# An Efficient EDDR Architecture for Video Coding Testing Applications

S.Shanavaz [1], S.Mahaboob Basha [2], I.V.Rameswar Reddy [3]
[1]PG Student, M.Tech VLSI, AVR & SVR Engineering College, Kurnool
[2]Associate Professor Department of ECE, AVR & SVR Engineering College, Kurnool
[3]Associate Professor, HOD of ECE, AVR & SVR Engineering College, Kurnool

## Abstract:-

*Motion Estimation (ME) has a critical role in the video coder .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. Processing Equipments (PEs) are the key components of Motion Estimation (ME). An error in PE affects the visual quality and PSNR (peak signal to noise ratio) of the high definition video. The error can be detected and recovered effectively by using proposed Error Detection and Data Recovery (EDDR) design based on Residue and Quotient (RQ) code. Thus the reliability and throughput of Motion Estimation will be improved. The experimental result shows the design achieve 100% fault coverage. The advantages of this scheme are minimal performance degradation, small cost of hardware.*

***Keywords:*** *- Motion Estimation, Processing element, Residue and quotient code, TCG, Throughput*



## 1. INTRODUCTION

Multimedia applications are more flexible and reliable when we used Advances in semiconductors, digital signal processing, and communication technologies. Video compression reduces storage space and increases transmission capacity. It is used in a wide range of applications. A new video coding standard called MPEG4 was developed by Video Coding Experts Group (VCEG) and Moving Picture Experts Group (MPEG). MPEG4 is also called Advanced video coding (AVC) or H.264.Among the coding systems, a ME in a video coder is the critical role so testing such a module is of priority concern. The visual quality and peak signal-to-noise ratio (PSNR) at a given bit rate are influenced if an error occurred in ME process. In 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.

For video coding systems, motion estimation is the most computationally demanding components in a video encoder. ME is of priority concern in exploiting the temporal redundancy between successive frames. In video coder 60% to 90% of the total of computation time is consumed in motion estimation. The motion estimation algorithm used also profoundly influences the visual quality of reconstructed images.

Generally ME consists of processing elements with a size of 4x4. As a commercial chip, it is necessary for the ME to introduce design for testability (DFT). There are many approaches to DFT, which can be divided into three categories Adhoc, structured and BIST. Among them, BIST has obvious advantages because there is no need to buy expensive test equipment, like Automatic Test Equipment, so it can efficiently reduce the test cost. BIST schemes not only detect faults but also specify their location for error correcting. BIST can generate test simulation and test responses without outside support. The extended BIST schemes generally focus on memory circuit, testing-related issues of video coding have been addressed.

**Disadvantages of Existing:**

- Poor performance in terms of high accuracy design for real time applications in DCT core on FPGA implementation.
- Does not achieve in terms of implementation on CMOS technology DCT core.

**Advantages of Proposed:**

- Fit for Real Time application in DCT Core
- More Reliable
- Less no of gate counts

This paper is organized as follows. Section 2 describes the RQ code generator. Section 3 describes proposed EDDR architecture. Conclusion is described in Section 4 and simulation results are shown in Section 5.

## 2. RQ CODE GENERATION

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 word, and $_m$ refers to the modulus. A

separate residue code of interest is one in which N is coded as a pair (N, $N_m$ ). Notably $N_m$ is residue of N modulo m .Error detection logic for is typically derived using a separate residue code detection logic is simply and easily implemented . Residue code detects only one bit error and the error cannot be recovered by residue code. 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. The mathematical model of RQ code is simply described as follows. Assume that binary data Y is expressed as

$$Y = \{b_{n-1}b_{n-2}\cdots\cdots b_2b_1b_0\} = \sum_{j=0}^{n-1} b_j 2^j.$$

In the RQ code R=Y modulo m and Q=Y/m respectively.

## 3. PROPOSED WORK

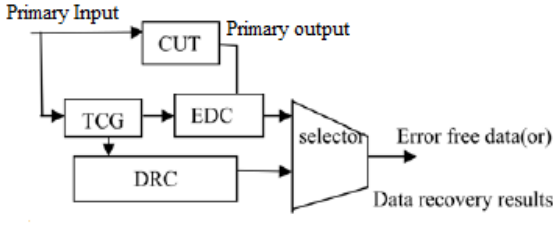



Fig.1.EDDR architecture

The block diagram of EDDR architecture is as shown in the Fig.1, 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.

### 3.1. Processing Element

The PEs are essential building blocks and are connected regularly to construct a ME. Generally ME consists of many PEs incorporated in a 1-D or 2-D array for video encoding applications. A PE generally consists of one 8-bit adder, one 12- bit adder and an accumulator (ACC). Next, the 8-bit ADD (a pixel has 8-bit data) is used to estimate the addition of the current pixel (Cur pixel) and reference pixel (Ref_pixel). Additionally, a 12-bit ADD and an ACC are required to accumulate the results from the 8-bit 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.

### 3.2. SAD TREE:

SAD Tree is a 2-D intra-level architecture called the Propagate Partial SAD. The architecture is composed of PE arrays with a 1-D adder tree. Current pixels are stored in each PE, and two sets of continuous reference pixels in a row are broadcasted to PE arrays at the same time. In each PE array with a 1-D adder tree, distortions are computed and summed by a 1-D adder tree to generate one-row SAD. The row SADs are accumulated and propagated with propagation registers in the vertical direction. 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.

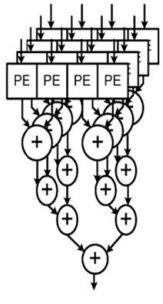



Fig.2. Partial SAD Tree

In Propagate Partial SAD, by broadcasting reference pixel rows and propagating partial-row SADs in the vertical direction, it provides the advantages of fewer reference pixel registers and a shorter critical path. Since Rt(Qt) is equal to RPEi (QPEi) EDC is enabled and a signal “0” is generated to describe a situation in which the specific PEi is error-free. Conversely, if SA1 and SA0 errors occur in bits 1 and 12 of a specific, PEi i.e. the pixel values of PEi= 2124. for video coding systems, motion estimation (ME) can remove most of temporal redundancy, so a high compression ratio can be achieved. Among various ME

algorithms, a full-search block matching algorithm (FSBMA) is usually adopted because of its good quality and regular computation. In FSBMA, the current frame is partitioned into many small macro blocks (MBs) of size. For each MB in the current frame (current MB), one reference block that is the most similar to current MB is sought in the searching range of size in the reference frame. The most common used criterion of the similarity is the sum of absolute differences (SAD).The absolute difference between each current pixel and the corresponding reference pixel in the original block and the block used for comparison is calculated. The differences are summed to create a simple metric of block similarity.

A numerical example of the 16 pixels for a 4X 4 mac-roblock in a specific PEi of a ME is described. An ex-ample of pixel values of the Cur_pixel and Ref_pixel. Based on, the SAD value of the 4X 4 macroblock of Fig 3

| | 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

Fig.3. Example of pixel values

SAD =

$$\begin{aligned}\text{SAD} &= \sum_{i=0}^{3}\sum_{j=0}^{3}|X_{ij}-Y_{ij}\\ &= X_{00}-Y_{00}+|X_{01}-Y_{01}|+\ldots+X_{33}-Y_{33}\\ &=(128-1)+(128-1)+\ldots+(128-5)\\ &=2124.\end{aligned}$$

### 3.3. TCG DESIGN

TCG is the combination of PE and RQCG. It will produce $R_T$ and $Q_T$ value and given to EDC circuit to detect errors. 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 4. 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 PEi in Fig estimates the absolute difference between the Cur_pixel of the search area and the Ref_pixel of the current macro black. It consists of 5 RQCG block and comparator, single accumulator block and subtractor

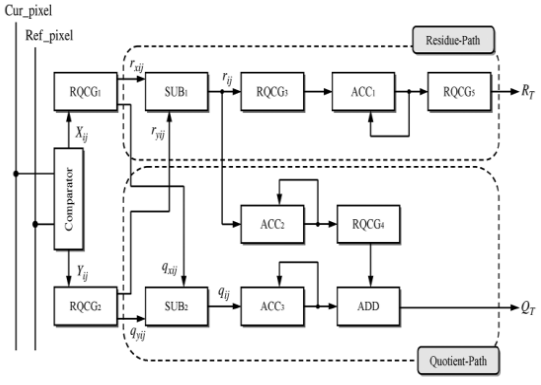



Fig.4.Circuit design of TCG

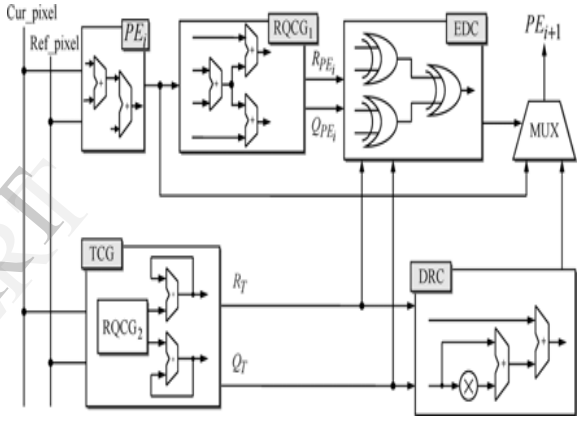



Fig.5. A specific PEi testing processes of the proposed EDDRarchitecture

### 3.4. EDDR PROCESS

EDC is designed for error detection in a specific PEs. Which is utilized to compare the outputs between TCG and in order to determine whether errors have occurred or not. The EDC output is then used to generate a 0/1 signal to indicate that the tested PEi is error-free/errancy. Using XOR operation can be identify the error if any variation in terms of residue and quotient value.

DRC was designed to generate error free output. In this quotient multiplied with a constant value (64) and add with reminder 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 PEi or data recovery that results from DRC is selected by a multiplexer (MUX) to pass to the next specific PEi+1for subsequent testing

.Thus, an error signal "1" is generated from EDC and sent to mux in order to select the recovery results from DRC.

### 3.5 Overall Test Strategy

The Fig.6 illustrates the overall EDDR architecture design of a ME. First The input data of Current pixel and Reference pixel are applied simultaneously to PEs and TCGs in order to estimate the SAD values. TCG generate the test RQ code $R_T$ and $Q_T$. Second, the SAD value from the tested object PEi , which is selected byMUX1, is then sent to the RQCG circuit in order to generate $R_{PEi}$ and $Q_{PEi}$ codes. Meanwhile, the corresponding test codes $R_{Ti}$ and $Q_{Ti}$ from a specific TCGi are selected simultaneously by MUXs 2 and 3, respectively. Third, the RQ code from TCGi 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 $R_{PEi} = R_{Ti}$ and $Q_{PEi} = Q_{Ti}$. Additionally, DRC is used to recover data encoded by TCGi , i.e. the appropriate $R_{Ti}$ and $Q_{Ti}$ 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) . 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 $PE_{i+1}$; otherwise, the final result is exported.

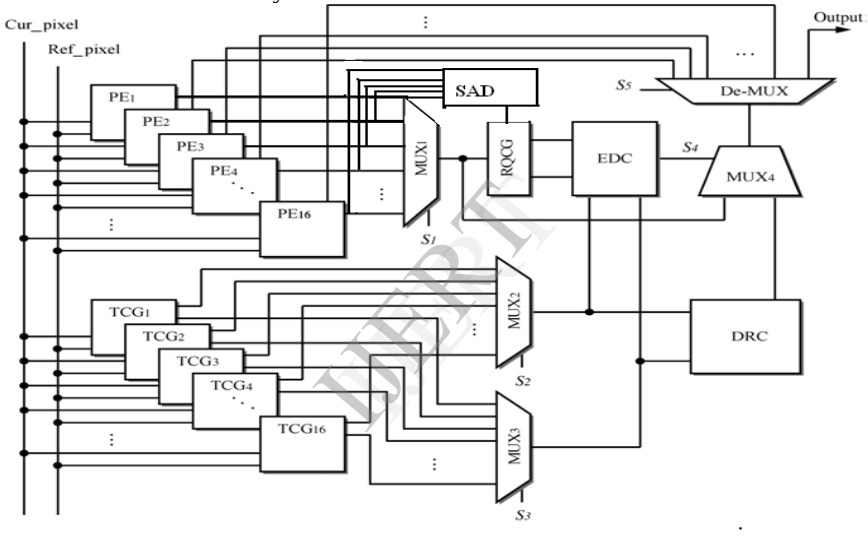



Fig .6.EDDR architecture design for a Motion Estimation

## 4. CONCLUSION:

This work presents 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 test codes to detect errors and recover data. The RQ code generation, test code generation was also discussed. The proposed EDDR architecture is also implemented by using Verilog and synthesized by the ModelSim and Xilinx. The complexity of RQCG code generation is reduced so less time is required to test the circuit.. The experimental result shows the design achieve 100% fault coverage

## 5. SIMULATION RESULTS:

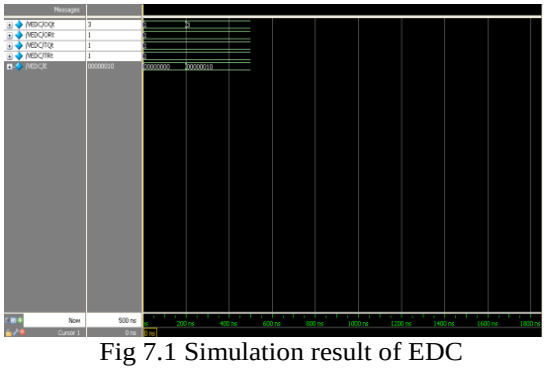


Fig 7.1 Simulation result of EDC

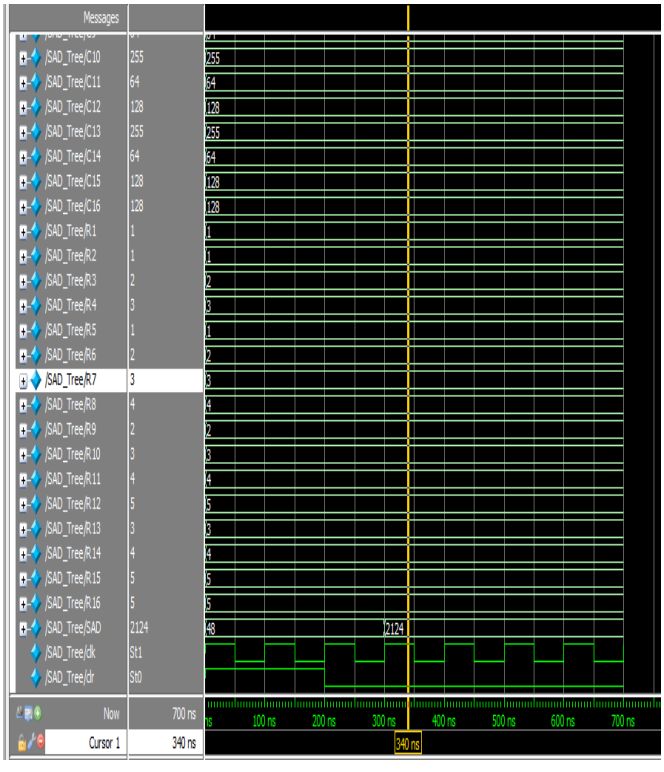


Fig 7.2 Simulation Result of SAD Tree

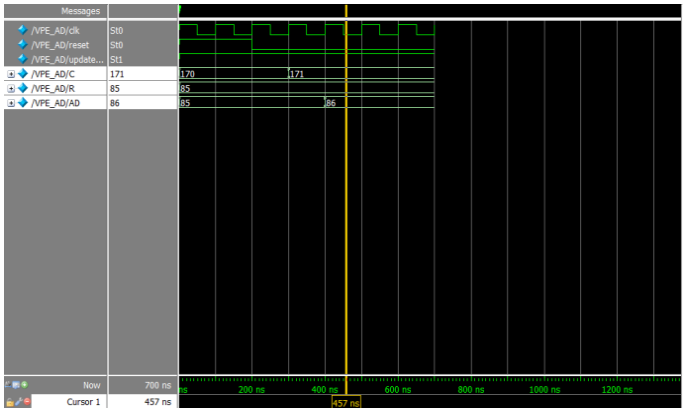


Fig 7.4 Simulation result of Processing Element

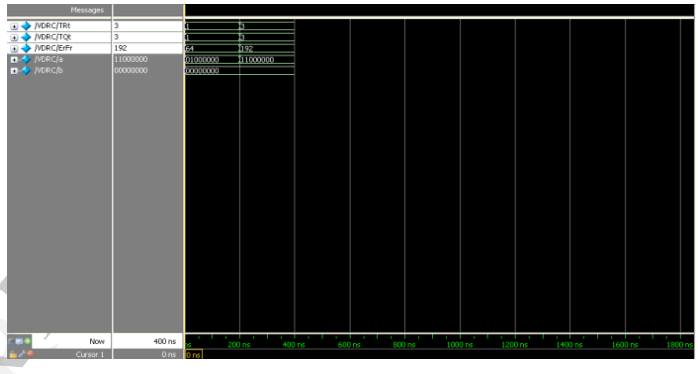
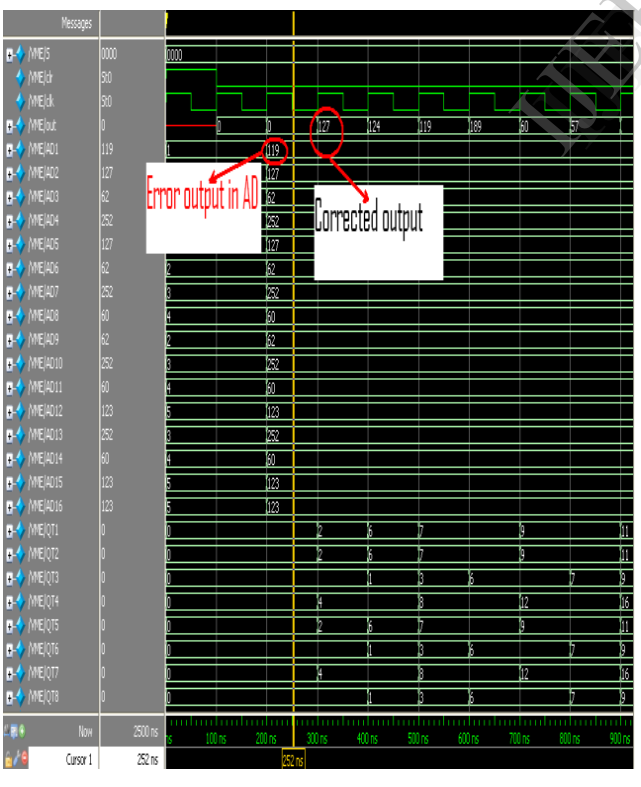


Fig 7.5. Simulation result of DRC




.Fig 7.3. Simulation result of Motion Estimation

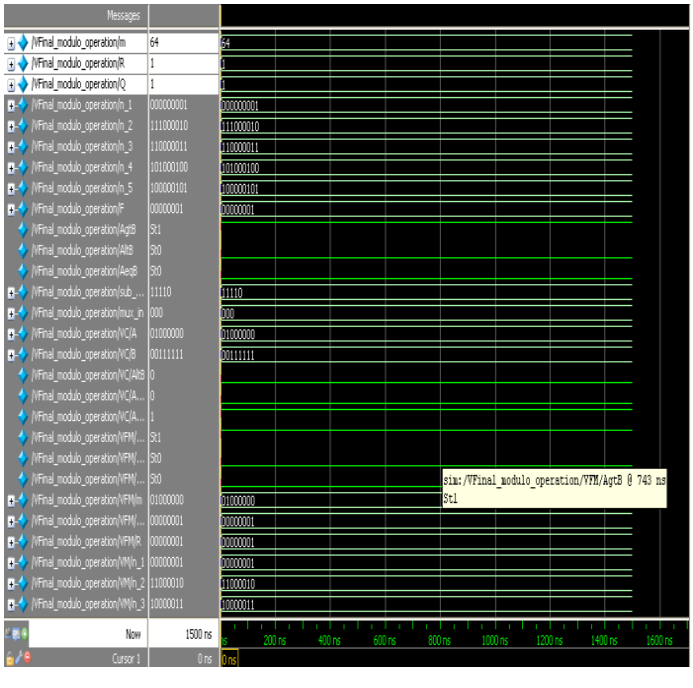


Fig 7.6. Simulation result of RQCG

## 6. REFERENCES:

[1] 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.

[2] J. F. Li and C. C. Hsu, "Efficient testing methodologies for conditional sum adders," in Proc. Asian Test Symp., 2004, pp. 319–324.

[3] 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 compu-ting array," in Proc. IEEE Asia Pacific Conf. Circuit Syst., Dec. 2008, pp.1751–1754.

[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] 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

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

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

[8] 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.

[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] . Tien-Ying Kuo, and Chen-Hung Chan, " Fast Variable Block Size motionEstimation for H.264 Using Likelihood and Correlation of Motion field", IEEE Transactions on circuits and systems for video Technology Vol. 16, No. 10, October 2006.