Low Latency, High Throughput, and Less Complex VLSI Architecture for 2D-DFT

Sohil Shah, Preethi Venkatesan, Deepa Sundar, and Muniandi Kannan
Department of Electronics Engineering, Anna University, India

Abstract: *This paper proposes a pipelined, systolic architecture for two- dimensional discrete Fourier transform computation which is highly concurrent. The architecture consists of two, one-dimensional discrete Fourier transform blocks connected via an intermediate buffer. The proposed architecture offers low latency as well as high throughput and can perform both one- and two- dimensional discrete Fourier transforms. The architecture supports transform length that is not power of two and not based on products of co-prime numbers. The simulation and synthesis were carried out using Cadence tools, NcSim and RTL Compiler, respectively, with 180 nm libraries.*

Keywords: *Digital signal processing chip, discrete Fourier transforms, systolic array, and very-large-scale integration circuit.*

Received February 22, 2008; accepted June 8, 2008

1. Introduction

The discrete Fourier transform has a vital importance in various domains like signal processing, image processing, communication and various other areas of science and engineering. In particular, two dimensional Discrete Fourier Transform (DFT) has a variety of applications in image processing domain like filtering, compression, pattern matching, magnetic resonance imaging and watermarking. Hence efficient implementation of the DFT architecture has become a challenge. Since many of these applications process real time data, computational speed has to increase with low latency. Fast Fourier Transform (FFT) based architectures were designed to acquire computational speed [1]. But they had many disadvantages such as the requirement of transform length N to be a power of some fixed radix which limits the choice of reachable values of N and the increase in computational complexity and latency in pipelined FFTs with the increase in the transform length N. Other architectures has been studied for years, among which systolic architecture is preferred as it is simple and offers rhythmic processing with repetitive elements or identical cells called processing elements. Hence there is a need for the development of a high performance systolic architecture for two- dimensional DFT that work on transform lengths that is not power of two.

In this paper, a less complex systolic architecture for the computation of 2d-DFT is proposed based on an algorithm which facilitates the computation of DFT for any transform length N, .i.e., N can be a prime number and need not be a power of any fixed radix. High throughput is achieved by using pipelining concept where 2d-DFT is computed via two pipelined 1d-DFT blocks provided with an intermediate buffer.

In Section 2, the previous works related to systolic architectures of DFT are summarized. Section 3 gives a mathematical framework for 1d-DFT computation for transform length N via four inner products of real valued data of length approximately equal to $N/2$. Section 4 discusses the proposed VLSI architecture which includes one-dimensional DFT block, intermediate buffer, floating point adders and multipliers. Implementation aspects and results for the proposed architecture are given in section 5 and 6.

2. Related Work

A variety of systolic array designs have been proposed for the direct computation of DFT [6, 3]. These designs offer simple architecture and can be used for any transform length N but they are computationally complex as they work on the direct algorithm of DFT where the arithmetic operations per DFT computation is of $O(N^2)$. Hence as the transform length N increases they offer a computation intensive architecture which does not support direct calculation of two dimensional, 2d-DFT.

Since computation of 2d-DFT from 1d-DFT requires a transposition operation, latency is reduced and hence transposition free two dimensional DFT architectures were developed [5, 7]. But this has been achieved at the cost of increased hardware and reduced performance which is undesirable. Computationally efficient architecture has been designed using two level transform factorization which provides low latency and high throughput [9, 10]. But this design is not

applicable for odd values of *N* and hence offers less choice in choosing *N*.

A reduced complexity algorithm that operates on any transform length *N* for 1-D DFT is given in [7].Based on that algorithm, a two dimensional architecture offering low latency and high throughput has been proposed in this paper where a dual port RAM is used in between two 1-D blocks. The dual port RAM is controlled in such a way that pipelining is achieved thereby reducing latency and increasing throughput.

3. Mathematical Background

3.1. General Formulation

DFT of an array {f (n, m) for n, m=0, 1, 2… N -1} of size (N x N) is defined as [8],

$$F(k,l) = \sum_{n=0}^{N-1}\sum_{m=0}^{N-1} f(m,n) W_N^{ln} W_N^{km} \quad (1)$$

for k, m, l, n = 0, 1, N -1,

$$W_N^{ab} = e^{-j2\pi ab/N} \quad (2)$$

and DFT of a sequence {f(n) n=0, 1, 2,..., N -1} is defined as,

$$F(k) = \sum_{n=0}^{N-1} f(n) W_N^{kn} \quad (3)$$

for k, n = 0, 1,…. ,N -1

The equation 1 can be split into two separate mathematical computations,

$$F'(m,l) = \sum_{n=0}^{N-1} f(m,n) W_N^{ln} \quad (4)$$

for l, n = 0, 1,…. ,N -1

and

$$F(k,l) = \sum_{m=0}^{N-1} F'(m,l) W_N^{km} \quad (5)$$

for k, m = 0, 1,…. ,N -1

The *N*-point 1D DFT of all rows of an input array of size (NXN) will result in intermediate matrix [F' (m, l)]. The 2D DFT is then computed by the inner product of the k^{th} column of the DFT kernel with the l^{th} column of the intermediate matrix [F' (m, l)] [14].

3.2. Formulation of a Proposed Algorithm

Since equations 4 and 5 indicates that 2D-DFT computation can be performed using two consecutive 1D-DFT, we hereby extend the algorithm which was proposed for 1D-DFT to help in 2D-DFT computation with the introduction of an additional hardware which we propose to avoid transposition operation and to enhance speed of computation. Thus equation 1 can be expressed as,

$$F(k) = f(0) + \sum_{n=1}^{M} (f(n) W_N^{kn} + f(N-n) W_N^{-kn}) \quad (6)$$

and

$$F(N-k) = f(0) + \sum_{n=1}^{M} (f(n) W_N^{-kn} + f(N-n) W_N^{kn}) \quad (7)$$

where k=1, 2, …., M and

$$F(0) = f(0) + \sum_{n=1}^{M} (f(n) + f(N-n)) \quad (8)$$

where M=(N -1)/2.

N can either be an odd or even number. Here we assume to be an odd number for hardware proposal and suitable modification can be done for even values of *N*. Modified equations 6 to 8 for even values of *N* are,

$$F(k) = f(0) + f(N/2)e^{-j\pi k} + \sum_{n=1}^{M} (f(n)W_N^{kn} + f(N-n)W_N^{-kn}) \quad (9)$$

and

$$F(N-k) = f(0) + f(N/2)e^{-j\pi k} + \sum_{n=1}^{M} (f(n)W_N^{-kn} + f(N-n)W_N^{kn}) \quad (10)$$

$$F(0) = f(0) + \sum_{n=1}^{M} (f(n) + f(N-n)) + f(N/2) \quad (11)$$

$$F(N/2) = \sum_{n=0}^{M} (f(2n) - f(2n+1)) \quad (12)$$

where M=(N/2)-1.

Splitting the real and imaginary coefficients of 6 and 7 we get,

$$F(k) = f(0) + (F_1(k) - F_2(k)) + j(F_3(k) + F_4(k)) \quad (13)$$

and

$$F(N-k) = f(0) + (F_1(k) + F_2(k)) + j(F_3(k) - F_4(k)) \quad (14)$$

where

$$F_1(k) = \sum_{n=1}^{M} \mathrm{Re}[x(n)]\,\mathrm{Re}[W_N^{kn}] \quad (15)$$

$$F_2(k) = \sum_{n=1}^{M} \mathrm{Im}[y(n)]\,\mathrm{Im}[W_N^{kn}] \quad (16)$$

$$F_3(k) = \sum_{n=1}^{M} \mathrm{Im}[x(n)]\,\mathrm{Re}[W_N^{kn}] \quad (17)$$

$$F_4(k) = \sum_{n=1}^{M} \mathrm{Re}[y(n)]\,\mathrm{Im}[W_N^{kn}] \quad (18)$$

where,

$$x(n) = [f(n) + f(N-n)] \quad (19)$$

and

$$y(n) = [f(n) - f(N-n)] \quad (20)$$

Re[.] and Im[.] are the real and imaginary part of a complex inputs respectively. Thus N-point complex inner products is reduced to 4N real inner products

which can be processed simultaneously each of length (N/2) points.

4. Architectural Aspects

4.1. Proposed 2D-DFT Structure

From the mathematical modification it is known that 2d structure can be easily arrived at by simply computing 1d-DFT concurrently for the rows and columns of the input array matrix and the intermediate matrix result, respectively. The resultant intermediate matrix columns are globally routed in a proper manner using the intermediate buffer block of size *(N x 2N)* cells each of length 32 bits. Figure 1 shows the fully pipelined general architectural implementation of 2d-DFT. It consists of M^2 number of Processing Cells (PCs), M number of Summing blocks 1 & 3 (SB1 & SB3) and one Summing block 2 (SB2), where M is given by *(N-1)/2* . Summing blocks form the first row and last column of the structure in both the DFT Blocks and Processing cells are placed on the left side of the structure below the first row. The functions of the processing cells and summing blocks are shown in the Figure 2. Each summing block consists of only floating point adders performing addition/subtraction operation, which in turn receives continuous inputs during every clock cycle. SB1 receives a pair of inputs *f(M + n)* and *f(M – n + 1)* where n=1, 2,..,M, and M denotes the total number of SB1 & SB3 blocks, from the preprocessing blocks (integer to FP converter) or from the intermediate buffer stage in case of second block. SB1 consists of 3 adder cells. Each SB2 unit receives the first element of each row/column from the blocks discussed above and it consists of only single adder, computing the DC component for each row of the input matrix. Each processing cells will consist of four floating point multiplier to multiply the two processed output from the summing blocks with the DFT coefficient W_N^{kn} producing real and imaginary outputs separately, thus in short it computes the operation shown in equations 11, 12, 13, and 14, and outputting four intermediate results. Multiplicative DFT coefficients are hardwired in the multiplier unit. This complex multiplication can also be performed using CORDIC multiplier by repeated shift and add operation or ROM based look up tables can also be used. Here each multiplier are replaced by ROM tables of size 2^L, where L is the word length based number of bits used for mantissa part for FP representation, which contain all the possible product values obtained on multiplying fixed DFT coefficients by all possible inputs values. Thus four adders are required in processing cells partially computing the operation of real and imaginary part separately as shown in equations 9 and 10. Final summing block SB3 takes four inputs from the processing cells in the previous column and adds the DC component received from SB2, thus performing five add/subtract operation, and giving out real and imaginary values of the computed 1d-DFT separately, of the input row/column of the matrix. Each PCs and SBs consists of latches at each output point for data transfer to the adjacent PCs/SB through both horizontal and vertical stream thus making it fully pipelined architecture and reducing latency.

The output from the SB2 block is available after (M+1) cycles and in the following next M clock cycle, a pair of output from each SB3 unit is available one after another from 1d-DFT unit. PCs form the critical path of the structure, which can be reduced using the techniques suggested above, and thus serve as a deciding factor for maximum frequency of operation. The inputs and outputs to the DFT unit are latched in order to store the processed values in correct sequence to buffer unit which in turn is controlled using FSM and thus this latch doesn't play any role in latency calculation. The number of latches increases with *N* and thus we have M latches following the M^{th} SB3 block. The input to the second DFT unit can be fed only after a column computed from the DFT of *N* row sequence is available in the buffer unit. Thus, only after (N+M) clock cycles, input is fed to second DFT block and thus output of the 2d-DFT sequence become available after *N* clock cycles from the time the input for which DFT has to be computed is fed to the first DFT block.

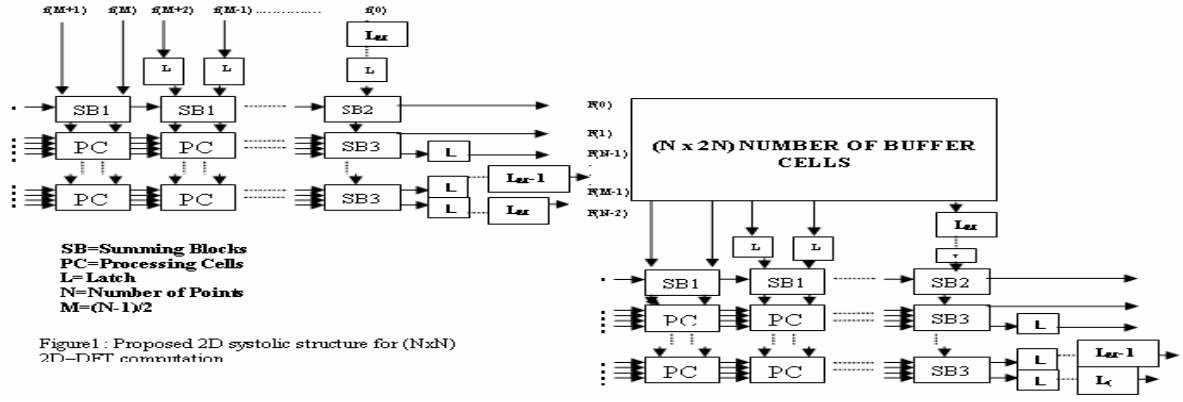
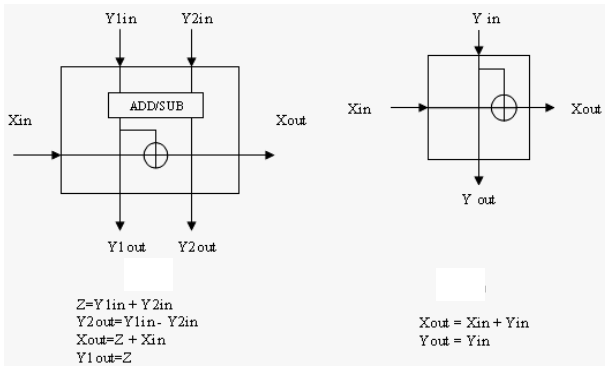
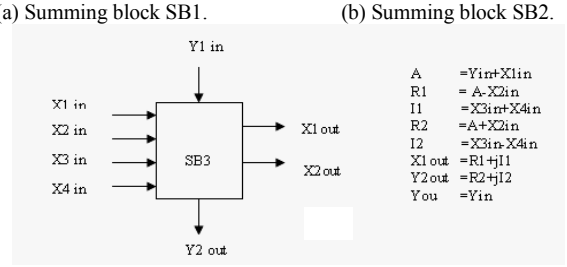


Figure1 : Proposed 2D systolic structure for (NxN) 2D–DFT computation

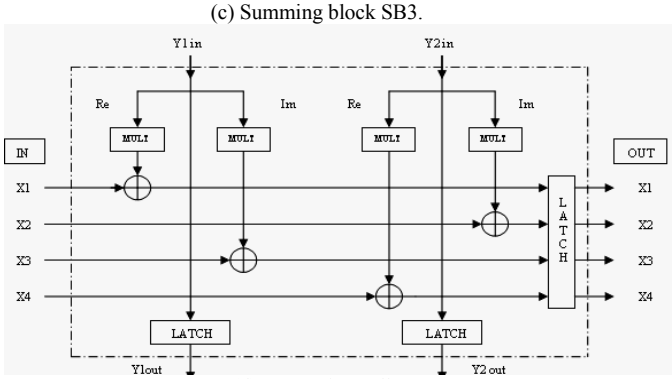
Figure 1. Proposed 2d systolic structure for (NxN) 2d-DFT computation.



(a) Summing block SB1. (b) Summing block SB2.



(c) Summing block SB3.



(d) Processing cell PC.

Figure 2. Structure and function of various units.

4.2. Buffer Unit

The internal structure and operation of buffer unit is shown in Figure 3. It consists of $2N^2$ cells each of size $2*2^L$ (word length considering both real and imaginary part). Twice the size of input matrix number of cells are required to store the 1d-DFT output of 2 input matrix, thus performing continuous computation without any halt. While output from first block is store continuously into one set of N^2 buffer cells and other set feed the input to second DFT block concurrently. Thus it can also be termed as Dual Port Ram where 2 ports are used to simultaneously write the input into one memory cell and read the output from another memory cell simultaneously in a single clock period. This buffer can be implemented as either D-FF or as RAM in ASIC and FPGA. Using it as RAM unit will add to extra fetching time and thus slow down the operation and using it as D-FF designed using standard cells will add overhead in terms of area and power. Clock gating can be applied to the cells using D-FF to save power. Thus there is a trade off in terms of area and speed.

The number inside the cell indicates the clock cycle number at which data to each cell is written and diagonal arrows indicate the flow in which input is stored. Vertical arrows indicate the flow at which the output is read out and number adjacent to rows indicate the clock cycle number at which the particular row of 5 cells each is read out.

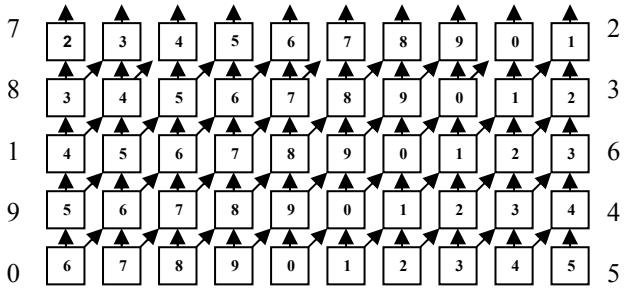


Figure 3. Structure of internal buffer for *N=5*.

4.3. Floating Point Block

Each 32 bit floating point input is split into a real and an imaginary part, with each part having a total of 16 bits, where mantissa is 10 bits, exponent is 5 bits and 1 sign bit. To multiply two floating point numbers, the mantissa of both the inputs are multiplied, while the exponents are added, and the result is rounded and normalized. When we add two floating point numbers, first their exponents have to be made equal and for this, shifters have been used. Then both the input's mantissa is added and the result is normalized. Carry look ahead addition technique has been used in both the floating point multiplication and addition. Modified Booth algorithm has been used for performing the multiplication. The output also consists of 10 bits mantissa, 5 bits exponent and 1 sign bit.

4.4. Scope for Future Work

This architecture can be even used for implementing prime factor DFT as well as for multidimensional DFT by simply repeating the same block over and over again with careful implementation of buffer unit which will further store 2N planes of (M-1) DFT output before computation in the last M^{th} block. Figure 4 shows the general structure for multidimensional DFT.

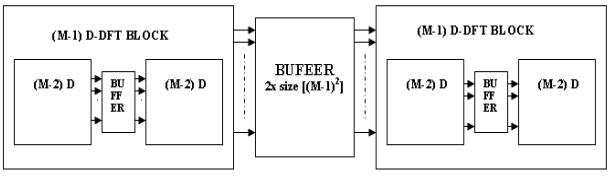


Figure 4. General structure for Multidimensional Implementation.

5. Implementation and Results

The whole 2d-DFT architecture for a 5x5 input has been coded in VERILOG HDL and simulation and synthesis has been carried out using Cadence tools, NcSim and RTL compiler respectively. The simulation is shown in Figure 5. Each input is a 16 bit integer. An integer to floating point converter is used to convert

the integer input to a complex number with floating point representation. A finite state machine has been used as a control circuit which controls the flow of 1d-DFT output into the buffers and from buffers into the other 1d-DFT block. All the synthesis reports have been obtained using the 180nm technology library. The synthesis report of the entire design has been summarized in Table1.Number of multiplication and addition needed for different transform length along with their latency and Average Cycles per Transform (ACT) for the proposed architecture has been given in Table 2.

Figure 5. Simulation using Cadence NcSim.

Table 1. Synthesis results.

2d-DFT block	Values
Speed	200Mhz
Power	61.81milliwatt
Area	0.727055mm^2
Gate Count	32734

Table 2. Latency, Computation for proposed architecture.

Number of points	Latency	ACT	Multipliers	Adders
5	10	5	16	42
32	64	32	961	1149
64	128	64	3969	4349
101	202	101	10,000	10602
256	512	256	65025	66557

Table 3. Comparison of proposed structure with existing architectures N=256.

Structure	Adders	Multipliers	Latency	ACT
[2]	131,072	131,072	1022	512
[3]	131,072	262,144	256	512
[8]	262,144	524,288	256	256
Proposed	66,557	65,025	256	256

The proposed architecture is compared with existing architectures of 2d-DFT for *N* being 256 in Table 3. Here Latency denotes the number of cycles within which the first output is obtained. ACT is defined as the number of clock cycles between corresponding results for successive data blocks. A lesser value of ACC denotes higher throughput. Number of adders and multipliers denotes the complexity of the structure. Hence from the table it is obvious that the proposed structure is less complex when compared to [5, 7] and [4]. Also, it offers low latency compared to [4] and high throughput when compared to [5] and [4].

6. Application

As stated previously discrete Fourier transform has a wide range of applications in signal processing and communication domain. Wireless communication has seen many advances in recent years in which channel estimation plays an important role to cancel out the channel noise and offer better performance. Two dimensional discrete Fourier transform offers an efficient method to estimate spatially correlated channels in MIMO communication systems [12] and offers an interesting area of research.

Pattern matching is another area where 2D DFT has found its importance. The phase component of discrete Fourier transform serves as an excellent classifier and has been used in areas like finger print recognition [2], dental radiography identification [11] and texture feature extraction [14].

7. Conclusion

Thus a systolic architecture for two dimensional discrete Fourier transform providing less complex design with low latency and meeting high throughput requirements has been implemented in this paper. Architectural techniques such as pipelining and parallelism have been used. Since the architecture is recursive, it can be extended to multi-dimensional DFT by using intermediate buffers whose size increases exponentially with the dimension. Inverse Discrete Cosine Transforms (IDCT) and Discrete Cosine Transform (DCT) can also be implemented using the same architecture with slight modification as they are based on the same algorithm. The front end design has been carried out. Placement and Routing on Cadence Encounter is proposed for the future.

References

[1] Chang C., Wang C., and Chang Y., "Efficient VLSI Architectures for Fast Computation of the Discrete Fourier Transform and its Inverse," *IEEE Transactions on Signal Processing*, vol. 48, no. 11, pp. 3206-3216, 2000.

[2] Ito K., Morita A., Aoki T., Higuchi T., Nakajima H., and Kobayashi K., "A Fingerprint Recognition Algorithm Using Phase-Based Image Matching for Low-Quality Fingerprints," *in IEEE International Conference on the Image Processing,* vol. 2, pp. 33-36, 2005.

[3] Kar D. and Rao V., "A New Systolic Realization for the Discrete Fourier Transform," *IEEE Transactions on Signal Processing*, vol. 41, no. 5, pp. 2008-2010, 1993.
[4] Lim H. and Swartzlander E., "A Systolic Array for 2D DFT and 2D DCT," *in Proceedings of the IEEE International Conference on Application Specific Array Processors*, pp. 123-131, 1994.
[5] Lim H. and Swartzlander E., "Multidimensional Systolic Arrays for The Implementation Of Discrete Fourier Transforms," *IEEE Transactions Signal Processing*, vol. 47, no. 5, pp. 1359-1370, 1999.
[6] Liu C., Jen C., "On the Design of VLSI Arrays for Discrete Fourier Transform," *in IEE Proceedings on Circuits, Devices and Systems*, vol. 139, no. 4, pp. 541-552, 1992.
[7] Meher P., "Design of a Fully-Pipelined Systolic Array for Flexible Transposition-Free VLSI of 2-D DFT," *IEEE Transactions on Circuits System II,* vol. 52, no. 2, pp. 85-89, 2005.
[8] Meher P., "Highly Concurrent Reduced Complexity 2-D Systolic Array for Discrete Fourier Transform," *IEEE Signal Processing Letters,* vol. 13, no. 8, 2006.
[9] Nash J., "Computationally Efficient Systolic Architecture for Computing the Discrete Fourier Transform," *IEEE Transactions on Signal Processing*, vol. 53, no. 12, 2005.
[10] Nash J., "Systolic Architecture for Computing the Discrete Fourier Transform on FPGAs," *in Proceedings of the 13th Annual IEEE Symposium on Field-Programmable Custom Computing Machines*, pp. 305-306, 2005.
[11] Nikaido A., and Ito K., Aoki T., Kosuge E., and Kawamata R., "A Phase-Based Image Registration Algorithm for Dental Radiograph Identification," *IEEE International Conference on Image Processing*, vol. 6, pp. 229-232, 2007.
[12] Tong H. and Zekavat S., "Spatially Correlated MIMO Channel: Generation via Virtual Channel Representation," *IEEE Communication Letters*, vol.10, pp. 332-334, 2006.
[13] Xianchao Z., Liusheng H., and Guoliang C., "A New Approach for Computing the Discrete Fourier Transform of Arbitrary Length," *in Proceedings of the 5th International Conference on Signal Processing*, vol. 1, pp. 81-84, 2000.
[14] Yu T., Muthukkumarasamy V., Verma B., and Blumenstein M., "A Texture Feature Extraction Technique Using 2D–DFT and Hamming Distance," *in Proceedings of the Fifth International Conference on Computational Intelligence and Multimedia Applications*, p. 120, 2003.

Sohil Shah has completed his BE in electronics and communication Engineering at Madras Institute of Technology, Anna University in the year 2008. He is pursuing his M.Tech in communication at IIT, Bombay. His areas of interest include VLSI design and optical communication.

Preethi Venkatesan has completed her BE in electronics and communication engineering at Madras Institute of Technology, Anna University in the year 2008. Her areas of interest include VLSI design and FPGA implementations.

Deepa Sundar has completed her BE in electronics and communication engineering at Madras Institute of Technology, Anna University in the year 2008. Her areas of interest include VLSI design and image processing.

Muniandi Kannan has been working as faculty in the department of electronics engineering, MIT Campus of Anna University since 1993. His areas of interest are computer architecture, digital electronics, computer networking and VLSI. His current area of interest is VLSI architecture for signal processing applications.