ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

An Implementation of CLA Adder and SAD Algorithm in Foldedtree Architecture

Binny Thomas

M. Tech Scholar (VLSI & Embedded Systems), Dept. of ECE, IIET, M. G. University, Kottayam, Kerala, India

ABSTRACT: Awireless sensor networks has many applicationswhich contains sensor nodes.Sensor nodes are battery driven and they are operating on an extremely low energy budget. They must have a lifetime on the order of months to years, since battery replacement is not an option for sensor networks, because the network containsthousands of physically embedded nodes. The radio communication exhibits the highest energy consumption in these nodes. Mainly the data communications are on-the-node computations. Currently, they are designed with low-power microcontrollers. By employing a more appropriate processing element or method, the energy consumption can be reduced. Here describes the design and implementation of the new folded tree architecture for on-the-node data processing, using parallel prefix operations and also implementing CLA adder and SAD algorithm in foldedtree architecture.

KEYWORDS: wireless sensor nodes; parallel prefix operations; CLA adder; SAD algorithm

I. INTRODUCTION

WSN nodes consist of four subsystems: a computing subsystem consisting of a microprocessor or microcontroller, a communication subsystem consisting of a radio for wireless communication, a sensing subsystem that links thesensor node to the physical world and it contains a group of sensors and actuators, and a power supply subsystem. Since theradio transmissions are very expensive in terms of energy and it should be kept in minimum in order to extend node lifetime [1].The goal is to design an area efficient folded tree architecture for WSN and implementing a CLA adder and SAD algorithm for motion estimation system in foldedtree architecture.

Most arithmetic operations such that multiplication and division are implemented using several addition and subtraction steps. Thus, improving the speed of addition will improve the speed of all other arithmetic operations and reducing the carry propagation delay of adders is the great importance. Different logic designs are employed to overcome this carry propagation problem. One widely used approach employ the use of carry look-ahead principle to solve this problem by calculating the carry signals in advance, on thebasis of input signals. This type of adder circuits are known as carry look-ahead adder (CLA adder). In the digital design world, the parallel prefix operations are best suited for the application in the class of carry look-ahead adders.

Recently, digital video systems, such as video-on-demand, video-phone, distance learning and digital TV, have become popular products because of their convenience and high quality. The advanced motion compensation techniques, such as various block size, multiframe reference and quarter sub-pixel accuracy motion estimation, uses to improve the coding performance. The variable block size motion estimation (VBSME) adoptssome fast algorithms to reduce the computational complexity. Most of these fast motion estimation (FME) algorithms are only for software implementation, but a few of them can be realizable in hardware implementation, because their searching routines are irregularity and unpredictable flow. The very large scale integration (VLSI) implementation of VBSME has been proposed in recent years .The Sum of Absolute Differences (SAD) algorithm for motion estimation is the one of the most time consuming and compute intensive part of image registration in Automated Video Surveillance (AVS) using non-stationary cameras and video compressionin Automated Video Surveillance (AVS).

II. RELATED WORK

The sensor node is a battery driven and they are operates in an extremely limited energy budget. So they must have a lifetime of months to years, because battery replacement is not an option for networks with thousands of physically

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

embedded nodes[2].There is no well-defined method to evaluate the performance and energy consumption of nodes. The historically used EPI (Energy Per Instruction) and MIPS (Millions of Instructions Per Second) metrics cannot provide an accurate comparison because they are depend on the nature of instructions, which differ across instruction set architectures. That is the current well-defined benchmarks do not represent typical workloads of sensor network systems, and they are not suitable to compare sensor processors. To address this problem, propose a set of basic application compositions that represent a duty cycle of a sensor processor. Furthermore, three new metrics [3], EPB (Energy Per Bundle), CFP (Composition Foot Print) and xRT (times Real-Time) are introduced to evaluate and compare such systems. The parallel prefix or scan operation is a versatile primitive and a basic building block in parallel algorithms for a variety of different problems[4]. Use three variants of this model: a) half-duplex where each communicating PE can either send or receive a message, b) telephone model, where a bidirectional communication is done between thematched pair of PEs, and c) full-duplex where a PE can send data to one PE and received data from a different PE simultaneously. They are Binomial tree, Simultaneous binomial tree, Pipelined binary tree, Doubly Pipelined binary tree.

A asynchronous processor architecture called BitSNAP that utilizes bit-serial datapaths with a dynamic compression to yield extremely lowenergy consumption. Based on Sensor Network Asynchronous Processor (SNAP) ISA, the BitSNAP [7] can reduce datapath energy consumption by 50% compared to a parallel word processor and still providing performance suited for the lowenergy sensor network nodes.Sensor network processors introduce a level of compact and portable computing. This small processing system is reside in the environment which they monitor, computation, storage,power supplies, communication, andcombining sensing into small form factors. The rapid decreases in size, cost andpower consumption, there is a dramatic increases in performance of sensing,communication and computation are led to the development of Smart Dust which is a millimeter scale autonomous systems. Smart Dust nodes contain one or more sensors, sensor interface circuits including an ADC,digital control and processing, wireless communication, energy storage and a power source, integrating these complete, complex systems into a millimeter scale volume[8].The growing wireless sensor network research field there is a lacks of standard method for evaluating in hardware platforms. Traditional benchmark suites do not sufficiently address the needs of sensor network designers. The TinyBench [9] provides motivation for a benchmark suite for benchmarking TinyOS compatible hardware devices.

III. PROPOSED METHOD

A. *Description of the Folded Tree architecture:*

Common on-the-node operations are performed on the input data collected directly from the node's sensors or through filtering, fitting, sorting, and searching.These types of algorithms can be expressed in terms of parallel prefix operations as a common denominator.The prefix operations can be calculated in a number of ways but we chose the binary tree approach because its flow matches with the desired on-the-node data processing. In binary tree the input data flows from the leaves to the rootthrough processing elements (PEs). But the binary tree implementation of Blelloch's approach [5] costs a significant amount of area as n inputs require $p = n - 1$ PEs.

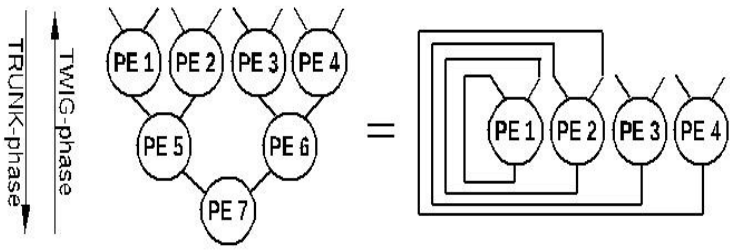

Fig.1. A binary tree (left, 7 PEs) and a folded tree (right, 4 PEs)

For reducing the area and power a new folded tree architecture is developed from binary tree.The idea presented here is to fold the binary tree back to itself for reuse the PEsmaximally. In doing so, p becomes proportional to n/2 and the area is cut in half. Note that also the interconnections are reduced. This newly proposed folded tree topology is

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

functionally equivalent to the binary tree (Fig.1).Then showing how Blelloch's generic approach for parallel prefix operator can run on the folded tree. As an example, the sum-operator is used to implement a parallel-prefix sum operation on a 4-PE folded tree, at first, the trunk-phase is considered. In this inputs are moves from leaves to root. The twig phase is just opposite used to check the results. The binary tree requires 7 PEs and a folded tree needs only 4 PEs, but they are functionally equivalent.

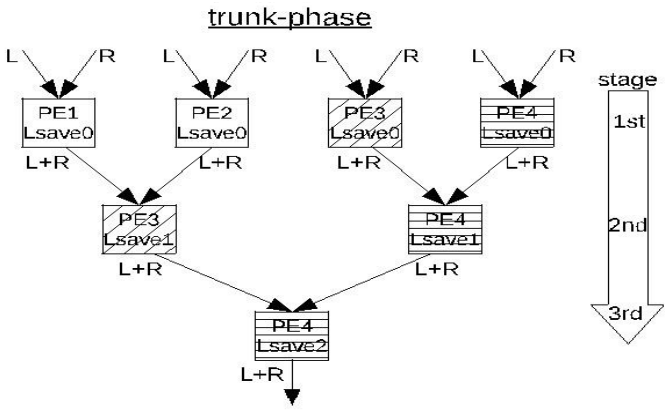

Fig.2. The trunk-phase prefix-sum on a 4-PE folded tree

The trunk-phase of a 4-PE folded treeis shown in *Fig.2* .The letters L and R is to indicate the left and right input value of inputs A and B. According to the Blelloch's approach, L is saved as Lsave and adding with right value then passing to root. Implications of using a folded tree (4-PE folded tree): some PEs must keep multiple Lsave's that is PE3 and PE4 are hatched differently.The folded tree functions as a binary tree, only difference in the case of number of PEs. As can be seen, PE1 and PE2 are used only once, PE3 is used twice and PE4 is used three times. This means that the decreasing number of active PEs while progressing from stage to stage. The first stage has all four PEs active. The secondstage has two active PEs,the PE3 and PE4. The third and last stage has only one active PEthat isPE4. More importantly, it can be seen that PE3 and PE4 have to store multiple Lsave values. PE4 can keep three, Lsave0 through Lsave2. The PE3 keeps two, Lsave0 and Lsave1 and PE1 and PE2 each only keep one that is Lsave0.

B. *Carry look-ahead (CLA) adders:*

In the digital design world the prefix operations are best known for their application in the class of carry look-ahead adders [6]. The addition of two inputs A and B in this case consists of three stages [1] : a bitwise propagate-generate (PG) logic stage, a group PG logic stage and a sum-stage areshown in Fig. 3.The output of bitwise stage is Pi=Ai xor Bi and Gi=Ai.Bi, Output of group PG logic is Ci+1=Gi + PiCi and the output of sum stage is Si =Pi xor Ci. Implementing the group PG stage in foldedtree architecture for reducing the number of PE and the area.

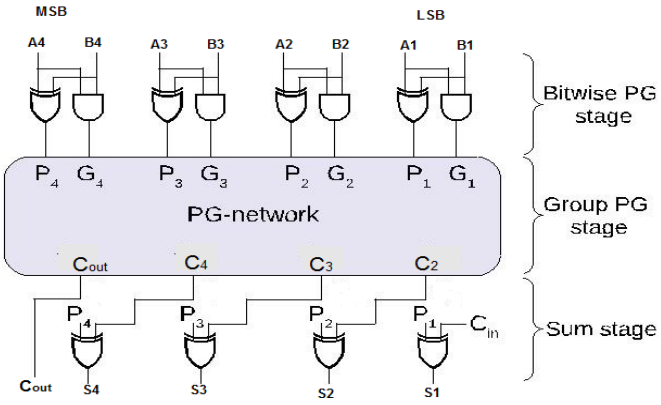

Fig.3. Addition with propagate-generate (PG) logic

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

C. *Motion estimation system:*

The advanced motion compensation techniques, such as various block size, multiframe reference and quarter sub-pixel accuracy motion estimation are uses to improve the coding performance. The variable block size motion estimation(VBSME) adoptssome fast algorithms.The SAD algorithm is widely used method for the motion estimation. In the process of video coding, the similarities between video frames could be used to achieve the higher compression ratio.

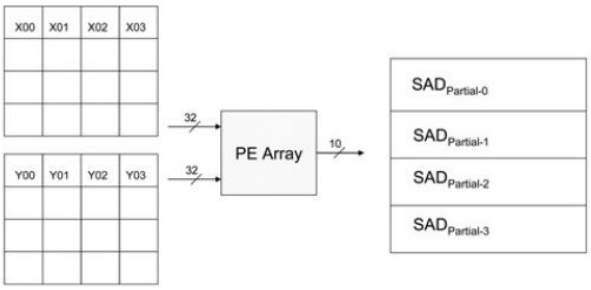

Fig. 4PE array access from buffers

The motion estimation on block based is performed on a set of pixels, each frame is divided into equally sized blocks. For each block in the current frame, there is a search for most resembling block in the reference frame and searching the whole reference frame for each block in the current frame makes this task computationally intensive. After the best matches are found for the current block and reference block the motion vectors are stored along with the SAD values[10]. To realize a regular VLSI architecture for VBSME, the design employs the base of the primitive 4 × 4 block.

The input pixels are from the temporal buffers with 32 bits, as shown in Fig. 4. Thus, this system can read four pixels with row-by-row per cycle. The partial SAD value is accumulated with an accumulator (ACC). After four cycles, one can achieve a complete SAD for one motion vector of the 4 × 4 block.The motion estimation on a video sequence using SAD uses the current video frame and a previous frame as the reference frame.

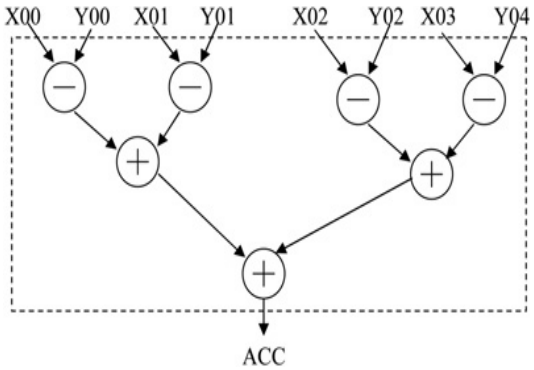

Fig. 5 Computational kernel of PE processing array

The two frames are compared pixel by pixel, summing up the absolute values of the differences of each of the two corresponding pixels. The result is a positive number that is used as the score. SAD reacts very sensitively to even small changes within a scene. Fig. 5 illustrates the computational kernel of the PE array with four PE [10]. The current coding pixels X00, X01, . and the reference pixels Y00, Y01 are read in parallel. With 4 subtractions and 3 additions, one can obtain the partial SAD from one row per cycle.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

IV. SIMULATION RESULTS

Xilinx ISE is a software tool produced by Xilinx for synthesis and analysis of HDL designs, which enables the developer to synthesize (compile) their designs, perform timing analysis, examine RTL diagrams and implement with FPGA kit.The following figure shows the Simulation Result of binarytree with 8 inputs and one output. There are 7 PEs numbered from pe1 to pe7 are required to produce the desired output.

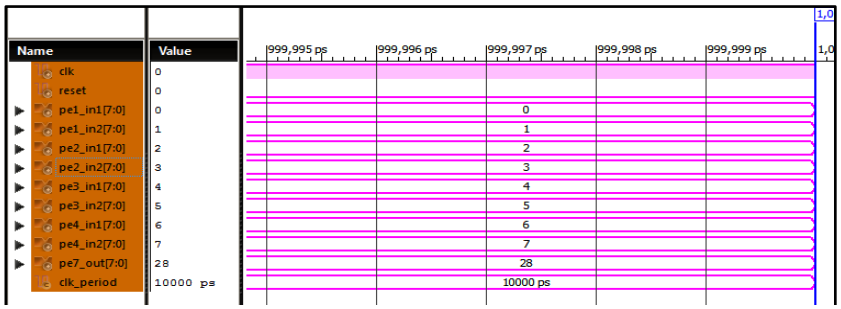

Fig.6 Simulation Result of binarytree

The Simulation Result of folded tree is shown below (Fig.7). Here also 8 inputs and one output but we can see that output of 7th PE of binary tree and the output of the 4th PE of folded tree are same. That is only 4 PE are required for folded tree and area is reduced compared to binary tree.

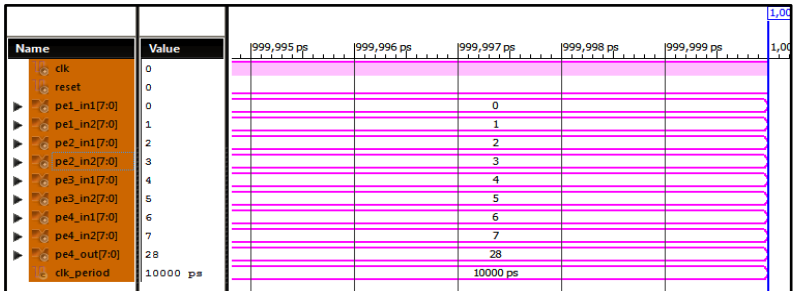

Fig.7 Simulation Result of foldedtree

Then comparing binary tree with foldedtree in the case of their input/output bits, number of PEs required and area that is number of slice FFs. It is clear that the number of PEs and area is less for folded tree compared to binary tree architecture.

Design	I/O bit	No: of PE's	Area (no: of slice FF)
Binarytree	8	7	66
Foldedtree	8	4	42

Table 1 comparison of binary and folded trees

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

The simulation result of carry look ahead adder(CLA adder) in folded tree with two inputs and one outputis given below.Both inputs(a and b) are 16 bits and the output sum_out is 17 bit,one bit for carry.The group PG stage of CLA adder contains many PEs,this can be reduced by using foldedtree.

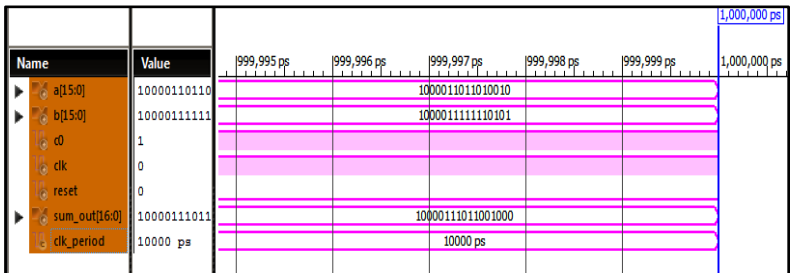

Fig.8 Simulation Result of CLA adder in foldedtree

The group PG stage of CLA adder contains many PEs and most of the area is consumed here. So implementing this stage on folded tree architecture we can reduce the number of PEs. In the case of 16 bit input CLA adder 15 PEs are required for binary tree implementation, but only 8 PE are required for a folded tree implementation .Thus area is also reduced in the case of foldedtree.Table 2 shows the comparison of CLA adder in binary and folded trees in the case of their input bits ,number of PE required and area.

Design	Input bit	No: of PE's	Area(no: of slice FF)
CLA adder in binarytree	16	15	55
CLA adder in foldedtree	16	8	50

Table 2 comparison of CLA adder in binary and folded trees

For motion estimation mainly SAD(sum of absolute difference) algorithms are preferred. In this finding the absolute difference of input pixel values and summing these results to produce final partial SAD value.The Fig.9 shows the simulation result of SAD algorithm for one row. It is clear that for same input and reference values the output partial SAD value is zero. In the case of different frame values there is a partial SAD output that is a motion is caused.

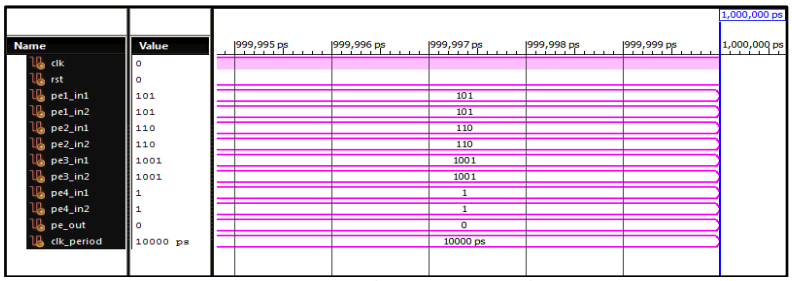

Fig.9 simulation result of SAD algorithm for one row

V. CONCLUSION AND FUTURE WORK

The folded tree architecture is used to limit the data set by pre-processing with parallel-prefix operations, to reuse of the binary tree as a folded tree, and the retaining sufficient flexibility. The simplicity of the programmable PEs that constitute the folded tree network resulted in fast cycle time,high integration and lower power consumption. This foldedtree architecture is implemented in carry look ahead adders for performance evaluation and it also implemented

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

in an application level such as motion estimation in compressed video stream. The WSN application such as Medical Monitoring, Environmental sensing, Industrial inspection, Military surveillance in folded tree isthe future work.

REFERENCES

1. Cedric Walravens, and Wim Dehaene,"Low-Power Digital Signal Processor Architecture for Wireless Sensor Nodes," IEEE Transactions On Very Large Scale Integration (VLSI) Systems.,vol. 22,no: 2, pp. 313- 321, Feb. 2014.
2. V. Raghunathan, C. Schurgers, S. Park, and M. B. Srivastava, "Energyaware wireless microsensor networks," IEEE Signal Process. Mag.,vol. 19, no. 2, pp. 40–50, Mar. 2002.
3. L. Nazhandali, M. Minuth, and T. Austin, "SenseBench: Toward an accurate evaluation of sensor network processors," in Proc. IEEE Workload Characterizat. Symp., Oct. 2005, pp. 197–203.
4. P. Sanders and J. Träff, "Parallel prefix (scan) algorithms for MPI," in Proc. Recent Adv. Parallel Virtual Mach. Message Pass. Interf., 2006, pp. 49–57.
5. G. Blelloch, "Scans as primitive parallel operations," IEEE Trans. Comput., vol. 38, no. 11, pp. 1526–1538, Nov. 1989.
6. N. Weste and D. Harris, CMOS VLSI Design: A Circuits and Systems Perspective. Reading, MA, USA, Addison Wesley, 2010.
7. V. N. Ekanayake, C. Kelly, and R. Manohar, "BitSNAP: Dynamic significance compression for a lowenergy sensor network asynchronous processor," in Proc. IEEE 11th Int. Symp. Asynchronous Circuits Syst., Mar. 2005, pp. 144–154.
8. B. A. Warneke and K. S. J. Pister, "An ultra-low energy microcontroller for smart dust wireless sensor networks," in Proc. IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers. Feb. 2004, pp. 316–317.
9. M. Hempstead, M. Welsh, and D. Brooks, "Tinybench: The case for a standardized benchmark suite for TinyOS based wireless sensor network devices ," in Proc. IEEE 29th Local Comput. Netw. Conf., Nov. 2004, pp. 585–586.
10. S.C. Hsia, P.Y. Hong, "Very large scale integration (VLSI) implementation of low-complexity variable block size motion estimation for H.264/AVC coding", IET Circuits Devices Syst., 2010, Vol. 4, Iss. 5, pp. 414–424
11. C. Walravens and W. Dehaene, "Design of a low-energy data processing architecture for wsn nodes," in Proc. Design, Automat. Test Eur. Conf. Exhibit., Mar. 2012, pp. 570–573.
12. H. Karl and A. Willig, Protocols and Architectures for Wireless Sensor Networks, 1st ed. New York: Wiley, 2005.
13. J. Hennessy and D. Patterson, Computer Architecture A Quantitative Approach, 4th ed. San Mateo, CA: Morgan Kaufmann, 2007.
14. S. Mysore, B. Agrawal, F. T. Chong, and T. Sherwood, "Exploring the processor and ISA design for wireless sensor network applications, in Proc. 21th Int. Conf. Very-Large Scale Integr. (VLSI) Design, 2008, pp. 59–64.
15. J. Backus, "Can programming be liberated from the von neumann style?" in Proc. ACM Turing Award Lect., 1977, pp. 1–29.
16. G. E. Blelloch, "Prefix sums and their applications," Carnegie Mellon Univ., Pittsburgh, PA: USA, Tech. Rep. CMU-CS-90, Nov. 1990.
17. M. Hempstead, J. M. Lyons, D. Brooks, and G.-Y. Wei, "Survey of hardware systems for wireless sensor networks," J. Low Power Electron., vol. 4, no. 1, pp. 11–29, 2008.
18. V. N. Ekanayake, C. Kelly, and R. Manohar "SNAP/LE: An ultra-lowpower processor for sensor networks," ACM SIGOPS Operat. Syst. Rev. ASPLOS, vol. 38, no. 5, pp. 27–38, Dec. 2004.

BIOGRAPHY

Binny Thomas is an M-Tech scholar in VLSI and Embedded System in the Electronics and Communication Department, IIET, M. G. University. She received B-Tech degree in 2009 from M. G. University, Kottayam, Kerala. Her research interests are VLSI, communication field etc.