Project report submitted for

**Vth Semester Minor Project-I**

**in**

**Department of Electronics and Communication Engineering**

By,

**Group No. 7**

**Ananya Garg (16101008)**

**Purvi Agrawal (16101042)**

**Ruchi Dhamnani (16101045)**

****

**Department of Electronics and Communication Engineering**

**Dr. Shyama Prasad Mukherjee**

**International Institute of Information Technology, Naya Raipur**

**(A Joint Initiative of Govt. of Chhattisgarh and NTPC)**

**Email:** [iiitnr@iiitnr.ac.in](mailto:iiitnr@iiitnr.ac.in)**, Tel: (0771) 2474040, Web:** [www.iiitnr.ac.in](http://www.iiitnr.ac.in)

**CERTIFICATE**

This is to certify that the project titled “Approximate Computing Techniques for Deep Neural Networks” by “Purvi Agrawal, Ananya Garg , Ruchi Dhamnani” has been carried out under my/our supervision and that this work has not been submitted elsewhere for a degree.

(Signature of Guide)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dr. Santosh Kumar

**Assistant Professor**

**Department of Computer and Science Engineering**

**Dr. SPM IIIT-NR**

**Month, Year**

**Declaration**

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature of Author)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Author Name**

**(Roll Number)**

**Date : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Approval Sheet**

This project report entitled “Approximate Computing Techniques for deep Neural network” by “Ananya Garg, Ruchi Dhamnani, Purvi Agrawal” is approved for Vth Semester Minor Project.

(Signature of Examiner - I)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dr Anurag Singh

(Signature of Examiner - II)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dr Santosh Kumar

Signature of Chair)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dr Ankit Chaudhary (Chair)

Date: \_\_\_\_\_\_\_\_\_\_\_\_ Place: \_\_\_\_\_\_\_\_\_\_\_\_

**ABSTRACT:**

Deep neural networks are widely used for many artificial intelligence (AI) applications including computer vision, speech recognition, and robotics. DNNs are employed in a myriad of applications but comes at the cost of high computational complexity. Accordingly, techniques that enable efficient processing of DNNs to improve energy efficiency and accuracy should be used. To minimize energy consumption while maintaining throughput, we have used approximate computing technique which possibly results in an inaccurate result but can be useful for applications where approximate computations are enough for its purpose and they don't require perfect accuracy. The approximate techniques which have used reduce the power consumption significantly although accuracy is compromised to some extent.

At the same time it is very much useful for improving efficiency and reducing energy consumption. We have designed different multipliers for calculating power consumption using Xilinx Vivado.

The proposed design of multiplier uses multiplexer so that allows us to highly simplify calculations as it allows inclusion and exclusion of any number of bit so it increases the flexibility as well as reduces power consumption.

**Table of Contents**

**Title Page No.**

**ABSTRACT………………………………………………………….6**

**TABLE OF CONTENTS………………………………………....... 7**

**LIST OF TABLES…………………………………………………..8**

**LIST OF FIGURES………………………………………………....9**

**CHAPTER 1 INTRODUCTION 10**

* 1. Motivation Behind Our Work……………………………………………………………....10
  2. CNN………………………….……………………………………………………………...11
  3. Max Pool Layer……………………………………………………………………………..12

1.4 ReLu ………………………………………………………………………………………...12

**CHAPTER 2 LITERATURE REVIEW 13**

* 1. Low power 1-Bit CMOS Full Adder Cell.……………………………………………………...14
  2. A Two's Complement Parallel Array Multiplication Algorithm……………………………….14

2.3 Area Efficient Multipliers for Digital Signal Processing Applications………………………………15

**CHAPTER 3 PROPOSED SOLUTION 17**

* 1. Bit Width Adaptive Computing Unit……………………………………………………….18
  2. Near Zero Approximation Techniques …………………………………………………..…18

**CHAPTER 4 RESULTS 19**

**ACKNOWLEDGEMENT**

**REFERENCES**

**List of All Tables**

|  |  |  |
| --- | --- | --- |
| **Table No.** | **Table Title** | **Page Number** |
| 1.1 | Summarisation of Literature Review | 13 |
| 2.1 | Comparison of Power | 19 |

**List of All Figures**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Figure Title** | **Page Number** |
| 1. | Neural Network Mapping | 11 |
| 2. | Maxpool Layer | 12 |
| 3. | Relu | 12 |
| 4. | Conventional Architecture of CNN | 12 |
| 5. | Adder Cell | 14 |
| 6. | Two’s Complement Multiplication | 15 |
| 7. | Multiplier Diagram | 16 |
| 8. | Baugh Wooley Multiplier | 17 |
| 9. | Bit Width Adaptive Computing | 18 |
| 10. | Near Zero Approximation | 18 |
| 11. | Power consumption 2\*2 Multiplier | 19 |
| 12. | Power consumption MAC | 20 |
| 13. | Schematic of MAC | 20 |
| 14. | RTL Schematic MAC | 21 |
| 15. | VIO Schematic MAC | 22 |

**CHAPTER 1**

**Introduction**

Deep learning systems and all the more explicitly ConvNets or convolutional neural systems (CNNs) have come up as best in class arrangement calculations, accomplishing close human execution in applications in Computer vision and speech recognition. ConvNets have been utilized to accomplish exceptional precision for tasks extending from manually written digit acknowledgment, scene understanding and even video acknowledgment. Despite the fact that these systems are to a great degree amazing, they are additionally computationally and memory escalated, requiring several megabytes for channel weight storage and a huge number of activities per input. This mind-boggling expense makes them difficult to utilize on battery-constrained frameworks. In any case, the energy utilization of neural systems can be fundamentally decreased by exploring various algorithm. We will be using different multipliers for this purpose and comparing their energy consumption and accuracy. And we will be implementing it on Field Programmable Gate Array (FPGA). On later stage we will detect images and compare the power consumption using conventional multipliers and the multipliers which we have proposed.

First, two’s complement parallel multiplication can be explored in [1] which focuses on every partial product bit which has a positive coefficient but it requires complements of each multiplier and multiplicand bit for forming partial product bits. In [2] it saves the power consumption upto 15% but processed in 90nm technology and in [3] accuracy is reduced because of truncation of whole bits in right side. In [4] power consumption is significantly low but delay overhead is much more.

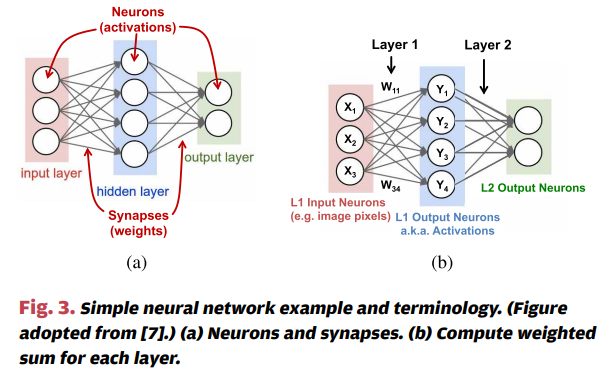
**1.1 Motivation Behind Our Work:**

Approximate computing technique which results in an inaccurate result and can be useful for applications where approximate computations are enough for its purpose and they don't require perfect accuracy. But at the same time it is very much useful for improving efficiency (approx double) and reducing energy consumption. We will be using different adders and multipliers for this purpose and comparing their energy consumption and accuracy. And later we will be implementing it on Field Programmable Gate Array (FPGA) and Zed Board.

1.2. **Convolutional Neural Network**

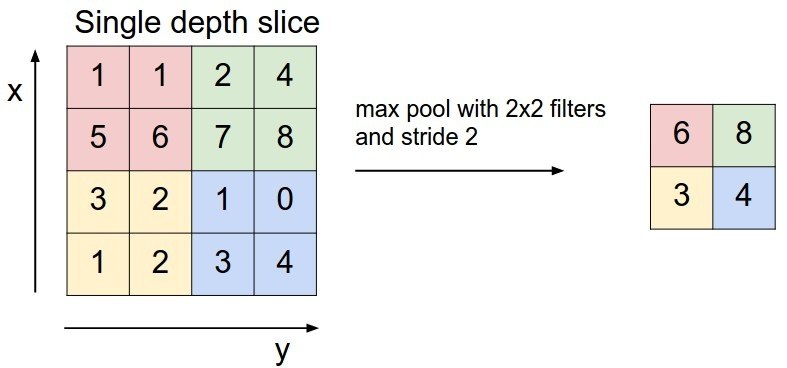
We have used CNN since ConvNets are fault tolerant and can be operated at low computational precision with very limited accuracy loss. CNN are made up of neurons which have weight and biases. Each neuron receives several inputs, takes a weighted sum over them, pass it through an activation function and responds with an output. The neurons in the input layer receive some values and propagate them to the neurons in the middle layer of the network, which is called as “hidden layer.” The weighted sums from one or more hidden layers are propagated to the output layer, which presents the final outputs of the network to the user. In the specific case of DNNs, this learning involves determining the value of the weights (and bias) in the network, and is referred to as training the network.

There are a few distinct types of Layers (e.g. CONV/FC/RELU/ POOL are by far the most popular). Each Layer accepts an input 3D volume and transforms it to an output 3D volume through a differentiable function.

Figure 1: Neural Network Mapping  


**1.3 MaxPool Layer**:

Max pooling is an example based discretization process. Its function is to dynamically decrease the spatial size of the portrayal to diminish the measure of parameters and calculation in the system. Pooling layer works on each element independently.

**Figure 2: Maxpool Layer**

**1.4 RELU (Rectified linear unit)**

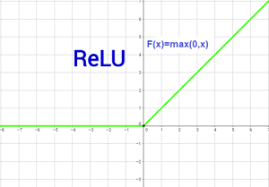
It is basically an activation function purpose of this is to determine output of neural network between 0 to 1 or -1 to 1.Convolution Layer:

Figure 3: RELU

**CHAPTER 2**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S No.** | **Title** | **Link** | **Year** | **Advantage** | **Disadvantage** |
| 1) | A Two's Complement Parallel Array Multiplication algorithm | <https://ieeexplore.ieee.org/document/1672241> | 1974 | Every partial product bit has a positive coefficient. | Need for the complements of each multiplier and multiplicand bit. |
| 2) | A Flexible Low Power DSP With a Programmable Truncated Multiplier | <https://ieeexplore.ieee.org/document/6329993> | 2012 | Conservative power savings of up to 15% | Architecture fabricated on the TSMC 90 nm process area consumption is more |
| 3) | Area efficient multipliers for Digital Signal Processing  Application | https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4864  55 | 1996 | Area consumption is reduced to 50% | Due to truncation accuracy is reduced |
| 4) | Design of Reconfigurable Low-Power Pipelined Array  Multiplier | <https://ieeexplore.ieee.org/document/4064379> | 2006 | Low Power Consumption | Delay overhead |
| 5) | Real-time Low-energy Fall Detection Algorithm  with a Programmable Truncated MAC | <https://ieeexplore.ieee.org/iel5/5608545/5625939/05626244.pdf> | 2010 | 23% power savings without any accuracy loss. | Truncation of low amounts of columns (< 15) result in robust gap values |

**Literature Review**

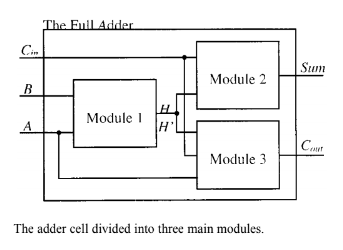
**Table 1: Summarization of Literature review**

**2.1 Low power 1-Bit CMOS Full Adder Cell**

The adder is part of the critical path that determines the overall performance of the system. That is why enhancing the performance of the 1-bit full-adder cell is a significant goal cells that exhibits

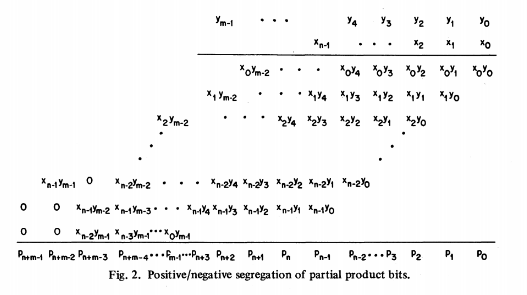
Different power consumption, speed, area, and driving capability are analyzed. We break the large circuit in smaller modules. For each module we try to minimize the dynamic and static power consumption.

Figure 5: Adder Cell



**2.2 A Two's Complement Parallel Array Multiplication Algorithm:**

* An algorithm for high-speed, two's complement, m-bit by n-bit parallel array multiplication is proposed. The difficulty in two's complement multiplication lies with the signs of the multiplicand and the multiplier. In this method the signs of all the partial product bits are positive. The product is formed by adding the first n - 2 partial product rows and subtracting the last two rows. Instead of subtracting the partial products that have negative signs, the negation of the partial products can be added.

 Figure 6: Two’s complement Multiplication

The two advantages of this uniformity are as follows:

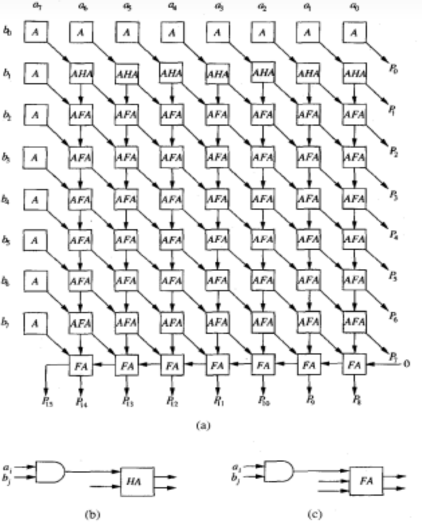
* The partial product bits are obtained by forming the AND of a multiplier bit and a multiplicand bit.
* Every partial product bit has a positive coefficient. Therefore, the product is formed with only the AND function and the ADD function. No subtraction is necessary, nor is the NAND function needed to form.

Disadvantage of the proposed algorithm is the need for the complements of each multiplier and multiplicand bit in forming the partial product bits.

2.3 Area Efficient Multipliers for Digital Signal Processing Applications

The proposed method a truncated multiplier is described which is an area-efficient parallel sign-magnitude multiplier that receives two N-bit numbers and produces an N-bit product. N bit signals are multiplied by N bit coefficients resulting into a 2N bit product which must be quantized to N bits by eliminating the N least significant bits (LSB ’s) by some quantization scheme (e.g.

truncation or rounding). The quantization of the product to N bits is achieved by omitting about half the adder cells needed to add the partial products but in order to keep the quantization error to a minimum, probabilistic biases are obtained and are then fed to the inputs of the retained adder cells.

Figure7: Multiplier Diagram.

**CHAPTER 3**

**Proposed Solutions**

We have implemented 2\*2, 4\*4, 8\*8 and 16\*16 signed multipliers and compared the power consumption of all of them. And later we have included the near zero approximation technique to compute how much lower the power consumption has been after introducing it. We have also implemented using two’s complement multiplication in that we have used Baugh Wooley multipliers and NAND gate for partial product of signed bit part. After that adding 1 to first and last row in the left side. It is found that signed multiplication consumes less power as compare to two’s complement method.

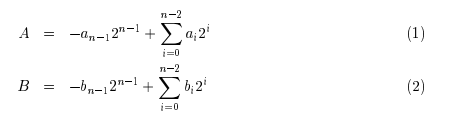
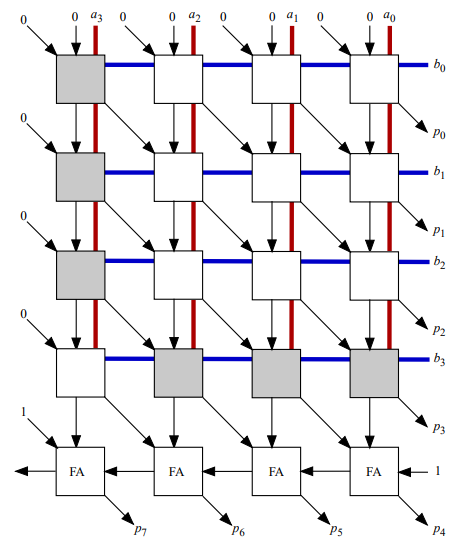
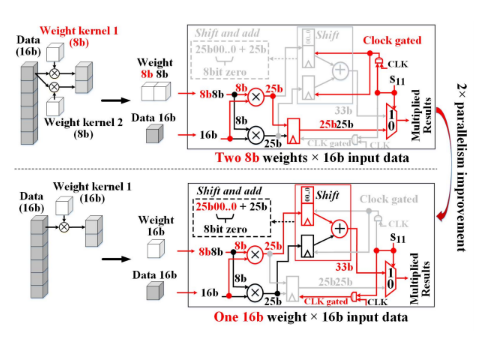


Figure 8: Baugh Wooley Multiplier****

**3.1 Bit Width Adaptive Computing Unit:**

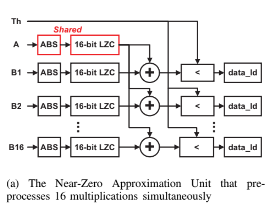
When the bit-width of weight is larger than 8-bit, these two multipliers are combined as one 16 × 16-bit multiplier. The results from two multipliers are sent to the shift and add unit. The result of high 25-bit part is shifted by 8-bit with zero filling. Then it adds with the low 25-bit part. By setting S11 as 1, the result is output. Compared to the multipliers in our design are all utilized in either 8-bit mode or 16-bit mode. Therefore, our design has **higher resource utilization.**

Figure 9: Bit Width Adaptive Computing

**3.2 Near Zero Approximation Techniques:**

* The power reduction is realized by avoiding multiplications of near-zero valued data.The near-zero approximation are proposed to predict and skip the near-zero multiplications under certain thresholds.
* Compared with skipping zero-valued computations, this design achieves 1.92X and 1.51X further reduction of the total multiplications with negligible loss of accuracy.
* The accelerator, designed using it consumes less energy approximately 717 times less energy and is comparatively 4 times faster.

Figure 10: Near Zero Approximation



**Chapter 4**

**Power Comparison of different Multipliers:**

| **SNo.** | **Power comparison of different multipliers** | |
| --- | --- | --- |
| ***Multipliers*** | ***Power Consumption*** |
| 1.) | 2x2 signed | 1.59W |
| 2.) | 2x2 using two’s complement | 2.868W |
| 3.) | 4x4 signed | 6.064W |
| 4.) | 8x8 signed | 11.45W |
| 5.) | 16x16 signed | 30.479W |
| 6.) | 16x1 signed | 4.612W |
| 7.) | 16x2 signed | 10.79W |
| 8.) | 16x4 signed | 18.28W |

**Table 2: Power Comparison of different multiplier**

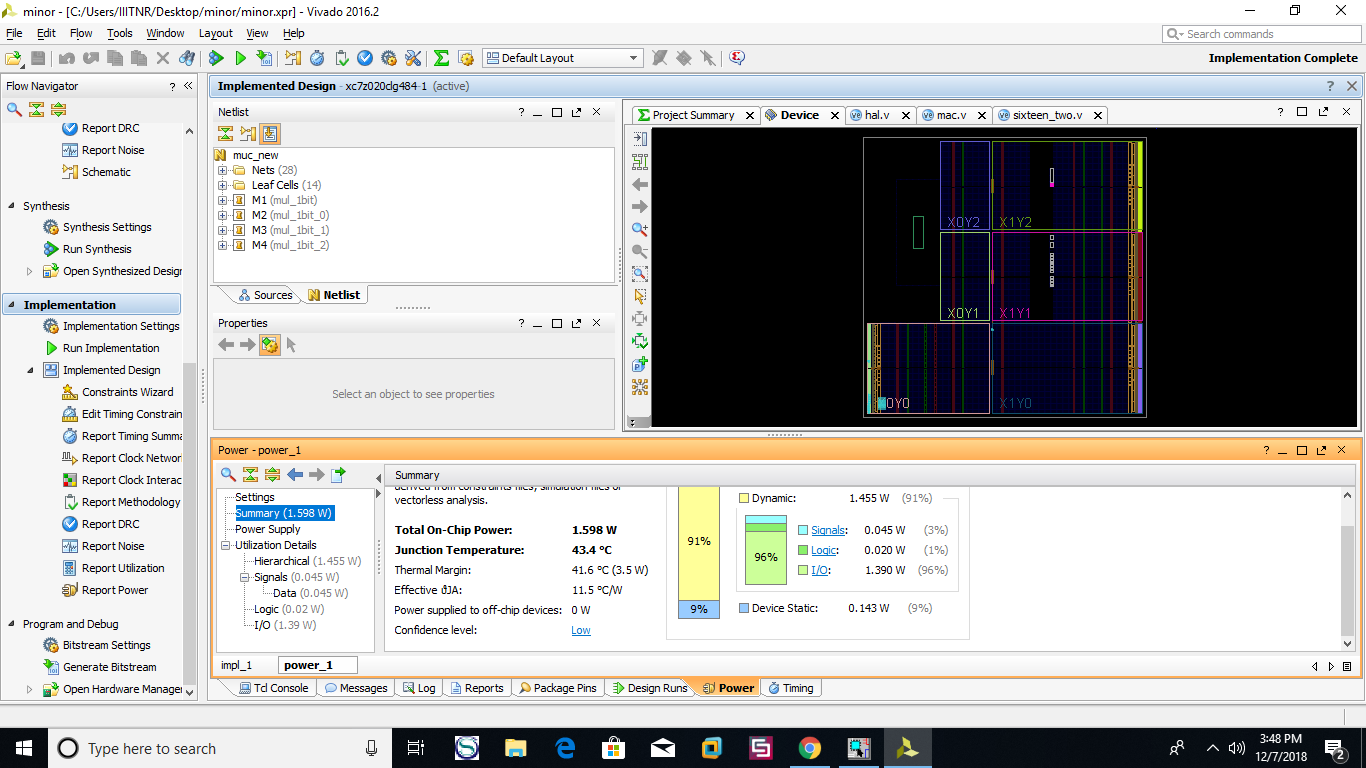
**Power consumption Result**:

Figure 11: Power consumption 2\*2 multiplier

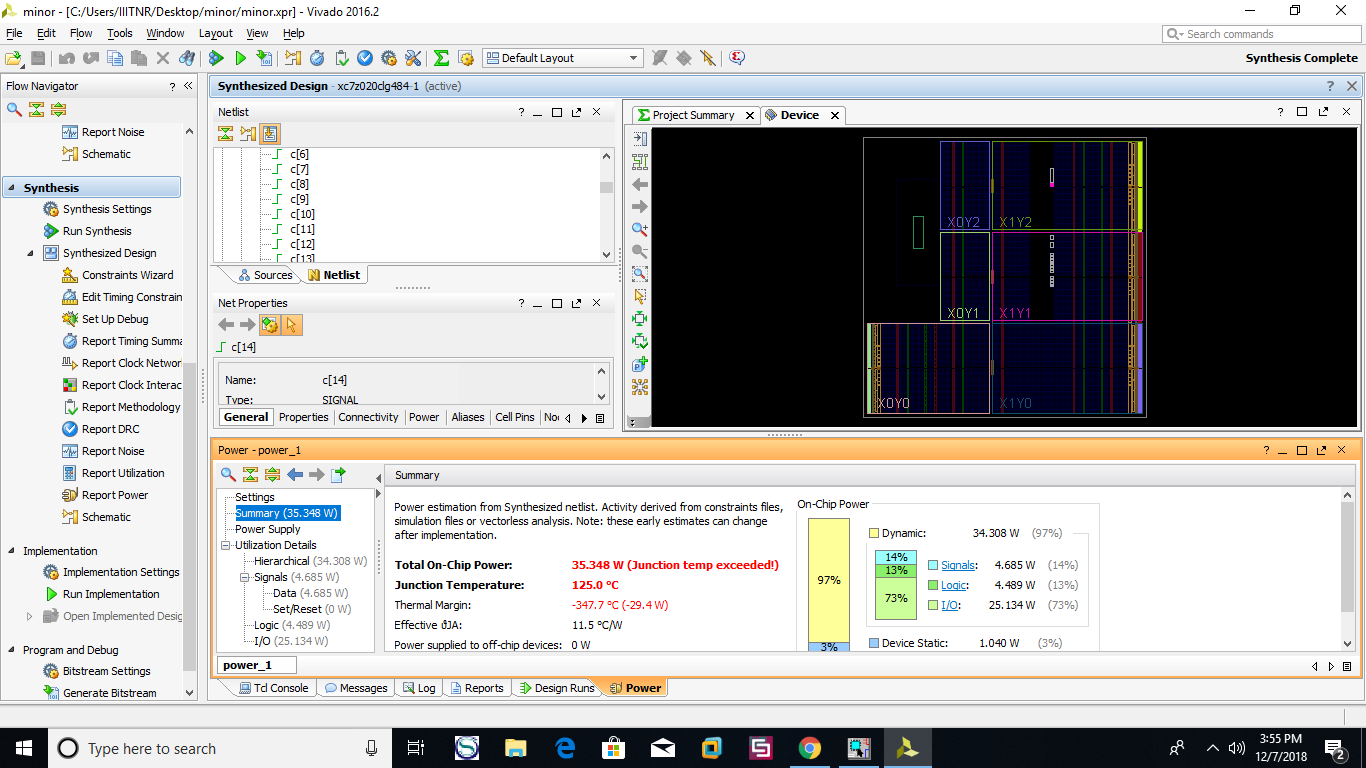
**Power Consumption of MAC**:

Figure 12: Power consumption MAC

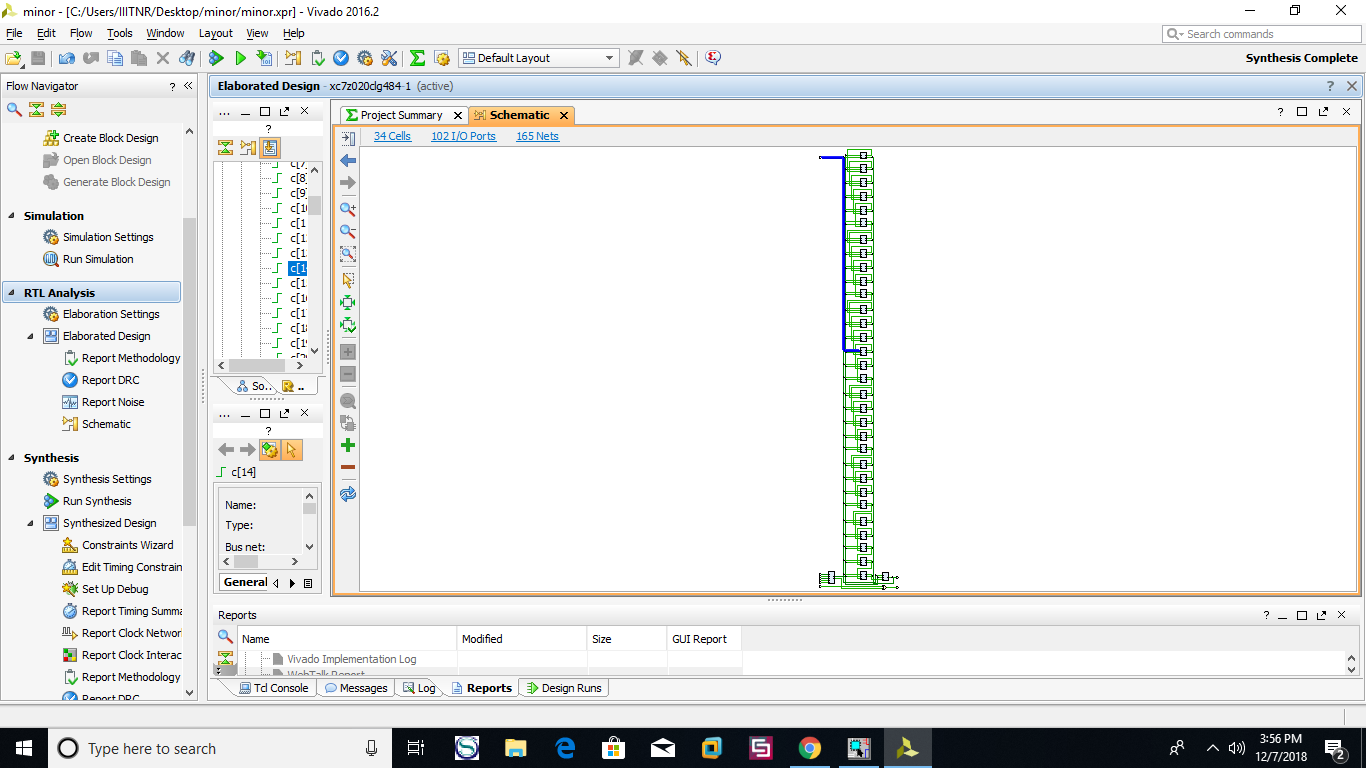
**Schematic of MAC**:

Figure 13: Schematic MAC

**Virtual Input Output (VIO)**:

module main\_vio(  
input clk,  
output [7:0]sum);  
wire [15:0] a;  
wire [15:0] b;  
wire [31:0] c;  
wire clk,asign,bsign,csign;  
wire [32:0] m;  
wire sign;  
mac g2(a,b,c,clk,asign,bsign,csign,m,sign);  
vio\_0 V1(clk,m,sign,a,b,c,asign,bsign,csign);  
assign m[7:0]=sum;  
endmodule

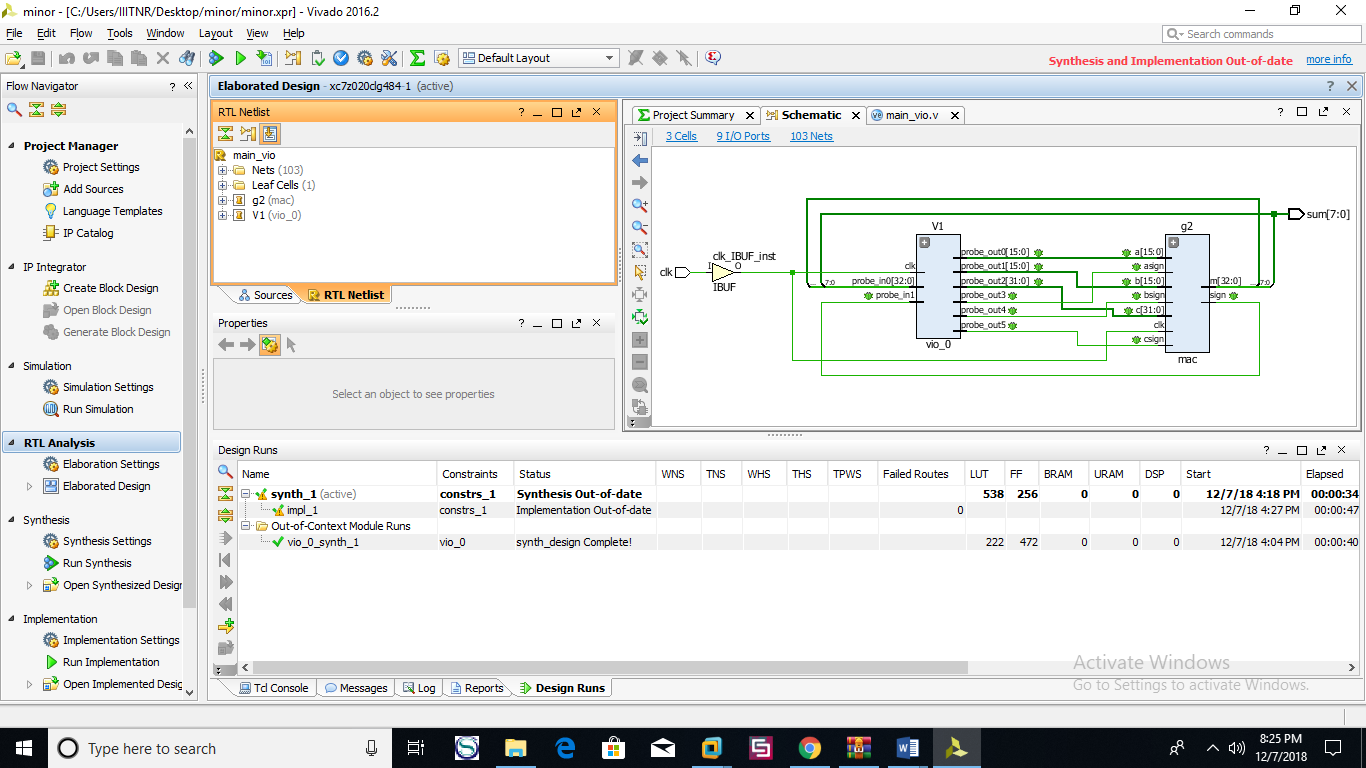
We have dumped the multiplier and accumulator in ZED Board using Virtual Input Output Pins. The results are as below:

Figure 14: RTL Schematic MAC

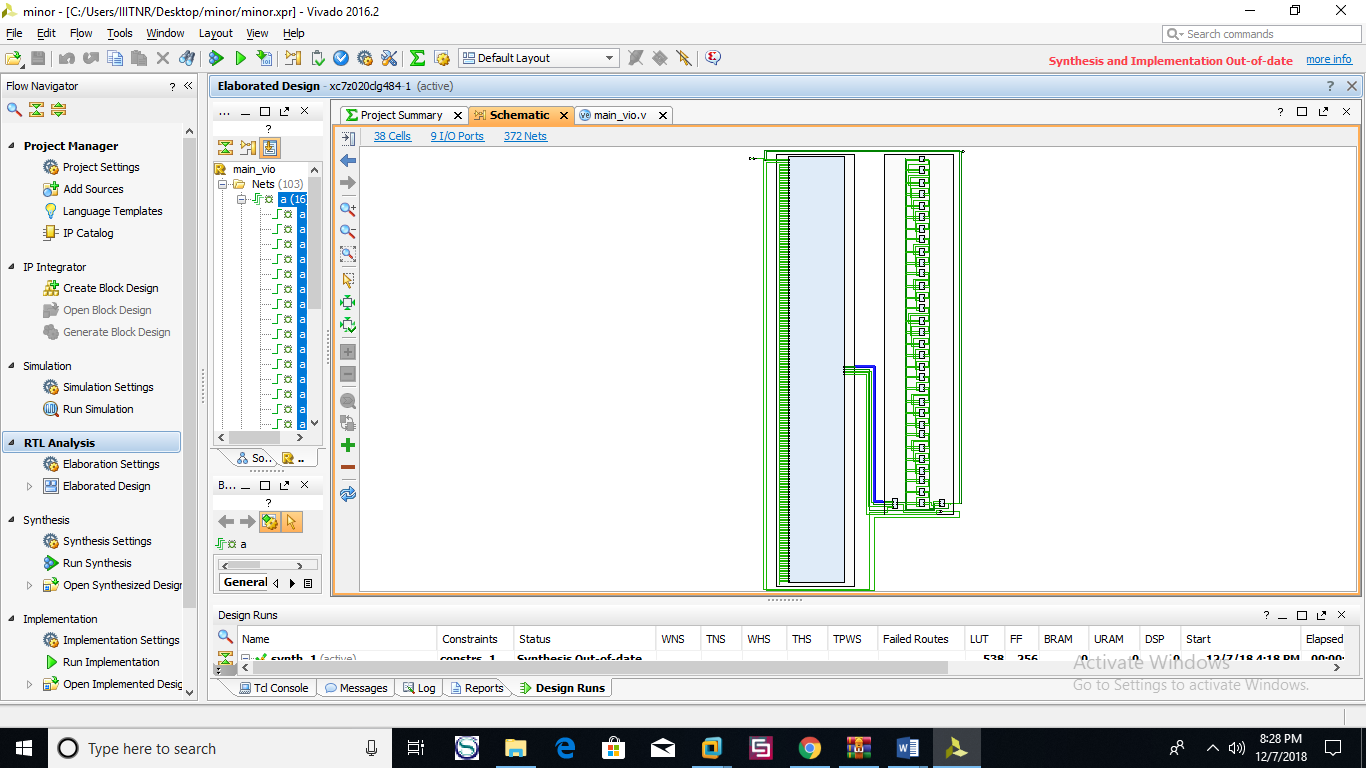


Figure 15: VIO Schematic MAC

Acknowledgement:

The success and outcome of this project till now required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along during our project.

We all respect and thank Dr. Ramesh Vaddi (Senior Research Fellow National University of Singapore) for providing us constant support and mentoring for this project work and giving us all this wonderful opportunity to work with him.

**References:**

[1] C.R. Baugh and B.A. Wooley, “A Two’s Complement Parallel Array Multiplication Algorithm,” IEEE Trans. Computers, vol. 22, no. 12, pp. 1045-1047, Dec. 1973.

[2] M.J. Schulte and E.E. Swartzlander Jr., “Truncated Multiplication with Correction Constant,” Proc. Workshop Very Large Scale Integration (VLSI) Systems Signal Processing, VI, pp.

[3]Tarun Vatwani, Arko Dutt, Debjyoti Bhatacharjee , Anupam Chattopadhyay, “Floating Point Multiplication Mapping on ReRAM based In memory Computing Architecture,” [2018 31st International Conference on VLSI Design and 2018 17th International Conference on Embedded Systems (VLSID)](https://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=8326586).

[4] [Jiun-ping Wang](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=%25252522First%25252520Name%25252522:%25252522Jiun-ping%25252522&searchWithin=%25252522Last%25252520Name%25252522:%25252522Wang%25252522&newsearch=true&sortType=newest) ; [Shiann-rong Kuang](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=%25252522First%25252520Name%25252522:%25252522Shiann-rong%25252522&searchWithin=%25252522Last%25252520Name%25252522:%25252522Kuang%25252522&newsearch=true&sortType=newest) ; [Yuan-chih Chuang](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=%25252522First%25252520Name%25252522:%25252522Yuan-chih%25252522&searchWithin=%25252522Last%25252520Name%25252522:%25252522Chuang%25252522&newsearch=true&sortType=newest) “Design of Reconfigurable Low-Power Pipelined Array Multiplier” [2006 International Conference on Communications, Circuits and Systems](https://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=4063797).