DESIGN AND IMPLEMENTATION OF SAD ALGORITHM FOR MOTION ESTIMATION IN H.264/AVC

Geethanjali G[1], William Thomas[2]

[1]*M.Tech IV SEM, Dept of ECE, BITM Bellary*
[2]*Associate Professor, Dept of ECE, BITM Bellary*

Abstract

Video – telephony, video conferencing and video streaming to mobile phones via internet, in order to use effectively, the video is often compressed for low memory and fast transfer of video and then decompressed for use Motion Estimation is the power hungry block in the Video Compression System (VCS). The motion estimation operation determines the motion vectors, giving the best direction of the motion, and the "fitness" of that motion vector. Here the new low power full adder cell for low power applications is identified and is used in the proposed sum of absolute difference algorithm, the designs are implemented using ASIC flow, which results in 28.74% improvement in Leakage Power (LP) 12.201% improvement in Dynamic Power (DP) and 13.143% improvement in the total power even though the no of cells increased from 3933to 4501.

This paper introduces a basic hardware component "comparator" for the compression architectures. Comparator augments as general purpose core to Sum of Absolute difference (SAD) architecture used for the object recognition, generation of disparity maps of the stereo images and for estimating the motion in videos.

Keywords: *H.264/AVC, SAD, ADDER, COMPARATOR, LP, DP, DIGITAL SIGNAL PROCESSING, LOW PWER VLSI*

--***--

1. INTRODUCTION

Video processing is one of the techniques in Image processing which contains the filters where video frames are the inputs and outputs. The video frames may have some parts to be in motion or the entire frame. Hence video frames forms the vectors for estimating the motion. The motion is a 3-D scene while the image will be a projection of 3D scene on to the 2D plane. In today's world compression ratio plays the major role in the field of image processing. But the motion in the video scene will reduce the efficiency of the compression ratio. By exploiting the similarities between the video frames, efficiency of the compression ratio can be increased. The SAD algorithm is the simple metric system where the absolute difference between the corresponding elements is added and the smallest SAD value among the SAD blocks is considered as the similarity image. A video is comprised of series of frames; there will be redundancy between the adjacent frames known as temporal redundancy, to achieve the video compression this redundancy can be exploited. The Motion estimation uses the reference block of pixels and computes the suitable match for the current block, which eliminates the re-encoding of the entire block as a result it can only transmit the difference across the channel, which saves the bandwidth.

In real time coding applications, the computational cost of the Block matching algorithm is a significant problem. Many VLSI architectures are developed to reduce the computational cost and its complexity by speeding up the associated arithmetic calculation [4]. The various coding systems don't provide the flexibility as suggested to be with VLSI implementations. The other reasons to go for VLSI design is their adaptability to newer developments, sufficient performance and also faster design times by re-use of the designed IP cores. There are varieties of video coding standards, the modern video coding standard H.264/AVC is using the Variable Block Size Motion Estimation (VBSME), in this new coding standard, the computation requirements are much higher than the previous coding standards such as H.263/ MPEG-IV.

Due to the increasing demand for the portable devices with low power consumption and high performance; many research organizations has put their effort on the development of architectures with such design characteristics. Such kind of architectures at the basic building block level would impact largely at the efficiency. In the proposed architectures, effort has been put to improve the efficiency of the architectures at the system level by optimizing the architectures at basic building component level. In this paper, efficient hardware architecture for the basic building block – "comparator" of the SAD architecture is implemented. In comparator, the subtraction part is optimized by exploring the parallel computation in the existing architecture. Logic optimization technique is utilized to achieve the power efficient architecture. The optimizations at the component and basic building block levels are addressed. The new comparator architectures are proposed which optimizes design metrics such as performance & power.

2. RELATED WORK

Several methods of finding the motion vectors have been presented in the literature, where there is a trade off between the power dissipation, area and the latency in the optimality of hardware implementation. The work presented by [1-6] shows that motion estimation aims at reducing the temporal redundancy between successive frames in a video sequence. Innovation has put much emphasis on improving the video-coding giving rise to new standard H.264/AVC [7, 8]. The coding efficiency in this new standard is increased to about 40–60% as compared with the Motion Picture Experts Group (MPEG)-2 and H.263 standards. The work in [9] presented H.264/AVC encoder which employs 1024 SAD processing units (PE) which use 305k gates. The work in [10] proposed that SAD architecture and compared much architecture in terms of area and delay. The authors in [8] proposed that the SAD architecture with 1 and 2stage pipeline, the [09] describes this parallel hardware implementation of the SAD operation in field-programmable gate arrays (FPGAs).

A novel SAD16 unit which performs a 16 x1 SAD operation is proposed by [14]. but the work done in [7]-[9] presented the SAD architectures in terms of gate count and delay optimization, but the aspects of power consumption focusing on low power devices were not presented. The work in [10] presents a variable block size full search motion estimation architecture which employs a 32-parallel SAD tree with 387.2k gates (79% of the total gate count) and the power consumption is showed, but no exploration of different design points was performed. The proposed work from this paper highlights the power dissipation (both leakage power and the dynamic power) and the existing and proposed 8X8 or 64 – parallel SAD architectures are compared and the results are discussed in section 5.

In SAD architecture the similarities between the images are measured by calculating the absolute differences between the image pixels and their corresponding ones within the block. Then these absolute differences are added within the block, and compared against such SAD blocks for the smallest value and result in the similarity block. The SAD algorithm is the simple parallel computation system which considers all the pixels in the block for computation separately. Hence its implementation is easy and faster due to its parallel computation. It is the most widely used technique in motion estimation and object recognition [5].

The SAD architecture can be implemented in many ways and also in many domains. In [2] the SAD algorithm has been addressed by implementing it on FPGA. In [4], the author modeled SAD architecture using VHDL and utilized for motion estimation system and implemented in FPGA. Similar kind of implementation can be obtained in [5] but in different context while it has been synthesized using Cadence RTL Compiler. MATLAB implementation of SAD algorithm for Visual Landmark detector is implemented in [6].

3. PROPOSED METHOD

Digital Image processing applications like multimedia, surveillance, medical electronics, space exploration and others are dealt with processing of images, frames and videos, in the compressed form. The idea behind the compression is to correlate the data from time domain to frequency domain to reduce the required storage space. Generally image coding algorithm will be used for compression where correlation between the pixels is reduced and then Quantized & entropy coding is followed. Figure 1 show the steps involved in Image Coding Algorithm [7].

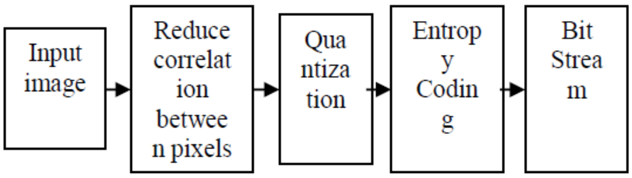

Fig 1 Image Coding Algorithm

But in Videos the frequency domain also needs to be correlated, which is done using Motion Estimation algorithm. Here the best motion part of the image will be searched which is the displacement of the best similar block in the previous block in the current block of frame and replaced [4].

Windowing technique plays an important role in improving the performance of the motion estimation system. In this technique matching is established only on interesting regions in the images or the frames. For instance, only high variation of intensity values in horizontal, vertical and diagonal directions are selected. Choosing the size of the window is critical part in this technique. Once the region is selected, a simple correlation scheme is applied in the matching process.

As mentioned in section I about the advantages of the VLSI hardware implementations; basic building block component "comparator" of SAD Processor of the Motion estimation system shown in Figure 2 is addressed in this context.

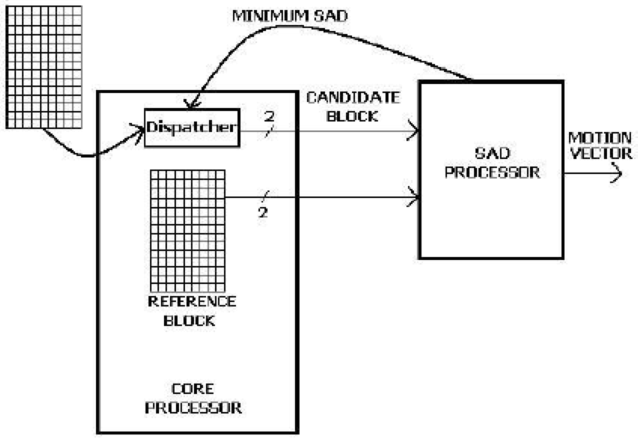

Fig 2: Motion Estimation System

Block diagram of SAD processor is shown in the Figure 3. The main blocks of the SAD processor are Absolute difference, SUM and Comparator blocks.

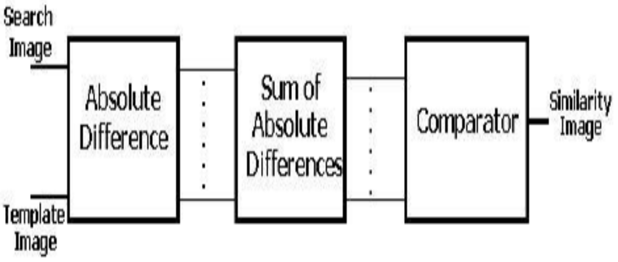

Fig 3: Block diagram of SAD Processor

In SAD processor, the Absolute difference block is used to calculate the absolute differences between the reference pixels of 4X4 block size and the corresponding current input pixel data of 4X4 block size; in the larger search area. 16 Absolute difference units are required for 4X4 block size. The outputs of each absolute difference units of 4X4 block are summed to form the single SAD value, and the process is repeated for next input 4X4 block size with reference pixel of 4X4 block size. The single SAD values of the each 4X4 block size in the search area are compared using comparator for the minimum SAD value. The corresponding 4X4 block sized current input pixel data of the minimum SAD value is considered as the block similar to the reference block against the other SAD blocks. The block size depends on the size of the reference block.

4. IMPLEMENTATION METHODOLOGY

The proposed and regular comparator architectures of 4 bit wide are designed and modeled using Verilog HDL and verified the functionality with the Mentor graphics Modelsim simulator using Waveform editor. The designs were synthesized using the Design Compiler and mapped to the TSMC 65nm technological library node. Standard ASIC methodology was considered for benchmarking the results.

Figure 4(a), Figure 4(b) and Figure 4(c) shows the Mentor Graphics Model-sim simulators basic steps of simulation flow, typical VLSI flow for synthesis and the Synopsys DC synthesis flow respectively. Modelsim is a simulation and Verification tool for Verilog HDL, VHDL, System Verilog, System C and mixed language designs. Initially working directory is created and all the design files are sourced in to it. Next the Design Units are compiled and simulator is loaded by invoking the top-module of the design. Finally the simulation will be run, and expected results are debugged.

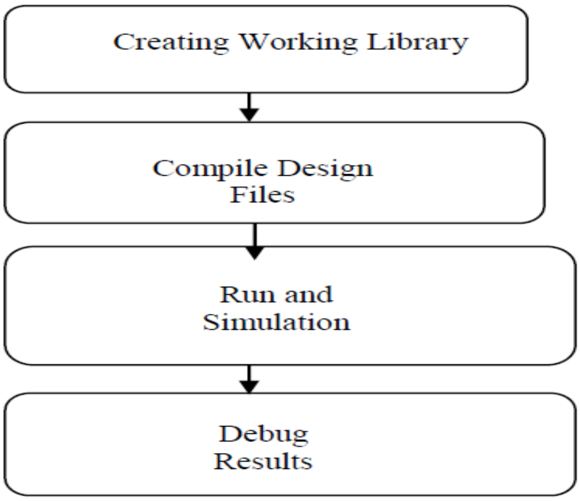

Fig 4 (a): Basic Simulation Flow

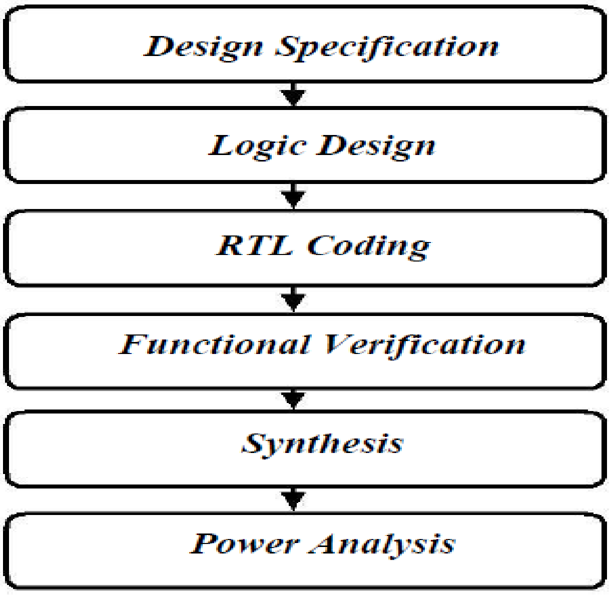

Fig 4 (b): Design & Power Analysis Flow

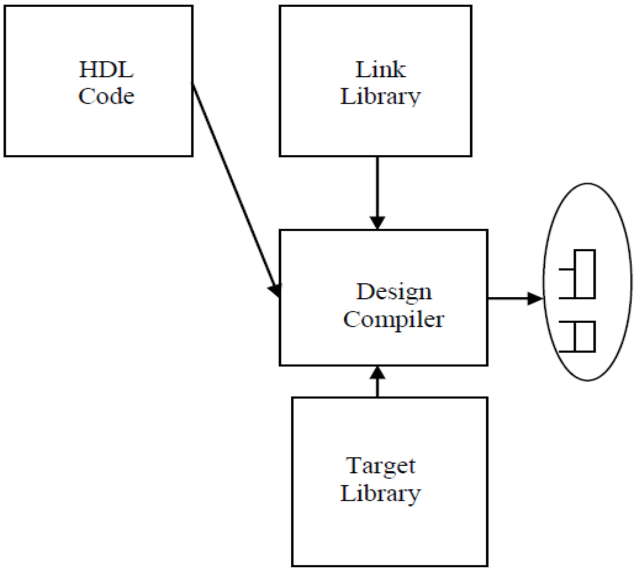

Fig 4 (c) Synopsys DC Synthesis Flow [14]

Synopsys DC Synthesisis is a complex task which consist of Several places and requires various inputs to arrive at the functionally correct netlists. Synthesises faces are reading the design, setting the constraints, optimizing the design, analyzing the design and saving the design data base. In the first phase it reads the input HDL and Checks for the synthatical errors and finally translate HDL objects into the technology independent design. Setting the constraints means instructing the design compiler to behave as per the requirement. The optimstion step translates the HDL description into gate level netlist using the cells available in the technology library. Final phase includes the generation of results in the synthesized design.

5. CONCLUSION

In this paper we implemented the existing and the proposed 8X8 (one of the subdivided macro block size used in the VBSME) sum of absolute differences. Here the new low power 1 bit full adder cell for low power applications is identified and is used in the proposed sum of absolute difference algorithm. The designs are implemented using ASIC flow, the simulations are done using modelsim and the synthesis is done using cadence-RC compiler, which results in 28.74% improvement in leakage power dissipation and 12.201% improvement in dynamic power dissipation and 13.143% improvement in the total power dissipation even though the no of cells increased from 3933 to 4501, with the area reduction from 30643 to 26080 but there is an increase of time delay of the critical path from 5059 Ps to 5473 Ps, as a future enhancement we need to work on further reduction of power and area so that the cost and area can be reduced. The basic building block component “Comparator” of SAD processor of Motion Estimation for Video compression is implemented. The impact of cell level computational logic at the block level is addressed by incorporating the Full adder in the comparator is addressed in this paper. The transistor stacking concept in the proposed architecture has reduced the leakage power by a significant amount (7- 43 %). The designs were modeled with the Verilog HDL and used 65nm technological library node for the synthesis in Design compiler tool. It can be observed that the proposed architectures are more power efficient than the counterpart regular architecture and enables the architectures to be analyzed with different corners of design constraints. The impact of Full adder at the comparator level proves that the power aware architectures at the component level can impact largely at the system level and can be carried over any level of abstractions as per the application requirement.

REFERENCES

[1]. Sumit K Chatterjee and Indrajit Chakrabarti, “A Fast and Low-power VLSI Architecture for Half-pixel Motion Estimation Using Two-step Search Algorithm for HDTV

[2]. FPGA Hardware”, International Journal of Electrical and Computer Engineering 4:9 2009.

[3]. C Hisham, K Komal, and Amit K Mishra, “Low power and less area architecture for integer motion estimation”, International Journal of Electrical and Computer Engineering 4:9 2009.

[4] Joaquin Olivares, Ignacio Benavides and et. al. , “Minimum Sum of Absolute Differences implementation in a single FPGA device”, Dept. of Electro-technics and Electronics, University of Cordoba, Spain.

[5]. D.V. Manjunatha, G. Sainarayanan, “Power Efficient Sum of Absolute Difference Algorithms for video Compression”, IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197 Volume 1, Issue 6 (Mar. – Apr. 2013), PP 10-18.

[6]. Craig Watman & David Austin, et. al, “Fast Sum of Absolute Differences Visual Landmark Detector”, Department of Systems Engineering, RSISE, Australian National University, Canberra, ACT 0200 Australia.

[7]. Lin, Pao-Yen. "Basic Image Compression Algorithm and Introduction to JPEG Standard." National Taiwan University, Taipei, Taiwan, ROC, 2009.

[8]. Weste, Neil, and David Harris. "CMOS Vlsi Design." A Circuits and Systems perspective, Pearson Addison Wesle, 2005.

[9]. Manjunatha, D. V., and G. Sainarayanan. "Low-Power Sum of Absolute Difference Architecture for Video Coding." Emerging Research in Electronics, Computer Science and Technology. Springer India, 2014. 335-341.

[10]. Yang, Shengqi, et al. "Accurate stacking effect macro-modeling of leakage power in sub-100 nm circuits." 18th International Conference on VLSI Design. IEEE, 2005.