

## A Build In Self Test Application For Motion Test Applications

**Geddham Haneesha**

MTech Student

Department of ECE

AnuBose Institute Of Technology(ABIT)

Paloncha, Khammam, India

**K.Vasavi**

Associate Professor

Department of ECE

AnuBose Institute Of Technology(ABIT)

Paloncha, Khammam, India

**ABSTRACT:** *Motion estimation plays a vital role in today's media applications. Hence testing of such a module is a significant concern. Even though several algorithms have been proposed in the past testing of motion estimators are seldom addressed. The proposed system describes an Error Detection and Data Recovery (EDDR) architecture that detects and recovers data in the motion estimator. The system uses the Mean Absolute Difference (MAD) method to compute the difference in the current and reference frames which overcomes some of the drawbacks in the existing techniques. The architecture comprises of an Error Detection Circuit (EDC) and a Data Recovery Circuit (DRC) to recover the original data. A Residue-Quotient code is used to compute the change in value between the error and expected values. Results show that the errors are effectively recovered with a delay of a single cycle.*

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

### INTRODUCTION:

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.

### Existing System:

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 simple 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.

### Proposed System:

The conceptual view of the proposed EDDR scheme comprises of two major circuit designs, error detection circuit (EDC) and data recovery circuit (DRC), to detect errors and recover the corresponding data in a specific Circuit Under Test (CUT) which is the systolic array. 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.

**Test Code Generator**

Figure 2, illustrates the structure of the TCG module in the proposed EDDR architecture. Notably, TCG design is based on the ability of the RQCG circuit to generate the corresponding test codes in order to detect errors and recover data. The specific PE<sub>i</sub> estimates the absolute difference between the Cur\_pixel of the search area and the Ref\_pixel of the current macroblock. Thus the Residue and Quotient value for an array size of N X N is given by equations (3) and (5).

$$R_T = \left\lfloor \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n |X_{ij} - Y_{ij}| \right\rfloor m \quad (2)$$

$$R_T = [r_{11}|m + r_{12}|m + r_{13}|m + \dots + r_{nm}|m] \quad (3)$$

$$Q_T = \left\lfloor \frac{(\frac{1}{n^2}) \sum_{i=1}^n \sum_{j=1}^n |X_{ij} - Y_{ij}|}{m} \right\rfloor \quad (4)$$

$$Q_T = [q_{11} + q_{12} + \dots + q_{nm}] + \left\lfloor \frac{r_{11} + r_{12} + \dots + r_{nm}}{m} \right\rfloor \quad (5)$$

Notably the error signal e is expressed as,

$$e = q_e.m + r_e$$

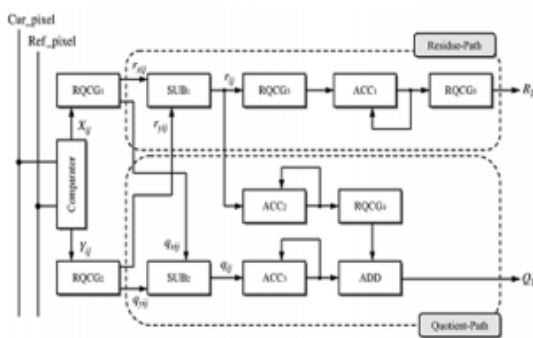


Figure 2. Test Code Generator

**Numerical Example**

A numerical example of the 16 pixels for a 4x4 macroblock in a specific PE<sub>i</sub> of a ME is described as follows. Fig. 5 presents an example of pixel values of the Cur\_pixel and Ref\_pixel. Based on (1), the MAD value of the 4x4 macroblock is,

$$\begin{aligned} MAD &= \frac{1}{n^2} \sum_{i=0}^3 \sum_{j=0}^3 |X_{ij} - Y_{ij}| \\ &= \frac{1}{16} [|X_{00} - Y_{00}| + |X_{01} - Y_{01}| + \dots + |X_{33} - Y_{33}|] \\ &= \frac{1}{16} \{(128-1) + (128-1) + \dots + (128-5)\} \\ &= 132.75 \end{aligned}$$

	0	1	2	3
0	128	128	64	255
1	128	64	255	64
2	64	255	64	128
3	255	64	128	128

Cur\_pixel

	0	1	2	3
0	1	1	2	3
1	1	2	3	4
2	2	3	4	5
3	3	4	5	5

Ref\_pixel

Figure 3. Example of pixel values

**RESULTS AND DISCUSSION**

The proposed architecture was executed on Windows XP operating system at an operating frequency of 2.80GHz using ModelSim for functional verification and synthesized using Xilinx ISE simulator. Table 1 gives the synthesis report provides the device utilization which is analyzed in terms of percentage. The timing summary of the code execution is also obtained. The proposed architecture was simulated on XC3S100E device of Spartan3E family provided the following synthesis results.

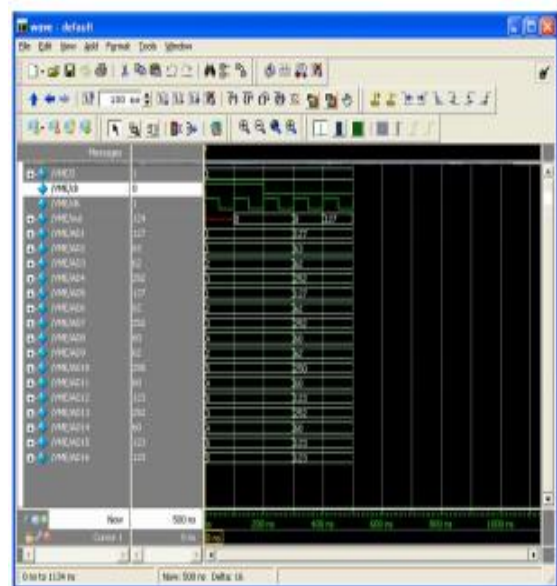


Figure 4. Screenshot of the operation of the proposed system

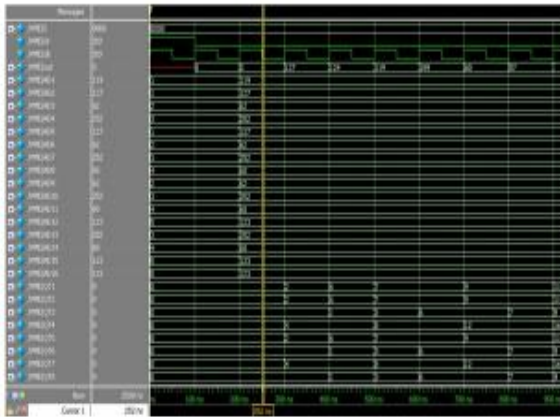


Figure 5. Pixel values set to the PE array and the TCG module

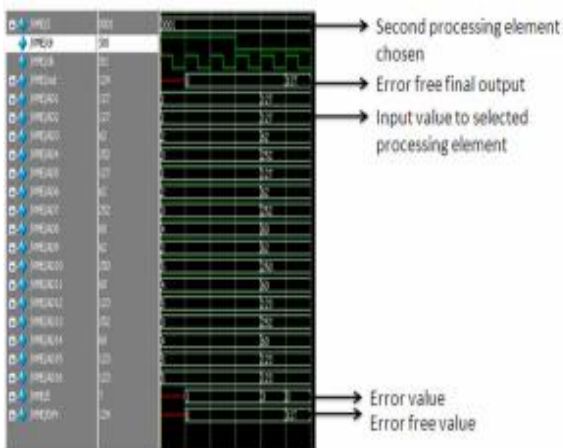


Figure 6. Comparison of error output with the recovered output

**CONCLUSION:**

Our work presents an EDDR 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 codes to detect errors and recover data. Experimental results obtained from ModelSim indicate that that the proposed EDDR architecture can effectively detect errors and recover data in PEs of a ME with reasonable area overhead and only a slight time penalty. Synthesis results show that the timing constraints and device utilization are minimum compared to error detection techniques available.

**REFERENCES:**

[1] Chang- Hsin Cheng, Yu Liu, and Chun-Lung Hsu, “Design of an Error Detection and Data Recovery Architecture for Motion Estimation Testing Applications” IEEE transactions on very large scale integration (VLSI) systems, VOL. 20, NO. 4, April 2012.

[2] Chun-Lung Hsu, Chang-Hsin Cheng, and Yu Liu, “Built-in Self-Detection/Correction Architecture for Motion Estimation Computing Arrays” , IEEE transactions on very large scale integration (vlsi) systems, vol. 18, no. 2, February 2010.

[3] 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.

[4] Che Wun Chiou, Chin-Cheng Chang, Chiou-Yng Lee, TingWei Hou, and Jim-Min Lin, “Concurrent Error Detection and Correction in Gaussian Normal Basis Multiplier over GF(2<sup>m</sup>)”, IEEE transactions on computers, vol. 58, no. 6, June 2009.

[5] 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.

[6] S. Dhahri, A. Zitouni, H. Chaouch, and R. Tourki, “Adaptive Motion Estimator Based on Variable Block Size Scheme,” in World Academy of Science, Engineering and Technology Jan. 2009

[7] Yu-Sheng Huang, Chen-Kai Chen and Chun-Lung Hsu, “Efficient Built-In Self-Test for Video Coding Cores: A Case Study on Motion Estimation Computing Array” - Dec 2008.

[8] 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.

[9] M. Y. Dong, S. H. Yang, and S. K. Lu, “Design-for-testability techniques for motion estimation computing arrays,” in Proc. Int. Conf. Commun., Circuits Syst., May 2008, pp. 1188–1191.

[10] D. K. Park, H. M. Cho, S. B. Cho, and J. H. Lee, “A fast motion estimation algorithm for SAD optimization in subpixel,” in Proc. Int. Symp. Integr. Circuits, Sep. 2007, pp. 528–531.

[11] S. Bayat-Sarmadi and M. A. Hasan, “On concurrent detection of errors in polynomial basis multiplication,” IEEE Trans. Vary Large Scale Integr. (VLSI) Systs., vol. 15, no. 4, pp. 413–426, Apr. 2007.

[12] T. H. Wu, Y. L. Tsai, and S. J. Chang, “An efficient design-for-testability scheme for motion estimation in H.264/AVC,” in Proc. Int. Symp. VLSI Design, Autom. Test, Apr. 2007,