International Journal of Pure and Applied Mathematics
Volume 119 No. 14 2018, 1387-1398
ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
Special Issue

ijpam.eu

DESIGN OF COARSE GRAINED ARCHITECTURE AND BIDIRECTIONAL ROUTING FOR LOW POWER AND HIGH SPEED APPLICATIONS

Yazhinian Sougoumar[1], Tamilselvan Sadasivam[2]
[1]Research Scholar, [2]Associate Professor
Department of ECE, PEC, Pondicherry, India.

ABSTRACT

In recent times, reconfiguration system is widely used due to their combination of efficiency and flexibility. Hardware based reconfiguration offers to use the same hardware for the different purpose of the optimized hardware utilization. Coarse grained (CGRA) and fine grained (FGRA) are the two types of reconfigurable architectures used in the present time. In the CGRA, the hardware functionality is specified at the word level. Likewise, the FGRA, hardware functionality is specified at a bit level. Coarse grained reconfigurable architecture (CGRA) needs many processing elements (PEs) and configuration memory for reconfiguration operation. CGRA reconfigure the processing element from one process to another process dynamically without any performance degradation. The CGRA system is designed with 16 processing elements; it's arranged in a 4X4 matrix structure. Each Processing elements are configured by more than one operation. The operation can be selected and changed by using the selection bits. According to the selection bit, the operation of the processing element is changed; likewise, reconfiguration bit is used to reconfigure the processing elements. Although the reconfiguration architecture provides higher performance and flexibility, it consumes a significant amount of power. Reducing power in the CGRA and FGRA is the very critical issue; avoid this kind of situation by proposed the clock gating and pipelining reuse techniques. The proposed techniques avoid the unwanted usage of the clock signal during the operation of processing elements. Interconnections between the processing elements are essential in the CGRA structure. Speed is one of the important issues during the configuration of the processor. Increase the speed of the processor and reducing the processing delay by proposed the routing technique named as Bi-directional routing. This technique improves the speed of interconnection link between the PEs. The proposed architecture was simulated by Modelsim 6.3c, synthesized and analyzed by Xilinx ISE simulator and Quartus with cyclone II device.

Keywords: Coarse Grained Reconfigurable Architecture (CGRA), Fine Grained Reconfigurable Architecture (FGRA), Clock gating, Pipeline reuse, Bi-directional routing, Quartus II, Xilinx ISE

INTRODUCTION

Reconfiguration provides high performance at low power. It is mostly used for mobile and embedded systems applications. Reconfigurable computing architecture is a process of joining the flexibility of software with the high performance of hardware by processing the high-speed computing like Field programmable gate array (FPGAs). FPGAs are also support the partial reconfiguration. It is the process of changing half of the portion of reconfigurable hardware, while the other part is still running. Advances in the silicon technologies, increase the number of transistors that can be integrated into a single device. Chip manufacturer uses this integration technique to fabricate the system on a single silicon device. FPGA manufacturers also using this technique to implement the DSP blocks, multipliers, a memory block and many logic elements along with the reconfigurable fabric system. The reconfigurations in the devices are reduced power consumption, flexibility and hardware reuse. Reconfiguration architecture contains fine grained and coarse grained reconfigurable architecture. Fine grained reconfigurable architecture programmable block and routing logic operated at a bit level. It is based on an array of homogeneous reconfigurable processing elements, which can be configured as logic interconnects. This design provides flexibility for distributing the hardware resources between the logic interconnects. It could obtain high performance for a huge number of algorithms, because of its bit level reconfiguration. It provides an efficient communication medium for

exchanging operations including memories. Likewise, the coarse grained reconfigurable architecture (CGRA) operated at the word level. CGRA is heterogeneous. CGRA is building an array of functional units which communicate using local and global interconnects. Functional units divided into logic and memory elements. All the functional units are dynamically reconfigurable to support the run time application. It is easier to program and fast reconfiguration when compared to fine grained reconfigurable architecture (FGRA). CGRA provides mapping flexibility to reduce the reconfiguration run time and achieve higher performance using the word level operation. Both the reconfiguration is constructed in the FPGAs for achieving high speed and low power applications. Modern FPGAs are used for much application like medical, industrial, video and image processing etc.

1. RELATED WORKS

Coarse grained reconfigurable architectures (CGRAs) need many logic elements (LEs) and a configuration memory block for reconfiguration of its LE array. It consumes more power; reducing power is very difficult for CGRA. Yoonjin Kim et al. [1] proposed the reusable context pipelining (RCP) architecture to reduce power consumption caused by reconfiguration. The characteristic of loop pipelining provides the power reduction. It is multiple instruction streams, and multiple data stream style execution model RCP efficiently reduces the power consumption in configuration memory without any performance changes. The proposed architecture is supported for low power reconfiguration and hybrid configuration. The architecture used to achieve power saving in the reconfigurable architecture; the performance is same as traditional CGRA.

The coarse grained architecture contains context memory and the data memory organizations. Performance, area and power are the parameters to consider, achieve a greater tradeoffs. Yansheng Wang et al. [2] proposed the two techniques named as hierarchical configuration context (HCC) and the lifetime based data memory organization (LDO) concentrating on the context memory and the data memory. The proposed techniques are to compress the on-chip memory space and reduce the reconfiguration time. Rahul Hiware and Dinesh padole [3] presented a configuration memory based dynamic coarse grained reconfigurable multicore architecture. CGRA is a reconfigurable system, using the byte computing. Coarse grained architecture changes the hardware configuration dynamically depending on the application. Proposed dynamic reconfigurable architecture changes its hardware based on configuration codes stored in the configuration memory.

NATURE architecture uses the concept of temporal logic folding and fine grained dynamic reconfiguration to increases the logic density. So Ting-Jung Lin et al. [4] have been proposed the fine-grain dynamically reconfigurable (FDR) that consists of an array of homogeneous reconfigurable processing elements (PEs). Each PEs can be configured into the LUTs. The proposed architecture eliminates most of the long distance and global wires, which occupy a huge area in the conventional FPGAs. Dawood Alnajjar et al. [5] proposed the novel concept named as flexible reliability in the coarse-grained reconfigurable architecture. It deals with soft errors and aging. TMR, DMR and SMS mode provides the error correction in the configuration. TMR mode provides error correction; likewise, DMR provides error detection. The aging reduces the circuit delay and power consumption by the system.

Conventional FPGAs have long reconfiguration latency. To overcome these limitation novel dynamically, reconfigurable architecture is proposed by Rakesh Warrier et al. [6]. It enables run time reconfiguration and hardware reuse. Modern FPGAs are used in computation intensive applications, coarse grain DSP blocks are needed to boost the performance. Reconfigurable DSP block design for dynamically reconfigurable architecture used for different arithmetic functions in different clock cycles. Shouyi Yin et al. [7] have been proposed the Nested loop is pipelining on CGRA. Loops are implemented on CGRA for acceleration. The nested loop pipelining used to exploit the parallelism of loops. To use of affine transformation and polyhedral model reduces the loop pipelining problem. Parallelism improves the performance of the CGRA.

Yoonjin Kim et al. [8] presented a coarse grained reconfigurable architecture (CGRA) based multi-core architecture provides high performance by kernel–level parallelism. Existing architecture suffer from power consumption and performance bottleneck. The proposed ring based sharing fabric (RSF) to boost their flexibility. Julio Oliveira Filho et al. [9] introduced an architecture description language targeted to describe the coarse grain

reconfigurable architecture template. This language allows fast modeling and analysis of the architecture. The proposed language enables a formal validation, analysis and estimation of hardware cost. The concept of reconfigurable computing and reconfigurable hardware was presented by konstantinos et al. [10] different types and classification of reconfigurable architecture is helpful for developing the architecture in the future.

2. GRANULARITY OF RECONFIGURATION

Reconfigurable systems are classified by the granularity. It shows which architecture is best for reconfiguration. Reconfiguration architecture can be classified into fine-grained, coarse-grained, medium-grained and mixed-grained. Here discuss the fine-grained and coarse grained reconfigurable architecture. Reconfigurable hardware with bit level is named as fine-grained reconfigurable hardware architecture; hardware operated with word level is called as coarse-grained reconfigurable hardware. Coarse grained architecture is easy to design and reconfigure but less flexible than fine-grained architecture. But the coarse-grained architectures are faster and more efficient because it runs in word level operation. CGRA system is trying to improve the performance and reduce the power consumption of the reconfigurable system.

In fine-grained architectures, the basic programmed building block consists of combinational and sequential circuits. The logic element in the architecture decides the operation of the reconfigurable system. Here the logic element can be programmed into a simple logic function, such as two-bit adder. These elements are connected to the reconfigurable interconnection. The task of the logic element is simple, so it provides more flexible. But comes to more critical task the performance and efficiency are very low. In case logic blocks are replaced by an 8-bit adder, the architecture provides low efficiency and occupies more space in the fine-grained implementation. Medium-grained reconfigurable systems use the logic block of large granularity. Operations are optimized in the medium-grained architecture. Because of these, the chip area is reduced when compared to fine-grained architecture.

CGRA architecture is shown in fig. 1; Coarse-grained architecture logic blocks are optimizing the large computations during the operation. It quickly performs these operations and utilizes a lesser area. Coarse-tissue construction is more efficient than fine-grained construction to perform tasks. It is implemented in a word-level operation. It requires less amount of reconfiguration information for reconfigurable hardware with fine granularity. The lower the amount of reconfiguration information that accelerates the speed of reconstruction. Reconfiguration time is short when compared to fine - grained architecture. Interconnection overhead is small in the coarse-grained architecture.

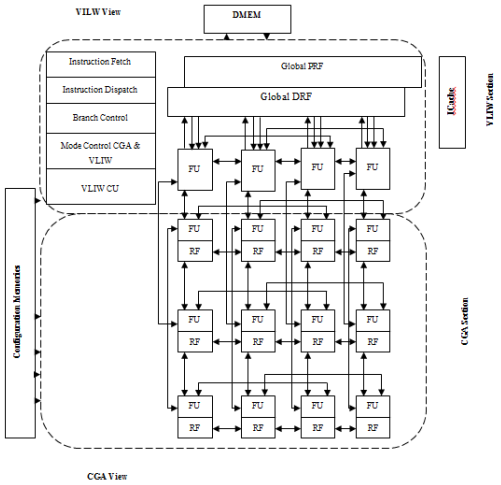

Fig.1 General CGRA Architecture

3. TYPES OF RECONFIGURATION

The Reconfiguration scheme of the reconfigurable system is divided into three categories: 1) Logic Reconfiguration, 2) Static Reconfiguration, 3) Dynamic Reconfiguration. Restructuring both logic elements and interconnections can be controlled by the transistors configuration using the SRAM memory bit. The functionality of the logic elements has been changed by obtaining a bit stream of reconfiguration data on I / O units and interconnection network hardware. The static reconfiguration changes the hardware configuration when the entire system stopping. Dynamic re-configuration allowed the process to change the hardware configuration while processing the system performance. The static reconstruction also refers to the reconfiguration of compile time. This is the simplest and most common approach for running the application. Similarly, dynamic reconstruction also refers to the reconfiguration of run time.

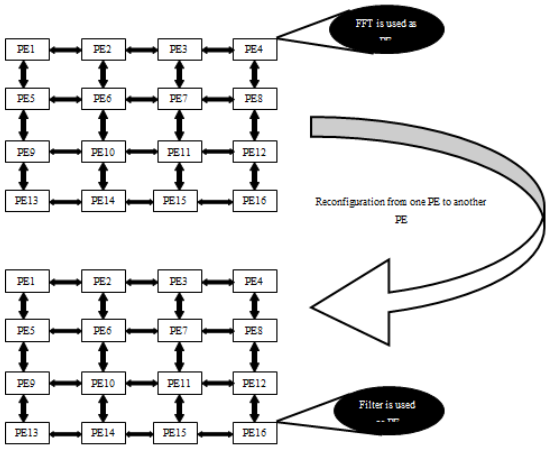

Fig. 2 Reconfiguration from one PE to another PE

It uses a dynamic allocation steps that reallocate the hardware at run time. This type of reconfiguration increases the performance, flexibility and functional density. It is based on the concept of virtual hardware. In fig. 2 shows the reconfiguration of the processing element (PE) from one operation to another operation. For example, the fourth processing element performs both the FFT and filtering operations, and it can be done by using dynamic reconfiguration without any performance degradation.

Table: 1 Comparison between Fine Grained and Coarse Grained Architecture

Characteristics	Fine Grained	Coarse Grained
Operation	Bit Level	Word Level
Flexibility	High	Medium
Performance	Medium	High
Reconfiguration Time	Large	Small
Interconnection	Long	Short

4. DESIGN OF PROCESSING ELEMENTS (PE)

Here designed the 16 processing elements into a single CGR Architecture. The CGRA system has been designed with the array of 4X4 PEs. The arrays of processing elements perform operations according to the opcode given. The lists of Processing Elements designed in the system are:

1. Data Encryption
2. Data Decryption
3. Advanced Russian Peasant Multiplier
4. FIR filters
5. Comparator
6. Fast Fourier
7. Transform (FFT) processor
8. Hamming encoder
9. Hamming decoder
10. Convolution
11. Factorial
12. Cyclic redundancy check (CRC)
13. ALU
14. QAM Modulation
15. QAM Demodulation.

These are processing elements used in the CGRA architecture.

Table: 2 List of Operations according to the Selection line

PE No.	Selection line	Operation
1	0000	Data Encryption
2	0001	Data Decryption
3	0010	Advanced Russian Peasant Multiplier
4	0011	FIR filter
5	0100	Comparator
6	0101	FFT
7	0110	Hamming Encoder
8	0111	Hamming Decoder
9	1000	Convolution
10	1001	Factorial
11	1010	Cyclic Redundancy Check
12	1011	ALU
13	1100	QAM Modulation
14	1101	QAM Demodulation

a. Data Security

In Real time all system needs security to protect the data from the attackers. Various data security techniques are available on the market; one of the efficient security techniques is Advanced Encryption Standard (AES). It is one of the commercial security algorithms used for satellite communication, net-banking, etc. It completes the operation by two processes, named as encryption and decryption. AES uses the symmetric block cipher technique. The Same key is used for both encryption and decryption operations.

b. FIR Filter

Digital Signal Processing (DSP) operations are widely used in wireless communication Technologies to control and guide the signal flows. Convolution, Correlation, Frequency Transformation and filtering are the important operations of DSP applications [14]. Finite Impulse Response (FIR) filter is used to filter the noise/unwanted signals at finite impulse durations. Multiplication and Accumulation (MAC) unit estimates the duration of periodic impulses. Designed FIR filter compromise the parameters like low power, less area and less delay while processing the signal.

c. Russian Peasant Multiplication

Modified Russian Peasant Multiplication (RPM) is the best multiplication algorithms which based on "Multiply Divide" principle. In the digital implementation perspective, the left-shift and right-shift are used to multiply and divide the 'n' bit binary data. Therefore, the shift-based digital circuit is designed to do the multiplication operation using the Russian Peasant Multiplication algorithm. Developed digital multiplication called as "Digital Russian Peasant Multiplier (DRPM)".

d. 128-Point FFT Architecture with Combined SDF-MDC Structure

To compute a 128-point FFT, Fast Fourier Transformation (FFT) is one of the digital signal processing operations used for frequency transformation. In nature, IFFT is used to analyze the frequency characteristics of discrete time domain signals. On the other hand, FFT is used to analyze the timing characteristics of discrete frequency response. The 128-point FFT architecture is designed as one of the processing element in the coarse grained architecture. This processing element operates like time to frequency signal conversion [11]. It includes Mixed Radix pipelined FFT with combined Single path delay feedback (SDF) – multipath delay commutator (MDC) structure and modified bit parallel multiplier for twiddle factor multiplication. SDF structure is used to reduce the delay in the FFT architecture [12]. Likewise, the MDC structure is used to reduce the hardware utilization and power consumption of the FFT architecture.

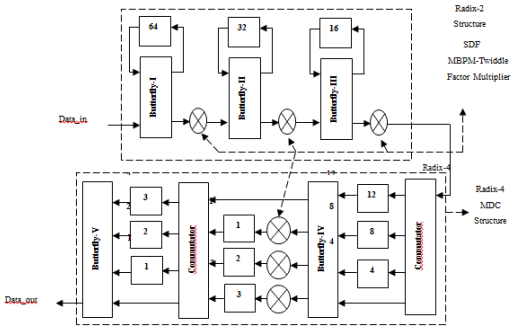

Fig. 3 Structure of 128-point new mixed radix-2 and radix-4 pipeline combined SDF-MDC FFT unit

Fig. 3 shows the combined structure of SDF and MDC. Commutator unit is used to change the position of real and imaginary input data points for optimizing or reducing the hardware complexity of the circuit. The proposed mixed radix-2 and radix-4 pipeline FFT architecture can process a continuous sequential flow of data. The proposed butterfly structures can read the both real and imaginary inputs and store both outputs in a single clock cycle. The mixed SDF-MDC structure has the less computational path and also enhances the performances of FFT processor. Developed 128-point Mixed SDF-MDC architecture has utilized to implement a single stage of Single-Path Delay Feedback, and all other stages are utilized in Multi-Path Delay Commutator.

5. CLOCK GATING TECHNIQUE

The clock signal has major sources of power dissipation because of high frequency. The clock signal is widely used for synchronization purpose; it does not compute any operations, and it does not carry any information in the logic circuits. Avoid the unnecessary clock activities in the logic circuits can be achieved by technique named as clock gating. Clock gating is used to control the power dissipation occurred by the clock net, reducing the unwanted switching on the part of the clock net by disabling the clock. It also reduces the unnecessary activities inside the gated module for saving power. Gated clock technique is easily accepted, to optimize the power and it can be applied at a system level, gate level and register transfer level (RTL). Power consumption in the circuit can be classified into dynamic power and leakage power. Leakage power is constant power

according to the device used. Dynamic power is not a constant power; it will be varied according to the process of the circuit. Clock signal dynamic energy abuse is the main source. In equilateral circuits, the clock of some sequences is disabled without affecting logic functionality. This technique is called clock gating. Clock Gating is the most effective and widely used technology for energy savings. Clock Gates performed at various levels of capture and granularities. Similarly, the power gating technique can effectively reduce the leakage power.

a. Architecture Level clock gating

This type of clock gating method is used in the system on chip design; because SoC design contains multiple processors. In this design, the clock signal of the entire processor or a particular module is disabled until it receives the request from the hardware power management unit. These are normally used while the processor is in sleep or wait for interrupt state.

b. Micro architecture level clock gating

This type of architecture is used for large architecture systems are composed of different logic blocks. Every block is mutually exclusive and it also not exploited at the same time. In the architecture many blocks are present in an idle condition, so the designers locate the idle blocks and include the clock gating technique for enabling or disabling the clock of these blocks.

c. Types of Clock Gating

There are different clock gating techniques are available in the system on chip design. These are: AND gate based technique, NOR gate based technique, latch based AND gate, latch based NOR gate, MUX based techniques are the different level of gating methods.

d. AND gate based Technique

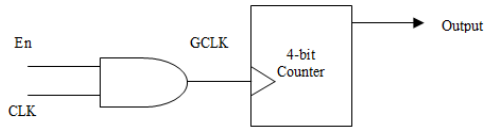

Fig. 4 AND based Clock Gating Structure

AND gate based clock gating technique is widely used in the system on chip processor, because of its simple logic. Most of the case two input AND gate is used in the design. One input of the AND gate is clock signal while the second input of the AND gate in control signal. The second input fully controls the output of the AND gate. Output clock from the AND gate is used to control the sequential system circuits.

e. NOR gate based Technique

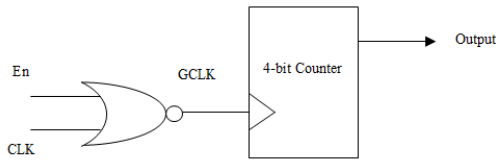

Fig. 5 NOR based Clock Gating Structure

NOR gate based technique is suitable for clock gating, but it performs the actions only on the positive edge of the global clock. Two input NOR gate is used to generate the output signal. It generates outputs when one of the inputs is positive.

6. PIPELINING REUSABLE TECHNIQUE FOR POWER REDUCTION

Coarse-grained reconfigurable architecture (CGRA) needs a lot of processing elements and a configuration memory for reconfiguration. The CGRA architecture takes more amount of power during the reconfiguration operations. Power saving is an important process in the reconfiguration approach.

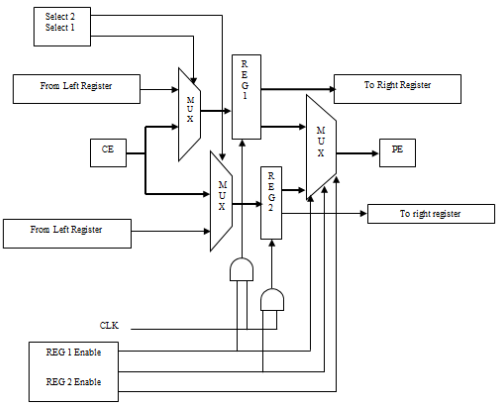

Fig. 6 Connection between CE and PE

Fig. 6 shows the connection between the processing element and cache element. The above architecture contains multiplexer for selecting input according to the selection line. Clock pipelining reusable architecture is the technique to save the power, also avoid the unwanted power usage without any performance degradation. Two mapping techniques: Spatial mapping and temporal mapping when mapping on loop construction with loop pipelining. In the spatial mapping, each PE maintains a stable operation with consistent configuration. This means that the loop is mapped to the body redevelopment line. The purpose of spatial mapping does not require rebuilding when running a loop due to the constant functionality of each PE. However, extending the loop body's all operations to a limited reconstruction range requires many resources. In the temporary mapping, PE performs multiple operations by dynamically changing configuration in a loop.

7. BIDIRECTIONAL NOC ROUTER

Network on Chip (NoC) router plays an important role in the system on chip (SoC) based applications. Normally routing operation is not easy to perform inside the Soc Chip. Soc contains millions of chip within single integrated circuits; each integrated circuit contains millions of transistor. Routing operation is important the SoC architecture, because the information transferred through routing logic. So need to design an efficient routing logic functions. NoC router consists of the following components, Network Interconnects (NI), crossbar switches, arbiters, buffer and routing logic. Unidirectional router and bidirectional router are the two types of router mostly used in the NoC architecture. In the unidirectional router operates in a single direction. It does not communicate on both sides; information travels to only one direction. Drawbacks of this routing logic are path failure, dead lock problem and live lock problem. The main aim of the routing logic creates the path between the source and the destinations. Routing logic prevents the deadlock, live lock and starvation problem. Deadlock is defined as the cyclic dependency among nodes requiring access to the collection of sources. Live lock is the process of circulating the packets to the network without ever making any progress towards their destinations. Starvation problem is occurred for the packet requesting the buffer when the output channel is allocated to another packet [16]. Routing algorithm can be classified into three criteria, a) where the routing decisions are taken, b) how the path defined, c) path length. The unidirectional router includes Round Robin arbiter, First in First out (FIFO) buffers and crossbar switches. Arbiter is used to access the data based on the priority. Higher priority data will be routed first in the architecture. A buffer is a temporary storage device. FIFO buffer is used to store the packet or data temporally. Both the input and output channels use the buffer. Data is transferred by using crossbar switches. Crossbar switches receive the control signal from the router with channel control logic [18].

A contention is one of the issues being done in the routing logic. A contention is nothing but competition for resources when two or more nodes are trying to transmit a message in the same channel at the same time. To avoid the contention situation by introducing the bidirectional network on chip router. The proposed bidirectional router consists of In Out port, static RAM, Round Robin Arbiter, Routing logic and channel control module. Arbiter chooses one output from the number of inputs based on the logic used. Arbiter present in the crossbar switch contains the same number of input and output. Data, request and destination are the three quantities come from the crossbar input. The input data will be routed at the output side, the output port address present in the destination. If two or more data send the request at the same time from the crossbar input, the round robin arbiter allows only one request at the time. This may cause the data losses, to avoid the data loss by using FIFO buffer and SRAM memory. The stored data in the memory transfer into the next clock cycle [17]. This process is called as contention free crossbar.

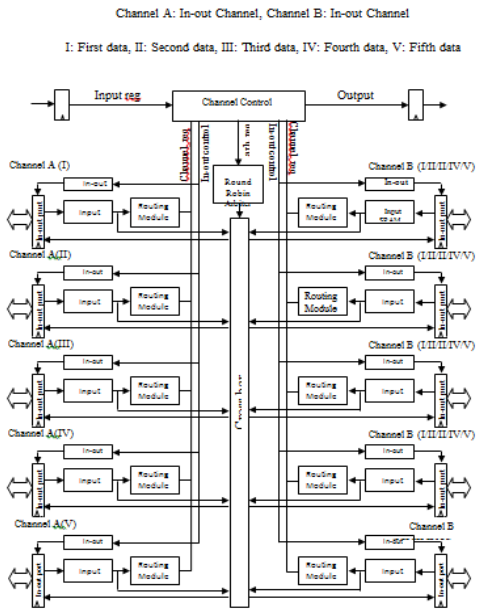

Fig. 7 Architecture of Bi-directional NoC Router

Static RAM and Dynamic RAM are the two type of memory element used in the routing logic. DRAM consists of transistor and capacitor, and it needs periodical revive to carry on the leak power of capacitor. SRAM contain more transistors, so it consumes more area, but it reduces the leakage power by increasing the number of the transistor to increase the reading capability. It provides high speed of operation when compared to the DRAM. The proposed bidirectional router is designed using the SRAM memory to speed up the router and avoid the unwanted leakage power. Virtual channel allocator and Source allocator is used to controlling the channels. Virtual channel allocator is used to virtually change the corresponding channel direction. Switch allocator is used to removing the path failures. Three methods are used to transfer the data in the bidirectional NoC router these are, all input and output channel act as master or slave. The second one is data from the input channel routes the data through same output channel; it eliminates the path failure. Third, all input data transfers through all the output port. It will remove the live lock and dead lock problems. The proposed bidirectional NoC router provides less area and high frequency than the conventional unidirectional router. The area and delay are reduced by introducing the efficient routing logic in the proposed bidirectional router.

8. RESULTS AND DISCUSSIONS

The structure of coarse grained reconfigurable architecture and fine grained reconfigurable architecture was designed by using Verilog HDL. Also, the number of processing of processing elements was done the same Verilog code. The simulation results of the CGRA system has been processed by using Modelsim XE 6.3C and those the results has been synthesized and analyzed using Xilinx ISE 10.1 and Quartus with Cyclone II device. A simulation result of the CGRA with clock gating is shown in fig. 8. Every processing element parameters are analyzed individually and shown in the table below. Table 3 shows the comparison results of conventional FFT and the proposed FFT. It explains the proposed FFT processing element is better in delay, area and power. Another processing element, Russian peasant multiplication and the FIR filter was compared with existing architecture, % reduction of delay, power; LUT and slices are shown in table 4 and table 5.

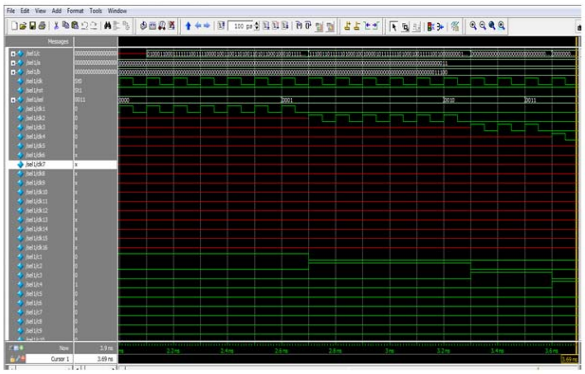

Fig. 8 Simulation Result of CGRA with clock gating architecture

Table 3: Comparison between Conventional FFT and Proposed SDF-MDC FFT

Parameters Noticed	Conventional FFT	Proposed FFT	% reduction
Look up Tables (LUTs)	10,542	10,317	2.1%
Number of occupied Slices	5,863	5,732	2.2%
Number of slice flip-flops	1,590	1,554	2.3%
Delay (ns)	14.625ns	14.18ns	3.4%
Power (W)	7.438w	6.093w	18%

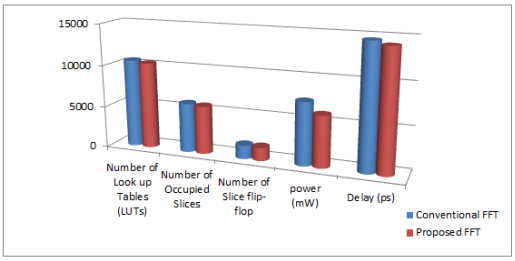

Fig. 9 Comparison graph of conventional and proposed FFT

Table 4: Comparison of Existing Multiplier with Proposed Multiplier

Parameters Noticed	Existing Russian Peasant Multiplier Design	Proposed Russian Peasant Multiplier Design	% reduction
Number of LUTs	725	687	5.2%
Number of occupied Slices	394	376	4.5%
Delay (ns)	47.848ns	46.765ns	2.2%
Power (W)	0.321w	0.303w	6.5%

Table 5: Comparison between Existing FIR filter and Proposed FIR filter

Parameters Noticed	Existing Filter	Proposed Filter	% reduction
Number of LUTs	89	67	24.7%
Number of occupied Slices	87	46	47%
Number of Slice flip-flops	94	35	62%
Delay (ns)	4.6ns	4.3ns	6.5%
Power (W)	0.252W	0.219W	13%

The parameters of the architecture like logic utilization, time period and power consumption was analyzed, and the results are taken from Altera Quartus II. Comparison results of coarse grained architecture by with clock gating and without clock gating is shown in table 6, and the graphical representation shown in fig 10. Likewise, comparison results of conventional routing and the bidirectional routing is shown in table 7, graphical representation of parameters difference shown in fig. 11. Different parameters are estimated for fine grained and coarse grained architecture, and the estimated results are shown in table 8.

Table 6: Comparison of Coarse Grained Architecture by with clock gating and without clock gating

Parameters Noticed	With clock gating	Without clock gating	% reduction
Total Logic Elements	15,056	15,568	3.2%
Total Registers	291	327	11%
Total Pins	387	389	0.5%
Total Thermal power dissipation (W)	33.013W	38.293W	13.78%
Frequency (MHz)	420.17MHz	358.55MHz	14.6%
Time Period (ns)	2.3ns	2.78ns	17.2%

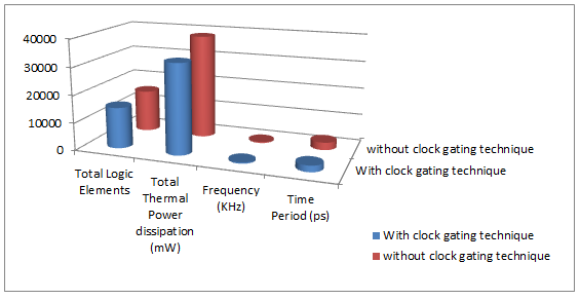

Fig. 10 Comparison Graph of CGRA with and without clock gating technique

Table 7: Comparison Between Conventional and Bidirectional Routing

Parameters Noticed	Conventional Routing	Bidirectional Routing	% reduction
Number of Slices Flip-flop	1579	1269	19%
Number of 4-input LUTs	1050	868	17%
Number of Occupied Slices	1204	981	18%
Minimum Time period (ns)	13.684ns	8.501ns	37%
Maximum frequency	73.076MHz	117.636MHz	~

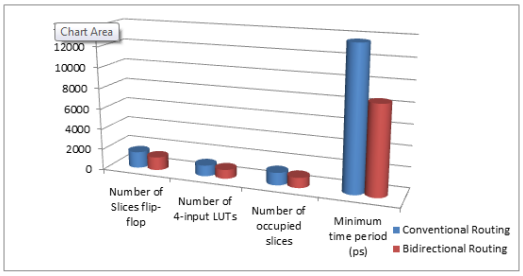

Fig. 11 Comparison graph of Conventional and Bidirectional Routing Technique

Table 8: Parameter Estimation of FGRA and CGRA

Estimated Parameters	Coarse Grained Reconfigurable Architecture (CGRA)	Fine Grained Reconfigurable Architecture (FGRA)
Total Logic Elements	14,199	14,076
Total Registers	390	256
Total Pins	387	387
Total Thermal power dissipation (mW)	238.35mW	238.49mW
Frequency (MHz)	420.17MHz	420.17MHz
Time Period (ns)	2.380ns	2.380ns
Delay (ns)	48.270ns	50.823ns

9. CONCLUSION

In this paper, the Coarse grained and the Fine grained reconfigurable architecture were designed. The flexible, reconfigurable architecture for efficient realization is suitable for applications. In the architecture, different processing elements were designed for different applications. Power consumption is very crucial for the coarse grained and fine grained reconfigurable architecture. Unwanted power consumption by the circuit can be avoided by clock gating and pipeline reuse approach. It offers 13.78% reduction of power consumption, 3.2% reduction of logic utilization and 14.6% improvement of operating frequency compared to the existing architecture. Delay is another important parameter communication between the processing elements. Delay of the circuit can be reduced by bi-directional routing method. It effectively reduced the delay occurring in the architecture. It offers 37% of delay reduced when compared to the conventional routing method. The synthesized report of the architecture is taken from Xilinx ISE and Quartus II tools.

REFERENCES

[1] Kim, Y., Mahapatra, R.N., Park, I. and Choi, K., "Low power reconfiguration technique for coarse-grained reconfigurable architecture." IEEE transactions on very large scale integration (VLSI) systems vol. 17, no. 5, pp. 593-603, 2009.

[2] Wang, Y., Liu, L., Yin, S., Zhu, M., Cao, P., Yang, J. and Wei, S., "On-chip memory hierarchy in one coarse-grained reconfigurable architecture to

compress memory space and to reduce reconfiguration time and data-reference time." IEEE Transactions on Very Large Scale Integration (VLSI) Systems vol. 22, no. 5, pp. 983-994, 2014.

[3] Padole, D. and Hiware, R., "Configuration memory based dynamic coarse grained reconfigurable multicore architecture." IEEE Region 10 Conference, pp. 1-5, 2013.

[4] Lin, T.J., Zhang, W. and Jha, N.K.,"A fine-grain dynamically reconfigurable architecture aimed at reducing the FPGA-ASIC gaps." IEEE Transactions on Very Large Scale Integration (VLSI) Systems vol. 22, no. 12, pp. 2607-2620, 2014.

[5] Alnajjar, D., Konoura, H., Ko, Y., Mitsuyama, Y., Hashimoto, M. and Onoye, T., "Implementing flexible reliability in a coarse-grained reconfigurable architecture." IEEE Transactions on Very Large Scale Integration (VLSI) Systems vol. 21, no. 12, pp. 2165-2178, 2013.

[6] Warrier, R., Hao, L. and Zhang, W., "Reconfigurable DSP block design for dynamically reconfigurable architecture." IEEE International Symposium on Circuits and Systems (ISCAS), pp. 2551-2554, 2014.

[7] Yin, S., Liu, D., Peng, Y., Liu, L. and Wei, S., "Improving nested loop pipelining on coarse-grained reconfigurable architectures." IEEE Transactions on Very Large Scale Integration (VLSI) Systems vol. 24, no. 2 pp. 507-520, 2016.

[8] Kim, Y., Joo, H. and Yoon, S., "Inter-coarse-grained reconfigurable architecture reconfiguration technique for efficient pipelining of kernel-stream on coarse-grained reconfigurable architecture-based multi-core architecture." IET Circuits, Devices & Systems vol. 10, no. 4, pp. 251-265, 2016.

[9] Oliveira Filho, J., Masekowsky, S., Schweizer, T. and Rosenstiel, W., "CGADL: An architecture description language for coarse-grained reconfigurable arrays." IEEE Transactions on Very Large Scale Integration (VLSI) Systems vol. 17, no. 9, pp. 1247-1259, 2009.

[10] Qian, Z. and Margala, M., "Low Power Split Radix FFT Processors using Radix-2 Butterfly Units", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 24, No. 9, pp. 3008-3012, 2016.

[11] Qiao, S. Hei, Y. Wu, B. and Zhou, Y., "An Area and Power Efficient FFT processor for UWB Systems", IEEE, pp. 582-585, 2007.

[12] Srivatsava, P. S. V. and Sarada, V., "Reconfigurable MDC Architecture Based FFT Processor", International Journal of Engineering Research and Technology (IJERT), Vol. 3, No. 3, pp. 1600-1604, 2014.

[13] Kakde, S. Khan, S. Dakhole, P. & Badwaik, S. " Design of area and power aware reduced Complexity Wallace Tree multiplier" IEEE International Conference on Pervasive Computing (ICPC), pp. 1-6, 2015.

[14] Lai, X. and Lin, Z. "Optimal Design of Constrained FIR Filters without Phase Response Specifications" IEEE Transactions on signal processing, Vol.62, No.7, pp.4532-4546, 2014.

[15] Adham Hadi, S. "Design of Hamming Encoder and Decoder Circuits For (64, 7) code and (128, 8) code using VHDL" Journal of Scientific and Engineering Research, Vol. 2, Issue. 1, 2015.

[16] Rajalingam, A. and Bechtel Brabi, "An Effective Bidirectional Network on Chip with Pipelining and Self Reconfigurable Channel". Research Journal of Applied Sciences, Engineering and Technology 7(22): 4714 - 4719, 2014.

[17] En-Jui Chang, Hsien-Kai Hsin, Shu-Yen Lin, and An-Yeu (Andy) Wu, "Path-Congestion-Aware Adaptive Routing with a Contention Prediction Scheme for Network-on-Chip Systems". IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 33, No. 1, January 2014.

[18] Anh T. Tran and Bevan M. Baas, "Achieving High-Performance On-Chip Networks With Shared-Buffer Routers". IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 22, No. 6, June 2014.