Designing Of Reconfigurable MPNOC On FPGA For Processing The Wireless Sensor Networks

*R.Ramachandran, J. Thomas Joseph Prakash

Abstract *- Designing of system on chip with the current algorithm and design methodology cannot meet the requirements of accommodating billion-transistor area in VLSI technology. There is a need of plat form based design and computing system design. It is to implement FPGA based reconfigurable Multiple Processor Network on Chip (MPNOC) which consists of Multiple Processing Units (MPUs),Communication controller (CC) and Memory Units (MU). The processing units are System on Chips; they are communicated each and other or connected with Routers. In this work NoC designed for processing the signals of wireless sensor networks, such as GPS, RF sensor, RFID, and Zigbee outputs. The proposed System was thus designed and simulated in ALTERA IDE's platform. In this work, the SOPC Builder component editor has been used to configure the node elements and to create Custom network interface component. In order to implement the designed Noc in FPGA chip, Altera Quartus II CAD tool was used, which compiles HDL written for configuring NoC , also generates RTL View and timing analyzer for the main components.*

Key words - MPNoC, SoC, reconfigurable Network on Chip, Wireless system, WSN

I. INTRODUCTION

Wireless sensor networks (WSNs)[1],[2],[3] consist of a large number of distributed communicating resource-constrained devices deployed to accomplishing monitoring and control goal. WSNs have some unique features, specifically handling the signals in terms of scale, communication pattern, resource level and mobility. E.g, they are typically orders of magnitude larger [4],[5] than other networks. The one more important task of the WSN is to manage the network traffic, usually it is asymmetric – mainly from sensor nodes to base stations. As a result, base stations are responsible for handling a large amount of data. Based on recent advances in WSN technology, the base stations will must have built-in computational abilities to be capable of dealing with the collected data in large-scale WSNs.In a platform based system design Network on Chip (NoC)[6],[7],[8] plays a vital role in the field of embedded system, and it is the replacement of bus based system architecture. The Noc architecture consists of Processing units, communication controllers and memory units with several routers. The routers are actually connected with one or more processing units (PU), also with neighboring routers. The NoC is thought to be future on chip interconnection. The concept of reconfigurable computing system means the processor with reconfigurable hardware. These system are having the feature of higher performance than general-purpose processors, also seems that more flexible than application specific integrated circuits (ASICs).Today Field Programmable Gate-arrays (FPGAs) are widely used in developing the applications of any sector, because of its low cost, low power consumption of novel reconfigurable hardware and high flexibility. Specifically the possibility of real-time reconfiguration of the hardware, FPGA architectures enables to introduce new ideas for adaptive hardware. Modern FPGAs support the partial dynamic run-time reconfiguration; this is the additional advantage of developing future applications demanding adaptive and flexible hardware. In this work we present network on chip concept for Wireless Sensor network whose architecture is tailored specially for handling information at wireless sensor network. NoC is intended to replace conventional processors in base stations and it can bring new levels of performance in information processing. This is the first attempt at optimizing this task and Results show that the novel architecture is at least 10 times faster than an architecture based on traditional RISC processor. The hardware/software co-design is a popular approach for accelerating various complex Algorithms and similar software applications. The time critical portion of the data processing algorithms can be implemented using hardware accelerator to reduce the processing time. FPGAs provide ideal template for run-time reconfigurable design. This paper is organized as follows 1. Related Work, 2.Proposed Network on Chip SystemArchitechture, 3.Functional Descrption, 4. Implementation and Result Analysis, 5.Conclusion.

Manuscript received December 14, 2012.

R.Ramachandran, Research Scholar, Applied Electronics – Physics (Interdisciplinary), Manonmaniam Sundaranar University, Tirunelveli, Tamilnadu, India.

Dr.J.Thomas Joseph Prakash, Assistant Professor, PG and Research Department of Physics, H.H. The Rajah's College, Pudukkottai – 1, Tamilnadu, India.

II. RELATED WORK

Wireless Sensor network plays vital role in monitoring and processing the information of the area or system. In which the studies particularly aimed at the processing information, Chu *et al* [9] introduce general information representation architecture for designing distributed inference algorithmin WSNs. The architecture comprises a graphical information representation with processing mechanisms guided by sensor

evidence and provides global view of the set of computations occurring in the system. In [10], Ganessan*et al* describes the Dimentions, a system provides a unified view of data handling in sensor networks . here, Dimension incorporates long term storage, multi resolution data access and exploits spatio-teporal correlation in sensor data. In[11], Andre Mota*etal* describes Wireless Sensor Network Processor (WISENEP), which is ASIC based processor. It is to develop FPGA based MPNOC for processing Wireless sensor Networks to optimize the information processing at base station.

III. PROPOSED SYSTEM

The Figure 1 illustrates proposed NoC, consists of heterogeneous processing elements connected in mesh network topology.

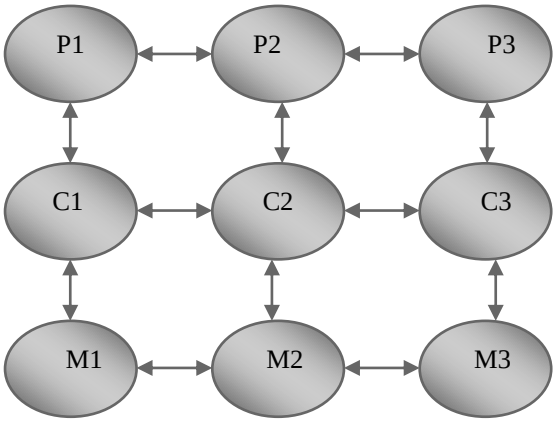

Figure 1: proposed NoC

The NoC module is designed in such a way to meet the requirements of the high speed application. In order achieve the speed, each processing units are connected with 4x4 router, which replace the bus for information sharing with other processing units, or communication controller or memory units. Modules programmed to construct the NoC system, are configured in field programmable gate array (FPGA), because it has reconfigurable property to meet the dynamically changing system demands. NoC framework contains of following components. They are processing unit, memory controller, serial communication controller, router. Each module in the NoC framework perform the following tasks;

- Decode the bit stream received from the network interface into the proper packet's fields
- Update the information stored on the memory on each received packet data; and
- Perform the computations on these information and define proper actions.

MPNOC is so designed to govern the data and to process it effectively.

The Processing units are connected to each other via mesh topology and use Wormhole (WH)[12] switching for message forwarding. Processing core (units) in each node consists of heterogeneous processing elements, such as state machine based CPU, A/D controller, serial communication controller and on-chip memory.

IV. PROCESSING UNIT (SOFTCORE PROCESSOR)

ATmega103(L)AVR core processor, which has many features, such as low-power, CMOS, 8-bit microcontroller based on the AVR RISC architecture. The AVR core is based on an enhanced RISC architecture that combines a rich instruction set with 32 general purpose working registers. The ATmega103(L) AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/ simulators. The RTL view of softcore processor is shown in Figure 2, and it is executed on the task based Finite state machine is shown in figure 3.Softcoreprocessor process the node data and send to memory controller and stored in EEPROM.

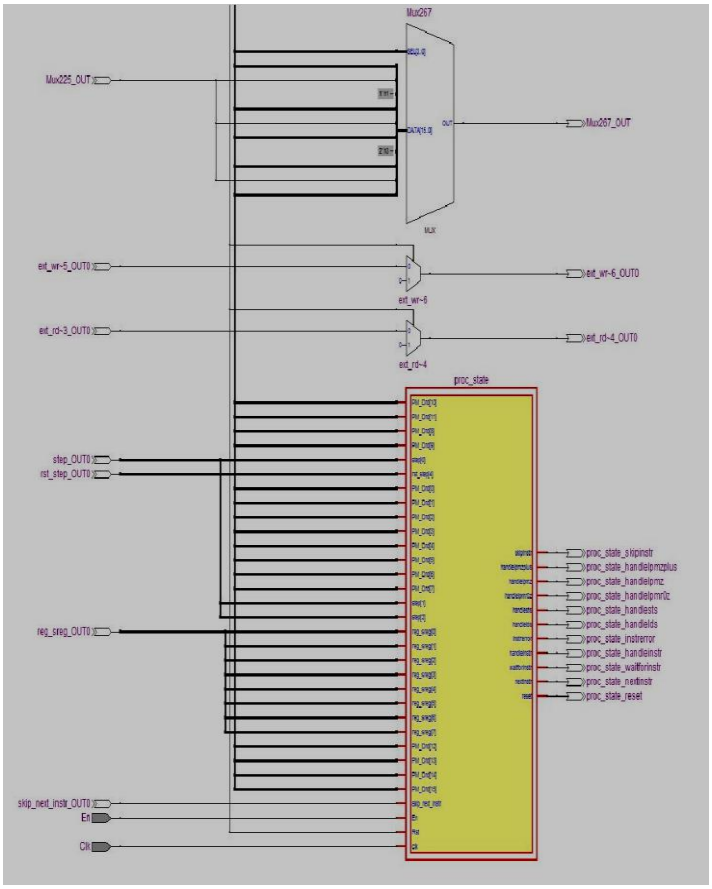

Figure 2: RTL View of the Soft core processor

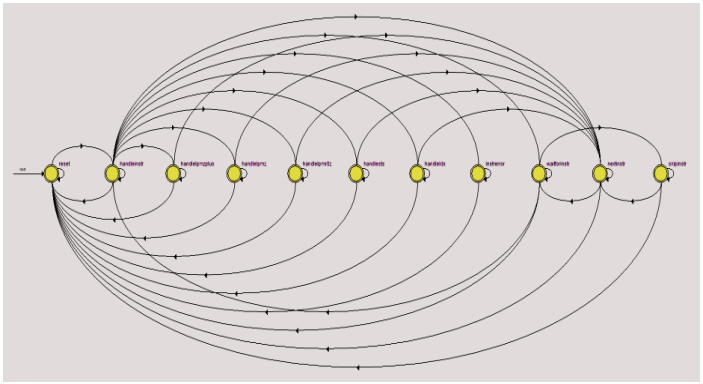

Figure 3: CPU State machine diagram

V. COMMUNICATION CONTROLLER

The communication controller is another important node in the design which consists of

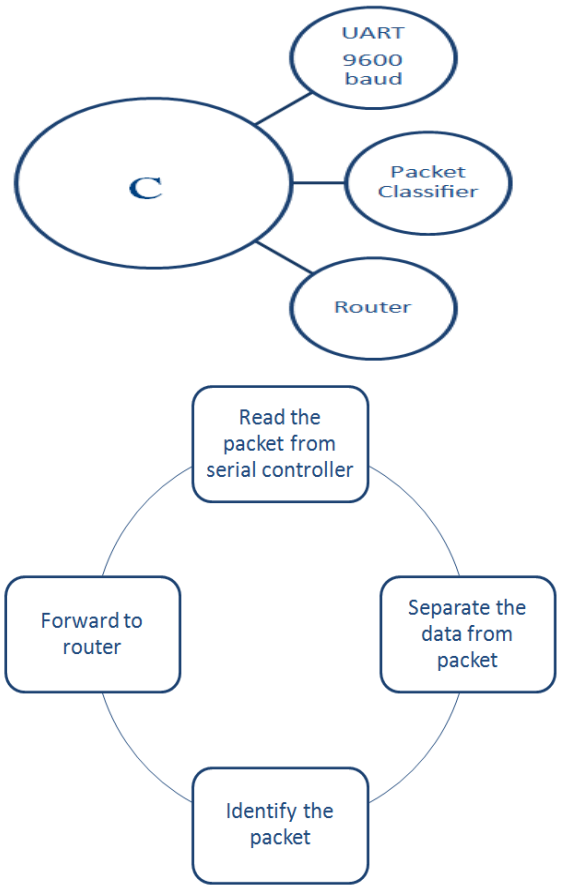

Figure (4) – Block diagram of communication controller

UART , packet classifier and router, illustrated in the figure (4) . UART controller is connected with RF module, which has many characteristics, It allows bidirectional full-duplex communication, It can communicates at a maximum speed, at the rate of 9.6 Kbps. This interface uses an "asynchronous" protocol. In asynchronous data transfer, handshake signals are used for testing the readiness of the receiver. The TxD line sends logic "1" as long as the line is idle. The start bit (logic "0") to be send before each byte of transmission. After the "start", data comes in the agreed speed and format, so the receiver can interpret it. The stop bit is usually logic "0".The common baud rates of RS 232 Serial interface are1200,9600,38400, and 115200. The speed can be easily calculated as, for example if the baud rate is 115200. T = 1/115200 = 8.7µs. If 8-bits data to be transmitted, that lasts 8 x 8.7µs = 69µs. But in the Asynchronous format of transmission, each byte requires an extra start and stop bit, so actually need of 10 x 8.7µs = 87µs. That translates to a maximum speed of 11.5KBytes per second. The Packet classifier reads the data from UART buffer, then it identifies the origin of the data, and take the decision to forward the data through router to the specified node. The router switches the data to the appropriate node.

VI. I2C CONTROLLER

I2C memory controller module used to interface I2C based EEPROM In this system, I2C based EEPROM memory used for store the node values. I2c is a two-wire, bidirectional serial bus it has two data line called SDA(serial data) and SCL(serial clock). It supports three modes of operation: I2C master controller, I2C slave controller, and an 8-bit parallel I/O (PIO) slave device. Three I2C bus transmission speeds are supported: 100 Kbps (normal), 400 Kbps (fast), and 3.4 Mbps (high speed).

VII. NETWORK INTERFACE MODULE

Network interface (NI) is an interface between network router and processing unit .It packetizes data sent by a directly connected processing unit and transmits the packet into the network through the router. Besides, it receives other packets sent by other processing unit through the router, de-packetizes the received packets and sends the data to the directly connected processing unit.

VIII. 4X4 ROUTER ARCHITECHTURE

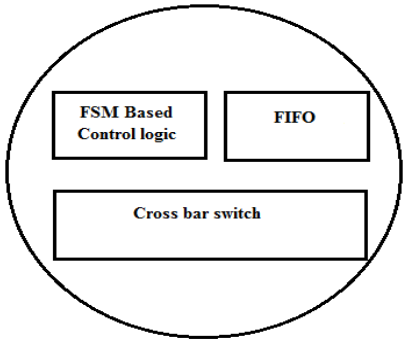

Figure 5: 4x4 Router architecture

The figure 5 shows the block diagram of 4 x 4 router architecture, which consists of FSM based control logic, FIFO, small cross bar switch, five ports east, west, north, south and local port and a central cross point matrix. Each port has its input channel and output channel. Data packet moves in to the input channel of one port of router by which it is forwarded to the output channel of other port. Each input channel and output channel has its own decoding logic which increases the performance of the router. Buffers are present at all ports to store the data temporarily. The movement of data from source to destination is called switching mechanism. The packet switching mechanism is used here, in which the flit size is 8 bits .Thus the packet size varies from 8 bits to 120bits.Every link in the network is full duplex, i.e., two messages can simultaneously travel on the link in opposite directions. A link is available for communication if its associated channel is available to accept the packet. A READY and SEND signal is used to communicate between adjacent nodes. Whenever the channel is busy it informs this to every adjacent nodes by setting the READY signal low.

A. Input channel

Figure 6 shows the block diagram of the input channel of the router.

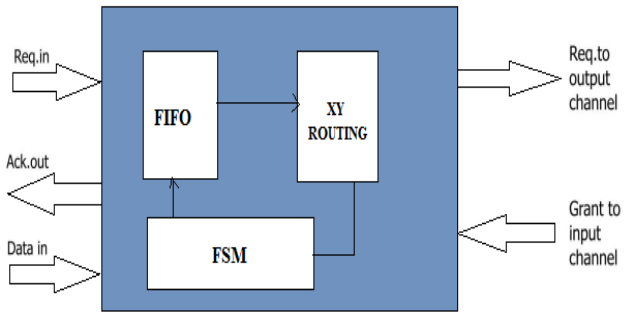

Figure 6: Input Channel

There is one input channel at each port and each has its control and decoding logic. It consists of three main parts i.e. FIFO, FSM, XY logic. FIFO is used as input buffer to store the data temporarily. The size of FIFO is 8 bits and depth is of 16 bits. The first 8 bits are the header which consists of coordinates of destination path. In this way the size of packet varies from 8 bits to 120 bits. The status of FIFO decides the communication can start or not. If the FIFO is empty the data can be written in it and communication can start.If FIFO is full, data can be read or can be forwarded to its destination router. Inside the router Grant / acknowledgement signals are used to access the FIFO. FSM controls the read and write operation of FIFO according to its status. If FIFO is empty and having space to store the data, FSM will generate acknowledgement signal in respect to the request coming to input channel, thus write operation starts. If FIFO is full or not having space to store the data, the write operation stops and the acknowledgement signal goes low. When the FIFO is full, FSM will send request to output channel of other port, if grant signal is received by it then read operation starts and continues until grant signal goes low or FIFO empties. Thus empty status of indicates the end of communication.

B. Routing method

XY routing first routes packets horizontally, towards their X coordinate, and then vertically, towards their Y coordinate. XY [13] is commonly used in NoCs XY logic is the deterministic logic which analyses the header of data and send it to its destination port. The first four bits of the header are the coordinates of destination port. In XY logic a comparator is used which compares the header of the data to the locally stored X and Y coordinate and send the packet according to its destination address.

C. Output channel

Each port of the router contains one output channel which has its control and decoding logic. It also consists of three parts i.e. FIFO, FSM and arbiter. The Figure 7 shows the block diagram of output channel.

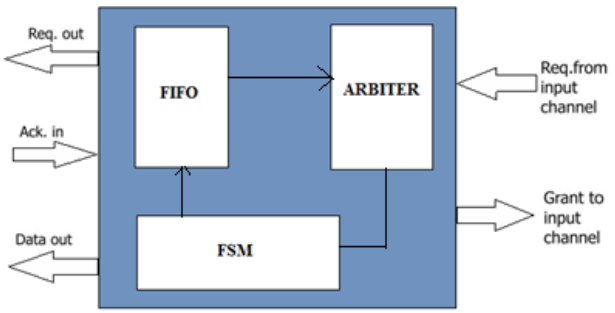

Figure 7: Output channel

FIFO and FSM are same as in input channel but in place of XY logic, arbiter is used in output channel. FIFO in output channels used as output buffer to store the data temporarily. FIFO is of size 8 bits and its depth 16 bits. The first 8 bits are the header which is the coordinates of destination router. Thus size of the packet varies from 8 bits to 120 bits. The status of FIFO decides the communication can start or not. If the FIFO is empty the data can be write and communication can start. If FIFO is full, data can be read or can be forwarded to its destination router. Inside the router the Grant/Acknowledgement signals are used to access the FIFO. The read and write operation of FIFO is controlled by FSM. FSM controls the read and write operation of FIFO according to its status. If FIFO is empty or having enough space to store the data, FSM will give acknowledgement signal in respect to the request coming from input channel, thus write operation starts. If FIFO is full or not having enough space to store the data, write operation terminates and the acknowledgement signal goes low. When FIFO is full, FSM will send request to other router, if grant signal is received by it, then read operation starts and continues until grant goes low or FIFO empties. Arbiter is used in output channel in place of XY logic in input channel. Arbiter is used to solve the problem of multiple requests coming at single output port. When there are more than one request coming from one input channels to a single output channel, arbiter selects one of the request and serve it. The Figure 8 shows the state machine diagram of output channel.

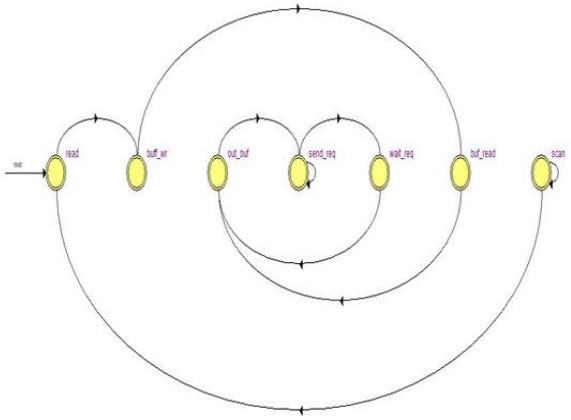

Figure 8: State machine diagram output channel

Arbiter is used in rotating priority scheme in which east has highest priority, then west, north, south and then local port. As it is rotating priority scheme, the priority of port reduces once it has been served. Thus this scheme increases the

performance of router as each port gets chance to send its data.

IX. IMPLEMENTATION & RESULTS

Thus the receiver module consisting MPNoC was designed successfully with ATMEGA 8 AVR [14] microcontroller, Communication controller, I2 C controller and router for the base station of wireless sensor networks. The MPNoC is capable of receiving the data serially from the transmitter nodes connected with appropriate sensors. The transmitter node may consists of microcontroller, A/D converter, UART controller etc.. or depends upon the need of sensor where the data has to be monitored. In the laboratory for evaluating the proposed system three avr processing core, communication controllers and i2c based memory controllers are used. Each wireless node at the transmitter end consist of ATMEGA 8 AVR microcontroller, zigbee transceiver and different analog sensor such us humidity, pressure, temperature etc. Each node has different addresses to identify the node name and each node has to send a data to receiver through zigbee transceiver serially.

X. SYNTHESIZE RESULTS

The extended RTL view of the softcore processor is shown in figure 9, and the chip planner view also illustrated in the figure 10. The Table 1 illustrates the consumption of the logic elements in the FPGA for implementing Soft core processor.

Quartus II Version	8.0Build 21505/29/2008 Sj full version
Revision name	AVR
Top level Entry name	Mcu_core
Family	Cyclone II
Device	EP2C35F672C6
Timing models	Final
Met timing requirements	yes
Total logic elements	2,075/33,216 (6%)
Total combinational functions	2,045/33,216 (6%)
Dedicated logic registers	390/33,216 (1%)
Total registers	390
Total pins	111/475 (23%)

Table 1: Compilation report

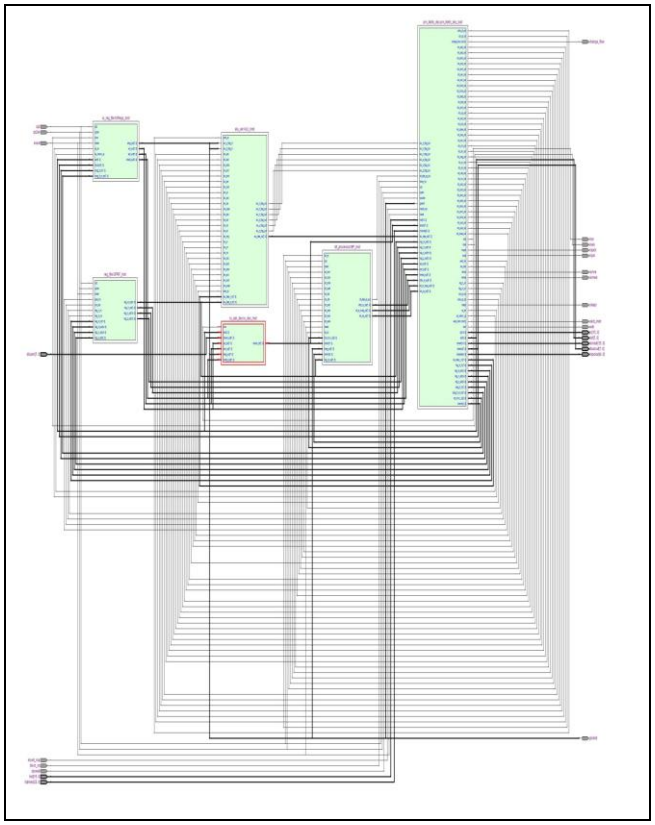

Figure 9: Extended view of RTL (processing core)

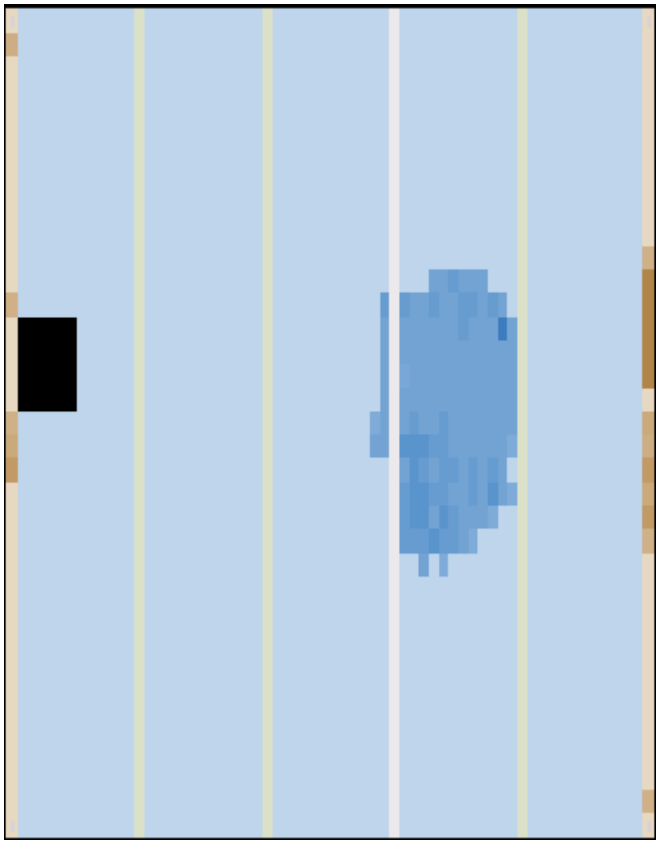

Figure 10: Chip planner of processing core

Another important module in the MPNoC is Serial Controller. Figure 11 shows the RTL View of serial controller. and compilation report shown in Table 2.

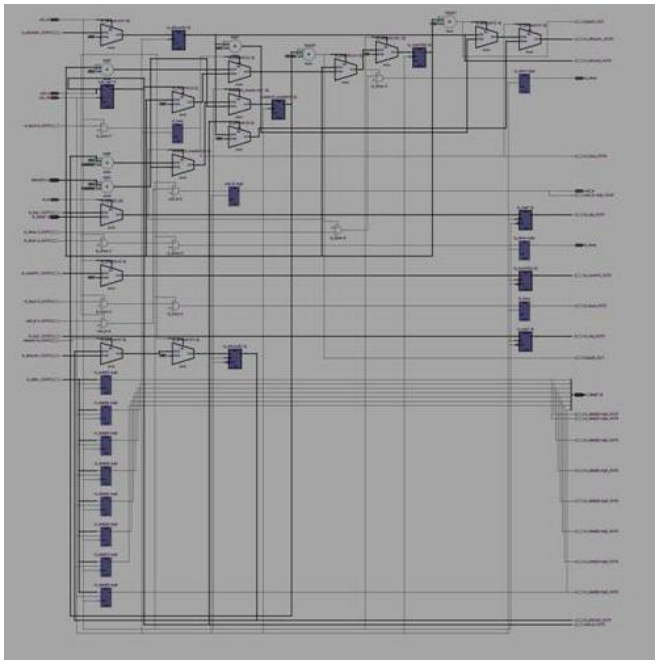

Figure 11: RTL view of the Seial controller

Quartus II Version	8.0 Build 21505/29/2008 SJ full version
Revision name	UART
Top level Entry name	UART_transceiver
Family	Cyclone II
Device	EP2C35F672C6
Timing models	Final
Met timing requirements	yes
Total logic elements	106/33,216 (<1%)
Total combinational functions	88/33,216 (<1%)
Dedicated logic registers	63/33,216 (<1%)
Total registers	63
Total pins	39/475 (8%)
Total Virtual Pins	0
Embedded Multiplier 9-bit elements	0/70 (0%)
Total PLLs	0/4 (0%)

Table (2) – Compilation report on Serial Controller

XI. CONCLUSION

Performance of proposed system is compared with ARMLPC2129 and 8051.and its results are tabulated in table 3.

Parameter	**NoC**	**ARM LPC2148**	**AT89c51**
UART Speed	**9600 -115200**	**9600**	**9600**
Core voltage	**1.8 v& 3.3v**	**3.3v**	**5v**
Onchip memory	**Configurable**	**16kb**	**256 byte**
Speed	**200 Mhz**	**Max 60 mhz**	**24 Mhz**
Number of Serial port	**User defined**	**Two**	**One**
GPIO	**User defined**	**46**	**32**
Execution	**Parallel**	**Sequential**	**Sequential**
I2C	**Configurable**	**In build**	**Not available**
Response time	**High**	**Medium**	**Low**

Table 3: Functionality Comparison of MPNoC and Microcontrollers

from this table, the illustrated system carryout the task three time faster than conventional hard core processors ARM LPC 2148 because all the node are execute the task in parallel manner and also consume less power compare to conventional microcontroller based WSN. The speed of the processor has been enhanced in this work. Functionality of the modules tested successfully. The NoC architecture can be further improved to get high speed Networks-on-chip (NoC) . The most important research areas includes, 1. NoC architectures for CMP /MPSoCs (topology, routing, switching, flow control. etc.), 2. Novel interconnect link/switch/ router designs, 3. Timing, Synchronization and ASynchronization communication, 4. Mapping of Applications on to NoCs, 5. Power, energy, and thermal issues, 6. Physical design of interconnect and NoC, etc… It is not limited in this area.

REFERENCES

[1] D. Estrin, R. Govindan, J. S. Heidemann, and S. Kumar, “Next century challenges: Scalable coordination in sensor networks,” in MobileComputing and Networking, Seattle, WA USA, 1999, pp. 263–270.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A Survey on Sensor Networks,” IEEE Communications, vol. 40, no. 8, pp. 102–114, 2002.

[3] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, “System architecture directions for networked sensors,” in Proceedings of the ninth international conference on Architectural support for programming languages and operating systems. ACM Press, 2000, pp. 93–104.

[4] V. Mhatre and C. Rosenberg, “Homogeneous vs. heterogeneous clustered sensor networks: A comparative study,” in IEEE International Conference on Communications (ICC 2004), Paris, France, June 2004.

[5] E. Altman, T. Basar, T. Jimenez, and N. Shimkin, “Competitive routing in networks with polynomial costs,” IEEE Trans. Automat. Control, vol. 47, no. 1, pp. 92-96, 2002.

[6] A. Jantsch and H. Tenhunen, Networks on chip. Kluwer Academic Publishers, 2003.

[7] Shasi Kumar et al “ A Network on chip architechture and design methodology” proceedings of the IEEE computer Society Annual Symposium on VLSI, 2002.

[8] T. Bjerregaard and S. Mahadevan, “A survey of research and practices of network-on-chip,” ACM Comp. Surveys, vol. 38, no. 1, pp. 1–51, Mar. 2006.

[9] M. Chu, S. K. Mitter, and F. Zhao, “An information architecture for distributed inference on adhoc sensor networks,” in Forty-first Annual Allerton Conference on Communication, Control, and Computing, Monticello, USA, 2003, oct.

[10] D. Ganesan, D. Estrin, and J. Heidemann, “Dimensions: why do we need a new data handling architecture for sensor networks?” SIGCOMM Comput. Commun. Rev., vol. 33, no. 1, pp. 143–148, 2003.

[11] Andre Mota, Leonardo B. Oliveira “WISENEP: A Network Processor for Wireless Sensor Networks”.

[12] William James Dally and Brian Towles (2004). "13.2.1". Principles and Practices of Interconnection Networks. Morgan Kaufmann Publishers, Inc.

[13] Mohapatra, P. Wormhole routing techniques for directly connected multicomputer systems. ACM Computing Surveys, 30(3), Sep. 1998, pp 374-410.

[14] http://www.atmel.com/images/doc2486.pdf.

Mr. R.Ramachandran, M.Sc., M.Phil., M.C.A., P.G.D.C.A., had been worked as a Assistant Professor in Thanthai Hans Roever College, Perambalur – 621 212., Tamilnadu for ten years. Author has completed his post graduation in National Institute of Technology, Trichy – 15 (Formerly R.E.C., Trichy), Tamilnadu. His research areas are Embedded systems, and VLSI design, he published papers in the various international journals in these fields. Presently, he is the Research scholar in Manonmaniam Sundaranar University, tamilnadu, India.

Dr .J.Thomas Joseph Prakash, M.Sc. ,M.Phil. M.Ed.,M.Phil.(Edn).,Ph.D. has been working as Assistant Professor in the Department of Physics, H.H. Rajah's College(Autonomous), Pudukottai, Tamilnadu, India. Author has 16 years of Teaching experience in handling U.G and P.G. of various subjects in the discipline of Physics. He is the member in International Association of computer Science and Information technology, and in Indian Association of Crystal Growth. His area of interest is Crystal Growth and Techniques, and another area of interest is Wireless communication. He Published numerous international papers and national level papers in these fields. Now he is guiding Ph.D research scholar in the emerging area of Crystal Growth and Applied Electronics.