

High Performance Multimedia Oriented Reconfigurable Architecture

Vigneshkumar C
Department of ECE-PG
Sona College of Technology
Salem, India

Hari Raj Kumar J
Department of ECE-PG
Sona College of Technology
Salem, India

Abstract—Multimedia Oriented Reconfigurable architecture (MORA) is a coarse-grained reconfigurable array of processor specially designed for accelerating multimedia processing applications. The reconfigurable architecture involves 2-D array of reconfigurable cells (RC) where the system has to provide a dense support for arithmetic operations. Booth multiplier is being introduced for this system which reduces the partial products that are generated during multiplication operations. Using this proposed method Discrete Cosine Transformation (DCT) is formulated, which is intended for fast streaming Image processing applications.

Keywords—Reconfigurable Array, Booth multiplier, Discrete cosine transform.

I. INTRODUCTION

Image processing, digital signal processing, video stream operations and others demand high performance computation where reconfigurable architecture makes attractive solutions. This architecture plays a key role between general purpose processors and applications specific circuits. The constantly growing area of application and algorithm to support, it becomes vital that these reconfigurable processors provide high level of flexibility. The MORA's main objective is to provide each reconfigurable cell with separate data and instruction memory to have better control over instruction execution and maximum memory bandwidth.

It also provides a varied distribution of each operation over a proper number of reconfigurable cells, thereby allowing each cell to perform tasks independently and at the same time obtaining maximum resource utilization.

The architecture consists of a two dimensional array of identical RC ordered in a 4*4 quadrants and linked through a hierarchical reconfigurable interlock network [1]. MORA with arrangement of Reconfigurable Cell providing each RC with inbuilt RAM memory supporting parallel computation required in various transformations for multimedia applications.

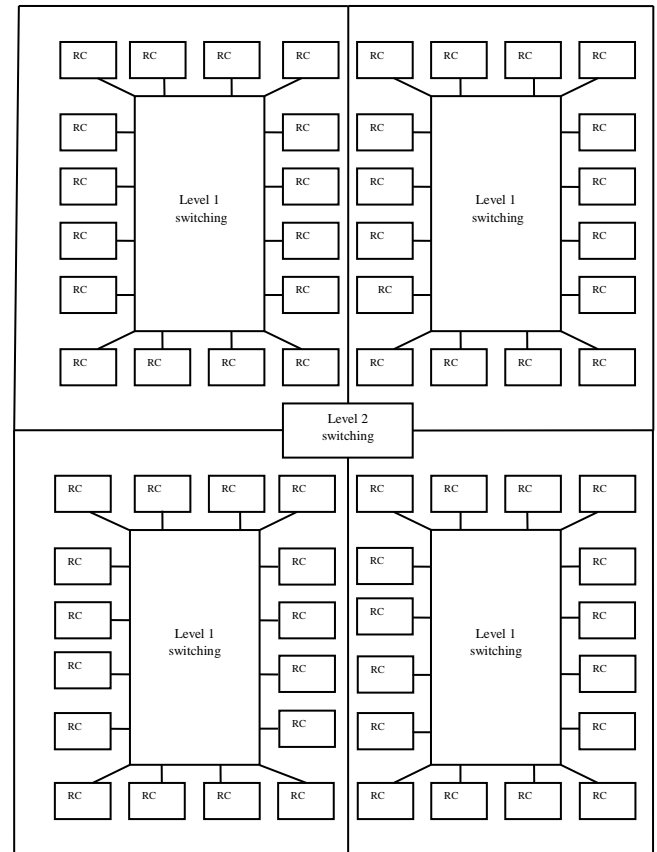


Fig.1. Top level organization of the MORA array.

Fig. 1, shows the top level organization of the MORA array. It can be compared to the configurable logic blocks (CLBs) in FPGA architecture replaced by the Reconfigurable Cell (RC) in DSP-style processor. The external data exchange is managed by a centralized Input/output data controller which accesses the internal memory of the RC by standard memory interface instructions. Between individual RC in the array of cells, internal data is controlled in a circulated manner through handshaking mechanisms. The switching helps to estimate the number of RCs required in each and every transformations.

II. RECONFIGURABLE CELL

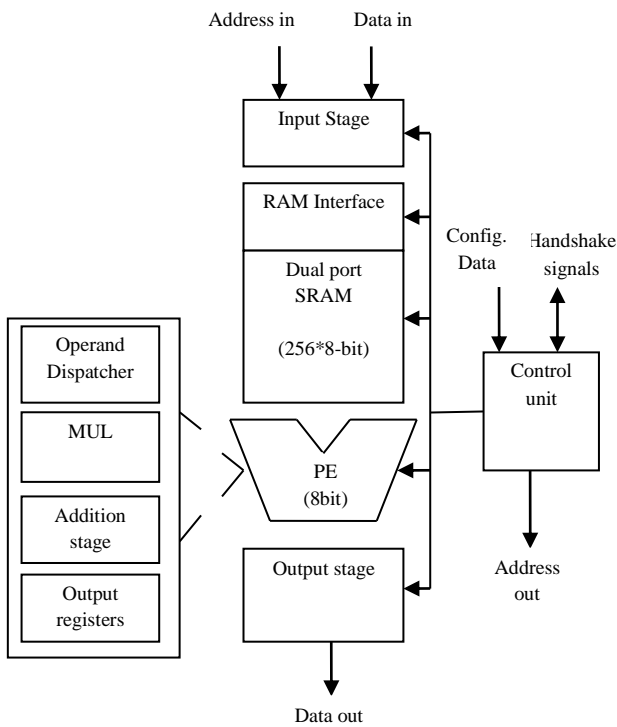


Fig.2. Reconfigurable Cell

Each Reconfigurable Cell as shown in Fig. 2, consists of a Processing Element (PE) for 8-bit arithmetic operations, data memory and control unit. In this Reconfigurable Cell (RC) the Input/output interface includes data/address input ports, two output data ports, single output address port and an additional interface signals needed to synchronize communication between the RCs [2]. The control unit allows the inter-cell communication between each and every RC with a certain degree of independence.

A. Arithmetic unit

The Arithmetic unit as shown in Fig. 3, consists of two's Complement Generator, Booth Encoding multiplier, Partial product generator and Carry Look-ahead Adder.

The array of reconfigurable cell performs parallel operations where fast arithmetic calculations are required for media processing applications. Booth encoding algorithm is a method that will reduce the number of multiplicand multiples. It is a dominant algorithm for signed-number multiplication, which makes both signed and unsigned numbers uniformly. The two 8-bit inputs A [7:0] and B [7:0] is given to the arithmetic unit. For multiplication purpose the multiplier bit B is encoded and the encoded signal is then multiplied with the multiplicand to provide certain intermediate partial products. The adders combine these partial products and produce the final result of the addition or subtraction operation.

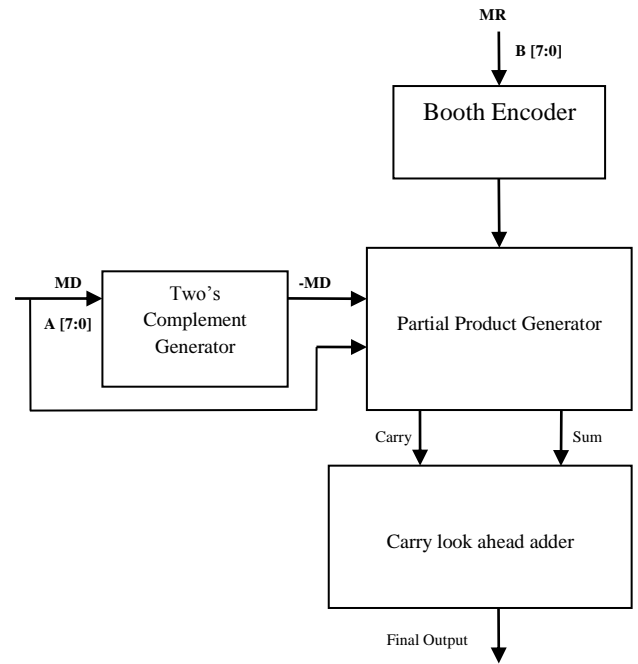


Fig.3. Arithmetic unit with Booth Encoding Method

For subtraction, the Multiplicand (MD) A is negated therefore performing 2's complement subtraction. Thus the booth's multiplication algorithm helps us perform multiplication much faster than any other conventional multiplier and also reduces hardware complexity.

The final results are available at the output of the 16-bit carry look ahead adder. These can then be sent to memory of the same or different RC as specified by the instruction. Media processing often requires that these arithmetic operations be carried out in a repetitive manner [3]. The registers also allow data from output register of one RC to be routed through to the input register of another RC directly, thus bypassing the memory.

B. Local data memory

Each and every Reconfigurable Cell consists of individual data memory for reducing the possibility of memory contention. The individual data memory helps to store the content of the arithmetic operations done by the Reconfigurable Cell. The RC during the program execution may require data from each other, hence a dual-port data memory is chosen which allows simultaneous operation for two 8-bit operands and also to provide easier memory interface between each and every Reconfigurable cell. Dual-ported RAM(DRAM) is a type of Random Access Memory that allows multiple read/write to occur at the same time unlike single-ported RAM which only allows single read/write operations for one memory cell accessed for one address during each clock cycle. Dual port RAM can read and write different memory cells simultaneously at different address.

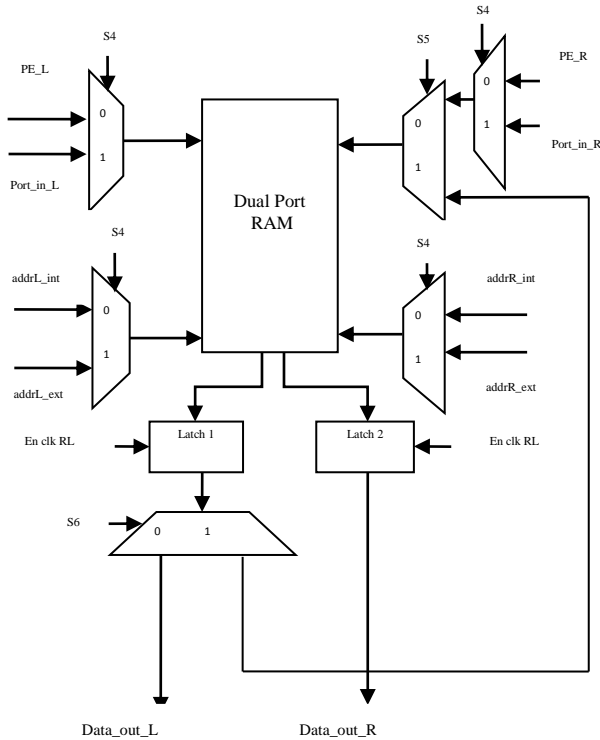


Fig.4. Data Memory

As shown in the Fig. 4, input data into the RAM memory are controlled using the signal S4 through an input multiplexer. These multiplexers write the result of the current PE operation or data from external RC into the specified memory address.

In the same way the flow of data out of a data memory into the PE of same RC or data memory of the neighboring RC or PE of the neighboring RC, is controlled by the signal S6 through output multiplexers. For media processing it requires operations based on the matrices and vectors, so the RC should be able to move data within its data memory.

Here signal s6 in the additional multiplexers allows RC to read data from a memory location through a left port and transfer to another memory location internally through right.

The read and write operation are done by rising and falling edges of the clock signal. This arrangement helps to write the 16-bit result of a multiplication operation into the location of the data memory which allows the RC to transfer data to another RC in a single clock cycle.

Both the internal and external handshake mechanisms are synchronized by the control unit thereby making each RC behave as a small independent DSP style processor [5].

C. Control unit

The decision making block of the Reconfigurable Cell is the control unit shown in the Fig. 5.

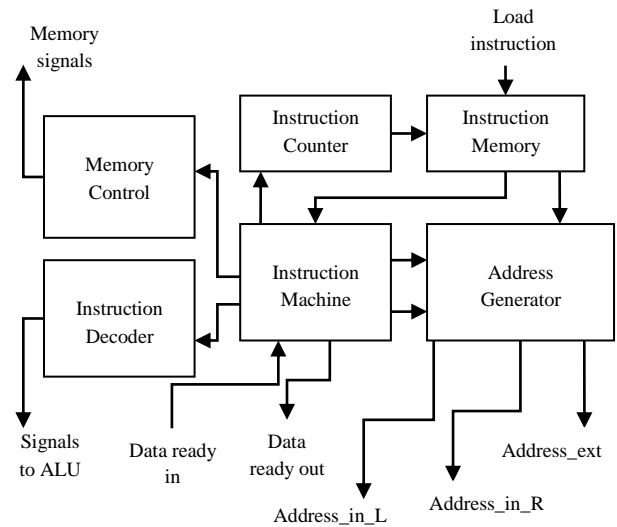


Fig.5. Control Unit

The control unit is liable for all the Reconfigurable Cell operations and also provides the handshaking signals between memory and data path, and ensures that they work in perfect synchronization with each other.

It has a small refreshable Instruction Memory, Instruction Machine, Instruction Decoder and address generator. The Instruction Memory stores the output address for operands A and B. The asynchronous handshake signals are controlled by the control unit to communicate with adjacent cells. The Instruction Machine controls the instruction fetch which synchronizes with the other integral part of the control unit. The instruction counter is used to consecutively step through the configured instructions.

In order to communicate between individual Reconfigurable cells the controller controls the external handshake mechanism unit. The RC checks the availability of the data either in its data memory or the output registers. The neighboring RC then sends an acknowledge signal, once the data transfer is completed. This will provide each RC to continue with its own independent execution cycle.

A unique feature for each and every individual Reconfigurable cell is the address generator. The address generator initially accepts the base address or initial memory address to fetch the operands. In address generators special descriptors specify the operands organization in the memory.

Depending upon the values of the special descriptors the address of the next memory location to be fetched by the data is calculated. Thus by generating the range of the address it becomes key feature for media processing applications.

III. SIMULATION OUTPUTS

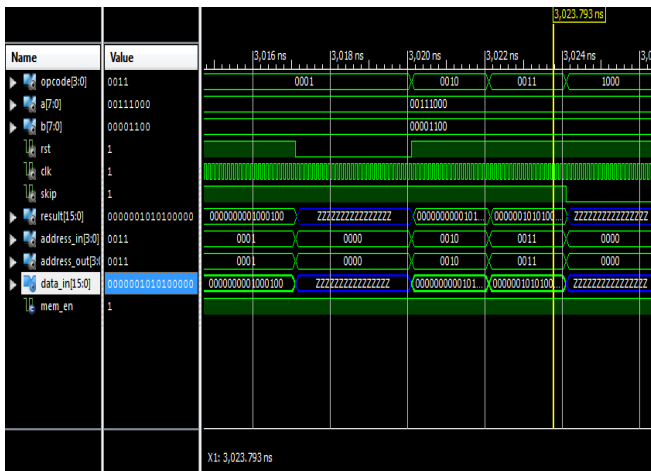


Fig.6. Simulation output of arithmetic unit for each RC block

The internal subsystem of the MORA block is analyzed and VHDL code is developed for the intentional purpose of each RC block. The Fig. 6, shows the Simulation output of arithmetic unit for each RC block where booth's algorithm is being introduced.

As the algorithm helps to reduce the intermediate products of the operations it helps to store less number of bits in the registers. The number of slices and number of four input LUT is less when compared to other conventional multiplier.

A. Discrete Cosine Transform Implementation

Discrete Cosine Transform (DCT) is a broadly used tool in image and video compression. DCTs are significant to various applications in science and engineering for compression of audio and images. The use of cosine rather than sine functions is precarious for compression, since it turns out that fewer cosine functions are needed to approximate a typical signal with limited boundary conditions.

DCT converts the information contained in a block(8x8) of pixels from spatial domain to the frequency domain. Before image compression data memory from image is divided into several blocks.

Each block consists of 8*8 pixels as a source then forward DCT is done from the input source followed by quantization, entropy encoder, entropy decoder, inverse quantization and inverse DCT to get the compressed image.

B. One dimensional DCT

The one dimensional Discrete Cosine Transformation is used to detect biomedical signals like Electroencephalograms(EEG) & Electrocardiograms(ECG) and also in speech information compression.

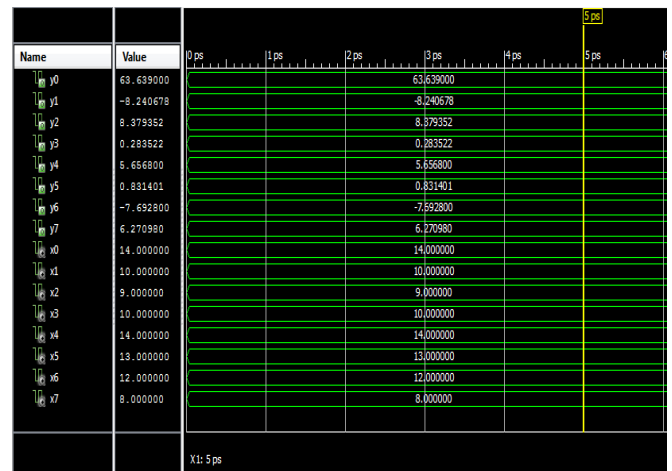


Fig.7. Simulation result of 1-D DCT

As shown in the Fig. 7, the one dimensional DCT simulation results with 1-D sequences helps to distinguish the two dimensional DCT with two 1-D image sequences.

C. Two dimensional DCT and its Inverse

The image is transformed into matrix format as input sequences. These sequences are then altered by using two dimensional Discrete Cosine Transform equations to get the compressed output sequences of the image.

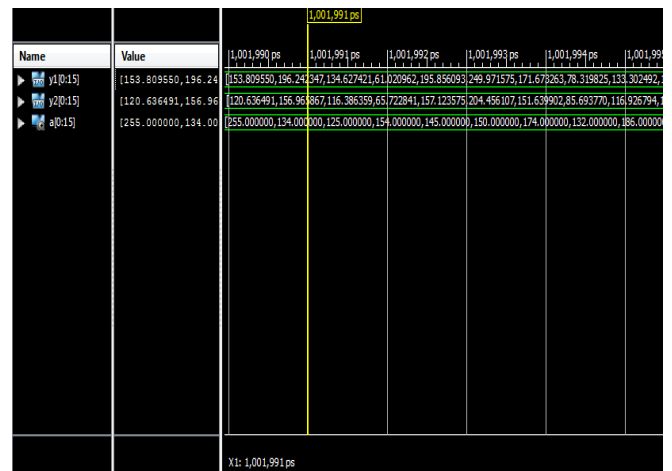


Fig.8. Simulation result of 2-D DCT & IDCT

The two dimensional simulation results are shown in the Fig. 8. These transformations are also used in Pattern Recognition and JPEG Encoders.

IV. CONCLUSION

In this proposed Multimedia Oriented Reconfigurable Architecture (MORA) a Booth multiplier is designed as a part of arithmetic unit for improving the speed of the arithmetic operations in executing multimedia functions. In addition the designed MORA is used for implementing 2-D

Discrete Cosine Transform, which can be utilized for image compression or any other image processing applications.

REFERENCES

- [1] Sohan Purohit, Sai Rahul Chalamalasetti, Martin Margala, and Wim Vanderbauwhede, "Throughput/Resource-Efficient Reconfigurable Processor for Multimedia Applications" in IEEE Transactions On Very Large Scale Integration (VLSI) Systems, Vol. 21, No. 7, July 2013.
- [2] W. Vanderbauwhede, M. Margala, S. Chalamalasetti, and S. Purohit, "Programming model and low level language for a coarse grained reconfigurable multimedia processor," in *Proc. Int. Conf. Eng. Reconfig.Syst. Algorithms*, 2009, pp. 375–380.
- [3] M. Lanuzza, S. Perri, P. Corsonello, "MORA- A New Coarse Grain Reconfigurable Array for High Throughput Multimedia Processing", Proceedings of International Symposium on Systems, Architecture, Modeling and Simulation, (SAMOS), pp-159-168, 2007.
- [4] C.Liang and X. Huang, "SmartCell: An energy efficient coarse-grained reconfigurable architecture for stream-based applications," *EURASIP J.Embedd. Syst.*, vol. 2009, pp. 1–15, Jan. 2009.
- [5] M. Butts, "Synchronization through communication in a massively parallel processor array," *IEEE Micro*, vol. 27, no. 5, pp. 32–40, 2007.
- [6] C. Ebeling, D. C. Cronquist, and P. Franklin, "RaPiD-reconfigurable pipelined datapath," in *Proc. 6th Int. Workshop Field-Program. LogicAppl.*, 1996, pp. 126–135.
- [7] Z. Yu, M. J. Meeuwsen, R. W. Apperson, O. Sattari, M. Lai, J. W. Webb, E. W. Work, D. Truong, T. Mohsenin, and B. M. Baas, "AsAP: Asynchronous array of simple processors," *IEEE J. Solid-State Circuits*, vol. 43, no. 3, pp. 695–705, Mar. 2008.