

October 2012

DESIGN FOR TESTABILITY TECHNIQUES FOR VIDEO CODING SYSTEMS

PARSHA SRIKANTH

KITS, Warangal, India, Parshasrikanth5@gmail.com

SD.RAZIYA SULTHANA

KITS, Warangal, India, razia14@gmail.com

Follow this and additional works at: <https://www.interscience.in/ijipvs>



Part of the [Robotics Commons](#), [Signal Processing Commons](#), and the [Systems and Communications Commons](#)

Recommended Citation

SRIKANTH, PARSHA and SULTHANA, SD.RAZIYA (2012) "DESIGN FOR TESTABILITY TECHNIQUES FOR VIDEO CODING SYSTEMS," *International Journal of Image Processing and Vision Science*: Vol. 1 : Iss. 4 , Article 13.

Available at: <https://www.interscience.in/ijipvs/vol1/iss4/13>

This Article is brought to you for free and open access by Interscience Research Network. It has been accepted for inclusion in International Journal of Image Processing and Vision Science by an authorized editor of Interscience Research Network. For more information, please contact sritampatnaik@gmail.com.

DESIGN FOR TESTABILITY TECHNIQUES FOR VIDEO CODING SYSTEMS

PARSHA SRIKANTHI, SD.RAZIYA SULTHANA2

KITS, Warangal, India
E-mail: Parshasrikanth5@gmail.com, razia14@gmail.com

Abstract- Motion estimation algorithms are used in various video coding systems. While focusing on the testing of ME in a video coding system, 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. 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. Therefore, paper describes a novel testing scheme of motion estimation. The key part of this scheme is to offer high reliability for motion estimation architecture. The experimental result shows the design achieve 100% fault coverage. And, the main advantages of this scheme are minimal performance degradation, small cost of hardware overhead and the benefit of a-speed testing.

Index Terms- Area overhead, data recovery, error detection, motion estimation, reliability, residue-and-quotient (RQ) code.

I. INTRODUCTION

The advent of VLSI technology, a large collection of processing elements can be assembled to achieve high-speed computation economically. Rather, the problem of testing a VLSI chip begins with introduction of a defect during the design or implementation phases[9]. 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 a video coding system [3].

✘

A ME generally consists of PEs with a size of 4x4. 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 increasingly 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 decreasing 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 reliability 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 analyse 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 [11] and built-in self-correction [12]–[14] 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) [15], [16]. 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 BIST architecture, Next, Section IV Results and Discussions. Conclusions are finally drawn in Section V.

II.METHODOLOGIES

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 in N which is coded as a pair (N,|N|m). Notably,|N|m is the residue of N modulus m[1]. 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. Therefore, 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. 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. the corresponding circuit design of the RQCG is easily realized by using the simple adders (ADDs). Namely, the RQ code can be generated with a low complexity and little hardware cost. The mathematical model of RQ code is simply described as follows[1]. Assume that binary data X is expressed as

$$X = \{b_{n-1}b_{n-2} \dots b_2b_1b_0\} = \sum_{j=0}^{n-1} b_j2^j.$$

The RQ code of X modulo m expressed as R=|X|m, Q=[X/m], respectively. Notably [i] denotes the largest integer not exceeding i.

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.

Definition 1:

$$|N1+N2|m = ||N1|m + |N2|m|m. \tag{2}$$

Definition 2: Let Nj = n1+n2+...+nj, then

$$|Nj|m = ||n1|m + |n2|m \dots + |nj|m|m. \tag{3}$$

To accelerate the circuit design of RQCG, the binary data shown in (1) can generally be divided into two parts:

$$\begin{aligned} X &= \sum_{j=0}^{n-1} b_j2^j \\ &= \left(\sum_{j=0}^{k-1} b_j2^j \right) + \left(\sum_{j=k}^{n-1} b_j2^{j-k} \right) 2^k \\ &= Y_0 + Y_12^k. \end{aligned} \tag{4}$$

Significantly, the value of k is equal to[n/2] and the data formation of Y0 and Y1 are a decimal system. If the modulus m = 2k - 1, then the residue code of modulo is given by

$$\begin{aligned} R &= |X|m \\ &= |Y_0+Y_1|m = |Z_0 + Z_1|m = (Z_0+Z_1)\alpha \end{aligned} \tag{5}$$

$$\begin{aligned} Q &= \left\lfloor \frac{X}{m} \right\rfloor \\ &= \left\lfloor \frac{Y_0 + Y_1}{m} \right\rfloor + Y_1 = \left\lfloor \frac{Z_0 + Z_1}{m} \right\rfloor + Z_1 + Y_1 \\ &= Z_1 + Y_1 + \beta \end{aligned} \tag{6}$$

where

$$\alpha(\beta) = \begin{cases} 0(1), & \text{if } Z_0 + Z_1 = m \\ 1(0), & \text{if } Z_0 + Z_1 < m. \end{cases}$$

Notably, since the value of Y0 + Y1 is generally greater than that of modulus m, the equations in (5) and (6) must be simplified further to replace the complex module operation with a simple addition operation by using the parameters Z0,Z1, α and β .Based on (5) and (6), the corresponding circuit design of the RQCG is easily realized by using the simple adders (ADDs). Namely, the RQ code can be generated with a low complexity and little hardware cost.

A.PROPOSED SYSTEM

The conceptual view of the proposed BIST Architecture, 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. utilizes the concepts of RQ code to generate the corresponding test codes for error detection and data recovery.

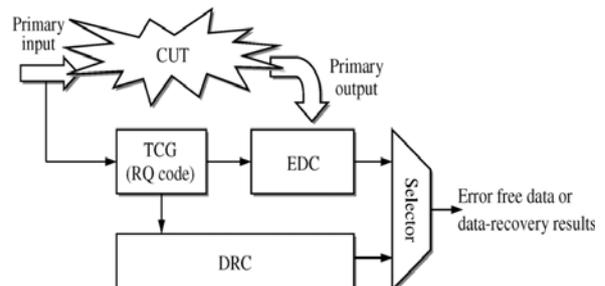


Fig. 1. Conceptual view of the proposed BIST architecture.

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 recovering data from TCG. Additionally, a selector is enabled to export error-free data or data-recovery results. 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 BIST architecture to detect errors and recover the corresponding data.

III. PROPOSED ARCHITECTURE

Fig. 1 shows the conceptual view of the proposed BIST 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 recovering data from TCG. Additionally, a selector is enabled to export error-free data or data-recovery results. 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.

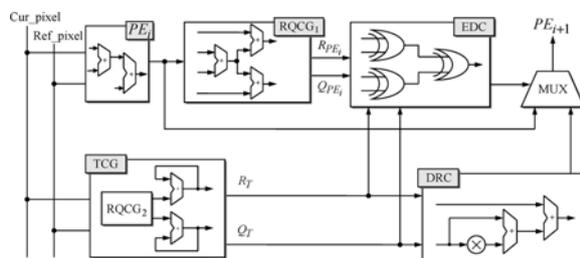


Fig. 2. A specific PE testing processes of the proposed BIST architecture.

A. CIRCUIT UNDER TEST (PROCESSING ELEMENT)

A ME (Motion Estimation) 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. Notably, some registers and latches may exist in ME to complete the data shift and storage. encoding applications. Notably, some registers and

latches may exist in ME to complete the data shift and storage. 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). 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. 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. 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 increasingly 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 decreasing 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). Motion estimation is the process of determining motion vectors that describe the transformation from one 2D image to another; usually from adjacent frames in a video sequence. It is an ill-posed problem as the motion is in three dimensions but the images are a projection of the 3D scene onto a 2D plane. The motion vectors may relate to the whole image (global motion estimation) or specific parts, such as rectangular blocks, arbitrary shaped patches or even per pixel. The motion vectors may be represented by a translational model or many other models that can approximate the motion of a real video camera, such as rotation and translation in all three dimensions and zoom. Closely related to motion estimation is optical flow, where the vectors correspond to the perceived movement of pixels. In motion estimation an exact 1:1 correspondence of pixel positions is not a requirement. Applying the motion vectors to an image to synthesize the transformation to the next image is called motion compensation. The combination of motion estimation and motion compensation is a key part of video compression as used by MPEG 1, 2 and 4 as well as many other video codecs.

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) [21]. 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.

B.TEST CODE GENERATION(TCG)

TCG design is based on the ability of the RQCG circuit to generate corresponding test codes in order to detect errors and recover data. According to Fig. 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 PE_i in Fig estimates the absolute difference between the Cur pixel of the search area and the Ref_pixel of the current macro block. It consists of 5 RQCG block and comparator, single accumulator block and subtractor. 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 in Fig. 2 estimates the absolute difference between the Cur pixel of the search area and the Ref_pixel of the current macro block Thus, by utilizing PEs, SAD shown in as follows, in a macro block with size of N X N can be evaluated:

$$SAD = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |X_{ij} - Y_{ij}|$$

$$= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |(q_{xij} \cdot m + r_{xij}) - (q_{yij} \cdot m + r_{yij})| \quad (7)$$

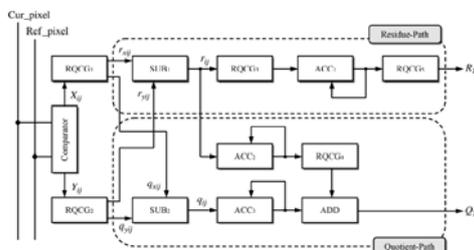


Fig. 3. Circuit design of the TCG.

where rx_{ij}, qx_{ij} and ry_{ij}, qy_{ij} denote the corresponding RQ code of X_{ij} and Y_{ij} modulo m . Importantly, X_{ij} and Y_{ij} represent the luminance pixel value of Cur_pixel and Ref_pixel, respectively. 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[1]. Namely, the circuit design of TCG can be easily achieved (see Fig. 3) by using

$$R_T = \left| \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (X_{ij} - Y_{ij}) \right|_m$$

$$= |(X_{00} - Y_{00})|_m + |(X_{01} - Y_{01})|_m + \dots$$

$$+ |(X_{(N-1)(N-1)} - Y_{(N-1)(N-1)})|_m$$

$$= |(q_{x00} \cdot m + r_{x00}) - (q_{y00} \cdot m + r_{y00})|_m$$

$$+ \dots + |(q_{x(N-1)(N-1)} \cdot m + r_{x(N-1)(N-1)})$$

$$- (q_{y(N-1)(N-1)} \cdot m + r_{y(N-1)(N-1)})|_m$$

$$= |(r_{x00} - r_{y00})|_m + |(r_{x01} - r_{y01})|_m + \dots$$

$$+ |(r_{x(N-1)(N-1)} - r_{y(N-1)(N-1)})|_m$$

$$= |r_{00}|_m + |r_{01}|_m + \dots + |r_{(N-1)(N-1)}|_m \quad (8)$$

and (9), to derive the corresponding RQ code.

$$Q_T = \left\lfloor \frac{\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (X_{ij} - Y_{ij})}{m} \right\rfloor$$

$$= \left\lfloor \frac{(X_{00} - Y_{00}) + (X_{01} - Y_{01}) + \dots + (X_{(N-1)(N-1)} - Y_{(N-1)(N-1)})}{m} \right\rfloor$$

$$= \left\lfloor \frac{(q_{x00} \cdot m - q_{y00} \cdot m) + (r_{x00} - r_{y00}) + (q_{x01} \cdot m - q_{y01} \cdot m) + (r_{x01} - r_{y01}) + \dots}{m} \right\rfloor$$

$$= \left\lfloor \frac{(q_{x00} - q_{y00}) + (q_{x01} - q_{y01}) + \dots + (r_{x00} - r_{y00}) + (r_{x01} - r_{y01}) + \dots}{m} \right\rfloor$$

$$= q_{00} + q_{01} + \dots + q_{(N-1)(N-1)} + \left\lfloor \frac{(r_{00} + r_{01} + \dots + r_{(N-1)(N-1)})}{m} \right\rfloor \quad (9)$$

C.ERROR DETECTION CIRCUIT(EDC)

In this module indicates that the operations of error detection in a specific PE_i is achieved by using EDC, which is utilized to compare the outputs between TCG and 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 PE_i is error-free/errancy. Using XOR operation can be identify the error if any variation in terms of residue and quotient value. Because a fault only affects the logic in the fan out cone from the fault site, the good circuit and faulty circuits typically only differ in a small region. Concurrent fault simulation exploits this fact and simulates only the differential parts of the whole circuit . Concurrent fault simulation is essentially an event-driven simulation with the fault-free circuit and faulty circuits simulated altogether. In concurrent fault simulation, every gate has a concurrent fault list, which consists of a set of bad gates. A bad gate of gate x represents an imaginary copy of gate x in the presence of a fault. Every bad gate contains a fault index and the associated gate I/O values in the presence of the corresponding fault. Initially, the concurrent fault list of gate x contains local faults of gate x. The local faults of gate x are faults on the inputs or outputs of gate x. As the simulation proceeds, the concurrent fault list contains not only local faults but also faults propagated from previous stages. Local faults of gate x remain in the concurrent fault list of gate x until they are detected. As we move to the nanometre age, we have begun to see

nanometre designs that contain hundreds of millions of transistors. We anticipate that the semiconductor industry will completely adopt the scan methodology for quality considerations. As a result, it is becoming imperative that advanced techniques for both logic simulation and fault simulation be developed to address the high-performance and high-capacity issues, in particular, for addressing new fault models, such as transition faults, path-delay faults, and bridging faults. At the same time, more innovations are needed in developing advanced concurrent fault simulation techniques, as designs today that are based on the scan methodology are still not 100% testable. Fault simulation using functional patterns remains important in order to meet excellent quality and parts per-million defect level goals. Test generation is the task of producing an effective set of vectors that will achieve high fault coverage for a specified fault model. While much progress has been made over the years in automatic test pattern generation (ATPG), this problem remains an extremely difficult one. Without powerful ATPGs, chips will increasingly depend on design for testability (DFT) techniques to alleviate the high cost of generating vectors. This chapter deals with the fundamental issues behind the design of an ATPG, as well as the underlying learning mechanisms that can improve the overall performance of ATPG.

D. DATA RECOVERY CIRCUIT

In this module will be generate error free output by quotient multiply with constant value (64) and add with remainder code. During data recovery, the circuit DRC plays a significant role in recovering RQ code from TCG. Notably, the proposed EDDR design executes the error detection and data recovery operations simultaneously. Additionally, error-free data from the tested PE_i or data recovery that results from DRC is selected by a multiplexer (MUX) to pass to the next specific PE_{i+1} for subsequent testing. Error concealment in video is intended to recover the loss due to channel noise, e.g., bit-errors in a noisy channel and cell-loss in an ATM network, by utilizing available picture information. The error concealment techniques can be categorized into two classes according to the roles that the encoder and the decoder play in the underlying approaches. Forward error concealment includes methods that add redundancy in the source to enhance error resilience of the coded bit streams. For example, I-picture motion vectors were introduced in MPEG-4 to improve the error concealment. However, a syntax change is required in this scheme. In contrast to this approach, error concealment by post-processing refers to operations at the decoder to recover the damaged images based on image and video characteristics. In this way, no syntax is needed to support the recovery of missing data. we have only discussed the case in which one frame has been damaged and we wish to recover damaged blocks

using information that is already contained in the bit-stream. The temporal domain techniques that we have considered rely on information in the previous frame to perform the reconstruction. However, if the previous frame is heavily damaged, the prediction of the next frame may also be affected. For this reason, we must consider making the prediction before the errors have occurred. Obviously, if one frame has been heavily damaged, but the frame before that has not been damaged, it makes sense to investigate how the motion vectors can be extrapolated to obtain a reasonable prediction from a past reference frame. Following this notion, we have essentially divided the problem of error concealment into two parts. The first part assumes that the previous frames are intact or are close to intact. This will always be the case for low BER and short error bursts. Furthermore, a localized solution such as the techniques presented in the previous subsection will usually perform well. However, if the BER is high and/or the burst length is long, the impact of a damaged frame can propagate, hence the problem is more global and seems to require a more advanced solution, i.e., one which considers the impact over multiple frames. In the following, we propose an approach that considers making predictions from a past reference frame, which has not been damaged. The estimated motion information which differs from the actual one may be recovered from that of neighbor blocks. Because a moving object in an image sequence is larger than the block size of a minimal block in many occasions, motion information of neighbor blocks are usually the same as, or approximate to, current blocks. The concept of global motion is discussed in many researches on motion estimation or related interests. In method which reconstructs the frame with the aid of neighbor motion vector is successfully applied to motion estimation. Thus, an error signal "1" is generated from EDC and sent to mux in order to select the recovery results from DRC.

Table I
ESTIMATION OF AREA OVERHEAD AND
TIMING PENALTY

Components	PE	RQCG	EDC	TCG	DRC
Area (Gate counts)	69482	1779	667	3265	2376
Operation Time (ns)	973.76	10.17	6.02	1016.56	17.99
Area Overhead (%)	5.13				
Time Penalty (%)	6.24				

IV. RESULTS AND DISCUSSION

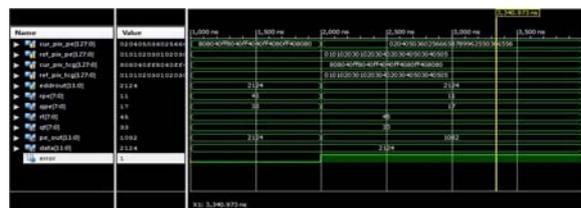


Fig. 4. Simulation Results of Proposed Architecture.

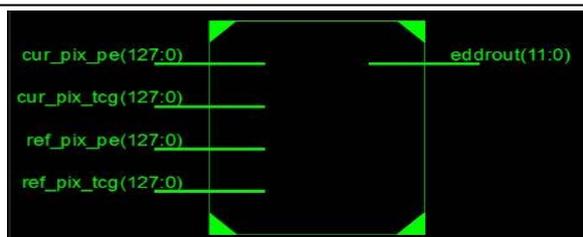


Fig. 5. RTL Schematic view of TOP module.

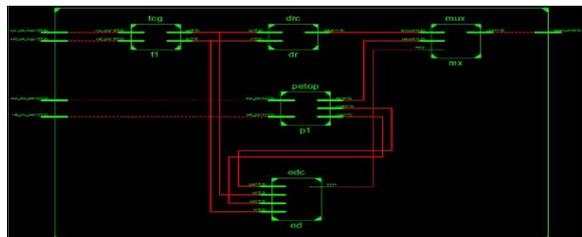


Fig. 6. RTL Schematic View of Internal modules.

V. CONCLUSION

This work presents an BIST architecture for detecting the errors and recovering the data of PEs in a ME. Based on the RQ code, a RQCG-based TCG design is developed to generate the corresponding test codes to detect errors and recover data. The RQ code generation, test code generation was also discussed and simulated using Xilinx 13.4 simulator.

REFERENCES

- [1] Design of an Error Detection and Data Recovery Architecture for Motion Estimation Testing Applications Chang-Hsin Cheng, Yu Liu, and Chun-Lung Hsu, Member, IEEE.
- [2] Analysis and Complexity Reduction of Multiple Reference Frames Motion Estimation in H.264/AVC-Chang-Hsin Cheng, Yu Liu, and Chun-Lung Hsu-Apr 2006.
- [3] Analysis and Architecture Design of Variable Block-Size Motion Estimation for H.264/AVC-Ching-Yeh Chen, Shao-Yi Chien, Yu-Wen Huang, Tung-Chien Chen, Tu-Chih Wang.
- [4] Design-for-testability techniques for motion estimation computing arrays- M. Y. Dong, S. H. Yang, and S. K. Lu - May 2008.
- [5] Efficient Built-In Self-Test for Video Coding Cores: A Case Study on Motion Estimation Computing Array- Yu-Sheng Huang, Chen-Kai Chen and Chun-Lung Hsu- Dec 2008.
- [6] A Built-In Self-Repair Design for RAMs with 2-D Redundancy- Jin-Fu Li, Jen-Chieh Yeh, Rei-Fu Huang, and Cheng-Wen Wu -Jun 2005
- [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 Circuits 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.
- [11] J. M. Portal, H. Aziza, and D. Nee, "EEPROM memory: Threshold voltage built in self diagnosis," in *Proc. Int. Test Conf.*, Sep. 2003, pp.23-28.
- [12] J. F. Lin, J. C. Yeh, R. F. Hung, and C. W. Wu, "A built-in self-repair design for RAMs with 2-D redundancy," *IEEE Trans. Vary Large Scale Integr. (VLSI) Syst.*, vol. 13, no. 6, pp. 742-745, Jun. 2005.
- [13] C. L. Hsu, C. H. Cheng, and Y. Liu, "Built-in self-detection/correction architecture for motion estimation computing arrays," *IEEE Trans. Vary Large Scale Integr. (VLSI) Syst.*, vol. 18, no. 2, pp. 319-324, Feb.2010.
- [14] C. H. Cheng, Y. Liu, and C. L. Hsu, "Low-cost BISDC design for motion estimation computing array," in *Proc. IEEE Circuits Syst. Int. Conf.*, 2009.
- [15] S. Bayat-Sarmadi and M. A. Hasan, "On concurrent detection of errors in polynomial basis multiplication," *IEEE Trans. Vary Large Scale Integr. (VLSI) Syst.*, vol. 15, no. 4, pp. 413-426, Apr. 2007.
- [16] C.W. Chiou, C. C. Chang, C. Y. Lee, T. W. Hou, and J. M. Lin, "Concurrent error detection and correction in Gaussian normal basis multiplier over GF," *IEEE Trans. Comput.*, vol.58, no. 6, pp.851-857, Jun. 2009.
- [17] L. Breveglieri, P. Maistri, and I. Koren, "A note on error detection in an RSA architecture by means of residue codes," in *Proc. IEEE Int. Symp. On-Line Testing*, Jul. 2006, 176-177.
- [18] S. J. Piestrak, D. Bakalis, and X. Kavousianos, "On the design of selftesting checkers for modified Berger codes," in *Proc. IEEE Int. Workshop On-Line Testing*, Jul. 2001, pp. 153-157.
- [19] S. Surin and Y.H. Hu, "Frame-level pipeline motion estimation array processor," *IEEE Trans. Circuits Syst. Video Technol.*, vol.11, no. 2, pp. 248-251, Feb. 2001.
- [20] D. K. Park, H. M. Cho, S. B. Cho, and J. H. Lee, "A fast motion estimation algorithm for SAD optimization in sub-pixel," in *Proc. Int. Symp. Integr. Circuits*, Sep. 2007, pp. 528-531.
- [21] J. F. Li and C. C. Hsu, "Efficient testing methodologies for conditional sum adders," in *Proc. Asian Test Symp.*, 2004, pp. 319-324.
- [22] D. P. Vasudevan, P. K. Lala, and J. P. Parkerson, "Self-checking carry select adder design based on two-rail encoding," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 12, pp. 2696-2705, Dec. 2007.
- [23] X. Yu, T. Meng, Z. Dai, and X. Yang, "Design and implementation of reconfigurable shift unit using FPGAs," in *Proc. IEEE Int. Symp. Pervasive Comput. Applic.*, Aug. 2006, pp. 543-545.
- [24] K. Neubeck, *Practical Reliability Analysis*. Englewood Cliffs, NJ: Pearson Prentice-Hall, 2004.
- [25] X. Li, J. Qin, B. Huang, X. Zhang, and J. B. Bernstein, "A new SPICE reliability simulation method for deep submicrometer CMOSVLSI circuits," *IEEE Trans. Device Mater. Reliabil.*, vol. 6, no. 2, pp. 247-257, Jun. 2006.

