Design of EDDR Architecture for High Speed Motion Estimation Testing Applications

Adusumalli Naga Jyothi, Prof. Kandula Ravi Kumar
RAO & NAIDU ENGINEERING COLLEGE

ABSTRACT
Motion Estimation (ME) plays a critical role in a video coder, testing such a module is of priority concern. 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. The proposed EDDR design for ME testing can detect errors and recover data with an acceptable area overhead and timing penalty.

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

1. INTRODUCTION
Multimedia applications are flexible and reliable when used advances in semiconductors and communication technologies. H.264 video standard is a good example, it is also known as MPEG-4(Motion Picture Experts Group-4) Part10 Advanced Video Coding, it is widely regarded as the next generation video compression standard. Video Compression reduces the total data amount required for transmitting or storing video data which is necessary in a wide range of applications.

Motion estimation (ME) explores the temporal redundancy, it is inherent in video sequences and represents a basis for lossy video compression. Other than video compression, motion estimation can also be used as the basis for powerful video analysis and video processing. A ME (motion estimation) generally consists of PEs (Processing elements) with a size of 4 x 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. As a commercial chip, it is absolutely necessary for the ME to introduce design for testability (DFT). Device testing is increased by using DFT which gives high reliability of a system. DFT methods rely on reconfiguration of a circuit under test (CUT) to improve testability. Systems have meant that built-in self-test (BIST) schemes have rapidly become necessary in the digital world. BIST does not expensive test equipment, ultimately lowering test costs. The built-in testing approaches not only detect faults but also specify their locations for error correcting. Thus, BIST extended scheme referred to as built-in self-diagnosis and built-in self correction have been developed recently. The extended BIST schemes generally focus on testing-related issues, memory circuits of video coding have seldom been addressed.

2. PROPOSED EDDR ARCHITECTURE DESIGN

Fig. 1 shows the proposed EDDR scheme, which depends on 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) utilizes RQ code to generate the corresponding test codes for error detection and data recovery. DRC is in charge of recovering data from TCG and a selector is enabled to export error-free data or data-recovery results. This work adopts the systolic ME as a CUT to demonstrate the feasibility of the proposed EDDR architecture. A ME consists of many PEs (processing elements) 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 [20]. 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 PE of a ME.
3. MAIN MODULES OF EDDR ARCHITECTURE

a) PROCESSING ELEMENT
b) RQ CODE GENERATION
c) TEST CODE GENERATION
d) ERROR DETECTION CIRCUIT
e) ATA RECOVERY CIRCUIT

3a. PROCESSING ELEMENT (PE)

ME consists of many PEs incorporated in a 1-D or 2-D array for video encoding applications. PE generally consists of two adders (i.e. an 8-b ADD and a 12-b ADD) and an accumulator (ACC). Next, the 8-b Adder (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.

The specific estimates the absolute difference between the Cur pixel and Ref pixel. Thus, by utilizing PEs, SAD shown in as follows, in a macro block with size of N X N can be evaluated:

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

\[
= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (q_{x_{ij}}m + r_{x_{ij}}) - (q_{y_{ij}}m + r_{y_{ij}})
\]

Where \(r_{x_{ij}}, q_{x_{ij}}, r_{y_{ij}}, q_{y_{ij}}\) denote the corresponding RQ code of and modulo.

3b. RQ CODE GENERATION

Coding approaches are like this as Berger code, parity code, and residue code which is destined to detect circuit errors. Residue code is a generally separable arithmetic codes which estimates residue for data and appending it to data. Separate residue code is typically derived through error detection logic for operations, which makes an easy and simply implemented. It can detect only a bit error and additionally an error cannot be recovered effectively by using the residue code. Therefore, this work presents a quotient code, it is derived from the residue code, which detect the multiple errors and recovering errors.

3c. TEST CODE GENERATIONS

In proposed EDDR architecture TCG is an important component. The design of TCG is based on the RQCG circuit to generate corresponding test codes in order to detect errors and recover data. The specific PE (processing element) estimates the absolute difference between the Cur pixel and the Ref pixel. The residue code generates the RQ code \(R_T\) and \(Q_T\) from TCG.

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

\[
= |(X_{00} - Y_{00}) + (X_{01} - Y_{01}) + \ldots + (X_{(N-1)(N-2)} - Y_{(N-1)(N-2)})|_m
\]

\[
= |(X_{00} - Y_{00}) + (X_{01} - Y_{01}) + \ldots + (X_{(N-2)(N-1)} - Y_{(N-2)(N-1)})|_m
\]

\[
= (q_{x_{00}} - q_{y_{00}}) + (q_{x_{01}} - q_{y_{01}}) + \ldots + (q_{x_{(N-2)(N-1)}} - q_{y_{(N-2)(N-1)}})
\]

And

\[
Q_T = \frac{\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (X_{ij} - Y_{ij})}{m}
\]

\[
= \frac{|(X_{00} - Y_{00}) + (X_{01} - Y_{01}) + \ldots + (X_{(N-1)(N-2)} - Y_{(N-1)(N-2)})|}{m}
\]

\[
= \frac{|(q_{x_{00}} - q_{y_{00}}) + (q_{x_{01}} - q_{y_{01}}) + \ldots + (q_{x_{(N-2)(N-1)}} - q_{y_{(N-2)(N-1)}})|}{m}
\]

\[
= \frac{|(q_{x_{00}} - q_{y_{00}}) + (q_{x_{01}} - q_{y_{01}}) + \ldots + (q_{x_{(N-2)(N-1)}} - q_{y_{(N-2)(N-1)}})|}{m}
\]
Fig. 2. A specific testing processes of the proposed EDDR architecture.

Fig. 3. Circuit design of the TCG.

3d. ERROR DETECTION CIRCUIT (EDC)

The operations of error detection in a specific PEi is achieved by using EDC, based on the outputs between TCG and RQCGB in order to determine whether errors have occurred. If the values of RPEi ≠ RT and/or QPEi ≠ QT, then the errors in a specific PEi can be detected. The EDC output is then used to generate a 0/1 signal to indicate that the tested is error-free/errancy.

3e. DATA RECOVERY CIRCUIT (DRC)

DRC generates error free output by the quotient multiply with constant value and add to reminder code. During the data recovery, the circuit DRC plays a significant role in recovering RQ code from TCG. The proposed EDDR design executes the error detection and data recovery operations simultaneously and also error-free data from the tested PEi or data recovery that results from the DRC is selected by a multiplexer (MUX) to pass to the next species PEi+1 for subsequent testing.

\[ SAD = Q_T \cdot m + R_T \]

4. Numerical Example

A numerical example of the 16 pixels for a 4 x 4 macroblock in a specific PEi of a ME is described as follows. Fig. 5 presents an example of pixel values of the Cur_pixel and Ref_pixel. The SAD value of the 4 x 4 macroblock is

\[ SAD = \sum_{i=0}^{3} \sum_{j=0}^{3} |X_{ij} - Y_{ij}| \]

\[ = |X_{00} - Y_{00}| + |X_{21} - Y_{21}| + \ldots + |X_{32} - Y_{32}| \]

\[ = (128-1) + (128-1) + \ldots + (128-5) \]

\[ = 2124 \]
5. CONCLUSION

This work presents EDDR architecture for detecting the errors and recovering the data of PEs in a ME. By using RQ code, a RQCG-based TCG design is developed to generate the corresponding test codes to detect errors and recover data. Experimental results 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.

REFERENCES