Low Power High Speed Accuracy Controllable Approximate Multiplier Design

Shivangi Ghanshyam Nagrikar[1], Prof. Dr. V. Jayashree[2]

[1]Student, Electronics Engineering, DKTE's Textile and Engineering Institute, Ichalkaranji, Maharashtra, India
[2]Professor, Electronics Engineering, DKTE's Textile and Engineering Institute, Ichalkaranji, Maharashtra, India

---***---

Abstract - Multiplication is a fundamental task for most error-tolerant applications. Estimated multiplication is considered an efficient technique for trading energy against performance and accuracy. This paper proposes accuracy-controllable multiplication whose final product is generated by a carry-maskable adder. The proposed plan can dynamically select the length of carry spread to meet the accuracy requirements. The partial product tree of the multiplier is estimated by the proposed tree compressor. An $8*8$ multiplier design is implemented by assigning the carry-maskable adder and compressor.

Key Words: accuracy-controllable, carry-maskable adder, multiplier, compressor, etc.

1. INTRODUCTION

Popular applications, such as image processing and recognition, can tolerate small inaccuracies. These applications require a lot of computations and multiplication is their fundamental arithmetic function, which creates requirement for trade off computational accuracy for reduced power consumption.

Traditionally there are different types of multipliers such as Booth multiplier, Wallace tree multiplier, Analog multiplier, Sequential multiplier, Array multiplier and Parallel multipliers etc. There have been many researches on multipliers which focuses on accuracy, power consumption and critical path delay and area. Depending on specific application some of them like Analog multipliers are used when product of two analog signals is required where accuracy can be compromised, Array multipliers and parallel multipliers are in high demand because of their high execution speed and consumes less area and circuitry is simple.

Approximate computing is best approach for fault-tolerant applications because it can compromise accuracy for power, and it currently plays a very vital role in such application sector. Different error-tolerant applications have different accuracy needs, as do different program phases in an application. If multiplication accuracy is constant, power will be wasted when high accuracy is not really needed. In different error-tolerant applications requiring different accuracies power can be compromised. For e.g. if accuracy for multiplication gains higher priority then power saving

above the set accuracy can be a boon to the application. Approximate computing has a great contribution here.

This means that approximate multipliers should be dynamically reconfigurable to cope up with the different accuracy requirements of different program phases and applications. This paper puts light on an approximate multiplier design that can control accuracy dynamically. A carry-maskable adder (CMA) that can be dynamically designed to function as a conventional carry propagation adder (CPA), a set of bit-parallel OR gates, or a combination of the two. This configurability can be achieved by masking carry propagation: the CPA in the last stage of the multiplier is changed by the proposed CMA. An approximate tree compressor is put to use to reduce the accumulation layer depth of the partial product tree. Our method introduces a term representing the power and accuracy requirements which simplifies the partial product reduction (PPR) component as needed. So, the literature review based on reduction of power and improved speed are discussed further.

2. LITERATURE SURVEY

K. C. Bickerstaff, E. E. Swartzlander, and M. J. Schulte[1] proposed a method on Analysis of column compression multipliers. The column compression technique reduces the delay in the circuit. Column compression multipliers are faster than array multipliers. The paper studies the area, delay and power characteristics of Dadda and Wallace multipliers and found that the ratios of power to area increases with operand word length due to longer interconnect lines and may also increase the chances of occurrence of fault.

M.S. Lau and et al. [3] presented a paper on design and analysis of energy aware probabilistic multiplier. In this they have incorporated the principle of relaxing the energy requirement by allowing incorrect computation which helped to achieve a good trade-off between energy consumption and correctness of the outputs.

A report on Bio-inspired Imprecise Computational blocks (BICs) structures to implement a three-layer face recognition neural network efficiently has been reported by H. R. Mahdiani et al[4]. These BIC structures are found to more efficient in terms of area, speed, and power consumption with respect to their precise rivals. They have presented a

sample BIC adder and multiplier structures with error behaviors and synthesis results.

J. Liang, J. Han, and F. Lombardi[5] proposed New metrics for the reliability of approximate and probabilistic adders. Using the new metrics several adders were implemented namely LOA(Lower part OR adder), AMA(Approximate mirror Adder), PFA(Probabilistic Full Adder) and compared to baseline implementation LIA(Lower Bit Ignored Adder). Simulation results have proven that compared to probabilistic adders PFA(Probablistic Full Adder) the approximate adders LOA, and AMAs are advantageous in terms of power saving but low in precision. Whereas PFA provide high precision. The proposed metrics may be used for inexact computing where minor fault can be tolerated.

S. Venkataramani et.al[6] have reported a method of Quality programmable vector processors for approximate computing. In this paper they have discussed Approximate computing techniques leave exact (numerical or Boolean) equivalence in the execution of some of the applications output quality is acceptable. Approximate Computing provides designers with new version of optimized design by exploiting the intrinsic resilience of applications. Results demonstrate that the levarging quality programmability leads to significant improvement in energy efficiency.

C. Liu, J. Han, and F. Lombardi[7] published A Low-Power, High-Performance approximate multiplier with configurable partial error recovery. This multiplier uses a new type of approximate adder that limits the carry propogation to the nearest neighbor for quick partial product accumulation, because of this the power consumption reduces as the critical path is shorter. The proposed multiplier has high accuracy and low error, lower power consumption as compared to traditional Wallace multiplier.

The Design and analysis of approximate compressors for multiplication by A. Momeni, J. Han and P. Montuschi et al.[8] deal with designing and and analysis of approximate 4-2 compressor for the use in multipliers. The proposed compressor is used in 4 different ways and studied for Dadda multiplier. The design thus implemented simulation results show that there is significant reduction in power dissipation, delay and the number of transistors compared to the exact multiplier design. This design can be used in fault tolerant applications where inexact computing can be used.

Z. Yang, J. Han, and F. Lombardi [9]proposed a method Approximate compressors for Error-Resilient multiplier design. In this paper 3 designs of approximate 4-2 compressor are proposed and these are used in partial product reduction circuit of multiplier, which gives very high accuracy, low power, less delay and reduced area with minimum error it can be used in image sharpening application.

S. Hashemi, R. I. Bahar, and S. Reda, [10] deviced a DRUM: A Dynamic Range Unbiased Multiplier for approximate applications. This design allows the designer to control power and and accuracy because of its unbiased nature there is great reduction in error and increased power saving. Utilizes core accurate multiplier. It uses the fact that not all bits of a number are important so we cut down the number of bits applied to the multiplier, which further reduces the hardware circuitry.

3. PROMBLE STATEMENT

Our aim is to design and implement an approximate multiplier which can provide accuracy but operate at high speed besides takes low power. Further our aim is to analyze the designed multiplier using Cadence tool for its power speed and delay.

4. OBJECTIVE

  i. To design and implement schematic and Layout of a power efficient high speed 8-bit multiplier using Virtuso of Cadence tool.
 ii. To perform DRC and LVS check to extract paracitics using ASSURA.
iii. To test performance of the implemented 8-bit for multiplier for different input combinations.
 iv. To analyze the power consumption, time delay and area using tools like Spectre, ADEL etc.
  v. To compare the performance with traditional 8-bit multiplier.

5. THEORETICAL BACKGROUND

Generally multiplier consists of three parts i) Generation of partial product using AND gate ii) PPR using an adder tree and iii) Use of CPA(Carry Propogation Adder) to generate addition to give the final result. But the Accuracy Controllable multiplier consists of i) (Incomplete Adder Cells(iCAC) ii) Approximate Tree Compressor(ATC) iii) Carry Maskable Adders(CMA).

i.   **CAC(Incomplete Adder Cells) :**
      An accurate half adder is shown in Fig.1 (a) equation for accurate half adder is as in (1);
      $\{c,s\}=a+b=2c+s=(c+s)+c$ -------------- (1)
      where $\{\}$ and + denote concatenation and addition, respectively. Where c is carry $c=a.b$ and s is sum $s=a\oplus b$, so $(C+S)$ can be generated by a+b.

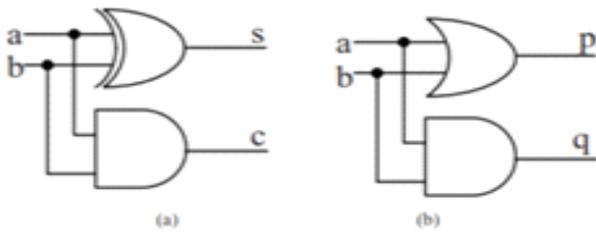**Fig-1:** (a) Accurate Half adder (b) Incomplete Adder Cell.

Inputs		Outputs			
		Accurate half adder		iCAC	
a	b	c	s	q	p
0	0	0	0	0	0
0	1	0	1	0	1
1	0	0	1	0	1
1	1	1	0	1	1

Table 1. Truth table for accurate half adder and Incomplete Adder Cell(iCAC).

$p=c+s$, $q=c$ and $\{c,s\} = a+b=p+q$ ………… (2)

This is called an incomplete adder cell (iCAC). Table1 shows the truth tables for an accurate half adder and an iCAC. The bit position of c and that of s, p, and q are different. We note that q=c. While p≠s, we can get the precise sum by p+q, so the iCAC is not an approximate adder but an element of a precise adder. By extending this concept to m bits, we get (3)

$S=A+B=P+Q$ …………………….………(3)

where A, B, P, and Q are n bit values, the bits of which correspond to a, b, p, and q, respectively. A row of eight iCACs, used for 8-bit inputs, is shown in Fig. 2.
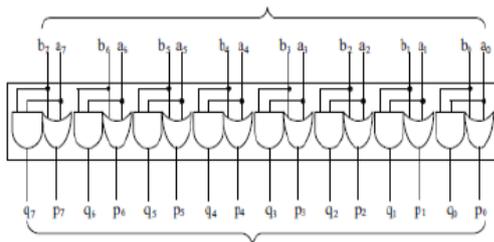

Fig. 2. A row of incomplete adder cells with two 8-bit inputs

Consider the example of an 8-bit adder with the two inputs. Consider an example A= 01011111 and B =00110110. The accurate sum S is 10010101, while the row of iCACs produces P = 01111111 and Q = 00010110. Again, it is evident that (4) holds.

$S=P+Q$ ………………..…... (4)

Here S is obtained from P+Q, P can be used as an approximation for S, and Q can be used as an error recovery vector for the approximate sum P.

ii.   Approximate Tree Compressor(ATC):
Consider a two 8-bit inputs A={a7,a6,a5,a4,a3,a2,a1,a0} and B={b7,b6,b5,b4,b3,b2,b1,b0}
Two 8-bit outputs are
Approximate sum : P={p7,p6,p5,p4,p3,p2,p1,p0} and
Error recovery vector :Q={q7,q6,q5,q4,q3,q2,q1,q0}

By extending the row of iCACs from two to n ie. 8 inputs, n/2 ie. 4 Ps and n /2 ie. 4 Qs are obtained. The number of Qs is decreased to one when the sum of the 4 Qs is used instead of the 4 Qs themselves. We see that always P≥S, and Q=C. By changing these facts, OR gates can be used to generate the approximate sum of the 4 Qs without significant loss of accuracy. This approximate sum is called the Accuracy Compensation Vector(ACV) and is referred as V. This method is named approximate tree compressor (ATC). An ATC with 'n' inputs is called an ATC-n , and the structure of an ATC with eight inputs (ATC-8) is shown in Fig. 3.
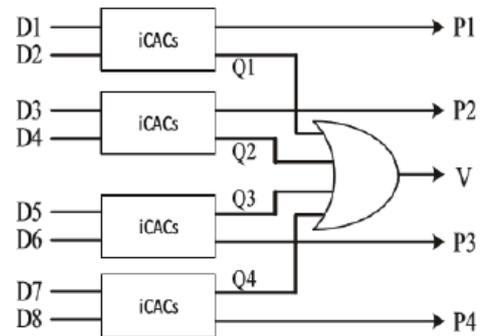

Fig. 3. Structure of an approximate tree compressor with eight inputs.

The rectangles represent rows of iCACs and the number of iCACs in each row (rectangle) is dependent on the bit width of the inputs. Consider, if there are 8-bit inputs (D1, D2, …, D8), four rows of m iCACs are required to build a 8-bit ATC-8. This reconstruction gives P1, P2, P3, and P4 as approximate sum and Q1, Q2, Q3, and Q4 as Error recovery vector. Accuracy compensation vector V is generated by OR gates. This is how, the eight inputs have been reduced to five.

iii.   Carry-maskable Adder(CMA):
A CMA controls the accuracy flexibly and dynamically. A k-bit CMA consists of (k-1) carry-maskable full adders and one carry-maskable half adder, and its structure is same as that of a k-bit CPA(Carry Propogation Adder). The structures of carry-maskable half and full adders are shown in Fig. 4. There working is as mentioned in the Table 2.
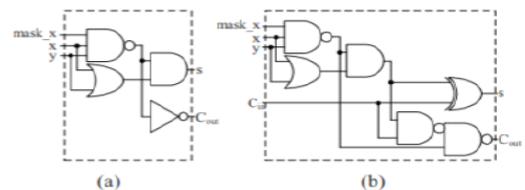

Fig. 4. (a) Carry-maskable half adder, (b) Carry-maskable full adder.

Type of Adder	inputs x y	mask_x	S	cout	Out-come
H.A.	-	0	x+y	0	Works as an OR gate with output S
	-	1	x⊕y	x.y	Accurate half adder
F.A.	-	0	x+y	Cin	Works as an OR gate with output S
	-	1	x⊕y	x.y	Accurate full adder

Table2. Working of CMA half adder and CMA full adder

6. OVERALL STRUCTURE

The working of overall structure of Approximate multiplier is as shown in Fig.5. It works in 4 stages. The block diagram shows what operations are performed in each stage. The operation in each stage is as discussed further. Stage 1 consists of (i) ATC-8 (ii)ATC-4 & (iii)iCACs. ATC-8 reduces 8 rows of Partial product(PP) to 4 PP and 1 Accuracy compensation vector(ACV), V1 and subsequently ATC-4 reduces 4 PP to 2 PP and ACV, V2 then iCAC gives 1 PP from 2 PP and Error recovery vector Q7. Stage 2 produces approximate sum of V1+ V2 using 7 OR gates and 4 rows are compressed to 3. In Stage 3, eleven FAs are used for bits 2 to12 to compress the 3 rows to 2 and 2 HAs are used for bits 1 & 13. In stage 4, to reduce the length of carry propagation, the bits are divided into 3 parts; the upper bits 12 to 14 are defined as accurate bits and accurate adders are put to use to produce these bits and the least significant bits are truncated since they are not so important the middle bits 5 to 11 are accuracy-controllable bits for which 7- bit CMA are used.
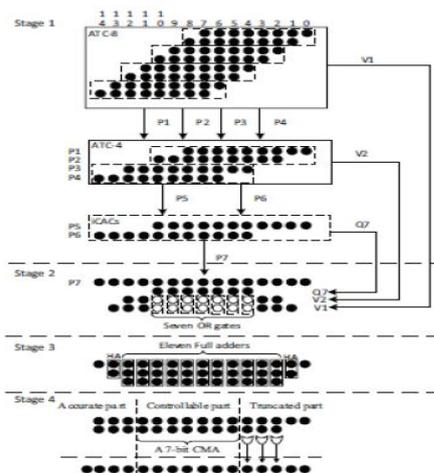

Fig 5(a) : Structure of a bit multiplier with 8*8 partial products

The part between truncated part and accurate part is the accuracy-controllable part. From the point of view of delay and accuracy this part plays a vital role. In stage 4 in CMA each bit S which is output of 2-inutp OR-gate the power consumed by the circuit is reduced as the switching activity is reduced in some logic gates.
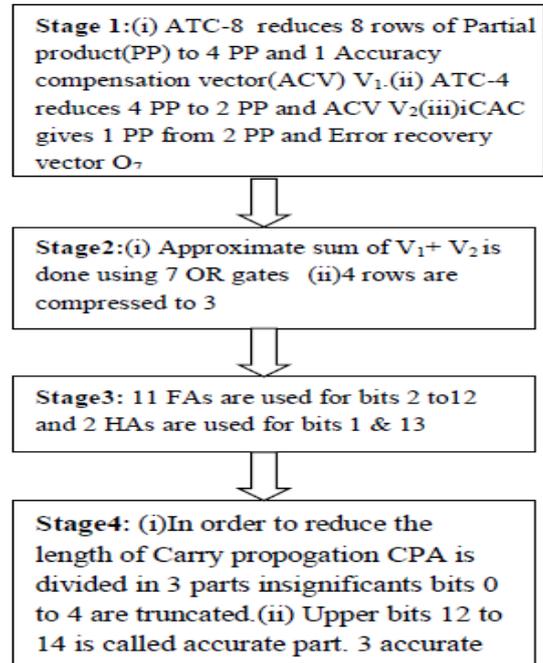

Stage 1:(i) ATC-8 reduces 8 rows of Partial product(PP) to 4 PP and 1 Accuracy compensation vector(ACV) V_1.(ii) ATC-4 reduces 4 PP to 2 PP and ACV V_2(iii)iCAC gives 1 PP from 2 PP and Error recovery vector O_7

Stage2:(i) Approximate sum of $V_1+ V_2$ is done using 7 OR gates (ii)4 rows are compressed to 3

Stage3: 11 FAs are used for bits 2 to12 and 2 HAs are used for bits 1 & 13

Stage4: (i)In order to reduce the length of Carry propogation CPA is divided in 3 parts insignificants bits 0 to 4 are truncated.(ii) Upper bits 12 to 14 is called accurate part. 3 accurate

Fig 5(b): Flow chart of a bit multiplier with 8*8 partial products

7. PERFORMANCE PARAMETERS

Various performance parameters such as error distance(ED), Mean Error distance(MED), Relative Error Distance (RED), Mean Relative Error Distance (MRED), Normalised Error distance (NRED) are used which are explained further.

Error Distance (ED): It is the arithmetic difference between the accurate product(S) and the approximate product(S').
$$ED=|S-S'| \qquad .... \qquad (5)$$
Mean Error distance(MED): It is the average ED for set of outputs.
$$MED=|S–S'|+|S1– S1'|+| S2 –S2' |+\cdots+| Sn –Sn'|n \quad (6)$$
RED: It is the ratio of Relative ED to Accurate output.
$$RED=|S–S'|S \qquad .... \qquad (7)$$
MRED: It is the Mean Relative ED.
$$MRED = RED1+RED2+....+REDnn \qquad ....(8)$$
NMED: Normalised Mean Error Distance
$$NMED= MEDSmax \qquad .... \qquad (9)$$
where Smax is the maximum output
ER: Error Rate is the percentage of inaccurate outputs among all outputs generated from all combinations of inputs.

8. METHODOLOGY OF IMPLEMENTATION

a. Design flow of low power high speed accuracy controllable multiplier is as shown in the flow chart of Fig. 6 which is self-explanatory.
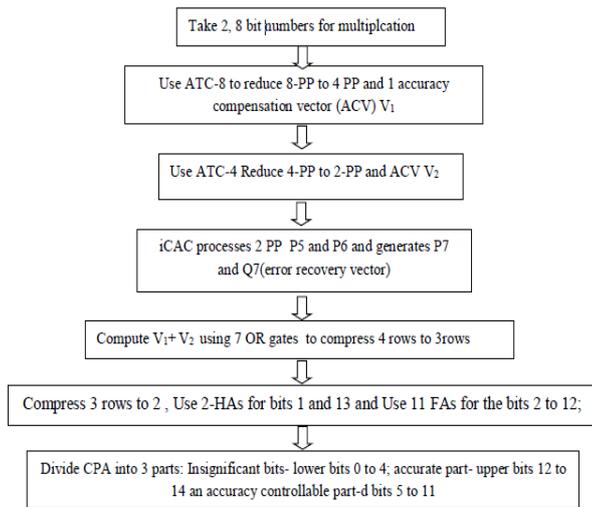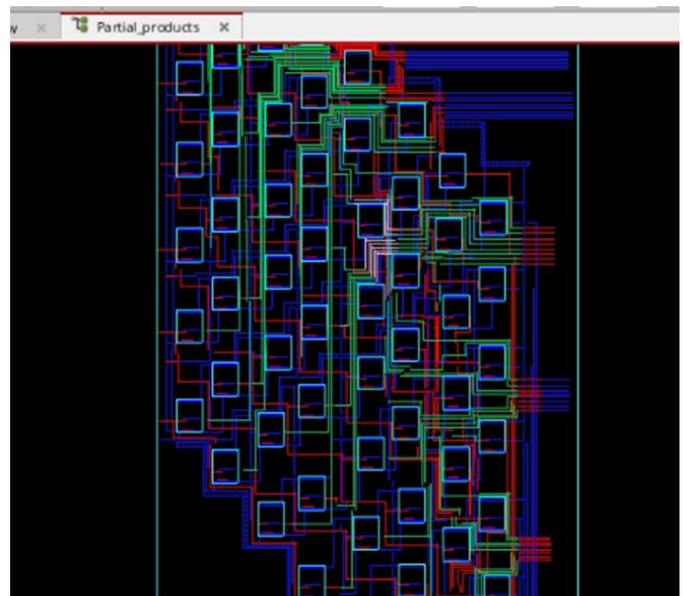
Fig 6. Design Flow chart of ACM

b. Steps for implementation of ACM:

The steps for implementation of a 8-bit low power high speed accuracy controllable approximate multiplier to be carried out in CADENCE EDA tool are presented further.

i. Design and implement 1bit iCAC, ATC, CMA, HA and FA in Cadence using AMS design flow or verilog ASIC flow and form components.
ii. Using the 4 iCACs as component and 4 bit OR gate build an 8-bit ATC.
iii. Build &7-bit CMA.
iv. Using 7-bit CMA build 3 Accurate adders.
v. Integrate the components in steps ii-iv to build a 8-bit low power high speed accuracy controllable approximate multiplier.
vi. Analyse the performance of this 8-bit multiplier.
vii. Compute the performance parameters such as ED, MED, RED, MRED, NMED, ER using the formula mentioned in equation (5)-(9).
viii. Compare the performances of 8-bit low power high speed accuracy controllable approximate multiplier with 8-bit Wallace-tree multiplier.
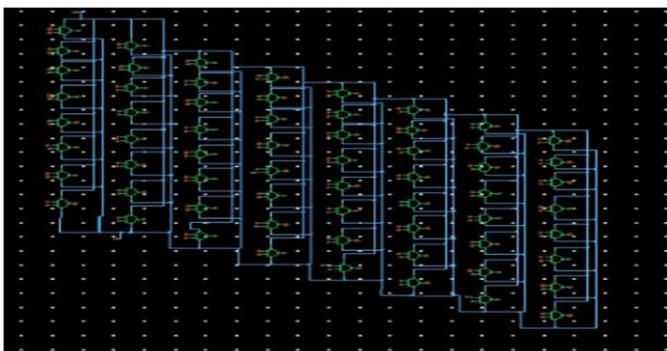
9. RESULTS

Fig-7: Schematic of Partial Product generation designed in virtuoso of Cadence tool

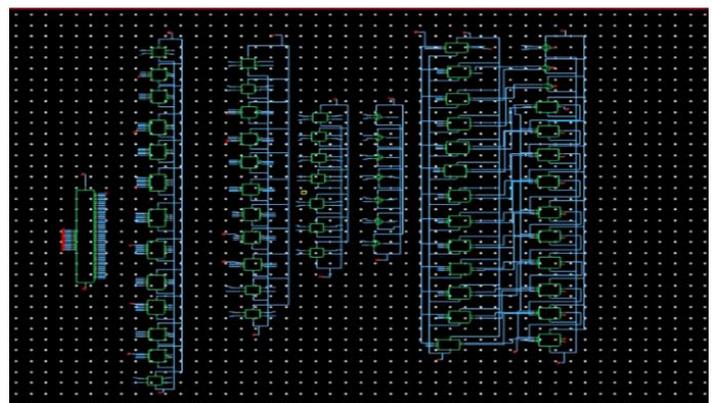
Fig-8: Layout of Partial Product design

Fig-9: Schematic of full module of the low power high speed accuracy controllable approximate multiplier design.
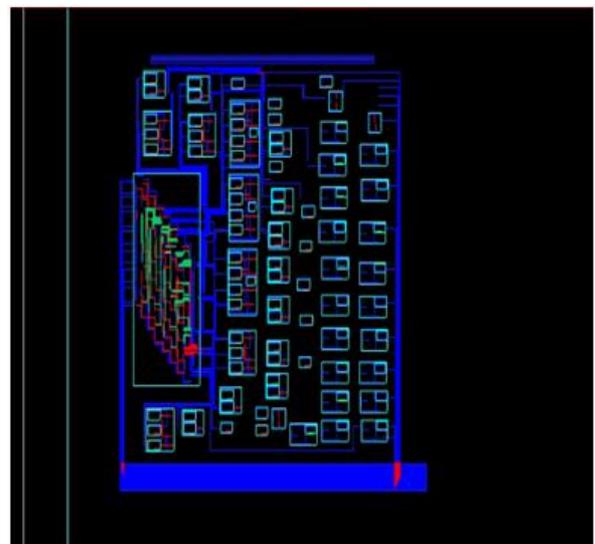

Fig-10: Layout of full module of the low power high speed accuracy controllable approximate multiplier design

REFERENCES

[1] K. C. Bickerstaff, E. E. Swartzlander, and M. J. Schulte, "Analysis of column compression multipliers," 15th IEEE Symposium on Computer Arithmetic, pp. 33-39, Jun. 2001.

[2] NanGate, Inc. NanGate FreePDK45 Open Cell Library, http://www.nangate.com/?page_id=2325, 2008.

[3] M. S. Lau, K. V. Ling, and Y. C. Chu, "Energy-Aware probabilistic multiplier: Design and Analysis," 2009 international Conferrence on Compliers, architeture, and synthesis for embedded systems, pp. 281-290, Oct. 2009.

[4] H. R. Mahdiani, A. Ahmadi, S. M. Fakhraie, and C. Lucas, "Bio-Inspired imprecise computational blocks for efficient VLSI implementation of Soft-Computing applications," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 57, no. 4, pp. 850-862, Apr. 2010.

[5] S. Venkataramani, V. K. Chippa, S. T. Chakradhar, K. Roy, and A. Raghunathan. "Quality programmable vector processors for approximate computing," 46th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO), pp. 1-12, Dec. 2013.

[6] C. Liu, J. Han, and F. Lombardi, "A Low-Power, High-Performance approximate multiplier with configurable partial error recovery," Design, Automation & Test in Europe Conference & Exhibition (DATE), Mar.2014

[7] Momeni, J. Han, P. Montuschi, and F. Lombardi, "Design and analysis of approximate compressors for multiplication," IEEE Transactions on Computers, vol. 64, no. 4, pp. 984-994, Apr. 2015.

[8] Z. Yang, J. Han and F. Lombardi, "Approximate compressors for error-resilient multiplier design," 2015 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS), Amherst, MA, 2015.

[9] J. Liang, J. Han, and F. Lombardi, "New metrics for the reliability of approximate and probabilistic adders," IEEE Transactions on computers, vol. 62, no. 9, pp. 1760-1771, Sep. 2013.

[10] S. Hashemi, R. I. Bahar, and S. Reda, "DRUM: A Dyanamica Range Unbiased Multiplier for approximate applications," IEEE/ACM International Conference on Computer-Aided Design (ICCAD), pp. 418- 425, Nov. 2015.

[11] T. Yang, T. Ukezono and T. Sato, "A low-power high-speed accuracy-controllable approximate multiplier design," 2018 IEEE 23rd Asia and South Pacific Design Automation Conference (ASP-DAC), Jeju, 2018, pp. 605-610.