International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-1 Issue-5, June 2012

# A Low Power VLSI Architecture for Image Compression System Using DCT and IDCT

**D.Preethi, A.M Vijaya Prakash**

***Abstract*— Image compression is an important topic in digital world. It is the art of representing the information in a compact form. This paper deals with the implementation of low power VLSI architecture for image compression system using DCT. Discrete Cosine Transform (DCT) is the most widely used technique for image compression of JPEG images[5] and is a lossy compression method.. The architecture of DCT is based on Lo-effler method[1] which is a fast and low complexity algorithm. In the proposed architecture of DCT multipliers are replaced with adders and shifters. Low power approaches like Canonic signed digit representation for constant coefficients and sub-expression elimination methods has been used. The 2D DCT is performed on 8x8 image matrix using two 1D DCT blocks and a transposition block. Similar to DCT, the IDCT is also implemented using the Lo.effler algorithm for IDCT. Verilog HDL is used to implement the design. ISIM of XILINX is used for the simulation of the design. CADENCE RTL compiler is used to synthesize and obtain the detailed power and area reports of the design. MATLAB is used as the support tool to obtain the input pixel values of the image and the results from both ISIM and MATLAB are compared.***

***Index Terms*— Discrete Cosine Transform (DCT), Low Power, Canonic Signed Digit (CSD), Common Sub expression Elimination (CSE), JPEG, VLSI.**

## I. INTRODUCTION

Image compression, the art and science of reducing the amount of data required to represent an image, is one of the most useful and commercially successful technologies in the field of digital image processing. Image data compression is the technique to reduce the redundancies in the image data representation in order to decrease the data storage requirement and hence the communication cost. Reducing the storage requirement is equivalent to increasing the capacity of storage medium and hence communication bandwidth. Thus the development of efficient compression techniques will continue to be a design challenge for future communication system and advanced media application. Image compression plays an important role in many areas of interest like tele-video conferencing, remote sensing, document and medical imaging. An increasing number of applications depend on efficient manipulation, storage, and transmission of binary, gray scale and colour image.

**Manuscript published on 30 June 2012.**
* Correspondence Author (s)
**D.Preethi**, M.Tech in VLSI Design and Embedded systems, Bangalore Institute of Technology, Bangalore, Karnataka, India.
**A.M.Vijaya Prakash**, Associatet. Professor, department of Electronics and communication, Bangalore Institute of Technology,Bangalore



Image compression algorithms can be broadly classified into lossy compression and lossless compression. The JPEG is one of the most popular and comprehensive continuous tone, still frame compression standard which is centred around Discrete Cosine Transform (DCT). In JPEG compression is performed in three steps viz, computation of DCT, quantization and Variable Length Coding. The DCT transforms the input image which is split into non-overlapping blocks of 8x8 matrix in spatial domain into frequency domain with different frequencies. During quantization the actual compression occurs wherein the less important and high frequency components are discarded and only the most important low frequency components which remain is used to retrieve the image in the decompression process. Once quantization is performed the quantized DCT coefficients are compressed using variable length codes. The JPEG standard uses the Huffman tables for variable length encoding.

DCT is computationally intensive since it takes on large number of multiplications. Many algorithms for DCT has been proposed [2] to reduce the number of computations and hence the power. Its also been proved that the lower bound of multiplications for the computation of DCT is 11[3].

In this paper architecture for DCT and IDCT is implemented using Lo-effler algorithm which requires 11 multiplications and 29 additions. This algorithm was selected as it uses minimum number of mathematical operations. In the design Canonic Signed Digit (CSD) representation for the constant coefficients is used which again reduces the effort of multiplications. Also the multipliers are replaced with adders and shifters. A technique known as Common Sub-expression Elimination [2] is used so as to obtain the utilization of the resources. Thus all of these making the design a low power implementation.

The rest of the paper is organised as follows: section 2 describes the DCT and IDCT algorithms, section 3 gives the description of Canonic Signed Digits (CSD); section 4 introduces to common sub-expression elimination (CSE); section 5 presents the proposed architecture; section 6 gives the implementation of the results and section 7 is the conclusion.

## II. DCT AND IDCT ALGORITHM

The 8x8 2D DCT for the input f(x,y) is given as


*Published By:*
*Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)*

$$c(u,v) = a(u)a(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{\pi(2x+1)u}{2N}\right]\cos\left[\frac{\pi(2y+1)}{2N}\right] \text{-(1)}$$

Where $a(u/v) = \begin{cases} \sqrt{\frac{1}{N}} \text{for u/v} = 0 \\ \sqrt{\frac{2}{N}} \text{for u/v} \neq 0 \end{cases}$ (2)

The 8x8 2D IDCT is given as

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} a(u)a(v)c(u,v)\cos\left[\frac{\pi(2x+1)u}{2N}\right]\cos\left[\frac{\pi(2y+1)v}{2N}\right] \quad (3)$$

Direct implementation of the above equations will require 1024 multiplications and 896 additions. Thus the design will be more complex and also expensive.

By making use of separability property of DCT the 2D DCT/IDCT can be implemented by using the cascade of two 1D DCT and a transposition block as shown below. The transposition block transposes the output of 1D DCT.

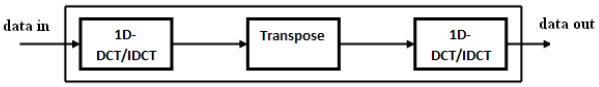



Figure 1. 2 D DCT/IDCT

The 1D DCT is as given

$$c(u) = a(u)\sum_{x=0}^{N-1} f(x)\cos\left[\frac{\pi(2x+1)u}{2N}\right] \quad (4)$$

and IDCT equations is as given below

$$f(x) = \sum_{x=0}^{N-1} a(u)c(u)\cos\left[\frac{\pi(2x+1)u}{2N}\right] \quad (5)$$

Where a(u) is as defined in equation (2).

### A. *Lo-effler Algorithm*

Christoph Lo-effler has proposed a 8-point DCT algorithm that requires only 11 multiplications and 29 additions [3]. In this scheme the number of multiplications has been reduced to a theoretical lower bound which is 11.

The Lo-effler algorithm for the DCT is as shown in the fig 2. It has four stages, each stage has to be executed in series due to the data dependencies. As is seen in the figure, stage 1 requires 4 additions and 4 subtractions. In the second stage the algorithm is split into two parts, one of which for even coefficients and the other half for the odd coefficients. Again in the third stage the even coefficients is separated into even and odd parts. The scaling factor k= √2.

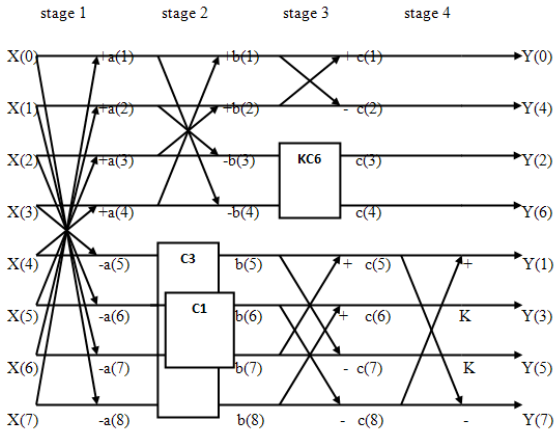
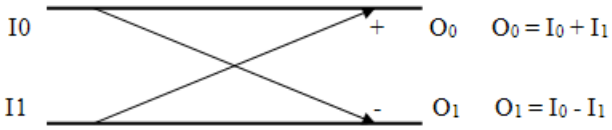
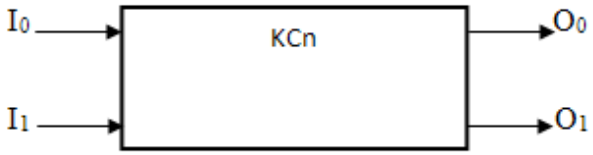



Figure 2. Flow graph of Lo-effler DCT

The building blocks are as shown below:




$O_o = I_0\, k\cos(n\pi/16) + I_1 k.\sin(n\pi/16)$

$O_1 = -I_0\, k\sin(n\pi/16) + I_1 k.\cos(n\pi/16)$

Similar to DCT. The IDCT is also as given by Lo-effler algorithm as below with k=1/√2.

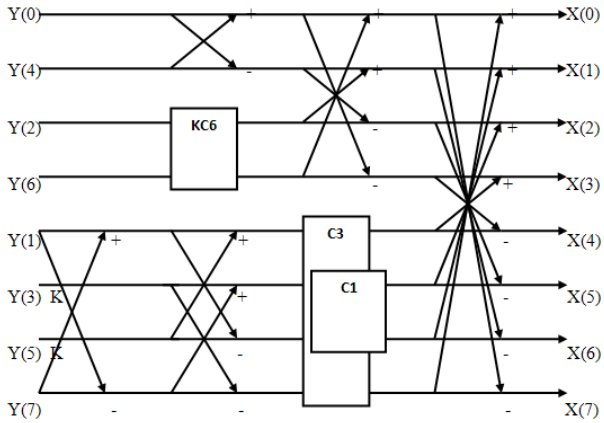
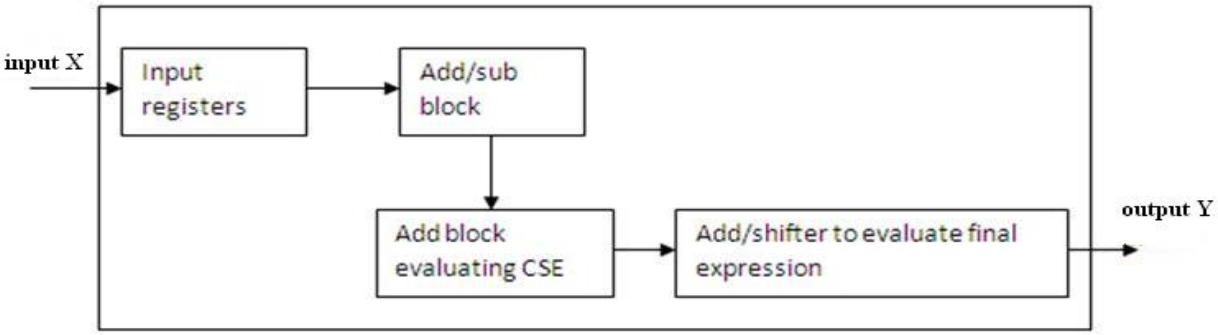



**Figure 3. Flow graph of Lo-effler IDCT**

The equations of the kcn is modified in IDCT and is given as

$O_o = I_0\, k.\cos(n\pi/16) - I_1 k.\sin(n\pi/16)$


*Published By:*
*Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)*




$O_1 = I_0 k.\sin(n\pi/16) + I_1 k.\cos(n\pi/16)$

## III. CANONIC SIGNED DIGIT (CSD)

Canonic Signed Digit was introduced by Avizienis, is a signed representation. Containing fewest number of non zero bits[4]. Thus for the constant multipliers, the number of toggles will be minimum and hence reducing the power consumption.

For a constant coefficient c, the CSD representation is as shown

$$c = \sum_{i=0}^{N-1} c_i 2^i \quad \text{where } c_i = \{-1,0,1\} \equiv \{-,0,+\}.$$

CSD numbers have essentially two properties

- no 2 consecutive bits in a CSD numbers are non zeros.
- The CSD representation of a number contains the minimum possible number of non zero bits.

The CSD representation contains about 33% fewer non zero bits than 2's complement number. Consequently, for constant multipliers the number of partial products are reduced. The CSD representation for the constant DCT coefficients is as shown in the table 1.

TABLE I:

CONSTANT DCT COEFFICIENTS IN CSD

| constant | Fractional value | Binary value | Csd equivalent |
|---|---|---|---|
| $\cos(6\pi/16)$ | 0.38268 | 00110001 | 0+0-000+ |
| $\sin(6\pi/16)$ | 0.92388 | 01110110 | +000-0-0 |
| $\cos(3\pi/16)$ | 0.83147 | 01101010 | +0-0+0+0 |
| $\sin(3\pi/16)$ | 0.55557 | 01000111 | 0+00+00- |
| $\cos(\pi/16)$ | 0.98079 | 01111110 | +00000-0 |
| $\sin(\pi/16)$ | 0.19509 | 00011001 | 00+0-00+ |

## IV. COMMON SUB-EXPRESSION ELIMINATION (CSE)

To replace the multipliers with adders and shifters CSE technique is used. CSE technique enhances the usage of adders and shifters by identifying the common expressions[2]. Thus by the use of CSE resource utilization is achieved. In the design the CSE has taken the advantage of CSD representation for identifying the common sub-expressions. Let us take for example the evaluation of Y(1).

Y(1) = c(5) + c(8);

= b(5) + b(7) + b(6) + b(8) ;

= a(5) * cos(3π/16) + a(8) * sin(3π/16) – a(6) * sin(π/16) + a(7) * cos(π/16) + a(6) * cos(π/16) + a(7) * sin(π/16) – a(5) * sin(3π/16) + a(8) * cos(3π/16) ;

= a(5) * ($2^7$- $2^5$+ $2^3$+ $2^1$) + a(8) * ( $2^6$+ $2^3$- $2^0$) – a(6) *($2^5$- $2^3$+ $2^0$) + a(7) * ($2^7$- $2^1$) + a(6) *($2^7$- $2^1$) + a(7) * ($2^5$- $2^3$+ $2^0$) – a(5) * ($2^6$+ $2^3$ - $2^0$) + a(8) * ($2^7$-$2^5$+$2^3$+$2^1$) ;

= $2^7$* (a(5) + a(8) + a(7) + a(6)) + $2^6$* (a(8) - a(5)) – $2^5$ *(a(5) + a(8) + a(6) - a(7)) + $2^3$ ( a(5) + a(8) + a(6) - a(7) - a(5) + a(8)) +$2^1$* (a(5) + a(8) - a(7) - a(6)) -$2^0$ * (a(5) + a(8) + a(6) - a(7)) ;

= $2^7$* (sb1 + sb4) + $2^6$*(sb2) -$2^5$* (sb1 + sb3) + $2^3$ *(sb1 +sb2 + sb3) + $2^1$ * (sb1 - sb4) – $2^0$ *(sb2 + sb3) ;

Where sb1=a(5)+a(8); sb2=a(8)-a(5); sb3=a(6)-a(7); sb4=a(6)+a(7).

Thus as can be seen sb1 occurs 4 times, sb2 occurs 3 times, sb3 for 3 times and sb4 occurs 2 times. Thus implementing these sub-expressions once we can share the hardware and hence power reduction can be achieved. Now the multipliers above can be replaced by the shift operations yielding

Y(1) = (sb1+sb4)<<7 + (sb2)<<6 – (sb1+sb3)<<5 + (sb1+sb2+sb3)<<3 + (sb1-sb4)<<1 – (sb2+sb3) .

## V. PROPOSED DCT AND IDCT ARCHITECTURE

The 2D DCT is implemented by using two 1D DCT blocks and a transposition block. The 8x8 image matrix having the values in the range of 0 to 255 is first converted from unsigned to signed representation which is called level shifting by subtracting from 128. The signed converted values are now given as the input to first 1D DCT in either rowwise or columnwise. The output of the first 1D DCT is given to the transposition block where the transpose of the input takes place. Then the output of this transposition block is given as input to the second 1D DCT to obtain the final 2D DCT.

The complete 2D DCT is as shown in figure 4.

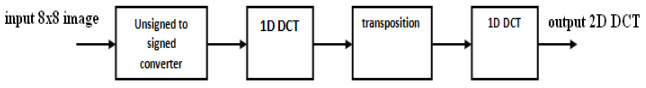



Figure 4. The 2D DCT architecture


Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Similarly to the DCT the 2D IDCT is also implemented using two 1D IDCT blocks and a transposition block. In the IDCT process to the obtained 2D IDCT an addition of 128 is done to reconstruct the original image.

The architecture for 1D DCT which is implemented using Lo-effler algorithm and using CSD and CSE is as shown in figure 5.

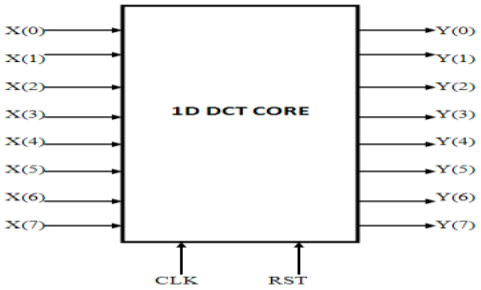



**Figure 5. The 1D DCT architecture**

The eight pixels in a column of 8x8 matrix is input in parallel to the input registers and then the registered inputs are sent for processing. The architecture is implemented in pipeline structure. The design implemented here is a parallel in and parallel out thereby reducing the latency of the design

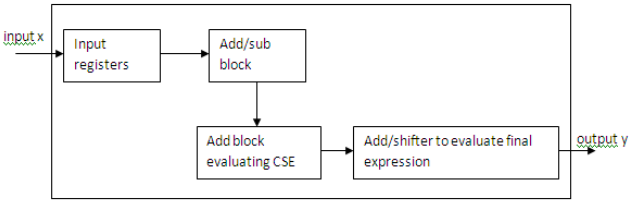
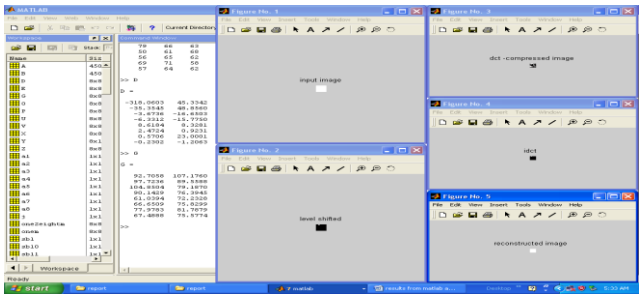



**Figure 6. The 1D DCT top module**



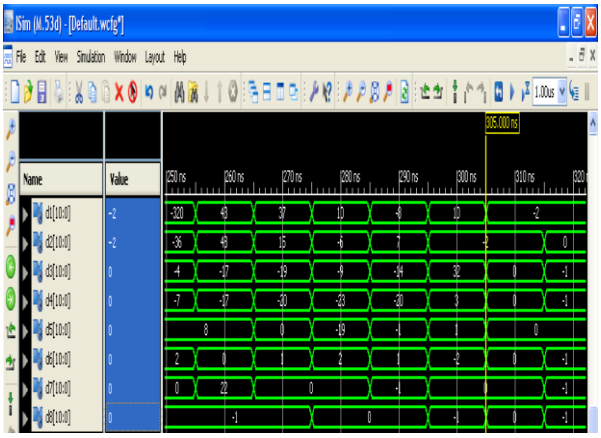


**Figure 7. The 2D DCT simulation results**

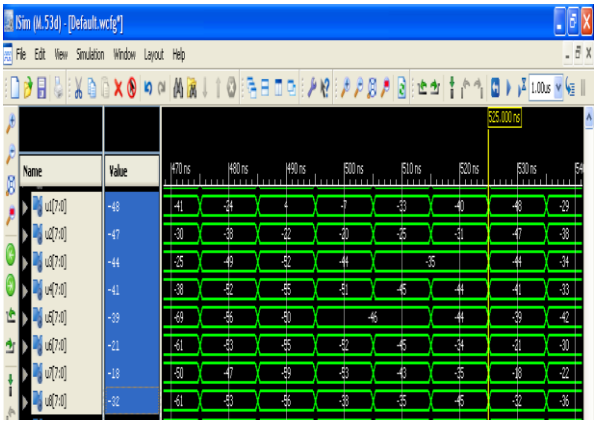


**Figure 8. The 2D IDCT simulation results**
**Fig 9. MATLAB results**

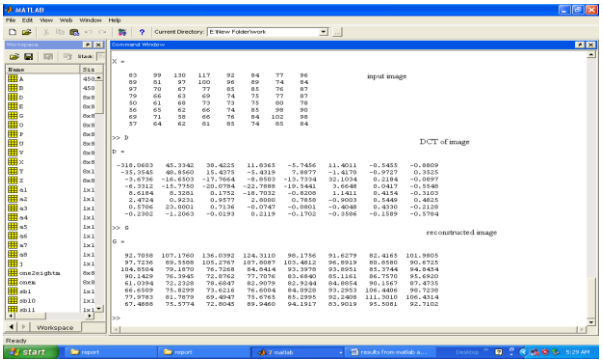


**Figure 10. MATLAB image results**

### A. *Power Analysis*

RTL compiler of CADENCE was used to synthesize the design. Once the synthesize was done the power and area reports were obtained by mapping the design to 180 nm TSMC library. The RTL schematic of the design was also obtained.


*Published By:*
*Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)*

## VI. IMPLEMENTATION AND RESULTS

The architecture for the proposed DCT/IDCT is modelled using Verilog HDL. MATLAB was used to obtain the input image pixels for the design and also to reconstruct the image after obtaining the outputs from the IDCT core. The functionality of the designs were verified by simulating the design in ISIM of XILINX. RTL compiler of CADENCE was used to synthesize the designs and the power and area reports were obtained. The simulation results for 2D DCT and 2D IDCT from ISIM simulator is as shown in figures 7 and 8. MATLAB results are as shown in figure 9 and 10.

### *B. Power Analysis*

RTL compiler of CADENCE was used to synthesize the design. Once the synthesize was done the power and area reports were obtained by mapping the design to 180 nm TSMC library. The RTL schematic of the design was also obtained.

TABLE 2
POWER AND AREA REPORT OF DCT AND IDCT

| **Features** | **DCT** | **IDCT** |
|---|---|---|
| Power | 2.488mW | 3.143mW |
| No.of cells | 8827 | 10901 |
| Cell area | 0.1033mm$^2$ | 0.1235mm$^2$ |

## VII. CONCLUSION

This paper proposed a low power VLSI architecture for DCT and IDCT for image compression system. Since there were no multipliers used in the design a very low power was obtained. Also due to pipelined structure and parallel input and parallel output the design had very low latency. Power reduction was achieved due to both CSD and CSE techniques and hence a low power design.

## REFERENCES

[1] Vimal P. Singh Thoudam, Prof.B.Bhaumik, Dr. s. Chatterjee, "Ultra Low Power Implementation Of 2D DCT For Image/Video Compression," presented in International Conference on Computer Applications and Industrial Electronics.
[2] M. Jridi, A. Alfalou, "Low Power, High-Speed DCT Architecture For Image Compression: Principle And Implementation", 18th IEEE/IFIP International Conference on VLSI and System on Chip.
[3] Christoph Loeffler, Adriaan Ligtenberg, George S. Moschytz "Practical fast 1D DCT algorithm with 11 multiplications," 1989 IEEE.
[4] Jeffrey O. Coleman, Arda Yurdakul "Fractions in The Canonical Signed Digit Number Systems," Conference On Information Science And Systems.
[5] Gregory K.Wallace, "The JPEG Still Picture Compression Standards", 1991 publication in IEEE transactions on consumer electronics.
[6] Ricardo Castellanos, Hari Kalva, Ravishankar, "Low Power DCT Using Highly Scalable Multipliers".
[7] Ankita Singla, A.P.Vinod, Deepu Rajan, Edmund M.K.Lai, "Low Power DCT Implementation Using Differential Pixels For On-Board Satellite Image Processing."
[8] Muhammed Yusuf Khan, Ekram Khan, M.Salim Beg, "Performance Evaluation Of 4x4 DCT Algorithms For Low Power Wireless Applications," First International Conference on Emerging Trends in Engineering and Technology.
[9] M. El Aakif, S.Belkouch, N.Chabini, M.M.Hassani, "Low Power and Fast DCT Architecture Using Multipliers-Less Method", 2011 Faible Tension Consommation.
[10] Vijay Kumar Sharma, K.K.Mahapatra, Umesh C. Pati, "An Efficient Distributed Arithmetic based VLSI Architecture for DCT", 2011 IEEE.
[11] A. Pradini, T.M.Roffi, R.Dirza, T.Adiono, "VLSI Design of a High Throughput Discrete Cosine Transform for Image Compression Systems", 2011 International Conference on Electrical Engineering and Informatics, Indonesia.
[12] S.V.V Sateesh, R.Sakathivel, K.Nirosha, Harish M.Kittur, "An Optimized Architecture to Perform Image Compression And Encryption Simultaneously Using Modified DCT Algorithms", Proceedings of 2011 International Conference On Signal Processing, Communication, Computing And Networking Technologies.

**D.Preethi** has received her Bachelor's Degree B.E in Electrical and Electronics from BMS Collage of Engineering, Bangalore of Visvesvaraya Technological University in the year 2008 and presently pursuing her M-Tech in VLSI Design and Embedded Systems from Bangalore Institute of Technology, Bangalore, India.

**Vijaya Prakash A.M** obtained his B.E degree from from UVCE Bangalore of Bangalore University in the year 1992, Post graduated in M.E from SDMCET Dhawad of Karnataka University in the year 1997 and presently pursuing Ph.D. from Dr.M.G.R University Chennai.

He has been actively guiding PG and UG students in the area of VLSI and Image Processing. He has around 10 technical paper publications in journals and international conferences. He has presented paper in Singapur. Currently he has been working as Associate Professor in Electronics and Communication Engineering Department, Bangalore Institute of Technology Bangalore. His research interests are Low Power VLSI, Image Processing, Synthesis and Optimization of Digital Circuits. He is a member of IMAPS and ISTE.


*Published By:*
*Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)*