International Journal of Engineering & Technology, 7 (1.5) (2018) 158-163

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Implementation of FPGA Based MRPMA for high performance Applications

T. Siva Sankara Phani *, M. Sujatha, K. Hari Kishore, M. Durga Prakash

Department of ECE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India-522502
**Corresponding author E-mail: phani.tammireddy@gmail.com*

Abstract

In the last few decay, Network on Chip's (NoC) are the powerful chips for high speed communications pertaining to 802.11 Ethernet protocol which is a need to be reconfigurable for successful data frame transmission. The existing architectures like coarse grained reconfigurable, ALU cluster and expression grain reconfigurable architecture and look-up-table used in fine grained reconfigurable devices requires a lot of storage memory, hardware resources such as slices, cell area and cell delay. To tackle these issues, Multigrained Reconfiguration and Parallel Mapping Architecture (MRPMA) is proposed and their performance analysis parameters are calculated. The MRPMA uses the four contributions to optimize Processing Elements (PE's) operations: 1) Fast Fourier Transformation (FFT) to perform fixed point numbers to the configuration words, 2) Discrete Cosine Transformation (DCT) to analyze the data in the frequency domain, 3) Finite Impulse Response (FIR) for parallel mapping the data and 4) Channel encoder and decoder to encode the data and to calculate the shortest route from source to destination switch.

Keywords: *FFT, DCT, FIR, Channel encoder, FPGA, MRPMA, NoC.*

1. Introduction

The MRPMA is an efficient parallel computing architecture and its combines the high performance of ASIC for Very Large Scale Integration (VLSI) design and its implementation on the Field Programmable Gate Array (FPAG), The MRPMA is a typically compose of the host controller, one or two Processing Elements (PE's) and data shared memory. The superior important function of MRPMA is the efficiently mapping the loops onto each PE's to optimize the FPGA target device resources. There are two important aspects which are suffering the distributed resources in the FPGA, first is that the large number of resources required to design a NoC on FPGA such as PE's and register files and second, the existing coarse grained reconfigurable architecture (CGRA) mainly has three ways to transfer the data (1) by providing route information to PE's (2) by sharing the registers (3) using the data in shared memory. The three procedures are high cost due to high latency [10].

Giovanni ansaloni.et.al [1] described the various ways to design & test the architectural methods of EGRA based on programmable complex cells. In this EGRA Technique we have incorporated different machine description parameter to provide a novel interface for architectural instance: we have investigated to get gamut of information/data and its implementation of fixed function units. To obtain linear output, subjected to non-linear topology with this the analysis of various RAC's architectures. The regulated in the scope of the invention of coarse-grained architecture to provide enhanced support for automated mapping technology at the kernel level of flexible architecture, which provides a rich Avenue for Research & development in Architecture issues [11].

Shouyi yin.et.al [2] proposed and described the need for nested loop pipeline technique to enhance the performance of CGRA based on various Architecture parameters it has observed that thenested loop depends on memory overhead & utilization rate: as an outcome better utilization rate & reduced memory overhead has been achieved which led to. Greater improvement in the execution performance of nested loop architecture from the results we can show that the by joint affine transformation & multi pipeline merging approach to enhance the parallelism & reduce the memory depend on to reduce the overhead.

Coarse-grained Reconfigurable Architectures (CRAs) [3], [4] attracts extra attention because it has a tendency the soaring crisis and expenses with custom hardware potencies. though, where CRAs posses the advantage of each hardware's performance and also the software's flexibility with the dearth of ok compilation generation of effective mapping applications (usually loops). For the Reconfigurable ALU Array (RAA) with extensive kinds of reconfigurable architectures, [4] the venture of mapping comes from the idiosyncrasies of the structure as nicely because the trends of the applying. to beat the limits and maximize the performance, a few mapping sub troubles have been addressed within the literature (e.g., temporal partitioning for regular array measurement and pipeline vectorization for throughput development [5], memory operation sharing for restricted memory bandwidth [6], and expertise context switching for loop-carried dependency). One of the disadvantage seen as core mapping of placing and routing the operations of a loop frame onto the ALU array inside the context of CRAs is one which still requires a continuous have a look at.

In comparison to FPGAs, the facts-route width of coarse grained reconfigurable architectures CRA's is multiple bit. Sincelast15 years, many projects were investigated and effectively designed

techniques wherein the reconfiguration is coarse-grained and is completed within a processor or in group of processors. In such kind of techniques the reconfigurable unit is a specialized hardware architecture which enables as common reconfiguration as an awful lot quicker than that of FPGAs. because of this truth, the acknowledged equipment area led to layout full customized information paths, which are appreciably prolonged and high in electricity [12].excellent-grained gadgets are typically based totally on small look-up tables (LUT) to permit bit-level manipulations. Those gadgets are extremely versatile and can be used to map certainly any set of rules. Generally excellent grained architectures are used to permit bit level manipulations that make those gadgets extraordinarily versatile and fit it to any algorithm used. In hardware utilization, great grained structure's may be inefficient however here for processing input factors, it is settled one at a time for more number of clock cycles. Whilst in comparison to excellent grained architectures, the coarse grained architectures makes use of length of ALU's to full scale processors size for constructing block factors. In those coarse grained architectures, as shown in discern-1, the big computational factors are constructed inside the shape of arrays or like small programmable kernels and nation machines. This creation of CGA requires less configuration information and improves reconfiguration time and additionally routing assets want decrease hardware overhead [13].

The drawbacks visible in FPGA are omitted in CGA with the help of a resource named as couple of bit wide knowledge paths. With the assist of silicon tricky operator's efficiency of CGRA is progressed for huge records guides. The architecture shown in figure 1 is a rough- grained reconfigurable architecture that consists an array of processing factors and a routing network. Consequently, the routing overhead is avoided which generated because of compilation of hard operators from bit-degree processing models. The coarse grain reconfigurable architectures additionally have unique capabilities like, connecting multiple bits extensive that generate utilization for a person line and some other capabilities that highlight CGRA while as compared to FPGA is defining processing element orders. This additional characteristic leads a international cutback place usage for routing. It highlights the better granularity and less reduces down region for strain conversations as additional communication belongings which may be inefficient for quality grained architectures. Examples suggests for such assets are time-multiplexed buses or international buses, which join every processing detail [7].the following three step methodology are mentioned in CGRA.

Step1: The designer conceives a everyday model: an array of coarse grained processing element (PEs) interconnected by using mesh like network that is surrounded with the aid of using enter and output resources and memory blocks. This model is known as "architecture mannequin".

Step2: The dressmaker writes down the architecture version as a parameterizable description that's called "shape template". The template outlines the granularity, range and disposition of PEs, the possible network interconnections, and the institution of the memory accessories. Templates are bendy descriptions considering the fact that they'll be able to be modified thru adjusting the well worth of parameters. Parameters alter precise traits of the structure, inclusive of the wide variety of traces and columns (width and top) of the array, the amount of available reconfiguration contexts, the range of inner registers in the PE, and the interconnection network.

Step3: A structure example is generated through solving the worth of every template parameter.

The CGRA is generally based totally on PEA, host controller, context memory, and records memory, as shown in determine 2. The PEA is commonly a 2-D PEA. each PE includes an arithmetic good judgment unit (ALU), several neighborhood registers, and a configuration sign in. The characteristic of ALUs may be configured to one of kind phrase-stage operations of fixed-point numbers in line with the configuration words. The interconnection of PEs has several types, along with mesh, mesh plus [8], and morphosys topology [9]. The manner of configuration, CGRA can be divided into two categories: 1) full-reconfigurable CGRA and a couple of) partial-reconfigurable CGRA. In Coarse grained structure inputs are fed from one processing element to some other processing detail while the output is within the same clock cycle whereas in CGRA architectures best ALU operations are accomplished. within the PE-stage mapping as shown in determine 3, the loop body is represented as expression trees of micro-operations. The micro-operation trees are blanketed with styles that may be implemented with a unmarried configuration of a PE, producing PE-level operation timber. A PE-level operation is an abstraction for a sample of micro-operations that can be implemented with one configuration of a PE like If a sequence of upload and store are used with greater than two reminiscence operations may be applied with one configuration.

2. Methodology

The existing CGRA results are suffering the full parallelism problems and high hardware utilization. Based on the literature survey analysis and their drawbacks for pipelining and data transfer, we found that the better solution to perform pipelining on MRPMA is adapted for data dependence transfer from one source to destination switches in a NoC router.

This paper gives the information about the following four contributions

1. The proposed the uses of FFT transformation to facilitate the inner first column pipelining process for first column, four PE operation and it will reduce the outer loop carried dependence.
2. From the analysis of hardware features of a MRPMA, the performance analysis module designed by a DCT is made for the second column of NoC to optimize the resources on FPGA.
3. Using the FIR and channel encoder for the third and fourth column approaches to dependence pattern for inner nested loop pipelining and the merging of FIR and channel encoder for the both columns can be further improve the parallelism of multiple pipelines simultaneously and reduce the memory accessing cost as compared to outer loops carries dependence.

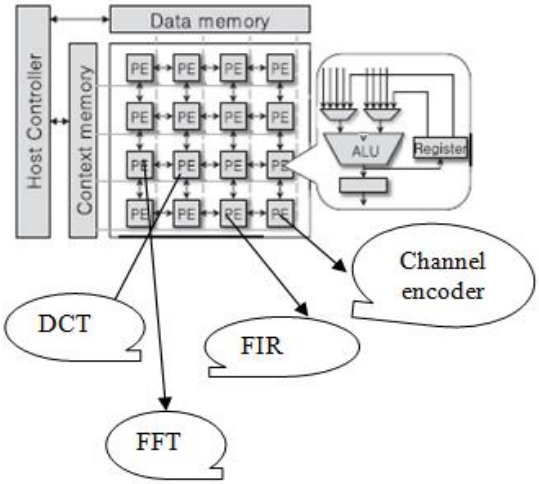

Fig. 1: Proposed Multigrained parallel mapping reconfigurable architecture

3. Background of MRPMA

The MRPMA is generally depends on PE's, controller of host, shared memory and data memory as shown in Fig.2. All the PE's are two dimensions arrays and each PE contains an ALU, internal register and programmable registers. The ALU modules are to configure to perform fixed or floating point number according to the IEEE754 standard. The data memory is used for storing the incoming and output data from each PE's and these PE's are

designed with larger number of register banks to improve the throughput and efficiency and can be calculated as follows

$$\text{Throughput} = \frac{\text{Input frequency x Size of input data bits}}{\text{No of clock cycles used for the design}} \quad and \; Efficiency = \frac{Throughput}{No\ of\ Slices}$$

For the proposed PE utilization rate is measured the performance of parallelism of pipelining of MRPMA and it can be represented as follows

$$Utilization\ rate = \frac{No\ of\ operations\ in\ the\ each\ PE\ per\ data}{Size\ of\ input\ data}$$

The designed and developed multigrained reconfigurable NoC consists the data paths and control channel for successful delivery of the packet to the destination. In general the NoC is developed by using processing elements (PE's) for processing of incoming data. In order to reduce the hardware resources on Field Programmable Gate Array (FPGA), the ALU has been used to perform the PE's operations, but still there is a tradeoff between power consumption and delay. To optimize these parameters MRPMA is proposed, where the ALU replaced with four multiple operations such as FFT, DCT, FIR and channel encoding to design a 4x4 NoC router. The designed router includes configuration memory, host controller, rows and column switches which are designed by FFT, DCT, FIR and channel encoding and also shared register files as shown in Fig.2. The MRPMA realized the each PE in NoC router to redistribute the data among the all nodes or switches and controller circuit is to scan the all processing nodes for the execution of common expression.

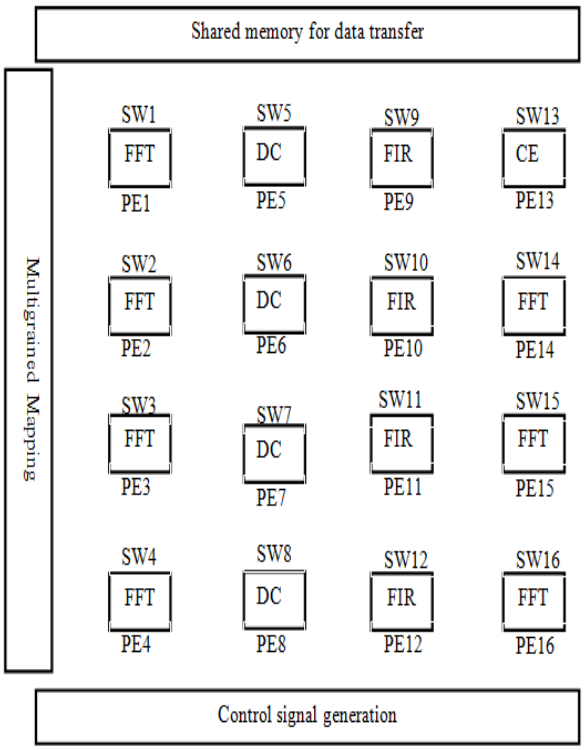

Fig. 2: Proposed MRPM Architecture for 4x4 NoC router using FFT, DCT, FIR and channel encoder

In the designed NoC router, having four columns, the first column designed by using FFT, Second column using DCT, third column by FIR and last fourth column designed using channel encoder. Each NoC router is a combination of the ALU, multiplexer and registers files. Since the MRPMA constructed with fast four different operations, it optimized the number of clock cycles, delay and power consumption. The data memory in the Fig.2. is to receive the number the data from any other host and then it will supply the same data to another host and then it will supply the same date to source PE in the NoC for processing and then transfer to the destination PE.

The following steps are the proposed 4x4 NoC router to process each column operation using four parallel reconfigurable techniques.

1. The data will be stored in the internal memory for further process based on the write and read commands.
2. FFT performs the ALU all associated arithmetic operations depending on the control signal generated by the control memory unit.
3. The FFT output transmits to the FIFO for the transmission of the data to the destination PE as shown in Fig.3.

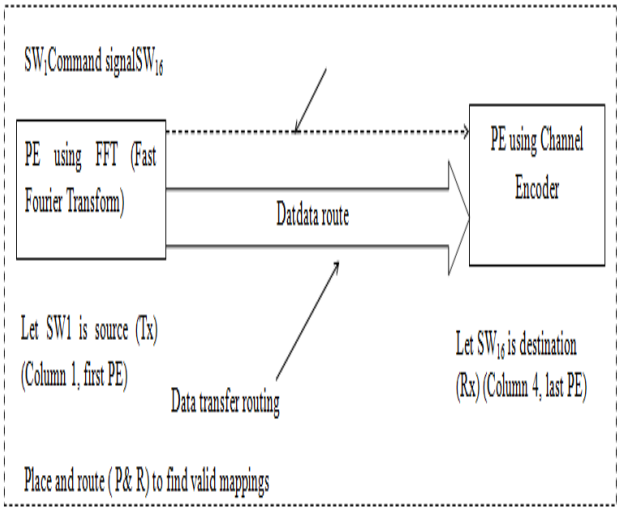

Fig. 3: Proposed MRPMA data transfer routing between SW_1 and SW_{16}

$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk}$ for 0≤ k ≤ N-1

Suppose the received data by data memory supplied to third PE in the second column then the following steps are listed

1. The data will be stored in the internal memory for further process based on the write and read commands.

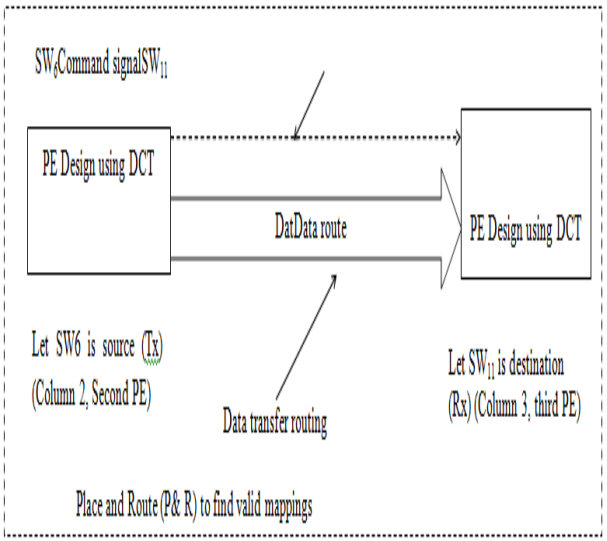

Fig. 4: Proposed MRPMA data transfer routing between SW_6 and SW_{11}

2. DCT performs the ALU all associated arithmetic operations depending on the control signal generated by the control memory unit.

3. The FFT output transmits to the FIFO for the transmission of the data to the destination PE as shown in Fig.4.

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\Pi}{N}\left(n+\frac{1}{2}\right)k\right] \quad \text{for k=0,1,…..,N-1.}$$

Suppose the received data by data memory supplied to first PE in the third column then the following steps are listed

1. The data will be stored in the internal memory for further process based on the write and read commands.
2. FIR performs the ALU all associated arithmetic operations depending on the control signal generated by the control memory unit.
3. The FFT output transmits to the FIFO for the transmission of the data to the destination PE as shown in Fig.5.

$y(n) = \sum_{i=0}^{N} bi \,.\, x(n-i)$ for $0 \leq n \leq N$

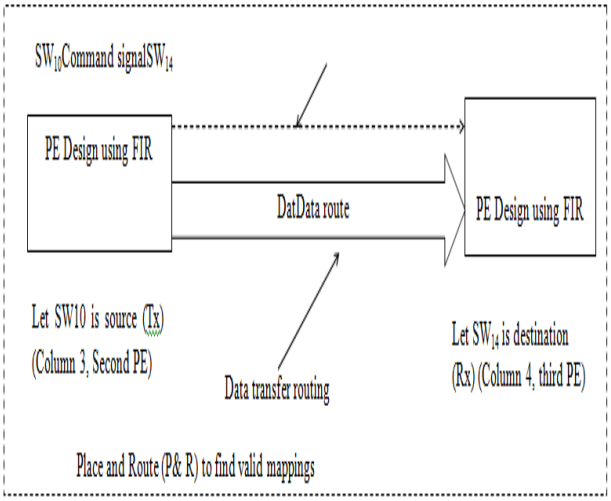

Fig. 5: Proposed MRPMA data transfer routing between SW_6 and SW_{11}

Suppose the received data by data memory supplied to fourth PE in the fourth column then the following steps are listed

1. The data will be stored in the internal memory for further process based on the write and read commands.

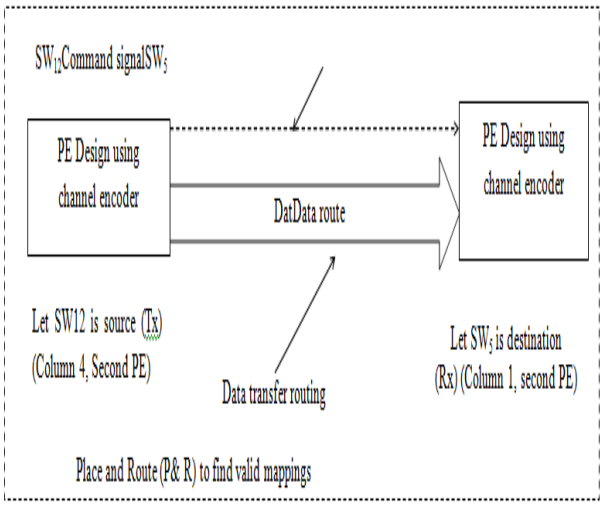

Fig.6: Proposed MRPMA data transfer routing between SW_{12} and SW_5

2. Channel encoder performs the ALU all associated arithmetic operations depending on the control signal generated by the control memory unit.
3. The FFT output transmits to the FIFO for the transmission of the data to the destination PE as shown in Fig.6.

Channel Capacity, $C = B \log_2\left(1 + \frac{p_s}{N_0 B}\right)$

Where p_s=transmit power in watts,
B=channel bandwidth,
N_0 =noise power spectral density.

4. Results and discussion

The first column is designed using FFT to obtain the complex operations like floating point numbers and transform the input signal from time domain signal to frequency domain signal for easy and better analysis of the complex data. The real and imaginary data is subjected to filter, both data are performing separately and their results are displayed in the Fig.7.

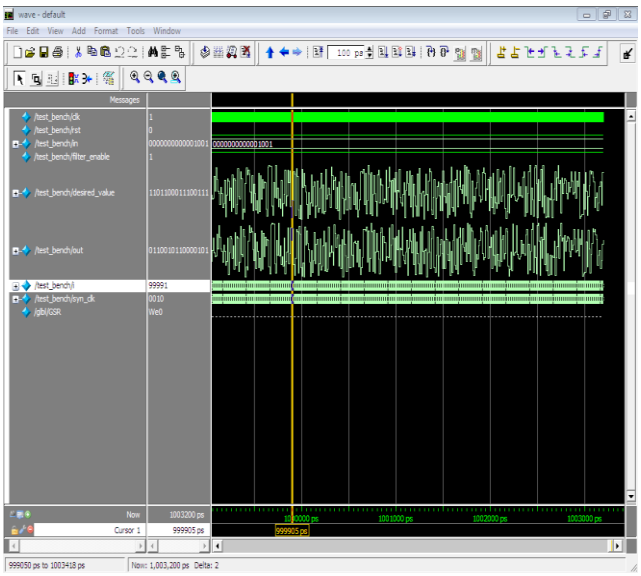

Fig. 7: Simulated results for FFT and filtered output

The second columns of four switches have been operated with DCT with 8 coefficients to perform cosine transformation data as shown in Fig.8. The coefficients are stored in memory of 8 locations and each coefficient is convolution with incoming data to each switch of second column.

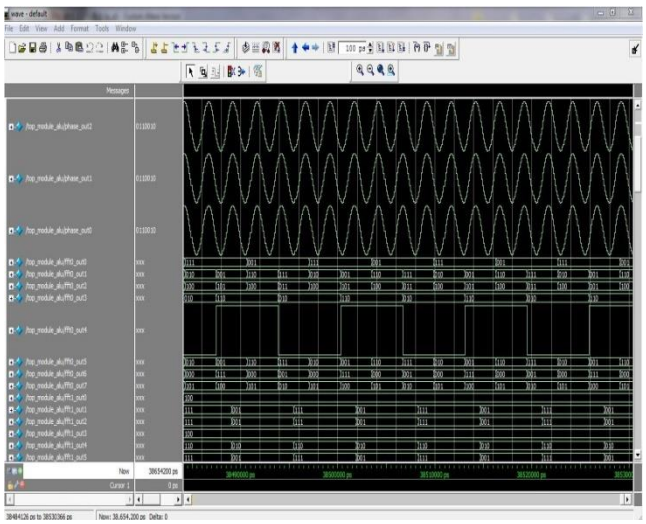

Fig. 8: Simulation results of DCT and its filtered signals

The third column of Multigrained parallel mapping NoC is designed with FIR with 4 tap operations, the main objective of FIR deign in this paper is to calculate the phase of the each switch signal as shown in Fig.9.

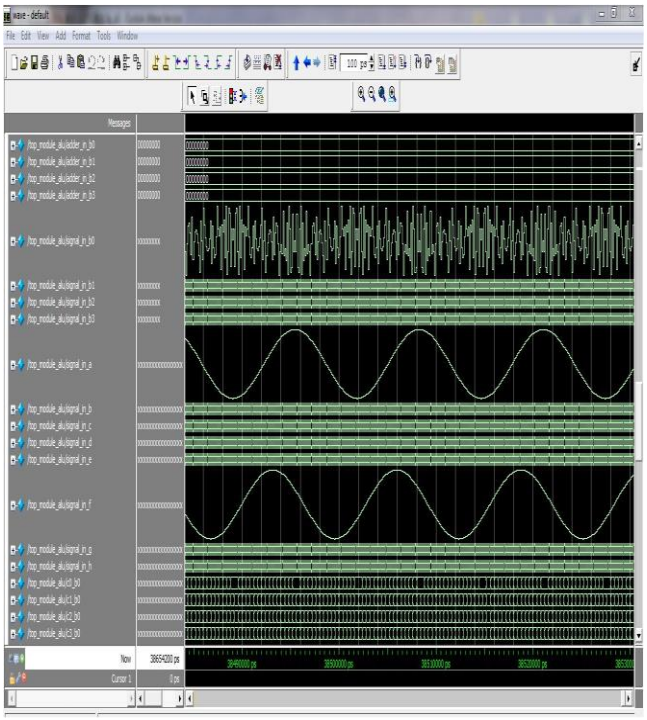

Fig. 9: FIR design for phase calculation in the third column of 4x4 NoC

Finally, the fifth column is designed using channel encoder for phase as well as magnitude calculation. Each switch incoming data is encoded and then transmitted to destination when it is source. If any switch is acting as destination, then it is act as a decoder to decode the original data as shown in Fig.10.

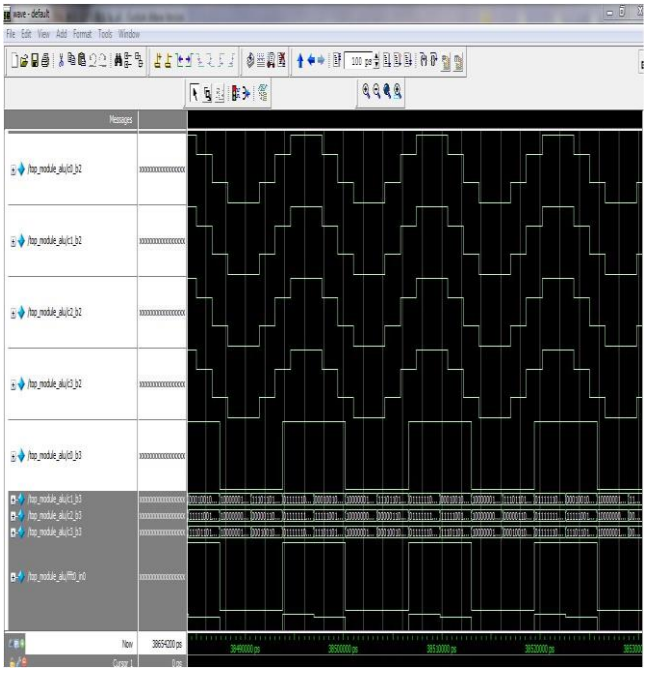

Fig.10: Simulated results of channel encoder for fourth column in the 4x4 NoC

The complete design is implemented on Virtex-5 FPGA using Chip scope pro tool and the results are shown in the Fig.11. Totally 8 control signal has been used to select each column operation. The first two control signal for FFT, next two signal for DCT, next two signal for FIR and last two signal for channel encoder as shown in Fig.11.

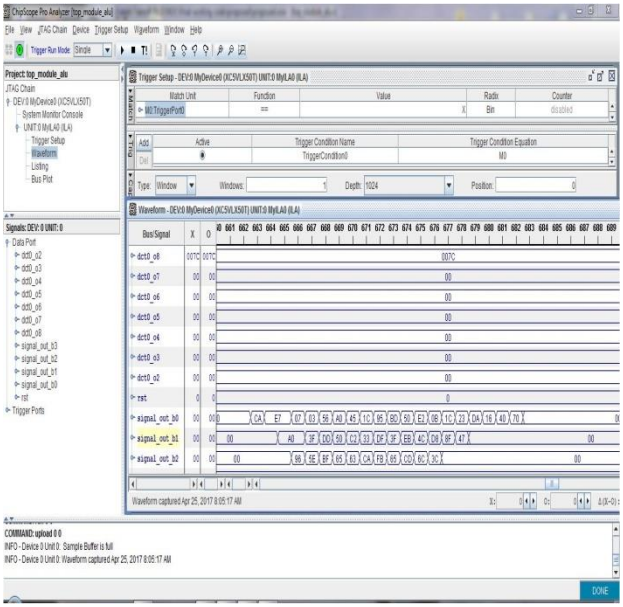

Fig. 11

5. Conclusion

The developed and designed NoC using MRPMA is promising architecture that will provide the high speed of data transmission, high performance and high power-efficiency product. This paper is introduced the some cost functions of delay on hardware synthesized like sequential circuits for mapping of different modules on FPGA area. The MRPMA is used for calculation of area in terms of slices and to minimize the delay and power consumption. The designed 4x4 NoC using MRPMA is a routing graph and that can perform easily to adapt the rotation and non-rotating. This NoC is exclusively for to obtain high latency
NoC router. The MRPMA flexibility became exploited in this paper to research overall performance of complicated RACs, displaying how they outperform simpler ones on a set of embedded system benchmarks; its assist of heterogeneous cells is instrumental in developing instances mapping entire computational kernels; sooner or later, its capability to instantiate multiple reminiscence types gives insights on tradeoffs in enforcing garage requirements.

References

[1] G. Ansaloni, P. Bonzini, and L. Pozzi, “Design and architectural exploration of expression-grained reconfigurable arrays,” in Proc.6th Symp. Appl. Specific Process., Anaheim, CA, Jun. 2008, pp. 26–33.

[2] Shouyi Yin, Xinhan Lin, Leibo Liu, Shaojun Wei, "Exploiting Parallelism of Imperfect Nested Loops on Coarse-Grained Reconfigurable Architectures", IEEE Transactions on Parallel & Distributed Systems, vol. 27, no. , pp. 3199-3213, Nov. 2016.

[3] R. Hartenstein, H. Grünbacher (Editors): The Roadmap to Reconfigurable computing Proc.FPL2000, Aug. 27-30,2000; LNCS, Springer-Verlag 2000

[4] A dynamically reconfigurable system-on-a-chip architecture for future mobile digital signal processing (1999) by Ahmad Alsolaim , Jürgen Becker European signal processing conference.

A. Bindra. Reconfigurable architectures chart a new course for DSPs. Electronic Design, pages 46–52, August 5 2002.

[5] R. Hartenstein. A decade of reconfigurable computing: A visionary retrospective. In Proc. DATE, pages 642–649, 2001.

[6] K. Compton, Architecture Generation of Customized Reconfigurable Hardware, Ph.D. Thesis, Northwestern University, Dept. of ECE, 2003.

[7] B. Mei, S. Vernalde, D. Verkest, H. De Man, and R. Lauwereins, “ADRES: An architecture with tightly coupled VLIW processor

and coarse-grained reconfigurable matrix," in Field Programmable Logic and Application. Berlin, Germany: Springer-Verlag, 2003, pp. 61–70.
[8] F. Bouwens, M. Berekovic, B. De Sutter, and G. Gaydadjiev, "Architecture enhancements for the ADRES coarse-grained reconfigurable array," in High Performance Embedded Architectures and Compilers. Berlin, Germany: Springer-Verlag, 2008, pp. 66–81.
[9] Ricardo Ferreira. Et.al, "An FPGA-based Heterogeneous Coarse-Grained Dynamically Reconfigurable Architecture", CASES'11, October 9–14, 2011, Taipei, Taiwan. Pp195-204, 2011 ACM 978-1-4503-0713-0/11/10
[10] Shuang Liang.et.al, "A Coarse-Grained Reconfigurable Architecture for Compute-Intensive MapReduce Acceleration", 10.1109/LCA.2015.2458318, IEEE Computer Architecture Letters, 1556-6056 (c) 2015 IEEE
[11] Vuk Vranjkovi.et.al, "Coarse-grained Reconfigurable Hardware Accelerator of Machine Learning Classifiers", IWEEIP 2016, 978-1-4673-9555-7/16, IEEE-2016.
[12] Amit Kulkarni.et.al, "A fully parameterized Virtual Coarse Grained Reconfigurable Array for High Performance Computing Applications", 2016 IEEE International Parallel and Distributed Processing Symposium Workshops, 978-1-5090-3682-0/16, 10.1109/IPDPSW.2016.13
[13] Dr. Seetaiah Kilaru, Hari Kishore K, Sravani T, Anvesh Chowdary L, Balaji T "Review and Analysis of Promising Technologies with Respect to fifth Generation Networks", 2014 First International Conference on Networks & Soft Computing, ISSN:978-1-4799-3486-7/14,pp.270-273,August2014.
[14] Meka Bharadwaj, Hari Kishore "Enhanced Launch-Off-Capture Testing Using BIST Designs" Journal of Engineering and Applied Sciences, ISSN No: 1816-949X, Vol No.12, Issue No.3, page: 636-643, April 2017.
[15] N Bala Dastagiri, Kakarla Hari Kishore "Reduction of Kickback Noise in Latched Comparators for Cardiac IMDs" Indian Journal of Science and Technology, ISSN No: 0974-6846, Vol No.9, Issue No.43, Page: 1-6, November 2016.
[16] A Murali, K Hari Kishore, D Venkat Reddy "Integrating FPGAs with Trigger Circuitry Core System Insertions for Observability in Debugging Process" Journal of Engineering and Applied Sciences, ISSN No: 1816-949X, Vol No.11, Issue No.12, page: 2643-2650, December 2016.
[17] Mahesh Mudavath, K Hari Kishore "Design of CMOS RF Front-End of Low Noise Amplifier for LTE System Applications Integrating FPGAs" Asian Journal of Information Technology, ISSN No: 1682-3915, Vol No.15, Issue No.20, page: 4040-4047, December 2016.
[18] P Bala Gopal, K Hari Kishore, B.Praveen Kittu "An FPGA Implementation of On Chip UART Testing with BIST Techniques", International Journal of Applied Engineering Research, ISSN 0973-4562, Volume 10, Number 14 , pp. 34047-34051, August 2015
[19] S Nazeer Hussain, K Hari Kishore "Computational Optimization of Placement and Routing using Genetic Algorithm" Indian Journal of Science and Technology, ISSN No: 0974-6846, Vol No.9, Issue No.47, page: 1-4, December 2016.
[20] N Bala Gopal, K Hari Kishore "Analysis of Low Power Low Kickback Noise in Dynamic Comparators in Pacemakers" Indian Journal of Science and Technology, ISSN No: 0974-6846, Vol No.9, Issue No.44, page: 1-4, November 2016.
[21] S.V.Manikanthan and V.Rama"Optimal Performance Of Key Predistribution Protocol In Wireless Sensor Networks" International Innovative Research Journal of Engineering and Technology ,ISSN NO: 2456-1983,Vol-2,Issue –Special –March 2017.
[22] S.V.Manikanthan and T.Padmapriya "Recent Trends In M2m Communications In 4g Networks And Evolution Towards 5g", International Journal of Pure and Applied Mathematics, ISSN NO:1314-3395, Vol-115, Issue -8, Sep 2017.
[23] Rajesh, M., and J. M. Gnanasekar. "CONGESTION CONTROL USING AODV PROTOCOL SCHEME FOR WIRELESS AD-HOC NETWORK." Advances in Computer Science and Engineering 16.1/2 (2016): 19.
[24] SHUBHANSHU GUPTA, S. KOLANGIAMMAL, T.PADMAPRIYA, "Smart Curtain Using Internet Of Things" International Innovative Research Journal of Engineering and Technology, Vol. 2, Special Issue, pp. 4-8.