International Journal of Engineering Research and Reviews ISSN 2348-697X (Online)
Vol. 4, Issue 1, pp: (1-8), Month: January - March 2016, Available at: www.researchpublish.com

Implementation of Built in self test Architecture for EDDR

[1]Govindarau.M, [2]Joseph.s, [3]Rajesh kumar.k

Abstract: **BIST schemes generally focus on memory circuit; testing-related issues of video coding have seldom been Addressed this work presents an Error Detection and Data Recovery (EDDR) design, based on the Residue-and Quotient (RQ) code, to embed into ME for video coding testing applications. Additionally, the reliability issue of numerous Processing Elements (PE) in a ME can be improved by enhancing the capabilities of Concurrent Error Detection (CED). Given the critical role of Motion Estimation (ME) in a video coder, testing such a module is of priority concern. While focusing on the testing of ME in a video coding system, BIST schemes generally focus on memory circuit, testing- related issues of video coding have seldom been addressed. Thus, exploring the feasibility of an embedded testing approach to detect errors and recover data of a Motion Estimation (ME) is of worthwhile interest. An error in Processing Elements (PEs), i.e. key components of a ME, can be detected and recovered effectively by using the proposed EDDR design. Experimental results indicate that the proposed EDDR design for ME testing can detect errors and recover data with an acceptable area overhead and timing penalty. Importantly, the proposed EDDR design performs satisfactorily in terms of throughput and reliability for ME testing applications.**

Keywords: **Motion estimation, processing element, residue-and quotient, error detection circuit, test code generation, circuit under test, error detection and data recovery.**

I. INTRODUCTION

Advantages in semiconductors, digital signal processing, and communication technologies have made multimedia applications more flexible and reliable. A good example is the H.264 video standard, also known as MPEG-4 Part 10 Advanced Video Coding, which is widely regarded as the next generation video compression standard [1], [2]. Video compression is necessary in a wide range of applications to reduce the total data amount required for transmitting or storing video data. Among the coding systems, a ME is of priority concern in exploiting the temporal redundancy between successive frames, yet also the most time consuming aspect of coding. Additionally, while performing up to 60%–90% of the computations encountered in the entire coding system, a ME is widely regarded as the most computationally intensive of video coding system [3].Thus, general schemes of BIST referred to as built-in self-analysis and built-in self-correction have been developed recently. While the extended BIST schemes generally focus on memory circuit, testing-related issues of video coding have seldom been addressed. Thus, exploring the feasibility of an embedded testing approach to detect errors and recuperate data of a ME is of worthwhile interest. The reliability issue of frequent PEs in a ME can be improved by ornamental the capabilities of Concurrent error detection (CED). The CED approach can sense errors through conflicting and undesired results generate from operation on the same operands. CED can also test the circuit at full operating rapidity without interrupt a system[3].

Moreover, BIST can produce test simulations and analyze test responses without outside support, consequently reformation the testing and diagnosis of digital systems. However, increasingly intricate density of circuitry requires that the built-in testing approach not only detect faults but also specify their location for error correcting.

A ME generally consists of PEs with a size of 4×4. However, accelerating the computation speed depends on a large PE array, especially in high-resolution devices with a large search range such as HDTV [4]. Additionally, the visual quality and peak signal-to-noise ratio (PSNR) at a given bit rate are influenced

if an error occurred in ME process. A testable design is thus in-caressingly important to ensure the reliability of numerous PEs in a ME. Moreover, although the advance of VLSI technologies facilitate the integration of a large number of PEs of a ME into a chip, the logic-per-pin ratio is subsequently increased, thus de-creasing significantly the efficiency of logic testing on the chip. As a commercial chip, it is absolutely necessary for the ME to introduce design for testability (DFT) [5]–[7]. DFT focuses on increasing the ease of device testing, thus guaranteeing high re-liability of a system. DFT methods rely on reconfiguration of a circuit under test (CUT) to improve testability. While DFT approaches enhance the testability of circuits, advances in sub-micron technology and resulting increases in the complexity of electronic circuits and systems have meant that built-in self-test (BIST) schemes have rapidly become necessary in the digital world. BIST for the ME does not expensive test equipment, ultimately lowering test costs [8]–[10]. Moreover, BIST can generate test simulations and analyze test responses without outside support, subsequently streamlining the testing and diagnosis of digital systems. However, increasingly complex density of circuitry requires that the built-in testing approach not only detect faults but also specify their locations for error correcting. Thus, extended schemes of BIST referred to as built-in self-diagnosis] and built-in self-correction have been developed recently.

While the extended BIST schemes generally focus on memory circuit, testing-related issues of video coding have seldom been addressed. Thus, exploring the feasibility of an embedded testing approach to detect errors and recover data of a ME is of worthwhile interest. Additionally, the reliability issue of numerous PEs in a ME can be improved by enhancing the capabilities of concurrent error detection (CED).

The CED approach can detect errors through conflicting and undesired results generated from operations on the same operands. CED can also test the circuit at full operating speed without interrupting a system. Thus, based on the CED concept, this work develops a novel EDDR architecture based on the RQ code to detect errors and recovery data in PEs of a ME and, in doing so, further guarantee the excellent reliability for video coding testing applications. The rest of this paper is organized as follows. Section II describes the mathematical model of RQ code and the corresponding circuit design of the RQ code generator (RQCG). Section III then introduces the proposed EDDR architecture, fault model definition, and test method. Next, Section IV evaluates the performance in area overhead, timing penalty, and throughput and reliability analysis to demonstrate the feasibility of the proposed EDDR architecture for ME testing applications. Conclusions are finally drawn in Section V.

II. RQ CODE GENERATION

Coding approaches such as parity code, Berger code, and residue code have been considered for design applications to detect circuit errors. Residue code is generally separable arithmetic codes by estimating a residue for data and appending it to data. Error detection logic for operations is typically derived by a separate residue code, making the detection logic is simple and easily implemented. For instance, assume that N denotes an integer, N1 and N2 represent data words, and m refers to the modulus.[4] A separate residue code of interest is one in which is coded as a pair N|NN|m) . Notably, |N|m is the residue of N modulo m . Error detection logic for operations is typically derived using a separate residue code such that detection logic is simply and easily implemented. However, only a bit error can be detected based on the residue code. Additionally, an error cannot be recovered effectively by using the residue codes. There-fore, this work presents a quotient code, which is derived from the residue code, to assist the residue code in detecting multiple errors and recovering errors[5]. The mathematical model of RQ code is simply described as follows.

According to the above RQ code expression, the corresponding circuit design of the RQCG can be realized. In order to simplify the complexity of circuit design, the implementation of the module is generally dependent on the addition operation. Additionally, based on the concept of residue code, the following definitions shown can be applied to generate the RQ code for circuit design.

III. PROPOSED EDDR ARCHITECTURE DESIGN

Fig. 1 shows the conceptual view of the proposed EDDR scheme, which comprises two major circuit designs, i.e. error detection circuit (EDC) and data recovery circuit (DRC), to detect errors and recover the corresponding data in a specific CUT. The test code generator (TCG) in Fig. 1 utilizes the concepts of RQ code to generate the corresponding test codes for error detection and data recovery. In other words, the test codes from TCG and the primary output from CUT are delivered to EDC to determine whether the CUT has errors. DRC is in charge of re-covering data from TCG. Additionally, a selector is enabled to export error-free data or data-recovery results.

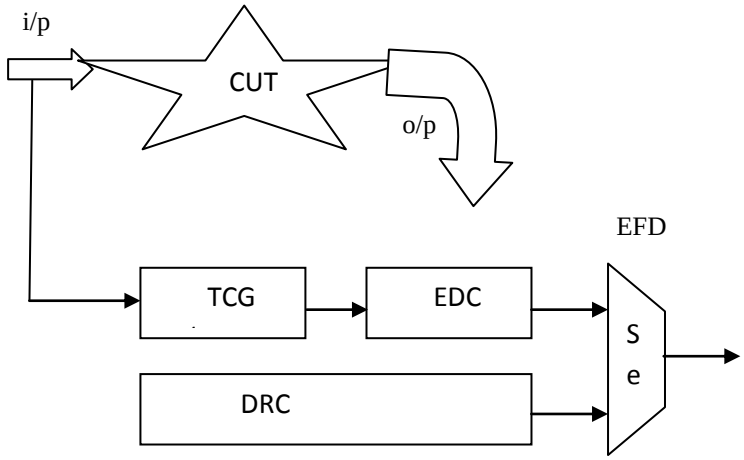


Fig.1: Conceptual view of the proposed EDDR architecture

Importantly, an array-based computing structure, such as ME, discrete cosine transform (DCT), iterative logic array (ILA), and finite impulse filter (FIR), is feasible for the proposed EDDR scheme to detect errors and recover the corresponding data.

This work adopts the systolic ME as a CUT to demonstrate the feasibility of the proposed EDDR architecture. A ME consists of many PEs incorporated in a 1-D or 2-D array for video encoding applications. A PE generally consists of two ADDs (i.e. an 8-b ADD and a 12-b ADD) and an accumulator (ACC). Next, the 8-b ADD (a pixel has 8-b data) is used to estimate the addition of the current pixel (Cur pixel) and reference pixel (Ref pixel). Additionally, a 12-b ADD and an ACC are required to accumulate the results from the 8-b ADD in order to determine the sum of absolute difference (SAD) value for video encoding applications [10]. Notably, some registers and latches may exist in ME to complete the data shift and storage. Fig. 2 shows an example of the proposed EDDR circuit design for a specific P Ei of a ME. The fault model definition, RQCG-based TCG design, operations of error detection and data recovery, and the overall test strategy are described carefully as follows.

IV. EVALUATE THE PERFORMANCE AREA OVER HEAD

A. Sum of Absolute Difference Calculation:

By utilizing PEs, SAD shown in as follows, in a macro block with size N X N of can be evaluated:

$$SAD = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (xij - yij)$$

$$SAD = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (qxij.m + rxij) - (qyij.m + ryij)$$

Utilizing the concept of the proposed SAD Tree Architecture. The proposed SAD Tree is a 2-D intra-level architecture and consists of a 2-D PE array and one 2-D adder tree with propagation registers Current pixels are stored in each PE, and reference pixels are stored in propagation registers for data reuse. In each cycle, current and reference pixels are inputted to PEs. Simultaneously, continuous reference pixels in a row are inputted into propagation registers to update reference pixels. In propagation registers, reference pixels are propagated in the vertical direction row by row. In SAD Tree architecture, all distortions of a searching candidate are generated in the same cycle, and by an adder tree, distortions are accumulated to derive the SAD in one cycle.

B. Fault Model:

The PEs are essential building blocks and are connected regularly to construct a ME. Generally, PEs are surrounded by sets of ADDs and accumulators that determine how data flows through them. PEs can thus be considered the class of circuits called ILAs, whose testing assignment can be easily achieved by using the fault model, cell fault model (CFM).

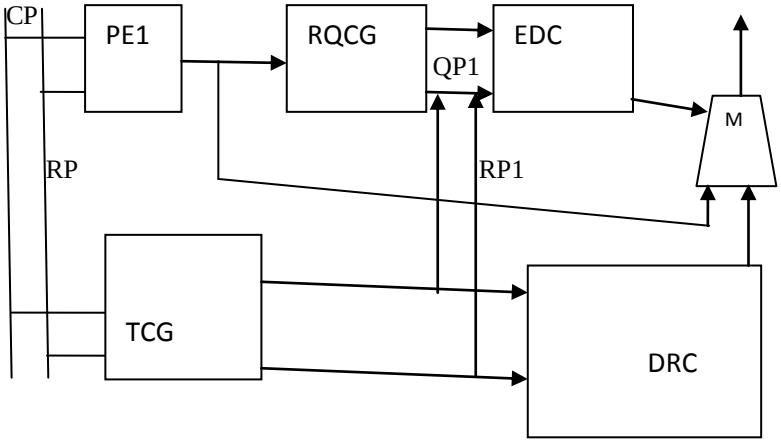


Fig.2: A specific PEi testing process of the proposed EDDR architecture

Using CFM has received considerable interest due to accelerated growth in the use of high-level synthesis, as well as the parallel increase in complexity and density of integration circuits (ICs). Using CFM makes the tests independent of the adopted synthesis tool and vendor library. Arithmetic modules, like ADDs (the primary element in a PE), due to their regularity, are designed in an extremely dense configuration.

Moreover, a more comprehensive fault model, i.e. the stuck-at (SA) model, must be adopted to cover actual failures in the interconnect data bus between PEs. The SA fault is a well-known structural fault model, which assumes that faults cause a line in the circuit to behave as if it were permanently at logic "0" (stuck-at 0 (SA0)) or logic "1" [stuck-at 1 (SA1)]. The SA fault in a ME architecture can incur errors in computing SAD values[7]. A distorted computational error (e) and the magnitude of e are assumed here to be equal to SAD, ⁄SAD, where SAD denotes the computed SAD value with SA faults.

C. TCG Design:

According to Fig. 2, TCG is an important component of the proposed EDDR architecture. Notably, TCG design is based on the ability of the RQCG circuit to generate corresponding test codes in order to detect errors and recover data. The specific PE1 in Fig. 2 estimates the absolute difference between the Cur pixel of the search area and the Refpixel of the current macro block. Thus, by utilizing PEs, SAD shown in as follows, in a macro block with size of NxN can be evaluated:

Where r xij and rij qij denote the corresponding RQ code of Xij and Yij modulo m. importantly, Xij and Yij rep-resent the luminance pixel value of Cur pixel (CP) and Refpixel (RP), respectively.

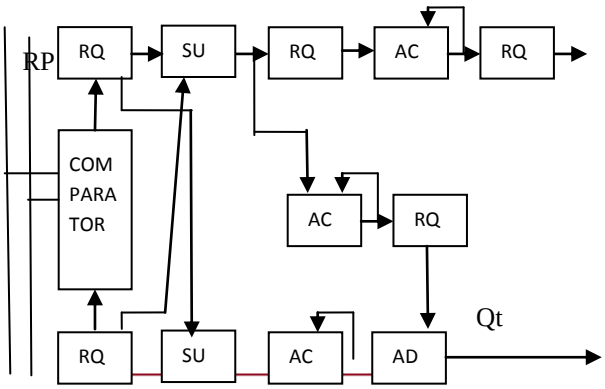


Fig.3: Circuit design of the TCG

. Based on the residue code, the definitions shown in (2) and (3) can be applied to facilitate generation of the RQ code (Rt and Qt) form TCG. Namely, the circuit design of TCG can be easily achieved

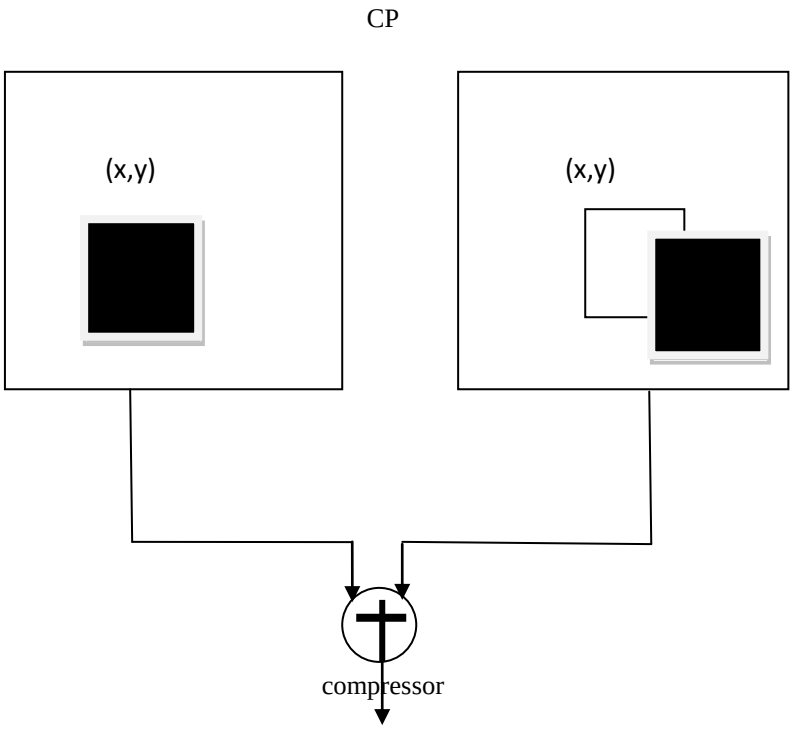


Fig.4: motion estimator

D. EDDR Processes:

Error detection in a specific PE is achieved by using EDC, which is utilized to compare the outputs between TCG and RQCG in order to determine whether errors have occurred. The EDC output is then used to generate a 0/1 signal to indicate that the tested is error free errancy. This work presents a mathematical statement to verify the operations of error detection.

Based on the definition of the fault model, the SAD value is influenced if either SA1 and/or SA0 errors have occurred in a specific PE. In other words, the SAD value is transformed to SAD'=SAD + e, if an error occurred. Notably, the error signal is expressed as

Additionally, DRC is used to recover data encoded by TCGi, i.e. the appropriate RTi and QTi codes from TCGi are selected by MUXs 2 and 3, respectively, to recover data. Fourth, the error-free data or data recovery results are selected by MUX. Notably, control signal S4 is generated from EDC, indicating that the comparison result is error-free (S4 = 0) or errancy (S4 = 1)[8]. Finally, the error-free data or the data- recovery result from the tested object PEi is passed to a De-MUX, which is used to test the next specific PEi+1; otherwise, the final result is exported

E. Overall Test Strategy:

By extending the testing processes of a specific PEi in Fig. 2, Fig. 6 illustrates the overall EDDR architecture design of a ME. First, the input data of Cur pixel and Refpixel are sent simultaneously to PEs and TCGs in order to estimate the SAD values and generate the test RQ code Rt and Qt. Second, the SAD value from the tested object PEi, which is selected by MUX1, is then sent to the RQCG circuit in order to generate RPE and QPE codes. Meanwhile, the corresponding test codes Rt and Qt from a specificTCG1 are selected simultaneously by MUXs 2 and 3, respectively. Third, the RQ code from TCG1 and RQCG circuits are compared in EDC to determine whether the tested object PEi have errors. The tested object PEi is error-free if and only if $Rp_E = Rt$ and $Qp_E = Qt$. Ad-ditionally[10], DRC is used to recover data encoded TCGi, i.e. the appropriate Rt and Qt codes from TCGi are selected by MUXs 2 and 3,

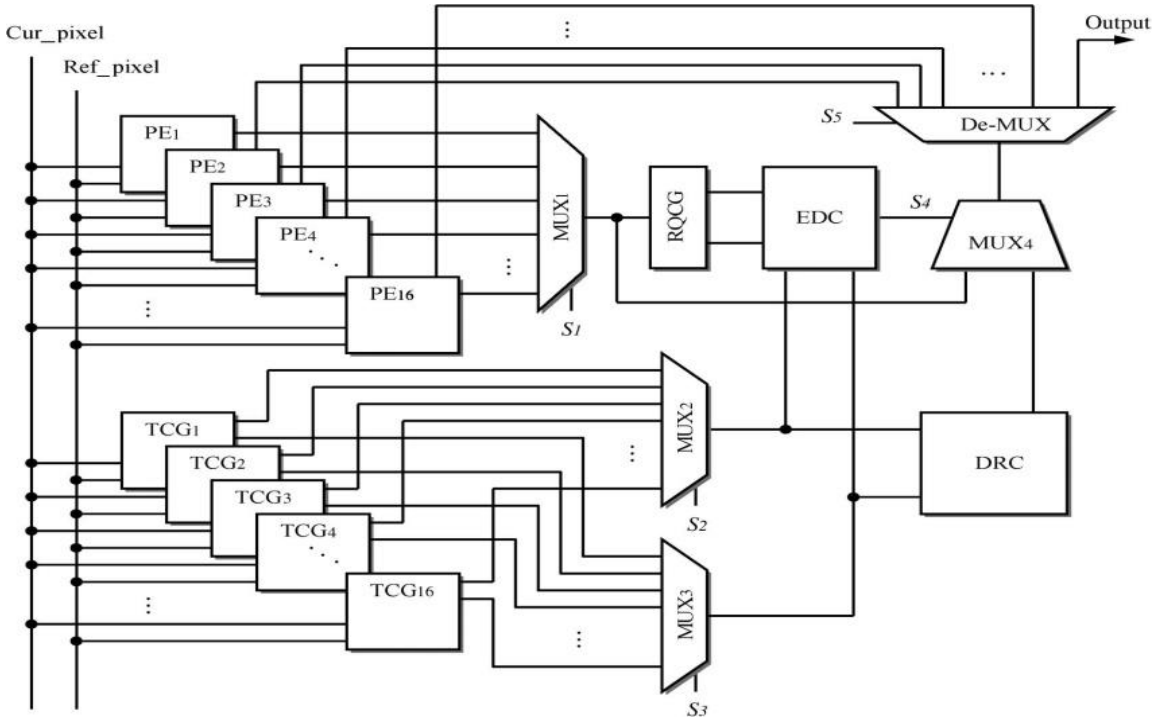
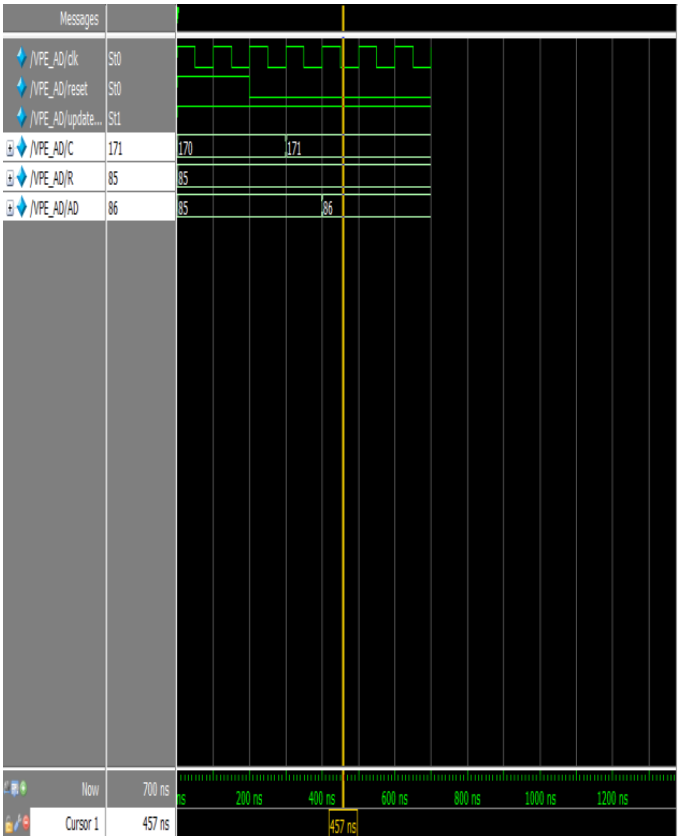
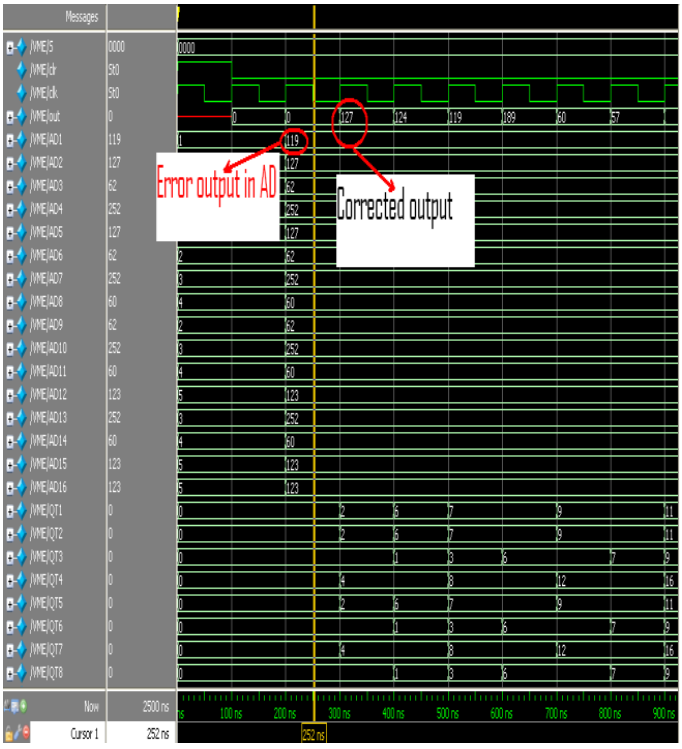


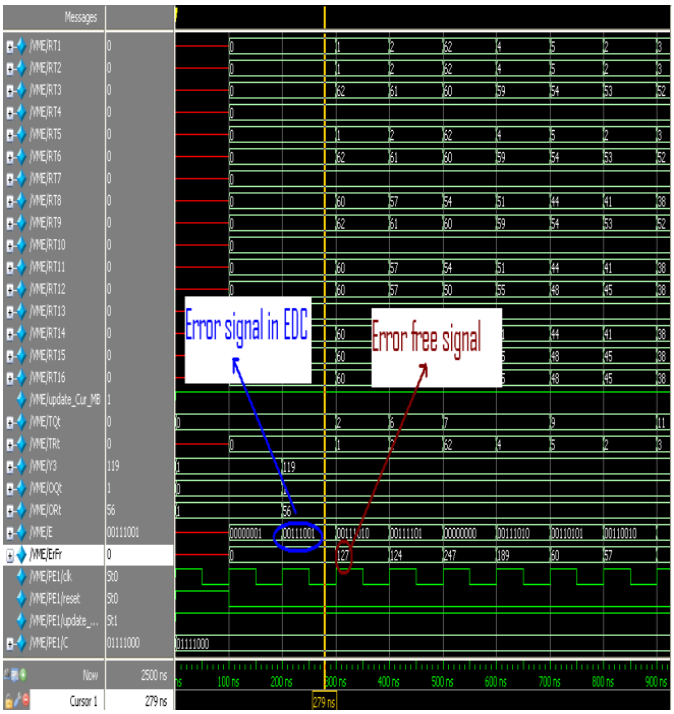
Fig.5: Proposed EDDR architecture design for a ME.

respectively, to recover data. Fourth, the error-free data or data recovery results are selected by MUX4. Notably, control signal S4 is generated from EDC, indicating.

V. RESULTS







VI. CONCLUSION

This work presents EDDR architecture for detecting the errors and recovering the data of PEs in a ME. Based on the Residue Quotient code, a RQCG-based Test Code Generation design is developed to generate the corresponding test codes to detect errors and recover data. The EDDR architecture is designed and tested using exhaustive test bench using Modelsim simulator and implemented by using Verilog on FPGA. Experimental results indicate that that the proposed EDDR architecture can effectively detect errors and recover data in PEs of a ME with with 3 clock cycles for simulation and area of 17% LUTs,4% Flip-flops and 12% of GCLKs. Throughput is demonstrate the satisfactory performance of the proposed EDDR architecture design for ME testing applications.

REFERENCE

[1] *Advanced Video Coding for Generic Audiovisual Services*, ISO/IEC 14496-10:2005 (E), Mar. 2005, ITU-T Rec. H.264 (E).

[2] *Information Technology-Coding of Audio-Visual Objects—Part 2: Vi-sual*, ISO/IEC 14 496-2, 1999.

[3] Y. W. Huang, B. Y. Hsieh, S. Y. Chien, S. Y. Ma, and L. G. Chen, "Analysis and complexity reduction of multiple reference frames motion estimation in H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 4, pp. 507–522, Apr. 2006.

[4] C. Y. Chen, S. Y. Chien, Y. W. Huang, T. C. Chen, T. C. Wang, and L. G. Chen, "Analysis and architecture design of variable block-size motion estimation for H.264/AVC," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 53, no. 3, pp. 578–593, Mar. 2006.

[5] T. H. Wu, Y. L. Tsai, and S. J. Chang, "An efficient design-for-testa-bility scheme for motion estimation in H.264/AVC," in *Proc. Int. Symp. VLSI Design, Autom. Test*, Apr. 2007, pp. 1–4.

[6] M. Y. Dong, S. H. Yang, and S. K. Lu, "Design-for-testability tech-niques for motion estimation computing arrays," in *Proc. Int. Conf. Commun., Circuits Syst.*, May 2008, pp. 1188–1191.

[7] Y. S. Huang, C. J. Yang, and C. L. Hsu, "C-testable motion estimation design for video coding systems," *J. Electron. Sci. Technol.*, vol. 7, no. 4, pp. 370–374, Dec. 2009.

[8] D. Li, M. Hu, and O. A. Mohamed, "Built-in self-test design of motion estimation computing array," in *Proc. IEEE Northeast Workshop Cir-cuits Syst.*, Jun. 2004, pp. 349–352.

[9] Y. S. Huang, C. K. Chen, and C. L. Hsu, "Efficient built-in self-test for video coding cores: A case study on motion estimation computing array," in *Proc. IEEE Asia Pacific Conf. Circuit Syst.*, Dec. 2008, pp. 1751–1754.

[10] W. Y Liu, J. Y. Huang, J. H. Hong, and S. K. Lu, "Testable design and BIST techniques for systolic motion estimators in the transform domain," in *Proc. IEEE Int. Conf. Circuits Syst.*, Apr. 2009, pp. 1–4.