not just more fault tolerant but also have fault recovery mechanism inbuilt inside. This has been designed keeping the size and area penalty minimum by reusing the already present scan chains in the design, to get fault recovery.

2. FAULT TOLERANT SYSTEMS

If system able to continue its normal operation even after failing of one or more of its components, then this property of the system is called fault tolerance. The operating quality is proportional to the severity of the failure i.e. operating quality decreases as the severity of the failure increases for naively designed systems [2]. Fault tolerance becomes substantial design criteria for the applications where the reliability of hardware is crucial. Medical, military and long-range missions are such applications that the fault tolerance of hardware became key issue. Fault tolerant systems are such systems wherein even if a certain part of a system fails; it doesn't affect the normal operation of that system [3]. The degree of normal operation depends on the degree or harshness of the fault. In other words the normal operation becomes more and more critical with the increase of harshness of the fault.

The main and crucial characteristics of a fault tolerant system are:

- The designed system must be intelligent enough to carry on the normal operation for the duration of the fault, i.e. until it is repaired
- The system must be able to identify the faulty components and isolate them
- All sorts of supplies and connection to the faulty component must be put off, i.e. the faulty component must be forced to go dead till it is repaired and becomes operational.

The critical components or functions of the system are duplicated or might be triplicated, so as to increase the reliability of the system. This process is called redundancy.

3. RECOVERY TECHNIQUES

If desired level of operational conditions is re-establish after occurrence of fault in the system then this phenomenon is called as fault recovery [11]. A system which is capable of recovering from the fault after error event is said to be fault recoverable system. Fault recovery can be achieved by many techniques depending upon nature of faults. Few faults are temporary and easy to recover like transient faults, struck at faults etc. while some are permanent faults and hard to recover [5]. Scan chains available in design for testability can be used to re-establish the sates of internal circuitry, and thus recover the fault in system.

Generally there are two famous recovery techniques used in VLSI designs: roll-back recovery and roll forwards recovery technique. In former the checkpoints are defined in the design and while executing if a fault occurs, then the system goes back to immediate previous point to loads the previous saved states into the logic circuitry and start functioning again from that point.

In case of roll forward technique, the states from another fault free module will be loaded into the faulty module and operation will be resumed from that point itself. This later one is much better and efficient as saves the time required for re-computation in the former.

4. PROPOSED DESIGN METHODLOGY

The following section contains all the integral information about the proposed design of Fault recoverable ALU System with all components and their description with block diagrams.

4.1. Triple Modular Redundancy

It has been known that the reliability of digital systems can be improved through the use of redundant components, if these additional components are properly employed.

The most common type of redundancy method is Triple Modular Redundancy (TMR). Triple modular redundancy, (TMR) is a fault-tolerant form of N-modular redundancy, in which three systems perform a process and that result is processed by a voting system to produce a single output [6]. If any one of the three systems fails, the other two systems can correct and mask the fault. If the voter fails then the complete system will fail. The majority voter uses voting logic as shown in Fig. 1.
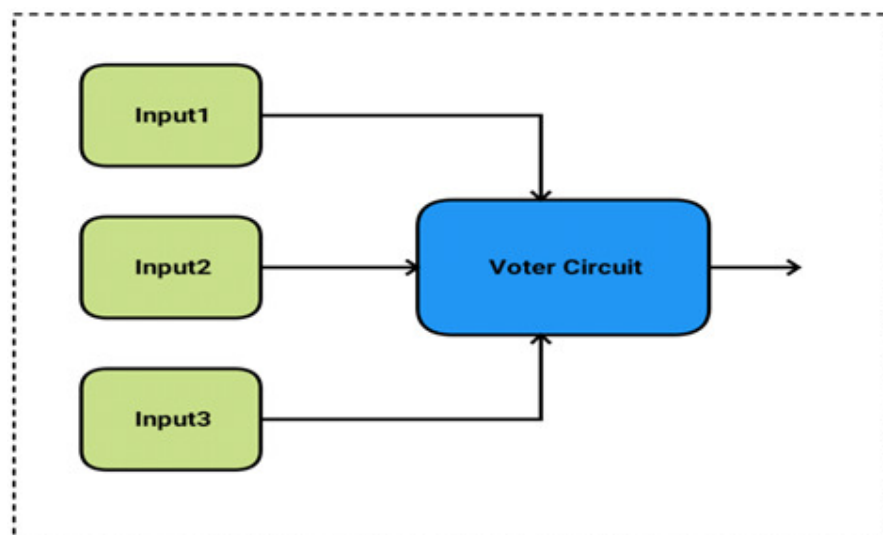

Figure 1. Block diagram of voting logic circuit used in designing TMR system

In TMR, all the three inputs are compared by the majority voter circuit and then the final output is decided upon. Suppose two out of three modules have same outputs then the majority voter decides which replication has error as two-to-one vote is pragmatic by the majority voter [8]. After this decision is made only two modules are left and the majority voter can switch to dual modular redundancy (DMR).

TMR can be employed for N number of imitations. The arrangement of redundancies will not fail if none of the three modules fails, or if just one of the three modules fails. The voting logic evaluates the output of all the modules and then it passes the majority output i.e. if all three outputs are equal then it turns out to be the final output and if two out of three outputs are equal then the two similar outputs turn out to be the final output. Also, if the two equal outputs are falsified output then it will turn out to be the final output.

4.2. Proposed ALU Design

Arithmetic Logic Unit (ALU) is the most critical part of any microprocessor or microcontroller. It is also the heart and soul of any central processing unit (CPU). ALU implements logic operations, such as AND, NOT, OR etc., and arithmetic operations, such as ADD, SUBTRACT, MULTIPLICATION etc. [7]. Most ALUs can perform following operations:

a. Bitwise logic operations: - NAND, NOR, XNOR

b. Integer arithmetic operation

c. Bit-shifting operations

The ALU module proposed in this paper have more sequential element to accommodate scan chains functionally. Pipelining is used to increase the throughput as well as for inclusion of DFT (design for testability) in design [9]. Generally only adder is used in ALU systems to realize other arithmetic logic like subtraction and multiplication etc [4]. In our proposed design the adder used is pipelined RCA (ripple carry adder) because of its compact design [10].

The scan chain has been added in the design by replacing registers with a scan cell named D-mux Cell. The scan cell works as a large shift registers while testing and for normal operation work as a simple D-Flip flop [9]. The scan logic is implemented in RTL code and later it will be used for fault recovery also by inbuilt recovery controller ScTMR.

4.3. Fault Tolerant ALU System

Triple modular redundancy has been used in order to design the fault tolerant ALU system. In this process the ALU system carried out is triplicated, each having the same input, thus making it triple mode redundant. The output of all the three ALUs is passed onto the Voting Circuit which contrasts the outputs and then passes the majority output.

It means that if any two ALUs are giving the similar output, then that output will be passed by the voting circuit and that turns out to be the final output of the entire circuit. In case of all the ALUs giving the similar outputs, then that output becomes the last output but in case of all the three ALUs giving dissimilar outputs then the voting circuit faces a dilemma and it fails. At this time the final output is undefined.
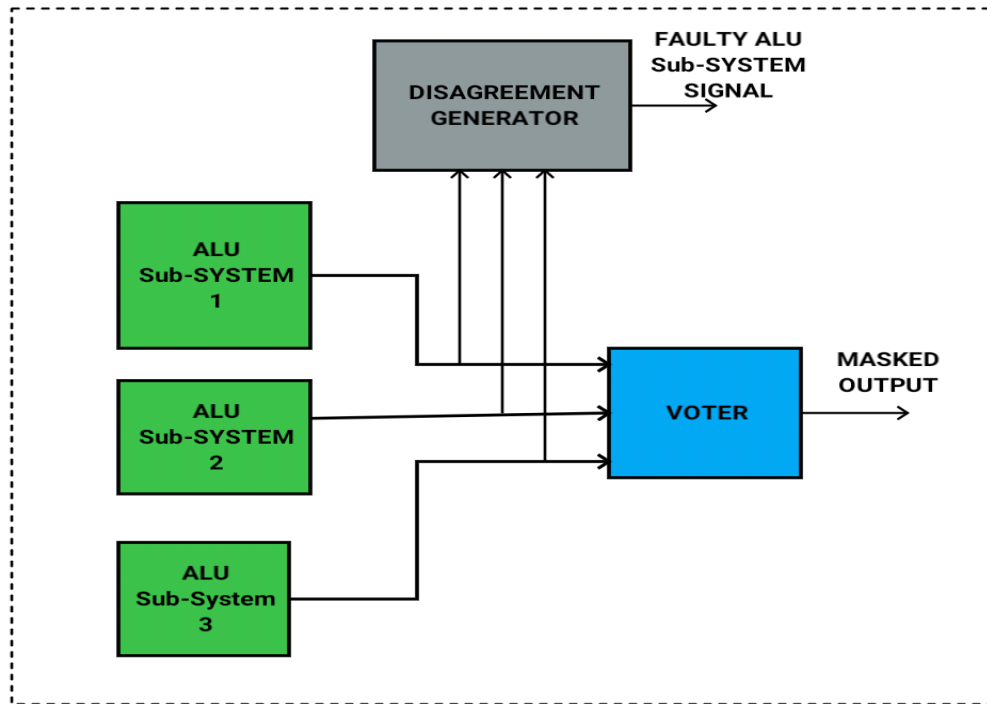
Figure 2. Block diagram of fault tolerant ALU system design using triple modular redundancy

Thus, we have made the ALU system fault tolerant to a great level but still the problem persists. It's because practically we are unable to make a 100% fault free system. We can reduce the level of fault occurrence but we cannot totally omit it. In the above "Fault Tolerant ALU System", there is a limitation i.e. it fails if N-1 systems become faulty.

 In other words, out of N systems (where N being odd no. of systems), if N-1 systems are faulty then our model fails. In case of ALU, out of three ALUs, if any two ALUs fail then the whole model fails. A further improvement can be done by incorporating fault recovery technique, to overcome any internal fault while system is in operation.

4.4. ScTMR Design for Fault Recovery

There are two methods for recovery using scan chains one is ScTMR and other is SMERTMR. Where SMERTMR provide better fault coverage, its time & logic overhead can't be neglected. While using simple ScTMR will not be apt utilization. The best the policy is to utilize the advantages of both of these, hence both has been incorporated in the proposed design to develop a new mixed custom technique It will use few new signals to implement the handshake between different blocks like SMERTMR plus all the basic features of ScTMR.

The ScTMR includes: Three redundant modules, a voter, a controller.

In the proposed design once an error is detected by the voter, the ScTMR controller triggers an appropriate recovery mechanism to eliminate the error from the system. This is achieved by copying the state of a fault free module to the detected faulty module using the scan-chain circuitry.

The recovery process is done through the scan-chain input (SCI), scan-chain output (SCO), and scan-chain enable signals instructed by the ScTMR controller. Once an error is detected by the

voter, it alerts the ScTMR controller using an error signal. In the proposed voter, an output selector circuit is used to route the error-free output to the ultimate output signal.
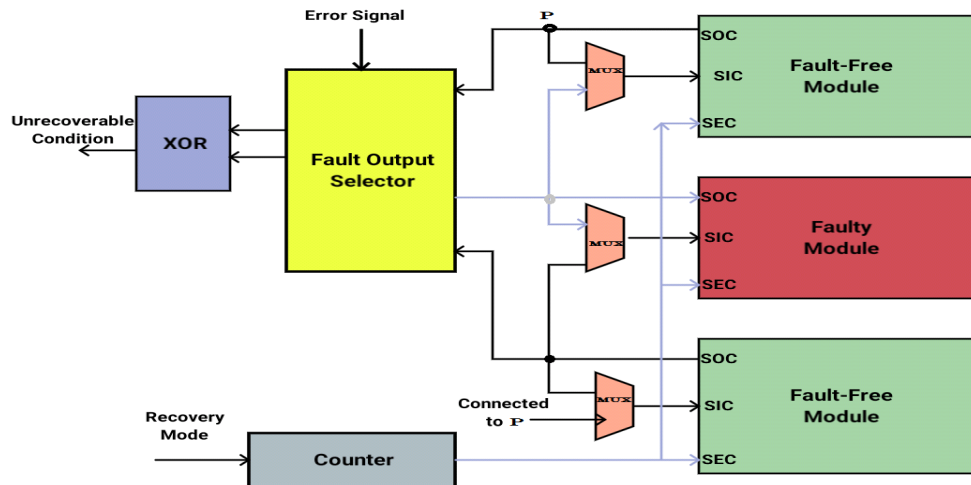
4.5. Recovery Process

Recovery is achieved by copying the state of a fault free module to the detected faulty module using the scan-chain circuitry.

The recovery process is done through the scan-chain input (SCI), scan-chain output (SCO), and scan-chain enable signals instructed by the ScTMR controller. Once an error is detected by the voter, it alerts the ScTMR controller using an error signal. In the proposed voter, an output selector circuit is used to route the error-free output to the ultimate output signal.

Figure 3. Recovery Process by ScTMR

5. SOFTWARE IMPLEMENTATION

The main top module in this project is "Fault Recoverable ALU System". It consists of two main sub blocks inside it: ALU_TMR Module and ScTMR Module.

Further, the ALU_TMR consists of five sub-modules: Three ALU Systems, One Voter, and One disagreement detector and in code named as ALU_1, ALU_2, ALU_3, Voter, and Disagreement Detector.

Verilog HDL has been used in designing all the sub-blocks, like ALU_TMR and ScTMR of Fault Recoverable ALU system. Structural model has been tried to combine with behavioural mode of Verilog programming.

The FSM implementation of ScTMR as well as the internal sub-modules of the ALU_TMR i.e., ALU_1, ALU_2, ALU_3, Voter and Disagreement detector, are designed as Synthesizable RTL code using Verilog HDL.

5.1. Algorithm for TMR_ALU

Algorithm for the fault tolerant ALU system is as follows:

Step1: Define Module name as ALU_TMR.

Step2: Define Inputs: clock, reset, a, b, command, oe, sce1, sce2, sce3, sci1, sci2, sci3, recover.

Step3: Define Outputs: sco1, sco2, sco3, out, err, x, y, z.

Step4: Instantiate the three ALU's.

Step5: Implementing the functionality of voter.

Step6: Implementation of Disagreement detector.

If output of ALU1 is not equal to the output of ALU 2 And ALU3, then the 'X' Signal goes high and if output of ALU2 is not equal to the output of ALU 1 And ALU3, then the 'Y' signal goes high, and if the output of ALU3 is not equal to the output of ALU 2 And ALU1, then the 'Z' signal goes high.

Step7: Make the "err" signal high, if any one of the three ALU modules are faulty.

Step8: The internal states of faulty ALUs output are sent to the ScTMR for the recovery   using scan chain output signals so that faulty ALU can be recovered by ScTMR.
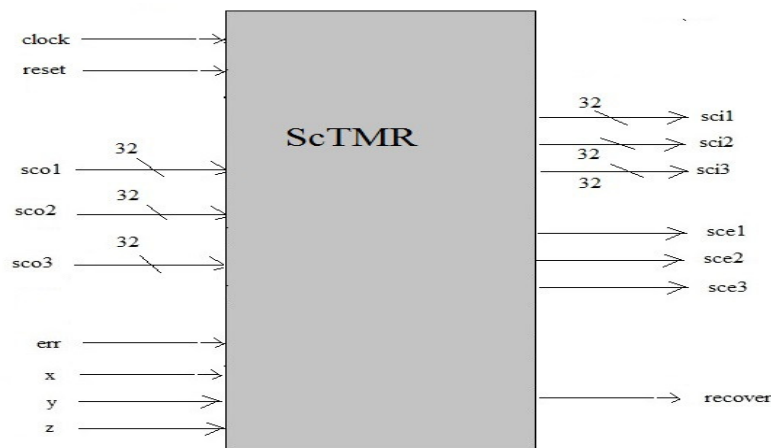
5.2. ScTMR Implementation

Figure 4. Block Diagram of ScTMR module

Algorithm for ScTMR:

Step 1: Module name is ScTMR.

Step 2: Input – clock, reset, sco1, sco2, sco3, err, x, y, z .

Step 3: Output – sce1, sce2, sce3, sci1, sci2, sci3, recover.

Step 4: If err signal is low, the ScTMR operates in the normal state and the output of the "Fault Recoverable ALU_System" is the output of the ALU_TMR.

Step 5: If the err signal is high, then the state changes from the normal state to the recoverable state. Based on the value of the x, y, z signals, and 'scan chain enable signal' of the corresponding faulty ALU goes high, after the recovery, the recovered signal is send to the ALU_TMR.

Step 6: After recovery, ScTMR goes to normal State and continues to work normally.

Step7: If the fault is permanent, ScTMR will enter in to the Master_checker state, it discards the faulty ALU, but still the system continues to work.

Step8: If more than one or all the three ALU's are faulty then, ScTMR enters into unrecoverable state, the system can't be recovered in this condition.
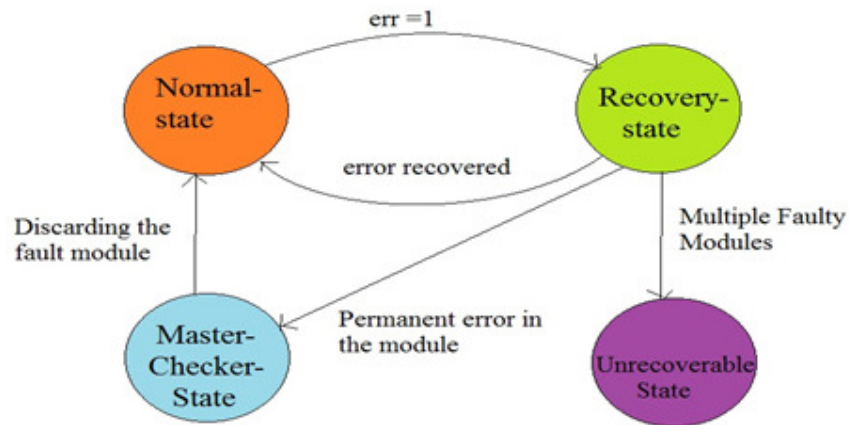
Figure 5. FSM Diagram of ScTMR

5. OUTPUT SIMULATION

5.1 Simulation Results of ALU_TMR

As shown in fig.6 the TMR module able to mask the fault when ALU1was faulty, hence, the overall output will not get affected by fault.
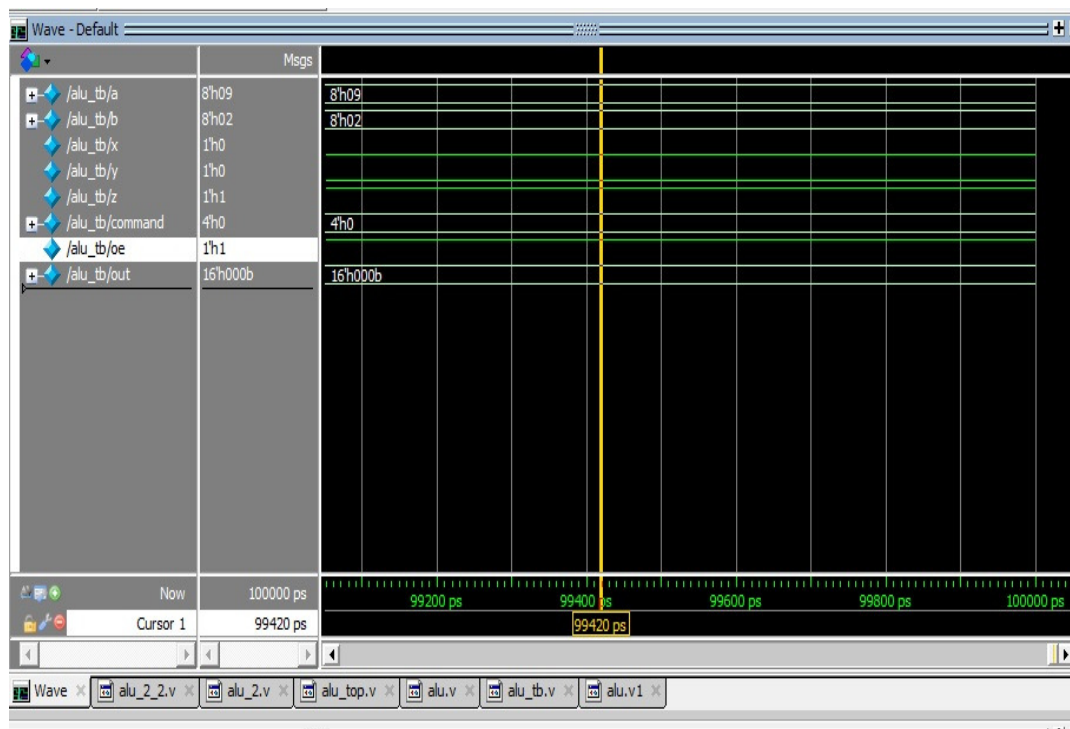

Figure 6. Simulation Waveform of TMR_ALU block

After detecting fault, now disagreement generator will send a signal to ScTMR to start the recovery process.

5.2 Simulation of ScTMR

The figure 7 shows the recovery process by the ScTMR to recover the fault, by connecting scan chains, the output will be unaffected due masking by Voter.
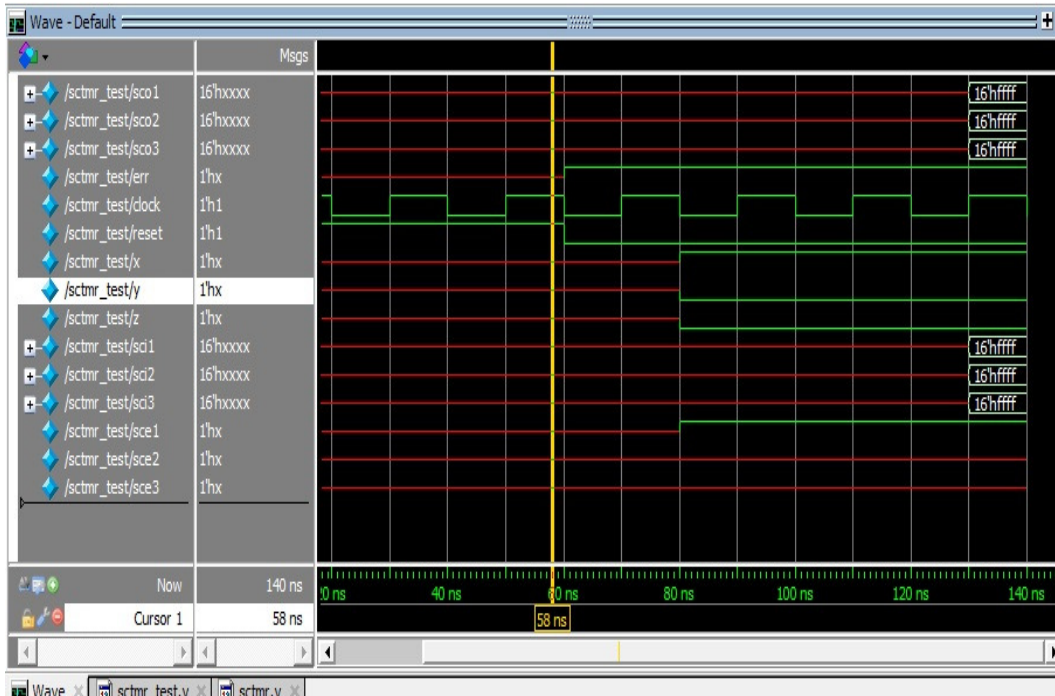

Figure 7. Waveform Recovery Process by ScTMR

5. RESULT AND CONCLUSION

The Final ALU system able to recover from faults induced during testing by testbench. Hence, the proposed deign is successful in bringing fault recovery in the critical ALU system and this is the most efficient and reliable ALU developed till now. Figure 8 shows the console window of the recovery result.
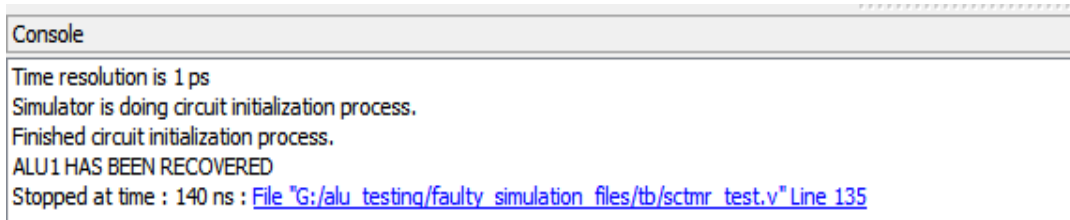

Figure 8. Console window showing Final Result of Recovery in ALU

The following conclusion has been derived from the output and results:

- The designed system is working as a fault tolerant and recoverable ALU.

- The ALU able to recover from multiple transient faults induced in testbench.

- The output of the designed TMR system is still uninterrupted even after getting one of the sub-module faulty, ensuring the high fault tolerant property of the system.

- The penalty of Area and power requirement of the system is just nearly 21-23 per cent, but the reliability has been increased by 9 folds, making this system apt for critical medical and defence application use.

ACKNOWLEDGEMENTS

REFERENCES

[1] Rakshith Saligram, Shrihari Shridhar Hegde, Shashidhar A Kulkarni, H.R.Bhagyalakshmi and M.K. Venkatesha, (2013) "Design Of Parity Preserving Logic Based Fault Tolerant Reversible Arithmetic Logic Unit", International Journal of VLSI Design & Communication Systems (VLSICS), Vol.4, No.3.

[2] Meshram, M. S. S., and Belorkar, M. U. A., (2011) "Design Approach for Fault Tolerance in FPGA Architecture", International Journal of VLSI design & Communication Systems (VLSICS), Vol. 2, No. 1.

[3] Denning, P. J., (1976) "Fault tolerant operating systems", ACM Computing Surveys (CSUR), Vol.8, No.4, pp 359-389.

[4] Bharathi, M., and Neelima, K, (2012) "Scope of reversible engineering at gate-level: fault-tolerant combinational adders", Int J VLSI Des Commun Syst (VLSICS), Vol.3, No.2, pp 85-98.

[5] Kakarla, H., Latha, M. M., and Khan H., (2012) "Self Correcting Memory Design For Fault Free Coding In Progressive Data Streaming Application", International Journal of VLSI Design & Communication Systems, Vol.3, No.1, 25.

[6] Samudrala, P. K., Ramos, J., and Katkoori, S. (2004) "Selective triple modular redundancy (STMR) based single-event upset (SEU) tolerant synthesis for FPGAs", Nuclear Science, IEEE Transactions on, Vol.51, No.5, pp 2957-2969.

[7] Dubey, N., & Akashe, S. (2014) "Implementation Of An Arithmetic Logic Using Area Efficient Carry Look-Ahead Adder", Int J VLSI Des Commun Syst (VLSICS), Vol.5, No.6, pp 29.

[8] Alagoz, B. B., (2008) "Hierarchical Triple-Modular Redundancy (H-TMR) Network for Digital Systems", OncuBilim Algorithm and Systems Labs, Vol. 08, Art. No.5.

[9] Laung Terng Wang, Cheng Wen Wu and Xiaoqing Wen (2008) VLSI Test Principles and architectures: Design for Testability, The Morgan Kaufmann Series in Systems on Silicon.

[10] Laxmi Kumre, Ajay Somkuwar and Ganga Agnihotri, (2013) "Power Efficient Carry Propagate Adder", Int J VLSI Des Commun Syst (VLSICS), Vol.4, No.3.

[11] Harikishore.Kakarla, Madhavi Latha.M, and Habibulla Khan, (2012) "Self Correcting Memory Design For Fault Free Coding In Progressive Data Streaming Application", International Journal of VLSI Design & Communication Systems (VLSICS), Vol.3, No.1.

AUTHORS

Ankit K V is a Graduate Scholar in Dept. of E&C, BIT, Bangalore, India. He received his B.Tech (Hons) ECE in 2012 and now involved in active research in VLSI Mixed Mode Designs, Logic Designing and Electronics Products Develop. etc.

S Murali Narasimham is Associate Professor in Dept. of E&C, BIT, Bangalore, India. He has received Master's degree in Electrical Engineering from Bulgaria. His research interests are Electrical Engineering and IC Fabrication etc.

Dr. Vijaya Prakash is Professor in Dept. of E&C, BIT, Bangalore, India. He has received PhD in VLSI and author of many research papers in renowned journals. His research interests are Low Power VLSI, Image Processing, Synthesis and Optimization of Digital Circuits. He is a member of IMAPS and ISTE