Heaven's Light is Our Guide

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
Rajshahi University of Engineering & Technology, Bangladesh

FPGA based CNN classifier for Bangali Handwritten Character and Digit Recognition

Presented by

Khaled Zinnurine
Roll No. 1703026

S M Riad Hossain
Roll No. 1703088

Supervised by

Dr. Md. Nazrul Islam Mondal
Professor
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology

OUTLINE

MOTIVATION

- Over 300 Million people all over the world uses Bangali as native language and it also is the sixth most spoken language. Therefore, Bengali characters are broadly used in every aspect as handwritten characters. [16]
- Moreover, detecting or recognizing something is a central task in the field of AI, as artificial intelligence (AI) tries to mimic the human brain from the beginning. [14]
- Machines are especially good at recognizing digits. Various algorithms are used to perform this task, like K Nearest Neighbors, Support Vector Machines, Neural Networks, and Convolutional Neural Networks. [14]
- Artificial neural network (ANN) and convolutional neural network (CNN) classification models performed fairly well in detection. Both architectures are parallel in nature. [15]
- FPGAs, on the other hand, outperform CPUs and GPUs. It has better performance, supports higher levels of parallelism, consumes less power, and has reconfigurable features. [18]
- Since FPGAs are reconfigurable devices , we can customize the CNN/ANN algorithms by changing the circuits in the FPGA. The result is increased speed and efficiency [17], allowing CNN/ANN models to run faster than traditional CPUs.

LITERATURE REVIEW

Table 1: Literature Review

#	Paper Title	Year	Used Method & Result
1	"FPGA acceleration on a multi-layer perceptron neural network for digit recognition ."[4]	2017	❑ FPGA acceleration on a scalable multi-layer perceptron (MLP) neural network for classifying handwritten digits. ❑ Accuracy On MNIST- 93.25%
2	"Design of FPGA-Based Accelerator for Convolutional Neural Network under Heterogeneous Computing Framework with OpenCL "[7]	2019	❑ Used Method: Parallel acceleration of CNN based on FPGA with OpenCL, by using Xilinx SDAccel tools. ❑ Result: system processing speed were improved 2 times, where the overall speed reaches 48.5 fps.
3	"Handwritten Bangla Numeral and Basic Character Recognition Using Deep Convolutional Neural Network "[2]	2019	❑ A nine layer sequential Convolutional Neural Network model to recognize 60 (10 numerals+ 50 basic characters) Bangla handwritten characters ❑ On the BanglaLekha Isolated Database, Recognition accuracy-99.44% ❑ On their own dataset - 95.16%
4	"Convolutional-Neural-Network-Based Handwritten Character Recognition: An Approach with Massive Multi-source Data"[5]	2022	❑ Used Method: 'ADAM' and 'RMS_prop' optimizer in the proposed CNN model for handwritten English alphabet recognition. ❑ Result: Kaggle alphabet dataset --Accuracy 99.516% and MNIST digit dataset 99.642%

Table 2: Literature Review

#	Paper Title	Year	Used Method & Result
5	"BDNet: Bengali Handwritten Numeral Digit Recognition based on Densely connected Convolutional Neural Networks "**[1]**	2020	❑ Used method: Trained End-to-End with Untraditional data preprocessing and augmentation techniques. ❑ Result: 99.78% on ISI Bengali handwritten numeral dataset.
6	"An Automated System for Recognizing Isolated Handwritten Bangla Characters using Deep Convolutional Neural Network ."**[6]**	2021	❑ Used method: A deep novel Convolutional Neural Network (CNN) of 11 layers is proposed. Among the layers, 6 are convolutional layers. ❑ Result: CMATERdb dataset — 98.03%
7	"Convolutional neural network based handwritten bengali and bengali-english mixed numeral recognition."**[3]**	2016	❑ Used moderate preprocessing technique to generate patterns from images and then employs CNN ❑ Test Set Recog. Accuracy (98.45%) for CNN based Bengali handwritten numeral recognition and (98.71%) for CNN based Bengali-English mixed recognition.
8	"BengaliNet: A Low-Cost Novel Convolutional Neural Network for Bengali Handwritten Characters Recognition"**[8]**	2020	❑ Used method: a low-cost novel convolutional neural network architecture for the recognition of Bengali characters with only 2.24 to 2.43 million parameters based on the number of output. ❑ Result: BanglaLekha Numerals— 98.11% NumtaDB Numerals —97.50%

PROBLEM STATEMENT

Those problems are summarised below

- ❑ CPU is not stand-alone
- ❑ The CPU does a variety of other activities in addition to performing classification.
- ❑ Limited parallelism

THESIS CONTRIBUTION

Our work is mainly inspired by this BDNet model [8]. Our aim is to implement ANN and CNN models on hardware capable of accurately recognizing Bengali handwritten number digits while increasing overall computation speed.

The significant contributions of our thesis work are as follows:

- We have implemented the ANN and CNN architecture using Verilog.
- We have used Verilog implemented ANN and CNN models for Bengali digit recognition.

THESIS OBJECTIVES

Our thesis objectives are listed below.

- Designing Verilog-implemented ANN and CNN classifier models that can recognize Bengali hand-written numeral digits with significant accuracy.
- Configuring those models on FPGA that can be used as a stand-alone FPGA-based classifier device.

DATASET DESCRIPTION

Table 3: ISI Bengali handwritten numeral dataset description [9]

	0	1	2	3	4	5	6	7	8	9	Total
Train Samples	1933	1945	1945	1956	1945	1933	1930	1928	1932	1945	19392 (83%)
Test Samples	400	400	400	400	400	400	400	400	400	400	4000 (17%)
Total	2333	2345	2345	2356	2345	2333	2330	2328	2345	2345	**23392**

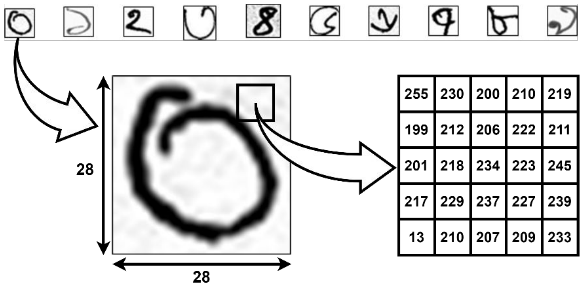


Fig. 1: Each sample

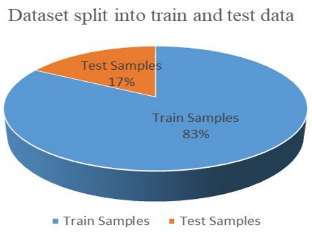


Fig. 2: Pie Chart for the train test dataset

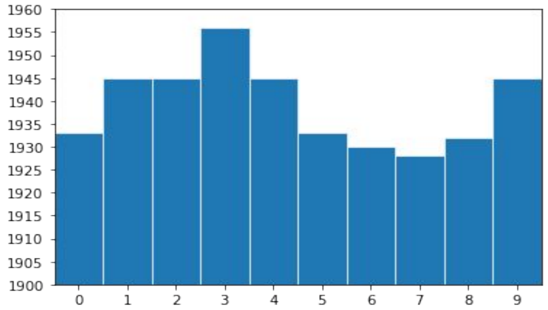


Fig. 3: Histogram of Train data

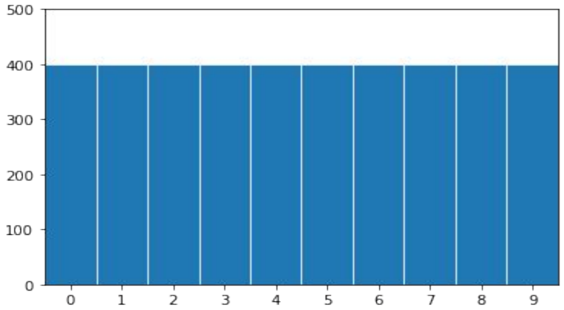


Fig. 4: Histogram of Test data

DATA PREPROCESSING

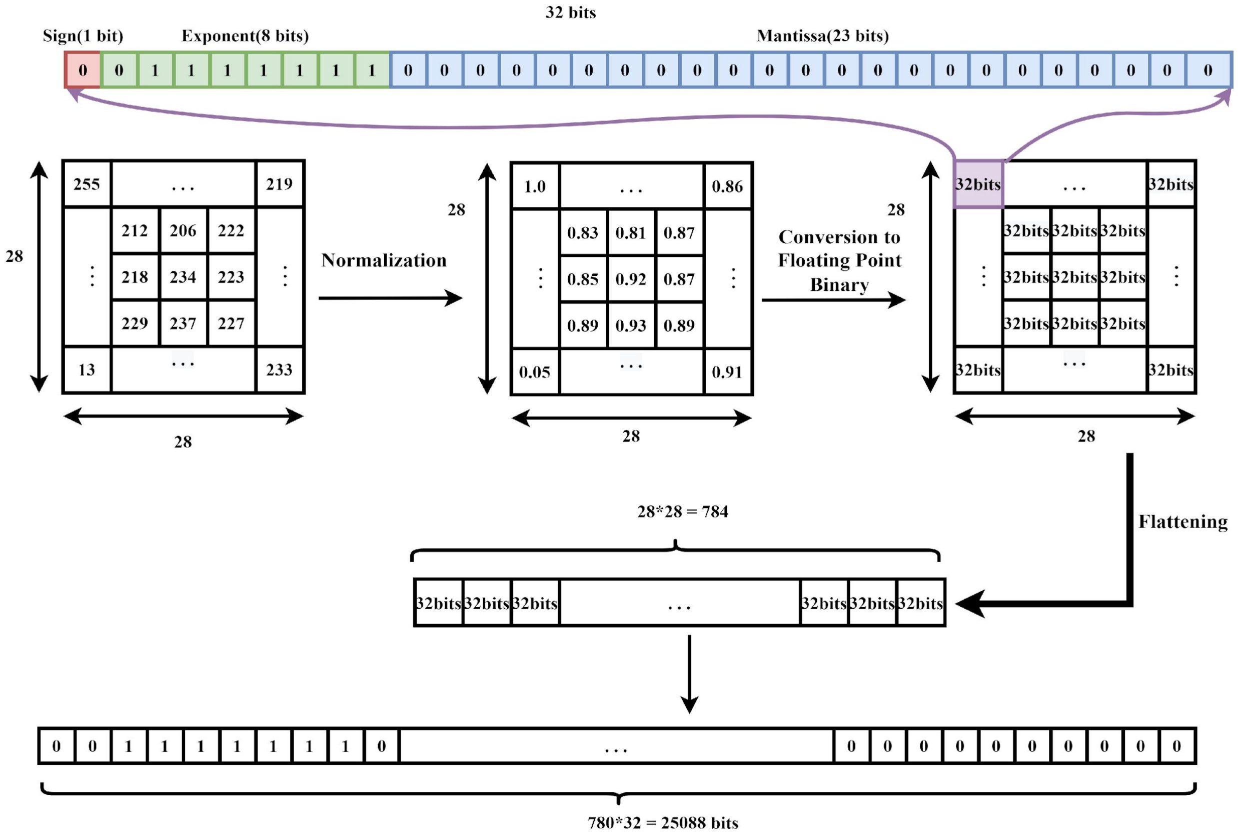


Fig. 5: Data Preprocessing

TOOLS

- **Hardware:**
  - FPGA: Xilinx Spartan-6 FPGA
- **Software:**
  - Visual Studio Code: For writing Verilog code.
  - Xilinx ISE (Integrated Synthesis Environment): For synthesis and analysis of HDL designs, performing the timing analysis, simulating a design's reaction to different stimuli, and configuring the target device with the programmer.
  - GTK Wave: To verify the test benches for Verilog modules
- **Language Support:**
  - Verilog Hardware Description Language
- **Algorithm:**
  - Convolutional Neural Network

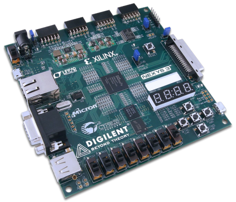


Fig. 6 : Digilent FPGA board [10]

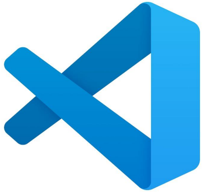
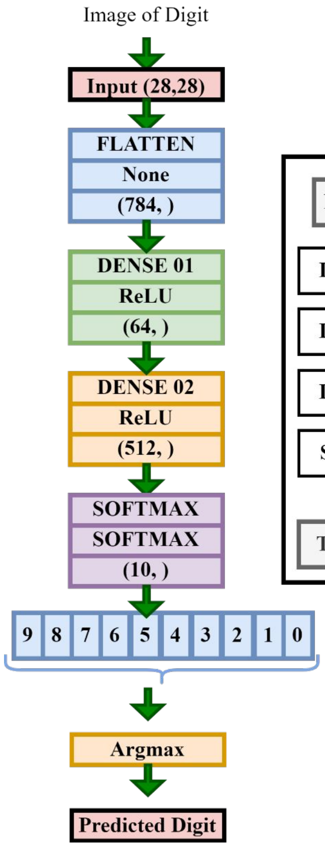


Fig. 7 : Visual Studio Code [11]

Fig. 8 : Xilinx ISE [12]

PROPOSED SYSTEM ARCHITECTURE



PARAMETERS(ANN)	
INPUT	0
DENSE 01	50240
DENSE 02	33280
SOFTMAX	5130
TOTAL	88650

Fig. 9: ANN architecture

PARAMETERS(CNN)	
CONV2D 01	320
MAXPOOL2D 01	0
CONV2D 02	18496
CONV2D 03	36928
MAXPOOL2D 02	0
FLATTEN	0
DENSE 01	102500
SOFTMAX	1010
TOTAL	159254

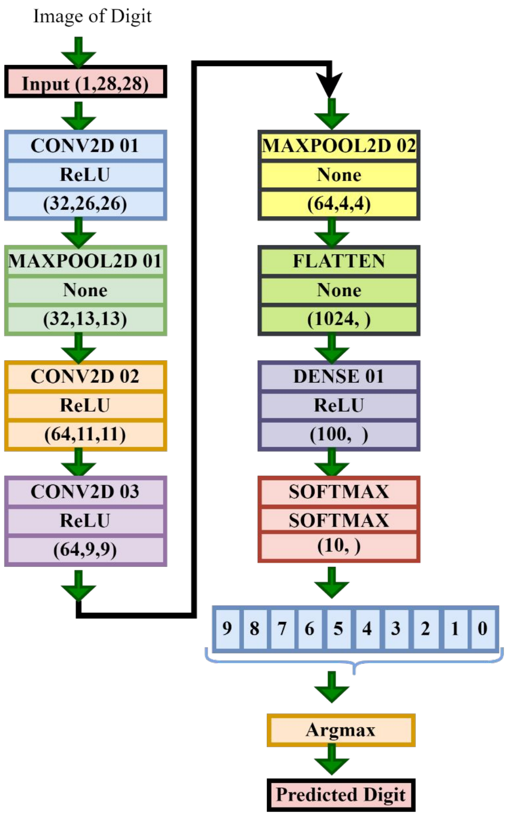


Fig. 10: CNN Architecture

IMPLEMENTATION

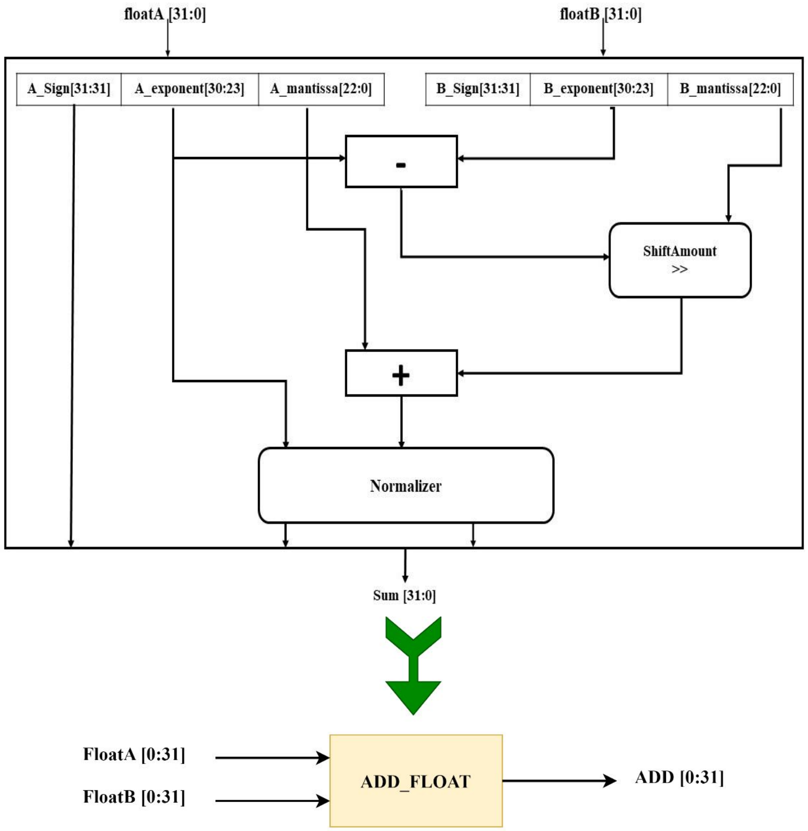


Fig. 11: Floating point addition

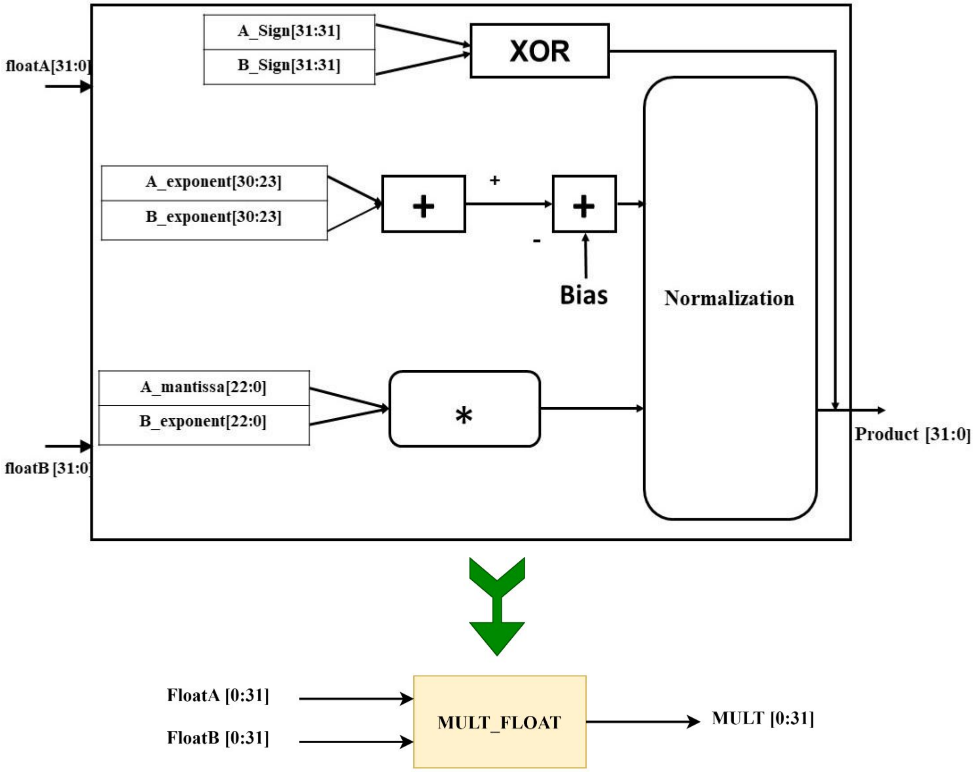


Fig. 12: Floating point multiplication

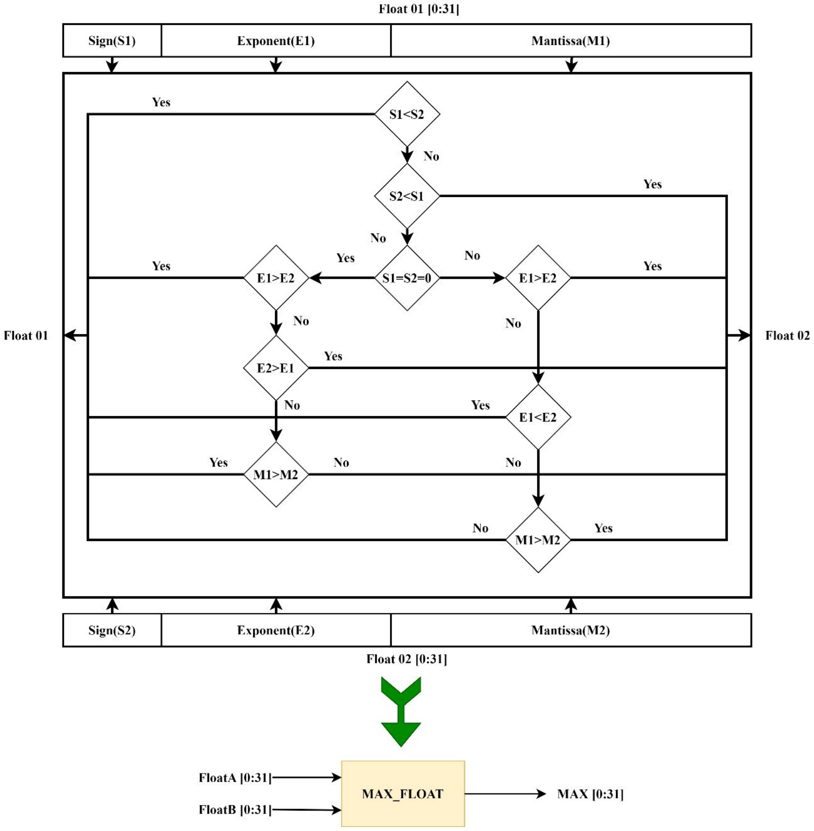


Fig. 13: Floating point max function

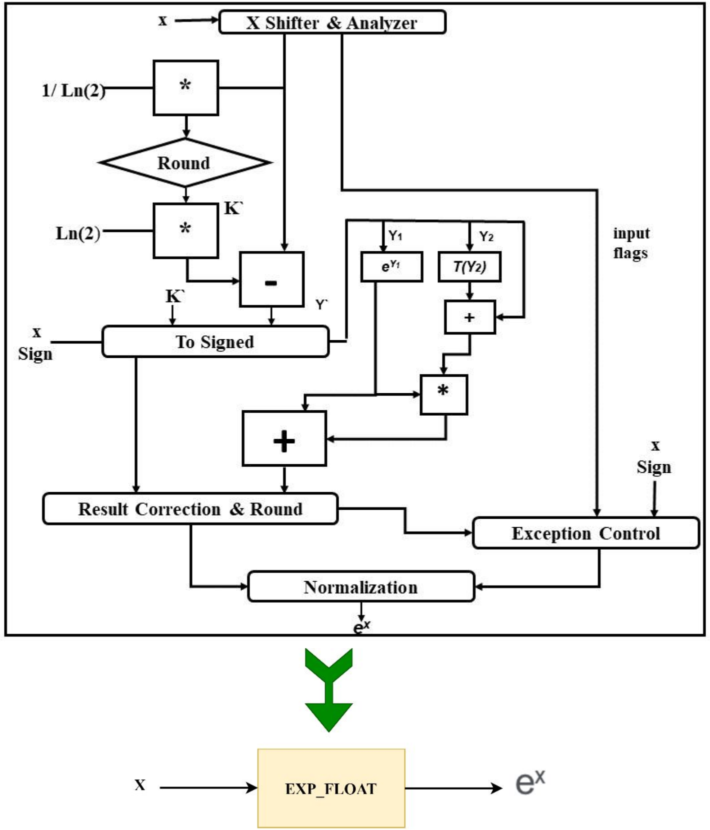


Fig. 14: Floating point Exponent [13]

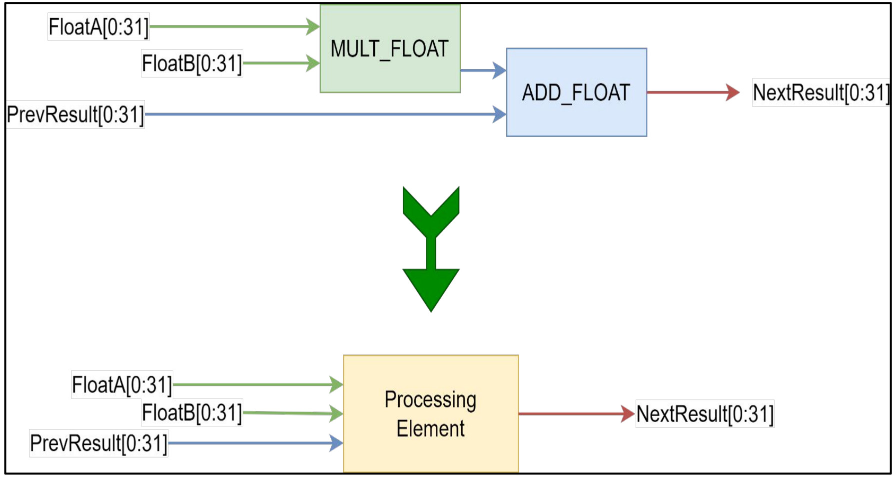


Fig. 15: Processing Unit

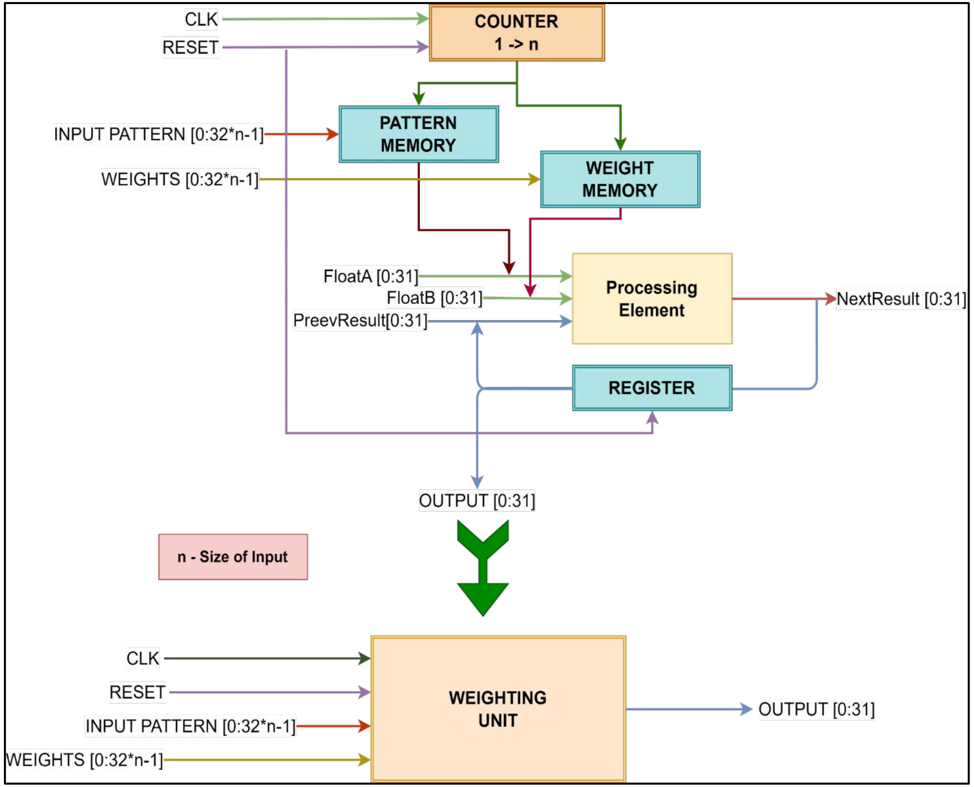


Fig. 16: Weighting Unit

IMPLEMENTATION

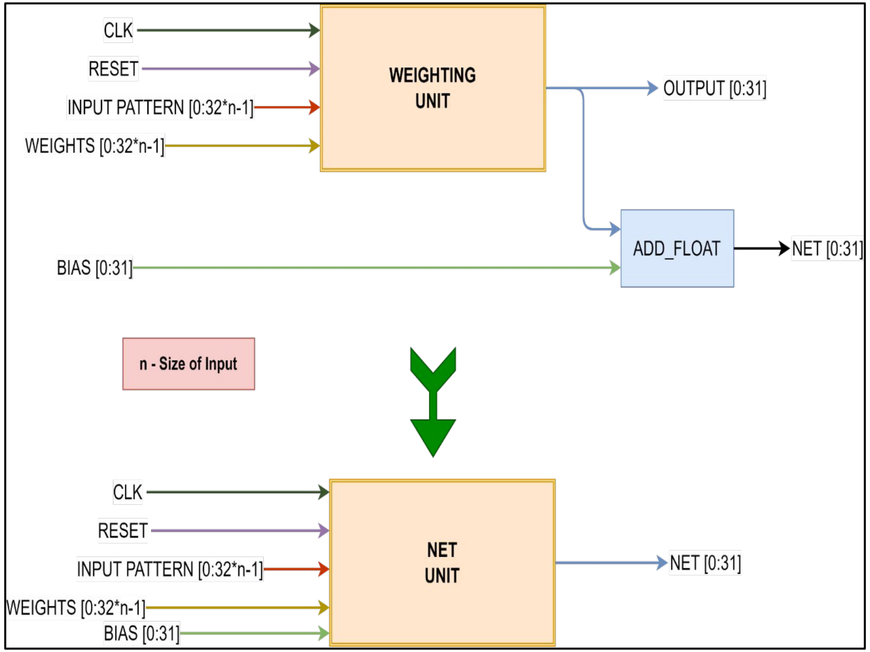


Fig. 17: NET Unit

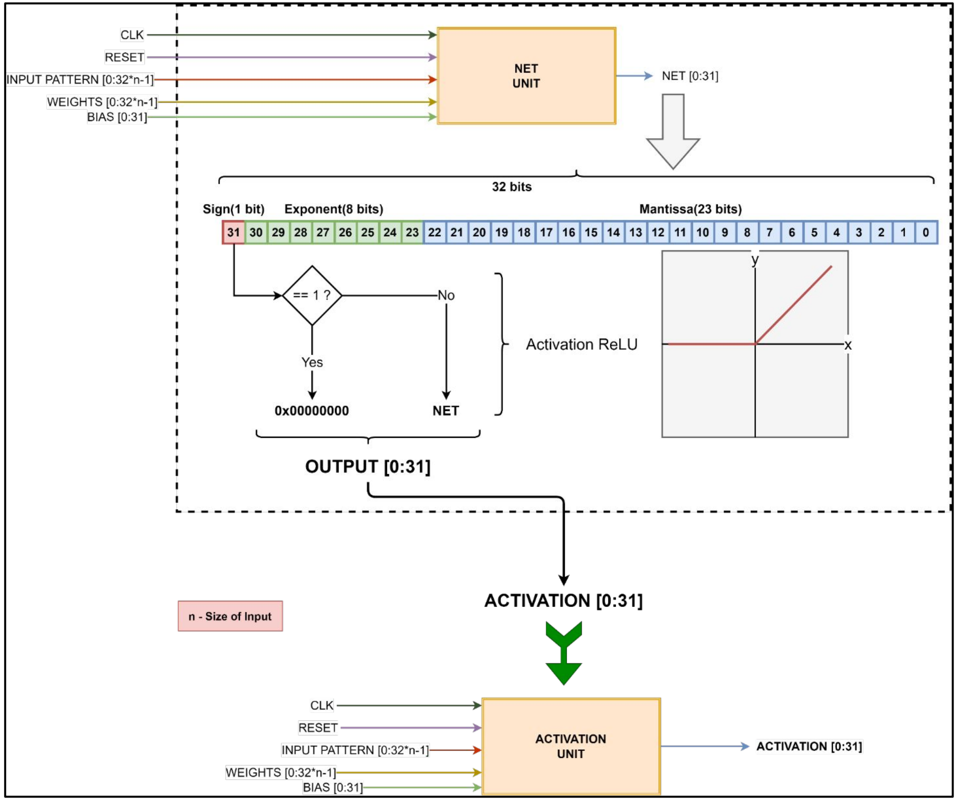


Fig. 18: Activation Unit

IMPLEMENTATION

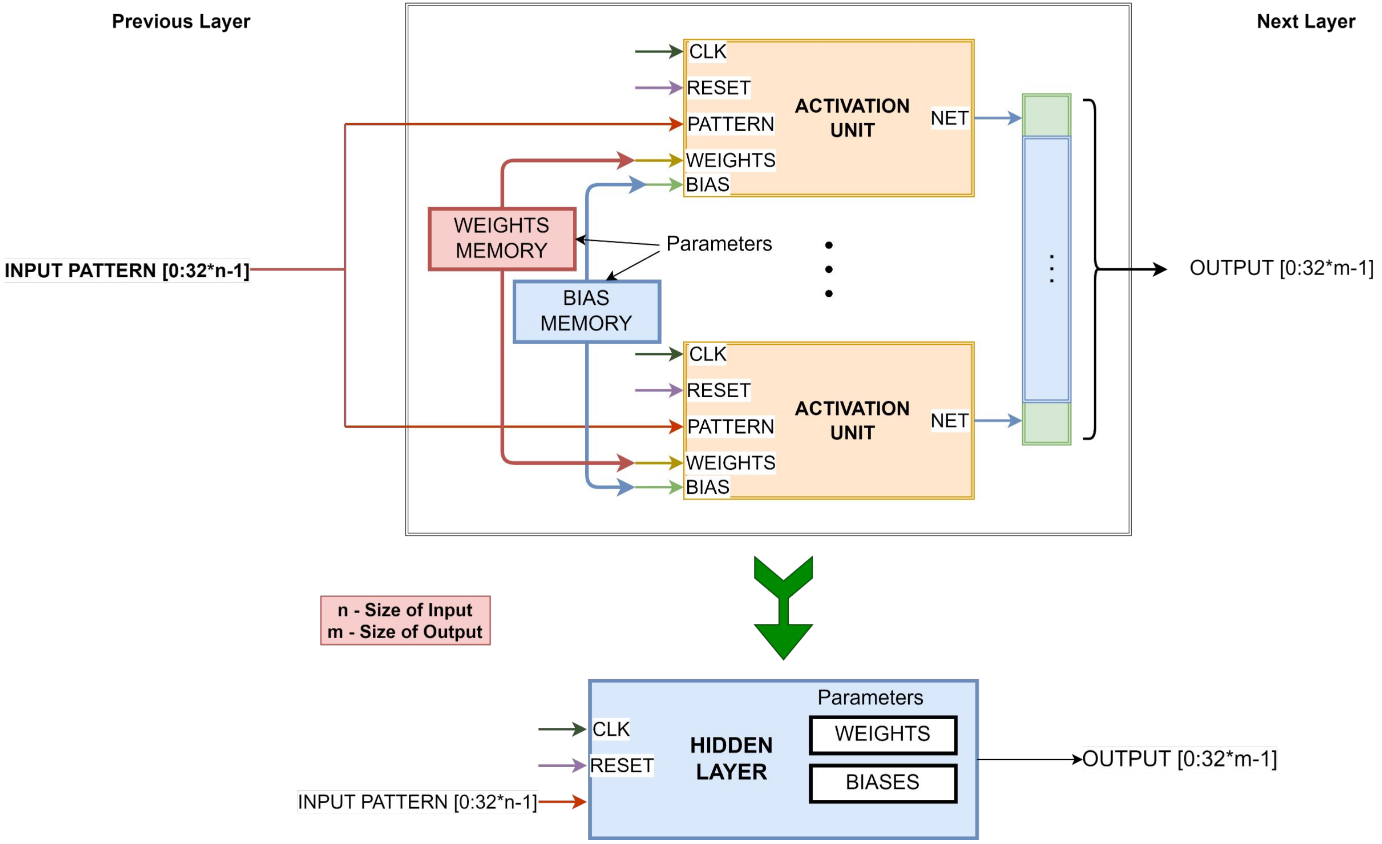


Fig. 19: Hidden/Dense layer

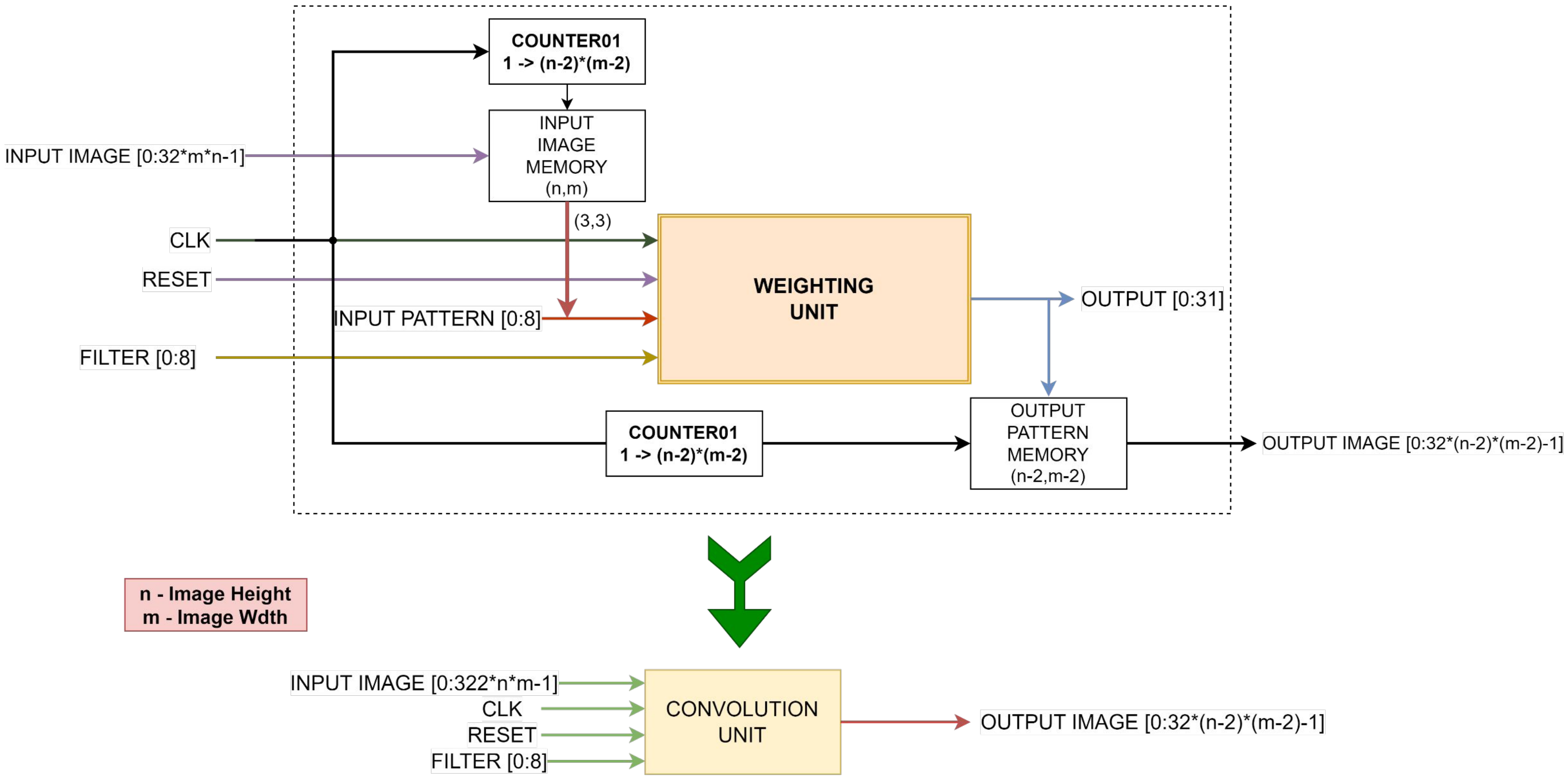


Fig. 20: Convolution Unit

IMPLEMENTATION

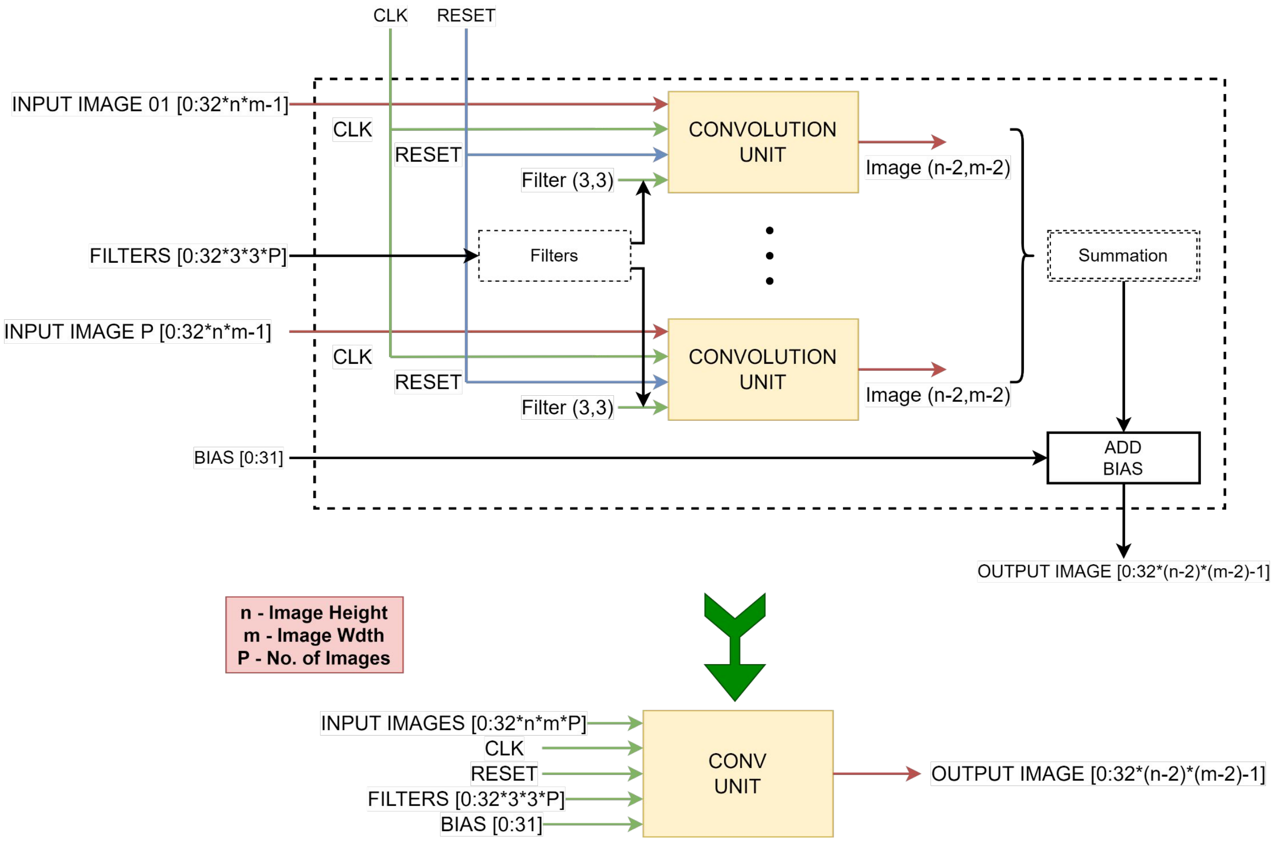


Fig. 21: CONV Unit

IMPLEMENTATION

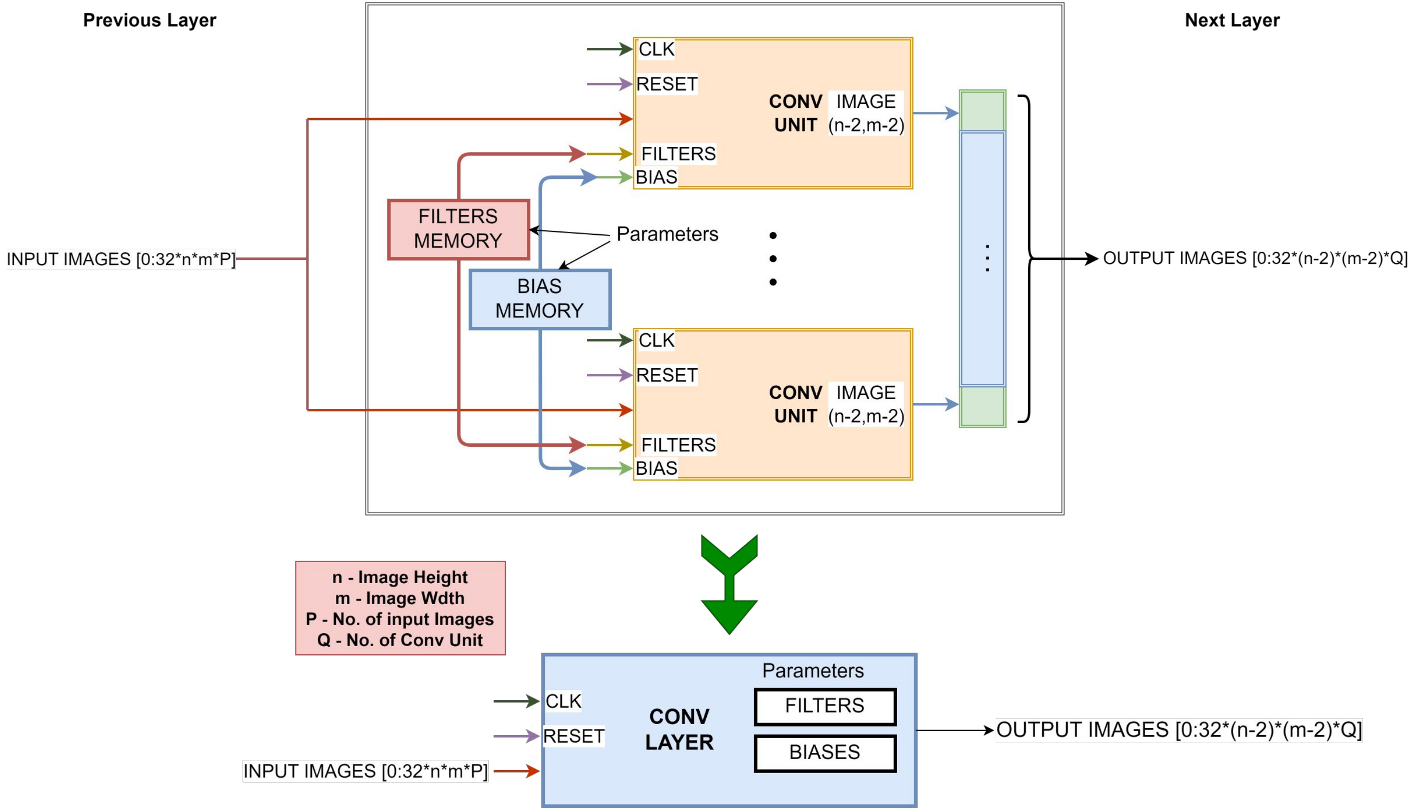


Fig. 22: Convolution Layer

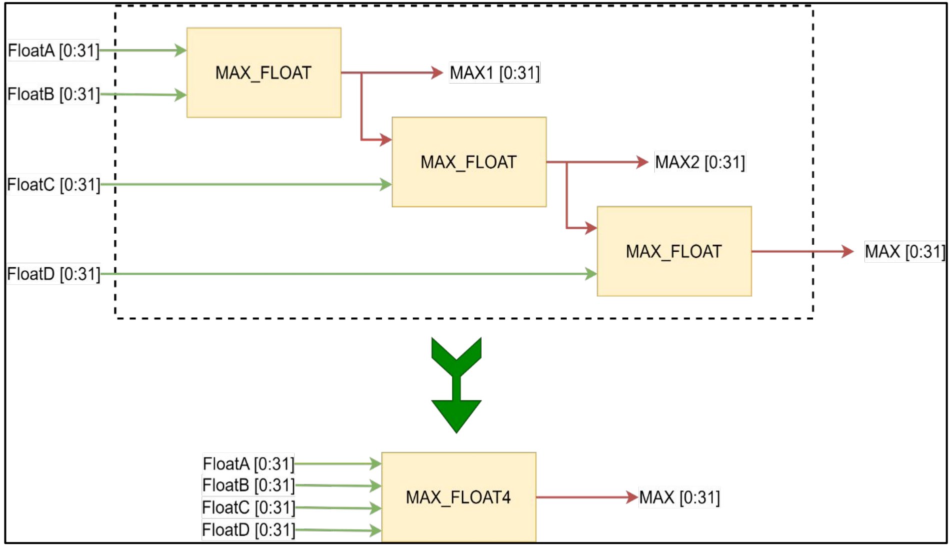


Fig. 23: Floating point max function 4

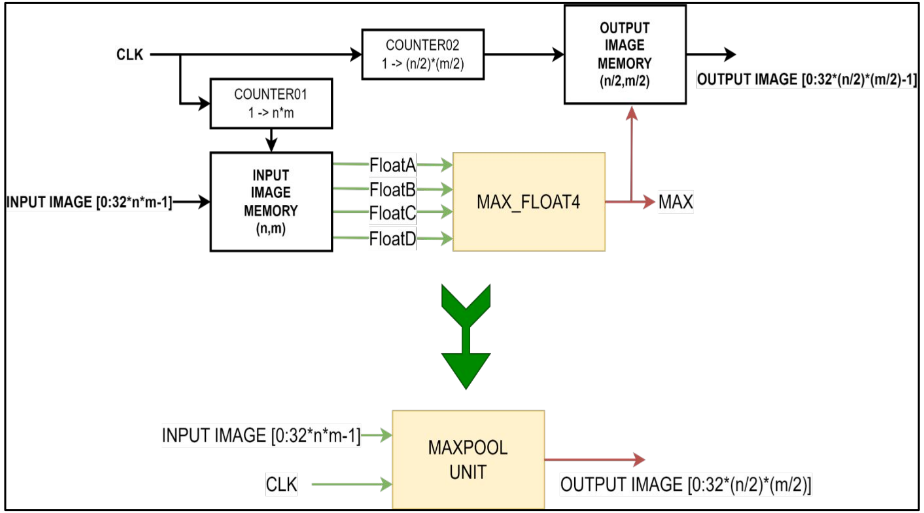


Fig. 24: Maxpool Unit

IMPLEMENTATION

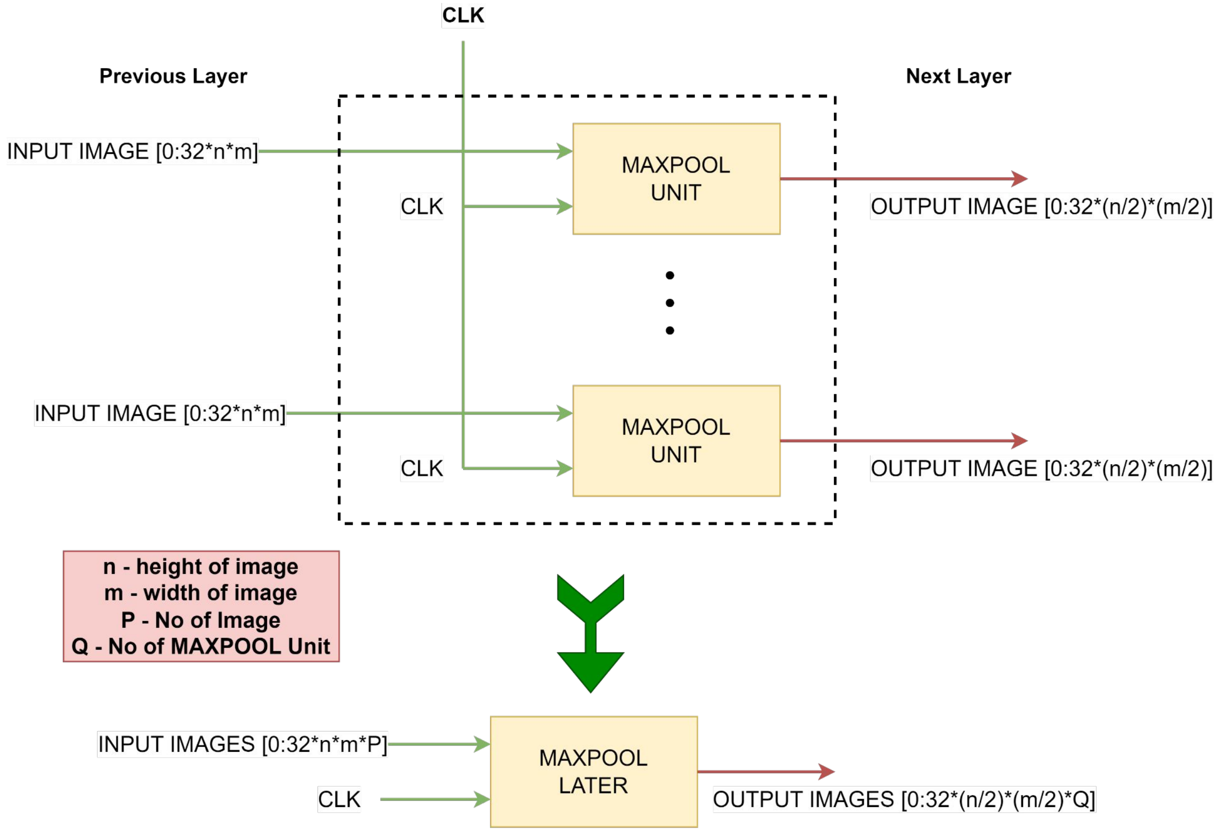


Fig. 25: Maxpool layer

RESULT & SIMULATION

Our dataset contains a total of 23392 samples of 10 classes. Dataset is split into train_set with 83% and test_set with 17%. We use k fold cross-validation technique with the value of k=4. So train_set is divided into 4 folds.

Table 4: Four fold distributions

Total train sample: 19392 samples			
Fold 01	Fold 02	Fold 03	Fold 04
4848 samples	4848 samples	4848 samples	4848 samples

Since the value of k is 4, so the model is evaluated 4 times. Each time 1 fold is used for the validation dataset and remain 3 folds are used for the training dataset.

Table 5: Four fold distributions

Iteration	Fold 01	Fold 02	Fold 03	Fold 04
01	Train	Train	Train	Validation
02	Train	Train	Validation	Train
03	Train	Validation	Train	Train
04	Validation	Train	Train	Train

Four folds validation accuracy of ANN and CNN are in fig4.6 and fig 4.11 respectively. The evaluation accuracy of ANN is **94.67%** and the evaluation accuracy of CNN is **97.75%**.

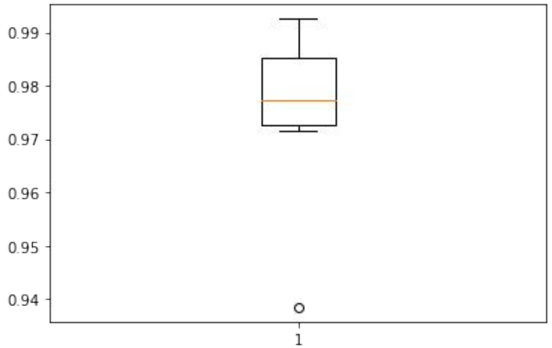


Fig. 26: Validation accuracy of ANN [mean=97.411 std=1.743]

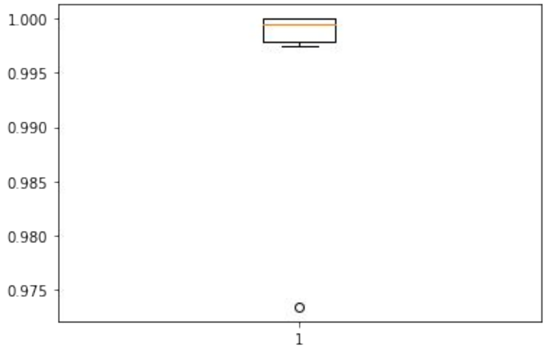


Fig. 27: Validation accuracy of CNN [mean=99.500 std=0.970]

RESULT & SIMULATION

DENSE 01

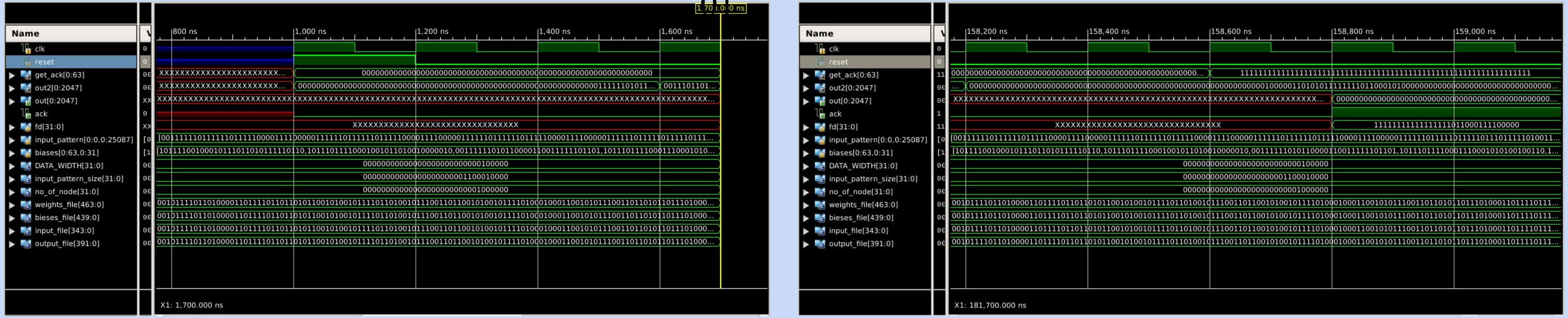


Fig. 28: Output of Dense 01 layer

DENSE 02

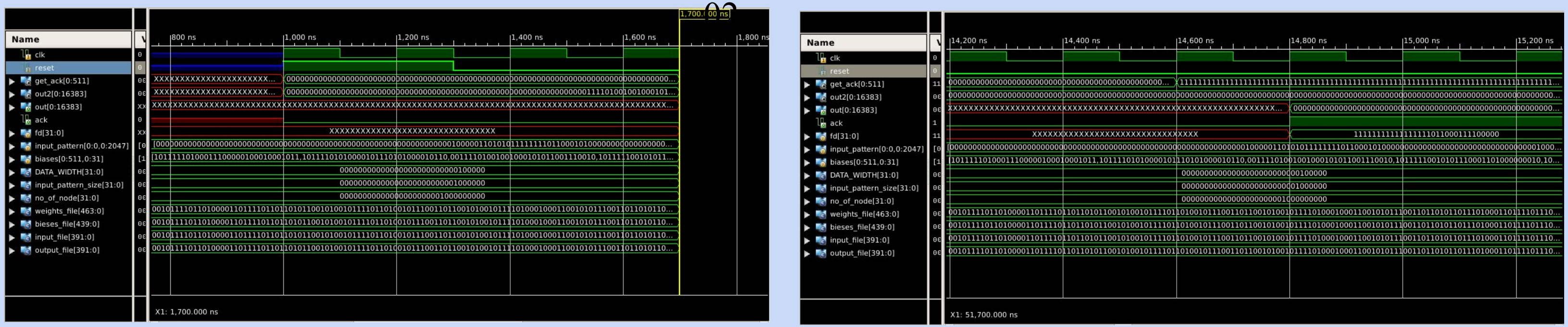


Fig. 29: Output of Dense 02 layer

SOFTMAX

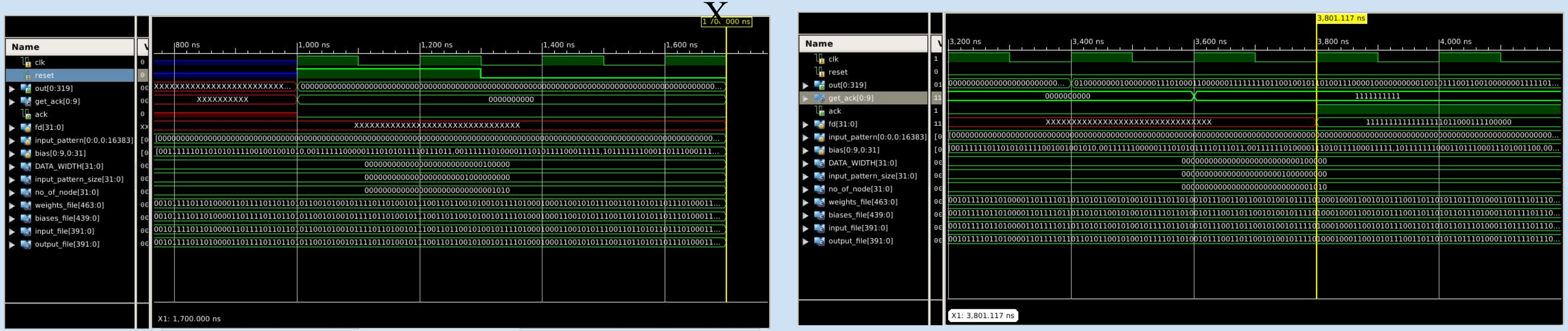


Fig. 30: Output of SOFTMAX layer

ARGMAX

0001

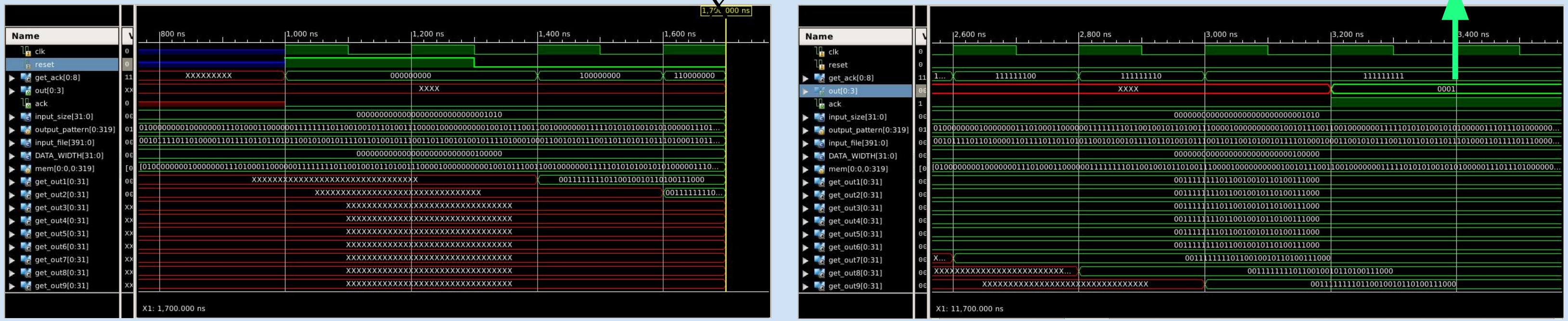


Fig. 31: Output of ARGMAX layer

CONCLUSION

- Implemented ANN and CNN using the library in python.
- Implemented ANN and CNN without using the library in python.
- Implemented ANN and CNN using Verilog.
- Simulated ANN and CNN circuit on Xilinx ISE.
- Literature Review Has been Done.

FUTURE WORKS

- ANN and CNN models can be extended for Bengali Character Recognition.
- Extend these models on various FPGAs, supporting a variety of network architectures.
- Characters Recognition.
- Mixed Characters and Digits Recognition.

REFERENCES

[1] N. Das, R. Sarkar, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu, “A genetic algorithm based region sampling for selection of local features in handwritten digit recognition application,” *Applied Soft Computing*, vol. 12, no. 5, pp. 1592–1606, May 2012, doi: 10.1016/J.ASOC.2011.11.030.

[2] U. Bhattacharya and B. B. Chaudhuri, “Handwritten numeral databases of Indian scripts and multistage recognition of mixed numerals,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 3, pp. 444–457, 2009, doi: 10.1109/TPAMI.2008.88.

[3] M. A. H. Akhand, M. Ahmed, and M. M. H. Rahman, “Convolutional Neural Network based Handwritten Bengali and Bengali-English Mixed Numeral Recognition,” *Image, Graphics and Signal Processing*, vol. 9, pp. 40–50, 2016, doi: 10.5815/ijigsp.2016.09.06.

[4] M. Shopon, N. Mohammed, and M. A. Abedin, “Image augmentation by blocky artifact in Deep Convolutional Neural Network for handwritten digit recognition,” *2017 IEEE International Conference on Imaging, Vision and Pattern Recognition, icIVPR 2017*, Mar. 2017, doi: 10.1109/ICIVPR.2017.7890867.

[5] C. L. Liu and C. Y. Suen, “A new benchmark on the recognition of handwritten Bangla and Farsi numeral characters,” *Pattern Recognition*, vol. 42, no. 12, pp. 3287–3295, Dec. 2009, doi: 10.1016/J.PATCOG.2008.10.007.

[6] Y. Wen and L. He, “A classifier for Bangla handwritten numeral recognition,” *Expert Systems with Applications*, vol. 39, no. 1, pp. 948–953, Jan. 2012, doi: 10.1016/J.ESWA.2011.07.092.

[7] A. S. A. Rabby, S. Abujar, S. Haque, and S. A. Hossain, “Bangla handwritten digit recognition using convolutional neural network,” *Advances in Intelligent Systems and Computing*, vol. 755, pp. 111–122, 2019, doi: 10.1007/978-981-13-1951-8_11/COVER/.

REFERENCES

[8] A. Sufian, A. Ghosh, A. Naskar, F. Sultana, J. Sil, and M. M. H. Rahman, "BDNet: Bengali Handwritten Numeral Digit Recognition based on Densely connected Convolutional Neural Networks," *Journal of King Saud University - Computer and Information Sciences*, Mar. 2020, doi: 10.1016/J.JKSUCI.2020.03.002.

[9] "GitHub - Sufianlab/BDNet." https://github.com/Sufianlab/BDNet (accessed Jun. 18, 2022).

[10] "FPIDroidHouse with a Nexys 3 – Digilent Blog." https://digilent.com/blog/fpidroidhouse-with-a-nexys-3/ (accessed Jun. 14, 2022).

[11] "Visual Studio Code - Code Editing. Redefined." https://code.visualstudio.com/ (accessed Jun. 19, 2022).

[12] "Logo del ISE Design Suite de Xilinx. | Download Scientific Diagram." https://www.researchgate.net/figure/Figura-32-Logo-del-ISE-Design-Suite-de-Xilinx_fig21_263850413 (accessed Jun. 14, 2022).

[13] P. Echeverría Aramendi and M. López Vallejo, "An fpga implementation of the powering function with single precision floating-point arithm," 2008.

[14] M. Javed, "The best machine learning algorithm for handwritten digits recognition," Nov 2020.

[15] H. Beniwal, "Handwritten digit recognition using machine learning," Dec 2018.

[16] "Bengali language," Oct 2022.

REFERENCES

[17] "Programming an fpga: An introduction to how it works."

[18] P. P. Chu, FPGA prototyping by Verilog examples: Xilinx Spartan-3 version. John Wiley & Sons, 2011.