



International Journal of Intellectual Advancements and Research in Engineering Computations

Power and area efficient modified booth multiplier for low power consumption

M.Chandraman¹, M.Ramyadevi²

Assistant Professor¹/ VLSI Design PG scholar ²

Department of ECE, Knowledge Institute of Technology, Salem.

ABSTRACT

The reconfigurable finite-impulse response (FIR) filters are one of the most widely implemented Components in Internet of Things systems that require flexibility to support several target applications while consuming the minimum amount of power to comply with the strict design requirements of portable devices. Due to the significant power consumption in the multiplier components of the FIR filter, various techniques aimed at reducing the switching activity of these multipliers have been proposed in the project. However, these techniques rarely exploit the flexibility on the algorithmic level, which can lead to additional benefits. In this project, FIR filter multipliers are (baugh-wooley multiplier, booth multiplier, modified booth multiplier) extensively characterized with power simulations, providing a methodology for the perturbation of the coefficients of baseline filters at the algorithm level to trade-off reduced power consumption for filter quality. The proposed optimization technique does not require any hardware overhead and it enables the possibility of scaling the power consumption of the filter at runtime, while ensuring the full baseline performance of any programmed filter whenever it is required.

INTRODUCTION

Finite Impulse Response (FIR) filter plays an important role in several signal processing applications in communication schemes, which performs interference cancellation, channel equalization, spectral shaping, matched filtering and more. Now a days, various implementation and architecture methods have been presented to improve the performance of filters in terms of system complexity and speed. The FIR filter used in the majority of the Digital Signal Processing (DSP) applications is based on electronic systems. The FIR filter coefficients change rapidly during execution time, in several application scenarios

such as a digital up-converter, digital down converter, multi-channel filter and software-defined radio systems. Programmable finite impulse response (FIR) filters are one of the most widely implemented accelerators and they are a fundamental building block for many DSP applications. Therefore, it can be expected that methods for reducing their power consumption can have a large impact on a variety of IOT systems and applications. When operating at high clock frequencies, dynamic switching energy generally dominates power consumption in digital systems. [1-5] The dynamic power consumption is described by the well known equation:

$$P_d = \alpha \cdot f \cdot C \cdot V_{DD}^2,$$

Author for correspondence:

Department of ECE, Knowledge Institute of Technology, Salem.

Where α is the switching activity factor, f is the operating frequency, C is the switching capacitance, and VDD is the supply voltage. Such an approach provides the capability to scale the power consumption of the filter at runtime. Based on the observation that most of the power consumed in FIR filters is due to multiplications, different techniques aimed to reduce power consumption in multipliers have been proposed. These techniques include optimizing the value of successive coefficients assigned to iteratively-decomposed FIR filters based on their Hamming distance; the design of approximate multipliers for low power operation; performing parallel multiplication by coefficient partitioning to enable voltage scaling; and reusing previously computed values through the factorization of the coefficients to reduce the complexity of the filter. [6-11]

OBJECTIVE OF THE WORK

Multipliers based on the number of non-zero generated partial product is presented. The dynamic power consumption of the considered multipliers has been characterized with accurate post-layout simulations. The power consumption of the multipliers contained in a programmable FIR filter is reduced by perturbing the baseline coefficients of the filter based on an extensive characterization of the implemented multiplier topology. The baseline performance is always ensured when operating with the reference

coefficients and the power consumption of the FIR filter can be reduced at runtime, when a less accurate operation of the filter is tolerated.

SCOPE OF THE WORK

A low power operation in reconfigurable FIR filter using efficient arithmetic circuits is proposed in this thesis. Many of the programmable filters in the literature focus on the coefficient multiplication. However, apart from the coefficient multiplication, there exists an adder for accumulation, which also has the significant role in the critical path delay. In this method, the final adder in multiplier and accumulator are replaced with a 3:2 compressors. The compressors in place of adders minimizes the critical path delay of the filter. The performance of the proposed diagram compared with better results in terms of delay and power.

MULTIPLIERS

A multiplication is very expensive and slows the overall operation. The performance of many computational problems is often dominated by the speed at which a multiplication operation can be executed. Consider two unsigned binary numbers X and Y that are M and N bits wide, respectively. To introduce the multiplication operation, it is useful to express X and Y in the binary representation.

$$X = \sum X_i 2^i \quad i = 0 \text{ to } M \quad (1.1)$$

$$Y = \sum Y_j 2^j \quad j = 0 \text{ to } N \quad (1.2)$$

With $X_i, Y_j \in \{0,1\}$. The multiplication operation is then defined as follows:

$$Z = X \times Y = \sum Z_k 2^k \quad k = 0 \text{ to } M + N - 1 \quad (1.3)$$

$$= (\sum X_i 2^i \quad i = 0 \text{ to } M) (\sum Y_j 2^j \quad j = 0 \text{ to } N) \quad (1.4)$$

$$= \sum (\sum X^i Y^j 2^{i+j}) \quad i = 0 \text{ to } M-1, j = 0 \text{ to } N-1 \quad (1.5)$$

The simplest way to perform a multiplication is to use a single two input adder. For inputs that are M and N bits wide, the multiplication tasks M cycles, using an N-bit adder. This shift –and-add algorithm for multiplication adds together M partial products. Each partial product is generated by multiplying the multiplicand with a bit of the multiplier—which, essentially, is an AND operation— and by shifting the result in the basis of the multiplier bit’s position. Similar to the familiar long hand decimal multiplication, binary multiplication involves the addition of shifted versions of the multiplicand based on the value and position of each of the multiplier bits. As a matter of fact, it’s much simpler to perform binary multiplication than decimal multiplication. The value of each digit of a binary number can only be 0 or 1, thus, depending on the value of the multiplier bit, the partial products can only be a copy of the multiplicand, or 0. In digital logic, this is simply an AND function.

LITERATURE SURVEY

Lei Wang (2003), briefly describes the A low-power technique for digital filtering referred to as *adaptive error-cancellation* (AEC) is presented. The AEC technique falls under the general class of *algorithmic noise-tolerance* (ANT) techniques proposed earlier for combating transient/soft errors. The proposed AEC technique exploits the correlation between the input and soft errors to estimate and cancel out the latter. It apply AEC along with *voltage over scaling* (VOS), where the voltage is scaled beyond the necessary for correct operation. It employ the AEC technique in the context of a frequency-division multiplexed (FDM) communication system and demonstrate that up to 71% energy reduction can be achieved over present-day voltage-scaled systems.

Rajamohana Hegde (2004), developed that an integrated circuit implementation of a low-power

digital filter in 0.35- m 3.3-V CMOS process. The low-power technique combines voltage over scaling (VOS) and algorithmic noise tolerance (ANT) to push the limits of energy efficiency beyond that achievable by voltage scaling alone. VOS refers to scaling the supply voltage beyond the limit imposed by the throughput constraints. ANT is an algorithmic level error-control technique that is employed to restore the algorithmic performance degradation in terms of output signal-to-noise ratio (SNR) caused by VOS. Measured results indicate 40%–67% reduction in energy dissipation *over optimally voltage-scaled systems* with less than 1-dB loss in SNR for a wide range of filter bandwidths.

RPR requires a reduced precision replica whose output can be employed as the corrected output in case the original system computes erroneously. When combined with voltage over scaling (VOS), the resulting soft digital signal processing system achieves up to 60% and 44% energy savings with no loss in the signal-to-noise ratio (SNR) for receive filtering in a QPSK system and the butterfly of fast Fourier transform (FFT) in a WLAN OFDM system, respectively. These energy savings are with respect to optimally scaled (i.e., the supply voltage equals the critical voltage) present day systems. Further, it show that the RPR technique is able to maintain the output SNR for error rates of up to 0.09/sample and 0.06/sample in an finite impulse response filter and a FFT block, respectively.

Brian Richards (2010), developed that energy-efficient reconfigurable distributed-arithmetic FIR filter for multi-mode wireless communication is fabricated in 7M1P 90nm CMOS and occupies 1.5mm². A 6- way parallel, 2-way time-multiplexed architecture with circuits for memory offset binary coding and memory partitioning enable input word length and tap configurability with 1–190MSample/s throughput and 10–130mW total power measured at 1.1V, 25°C.

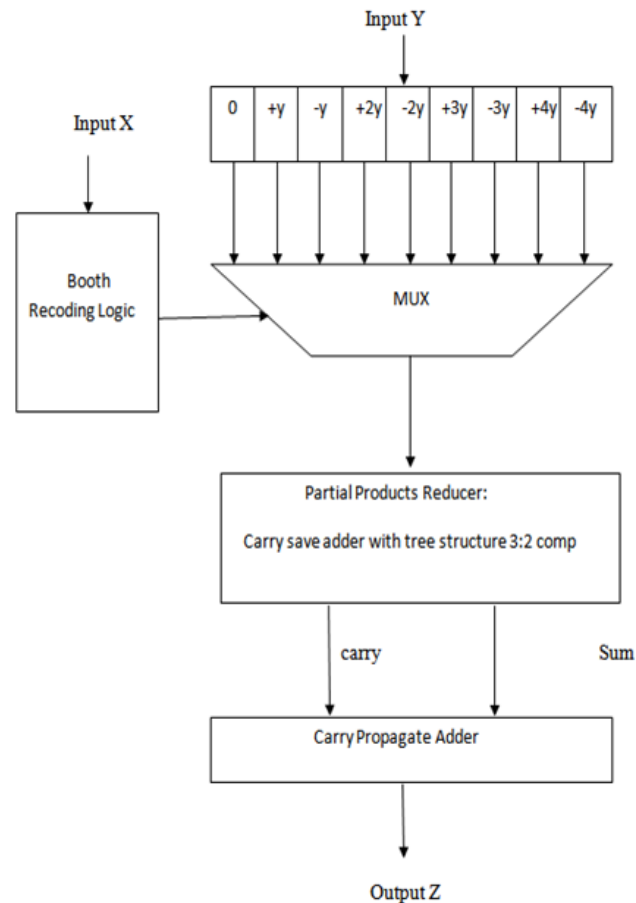


Figure Modified booth encoding multiplier block diagram

PROPOSED METHODOLOGY

Booth's encoding

Booth encoding reduces the number of partial products by a factor of two and thus considerably reduces the area as well as increasing the speed of our multiplier.

Booth's encoding or Booth's multiplication algorithm is a multiplication algorithm which can multiply two signed binary numbers in a two's complement notation. Booth's algorithm has the ability to perform fewer additions and subtractions in comparison to normal multiplication algorithm. It is an encoding process which can be used to minimize the no of partial products in a multiplication process. It is based upon the relation Modified booth's algorithm

One of the many solutions of realizing high speed multipliers is enhancing parallelism which

helps in decreasing the number of subsequent calculation levels. The original version of Booth algorithm (Radix-2) had two particular drawbacks. They were:

The number of add-subtract operations and shift operations become variable and causes inconvenience in designing parallel multipliers.

- The algorithm becomes inefficient when there are isolated 1's.
 - These problems are overwhelmed by using modified Radix4 Booth algorithm which scans strings of three bits using the algorithm given below:
 - Lengthen the sign bit 1 position if necessary to ensure that n is even.
 - Add a 0 to the right of the LSB of the multiplier.
- Corresponding to the value of each vector, each Partial Product will be $0, +M, -M, +2M$ or $-2M$.

The negative values of M are made by taking its 2's complement. The multiplication of M is done by shifting M by one bit to the left (in case it's multiplied with 2). Thus, in any case, in designing an n-bit parallel multiplier, only n/2 partial products are generated.

The overall multiplier speed does depend on the size of final CPA, but this may be optimized independently of the CSA array.

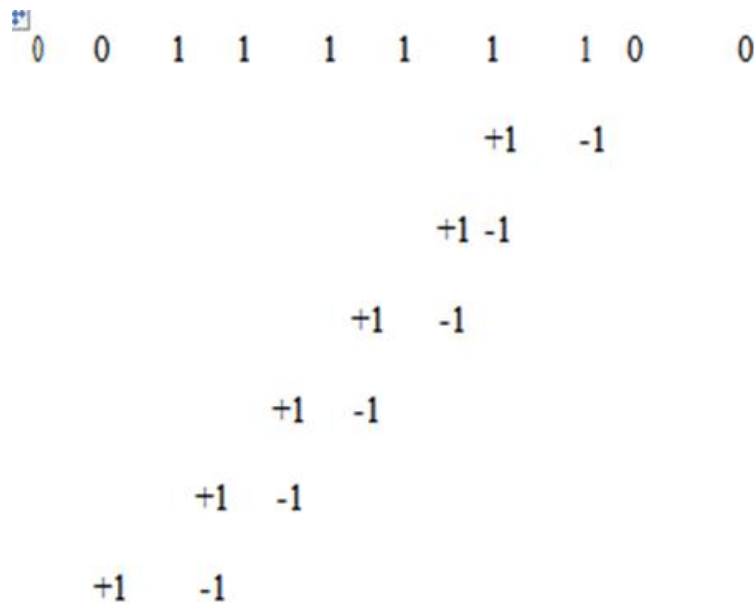
Power dissipation is reduced by the tree-based structures. The simplified carry-save logic produces fewer signal transitions and the tree structures produce fewer glitches than a chain

Radix -4 booth encoder multiplier

Since radix-2 Baugh- Wooley multipliers are rather slow, to study a fast multiplier that uses

Booth recoding. A modified booth recoded multiplier⁴ shown in figure 1.4, has been considered, where the PPR is fed with less than half of the partial products of those in the BW2 multiplier, there by providing a much shorter critical path. As opposed to the symmetric BW2 multiplier, in the BR4 topology the two input operands are processed differently, since x is passed to the recoding logic that decides which multiples of y should be fed to the PPR. For the considered implementation, a carry-save adder with (3:2) compressors is used for the PPR and a carry-propagate adder is used for the VMA, as in the BW2 multiplier.

$$2^n = 2^{n+1} - 2^n$$



Example

Booth's algorithm examines consecutive bits of the N-bit multiplier Y in signed two's complement representation, which includes an implicit bit below the least significant bit, $y_{-1} = 0$. For each bit y_i , as i runs from 0 to N-1, the bits y_i and y_{i-1} are

considered. When these two bits are equal, the product accumulator P stays unchanged. Where $y_i = 0$ and $y_{i-1} = 1$, the multiplicand times $2i$ is added to P; and where $y_i = 1$ and $y_{i-1} = 0$, the multiplicand times $2i$ gets subtracted from P. The final value of P will be the signed product.

Table 3.1.6 Modified Booth’s Recoding Table

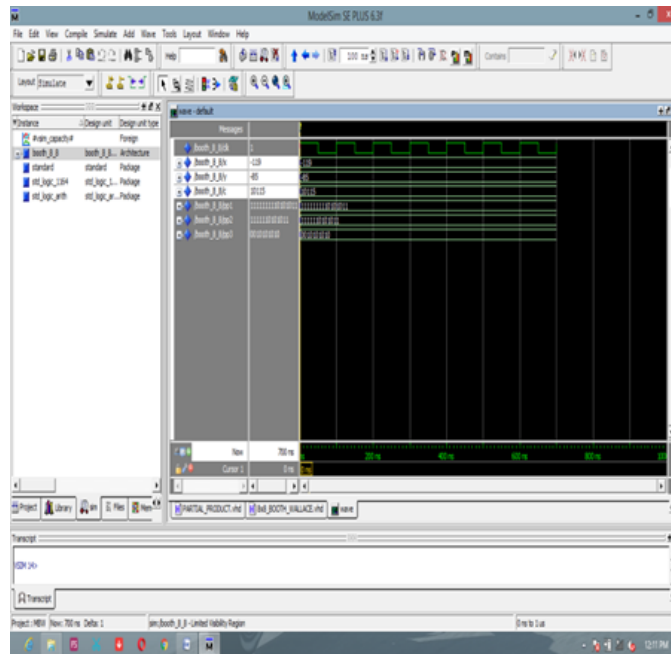
Multiplier bit - pair		Multiplier bit on the right	Booths Representation		Multiplicand selected at position i
i+1	i	i-1			
0	0	0	0	0	0*M
0	0	1	0	+1	+1*M
0	1	0	+1	-1	+1*M
0	1	1	+1	0	+2*M
1	0	0	-1	0	-2*M
1	0	1	-1	+1	-1*M
1	1	0	0	-1	-1*M
1	1	1	0	0	0*M

The functional operation of Radix-4 booth encode is shown in the table1.5. It consists of eight different types of states and during these states it

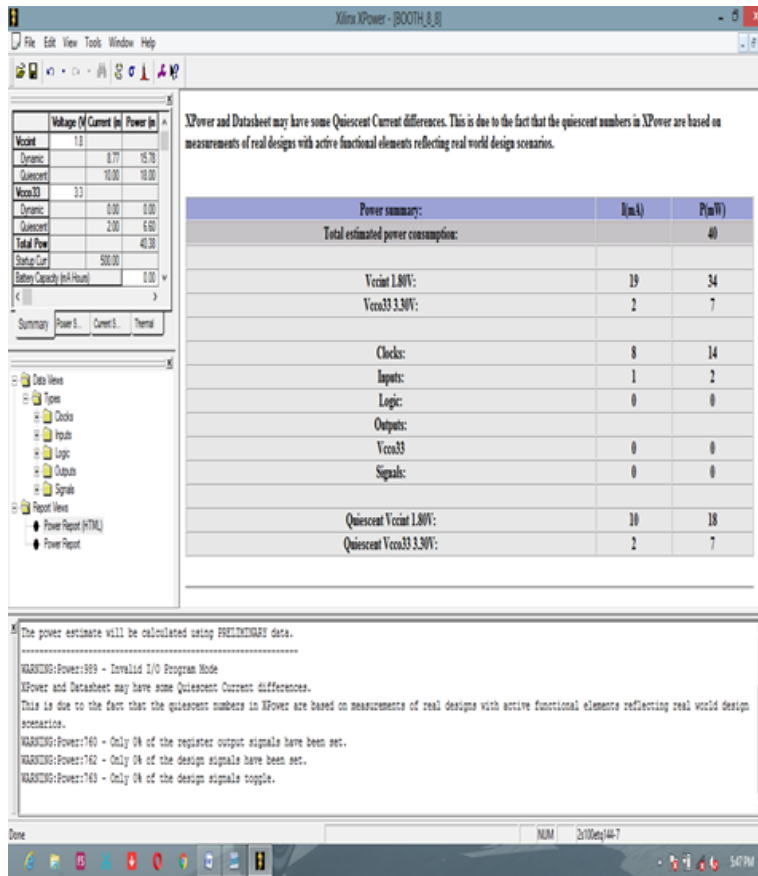
obtain the outcomes, which are multiplication with 0,-1 and 2 consecutively.

RESULTS AND DISCUSSION

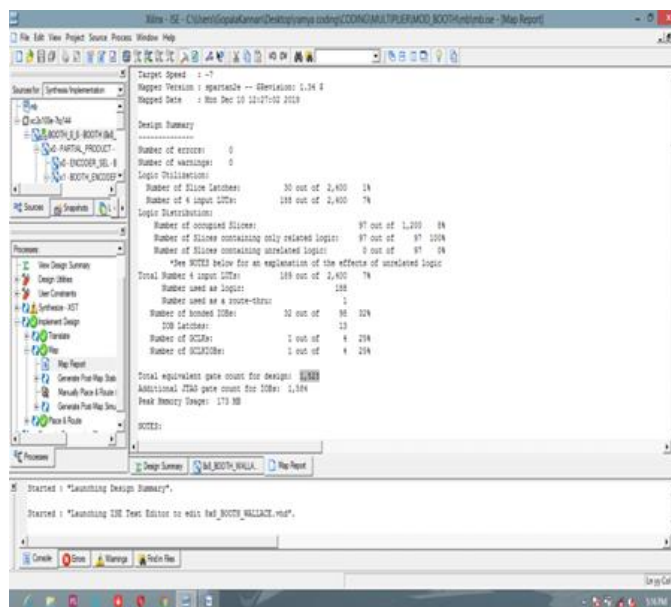
Modified booth multiplier



Booth multiplier of area



Booth multiplier of power consumption



MODIFIED BOOTH MULTIPLIER OF AREA

Modified booth multiplier power consumption

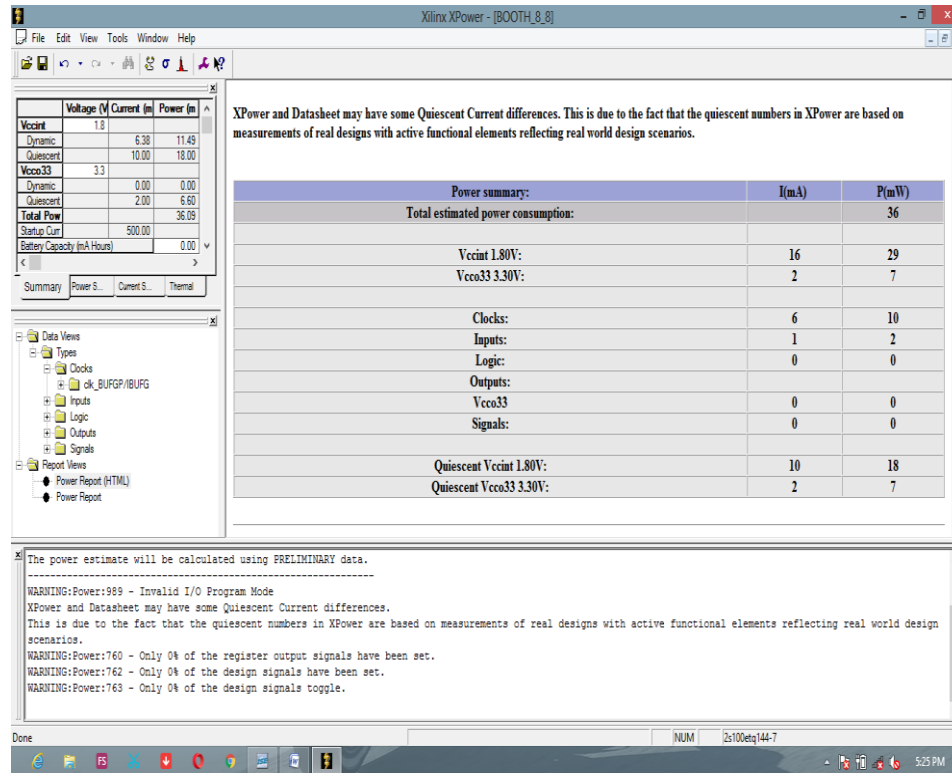


Table 4.1.7 Area and power value

MULTIPLIERS		AREA (μm ²)	POWER(μw)
Baugh - wooley	multiplier	1845	47
Booth multiplier		1528	40
Modified booth	multiplier	1523	36

The booth’s multiplier is then coded in VHDL tool, area and power analysis is performed on it. Radix-4 multiplier is then changed in the addition to reduce the number of partial product. Results of area and power are shown table 4.3. Finally it is found be efficient in terms of power consumption.

CONCLUSION

The power consumption of multipliers having one input port connected to a constant coefficient operand can significantly vary with the value of that coefficient. In programmable architectures where one coefficient is constant over many

cycles, this can be exploited for power savings by approximating the original coefficient with a similar coefficient that yields to lower power consumption. This technique can effectively be applied, as an example, to reconfigurable FIR accelerators. A simple greedy algorithm is used to modify the coefficients of a baseline filter to derive a new set of coefficients that are optimized for low power consumption while allowing for some degradation of the filtering quality. By exploiting the flexibility on the algorithm level, the proposed approximate computing technique does not require any design overhead for a programmable accelerator. At the same time, it

ensures the quality of the baseline filter whenever it is required, while it offers also the possibility of scaling the power consumption at runtime when energy is short and reduced accuracy is tolerated. .

It is mainly meant to reduce the time delay and finally it is found be efficient in terms of power consumption.

REFERENCES

- [1]. Lei Wang, Naresh R. Shanbha, "Low-Power Filtering via Adaptive Error-Cancellation" IEEE Transactions on signal processing, 51(23), 2003.
- [2]. Rajamohana Hegde and Naresh R. Shanbhag, "A Voltage Overscaled Low-Power Digital Filter IC" IEEE journal of solid-state circuits, 39(2), 2004.
- [3]. Byonghyo Shim, Srinivasa R. Sridhara and Naresh, "Reliable Low-Power Digital Signal Processing via Reduced Precision Redundancy" IEEE transactions on very large scale integration (vlsi) systems, 12(5), 2004.
- [4]. Brian Richards, Farhana Sheikh, Melinda Miller, Dejan Marković, Borivo, "A 1–190MSample/s 8–64 Tap Energy- Efficient Reconfigurable FIR filters for Multi-Mode wireless communication" 2010.
- [5]. Swaroop Ghosh, Debabrata Mohapatra, Georgios Karakonstantis, Kaushik Roy, "Voltage Scalable High-Speed Robust Hybrid Arithmetic Units Using Adaptive Clocking" IEEE transactions on very large scale integration (vlsi) systems, 18(9), 2010.
- [6]. Parag Kulkarni, Puneet Gupta, milos Ercegovac, "Trading Accuracy for power with an Under designed Multiplier Architecture" 2011.
- [7]. Khaing Yin Kyaw, Wang Ling Goh, and Kiat Seng Yeo, "Low-Power High-Speed Multiplier For Error Tolerant Application" IEEE International Conference of Electron Devices and Solid-State Circuits 2011.
- [8]. Dursun Baran, Mustafa Aktan and Vojin G. Oklobdzija, "Multiplier Structures for Low Power Applications in Deep-CMOS". 2011
- [9]. Paul N. Whatmough, Shidhartha Das, David M. Bull, and Izzat Darwazeh, "Circuit-Level Timing Error Tolerance for Low-Power DSP Filters and Transforms" IEEE transactions on very large scale integration (vlsi) systems, 21(6), 2013.
- [10]. Francesco conti, chuck pilkingtony, rea marongiu_z and lucabenini Z, "Architectural Heterogeneity Exploration on a Scalable Many-Core Platform." 2013.
- [11]. Pravin S Kartikeya Bhardwaj J'org Henkel, "Power-and Area-Efficient Approximate Wallace Tree Multiplier for Error-Resilient Systems" 2014