JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Implementation of systolic array architecture using distributive arithmetic logic for convolution neural networks (CNN)

M.Sony (ME – Scholar), Dr .B. Sarala (Professor)

Department of Electronics and Communication Engineering

Maturi Venkata Subha Rao (MVSR) Engineering College

Saroornagar Mandal, Badangpet – Nadergul Main Rd, Hyderabad, Telangana, 501510.

Abstract - Deep learning has proved to be a highly efficient technology because of its capability to process large amounts of data. Traditional methods are no longer as popular, especially when it comes to pattern recognition. Convolution Neural Networks are a type of deep neural network that are widely used. Designing a reconfigurable systolic array is used to modify the deep learning architecture's precision level (input-output bit width) and support for both signed and unsigned image convolution operations. Improved distributed arithmetic (DA) models are used to CNN that can speed up computation and configuration levels. Employing DA logic to replace all multiplier units with shifters and adders which reduces implementation complexity.

Index terms: Convolution neural network, Systolic array, Processing Elements

I INTRODUCTION

The intrinsic capability of deep neural networks (DNNs) to perform feature engineering autonomously at greater learning rates, and has propelled them past conventional machine learning and computer vision techniques in recent years. Deep neural networks are implemented with limited resources and Internet of things, which results low latency for critical tasks implementation. The primary goal of this field is to develop machines, which include picture and video recognition, image processing, etc. Such tasks may include picture and video recognition, image processing and classification, media restoration, recommendation systems, natural language processing, etc. People could then employ this knowledge for a wide range of tasks. Convolution neural networks have been developed and used specifically over time, resulting in enhancements in computer vision.

The convolution neural network is one of the most prominent networks in the field of deep learning. In the past few years, CNN has gained a significant amount of attention from both business and academics because of its tremendous accomplishments in a variety of fields, including natural language processing and computer vision processing.

A convolution Neural network contains several layers of neurons that connect to transmit data and are driven by the precision of the optic nerve. The enormous demand for CNN on new devices is a result of the huge expansion of new applications based on huge learning. This is especially true for the predicting part, which is expanding and growing on small smart devices

II RELATED WORK

Multiplication process using distributed arithmetic (DA) model

Multipliers are crucial component in digital signal processing applications which includes digital communications, spectral analysis, and digital filtering. Many current DSP applications are designed for portable, and low power dissipation for battery operated systems and reducing a multiplier's delay is significant for realizing the overall design of complex multipliers. They are usually employed at a high system clock rate and quite complex circuits.

Multiplications are slow down the process since they are so expensive. The speed of a multiplication procedure is an important factor, and using it how efficiently a computer problem performs. Consider the unsigned M-bit and N-bit binary values of X and Y. The binary form representation of X and Y as given below.

$$X = \sum Xi\ 2^{i} \qquad i = 0 \text{ to } M$$

$$Y = \sum Yj\ 2^{j} \qquad j = 0 \text{ to } N$$

Multiplication is performed in the quickest and easiest way by employing a 2-bit adder. For values that are M and N bits wide, M cycles of multiplication are carried out by employing an N-bit adder. The shift-and-add method of multiplication is used to add the M partial products. Each partial product is produced by adding a multiplier bit to the multiplicand, which is basically an AND operation, and shifting the output according to the position of the multiplier bit. Binary multiplication adds shifted copies of the multiplicand based on the value and position of each multiplier bit, much like the well-known long-hand decimal multiplication.

Relative to decimal multiplication, binary multiplication is actually simpler to perform. Due to the fact that each binary digit can only contain a value of 0 or 1. Depending on the multiplier bit's value, the partial products can only be a duplication of the multiplicand or 0.This is a digital logic AND function, to put it simply.

Multiplication can be performed more efficiently by using a DA technique than by manually computing it, and an array of the entire partial product is produced simultaneously. The output is frequently computed using addition with multiple partial products. A good illustration of the approach is shown in Figure 1. This set of operations can be directly mapped to hardware. An array multiplier, the resulting structure, combines the partial product output, partial product accumulation, and final addition operations. The adder unit is required for the design of any multiplier. Moreover, compared to

DA technique it has been shown that, matrix multipliers exceed

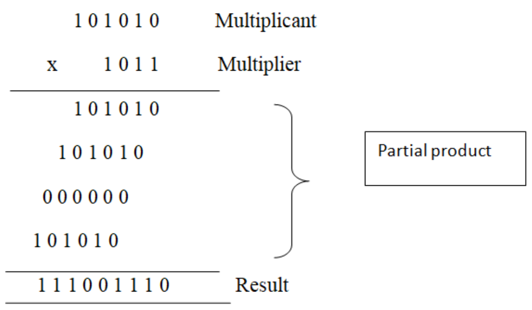


Fig1 : Manual multiplication

MOS transistor count in terms of speed, power consumption, and threshold loss.

MULTIPLICATION PROCESS

The digital multiplier are more responsible to produce power dissipation in digital signal processors. The switching of a large number of gates during multiplication is the major reason for such significant power dissipation in these circuits. Moreover, a lot of false transitions on internal nodes will be using lot of power. A new array connection design that minimizes power loss due to spurious transition activity is the result of the timing analysis of a full Adder, which is the essential component of multipliers.

Furthermore, the multiplier throughput is enhanced by DA connection architecture. The effectiveness of the processors can be analyzed using a variety of measures. As a result, it is necessary to take into consideration both the circuit's area and the processing duration. Depending upon the required speed and area, the employed digital multipliers can be of either a bit-serial or bit-parallel architecture. The bit-serial method of processing data provides one data bit to the processor per clock cycle for processing. The parallel approach, however, just requires one clock cycle to process the data bits simultaneously. Because of their simple layout and limited hardware requirements, bit-serial-based structures are often employed to produce the first designs, resulting in reduced system costs. The bit-parallel approach is chosen, however, for systems that demand exceptionally high speeds. However, this technique necessitates a sizable silicon area, significant connection costs, and pin out. Therefore, a decision needs to be made among these two methods based on the application's requirements. The two steps that comprise the multiplication process are as follows: 1) A partial product's evaluation and 2) Accumulation of the shifted partial products.

III BLOCK DIAGRAM OF PROPOSED

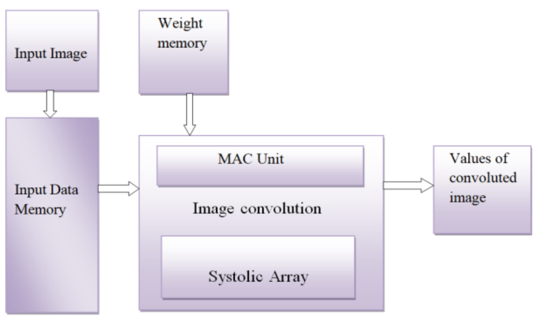
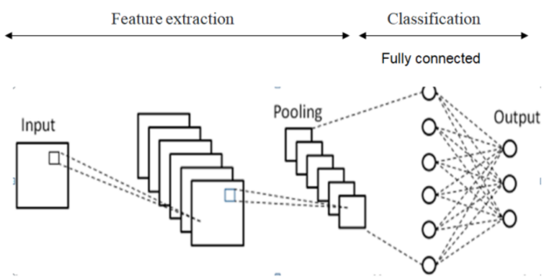


Fig 2: Block Diagram of modified design

The block diagram for the proposed work is shown in Fig 2. Apply the image as input, & then it will be converted to a digital file format. Basically, an image is a form of a matrix comprised of rows and columns. 8x8 macro blocks are formed by decomposing the input image. Every one of these macro blocks is read individually and given to CNN as input. A systolic array is utilized by CNN to perform convolution. The processing element (PE) is the fundamental core component of the systolic array, which utilizes specialized hardware to process the data elements in parallel. The size of each matrix is 8x8 , and it has 64 MAC units (Multiplication cum Accumulation). It performs the convolution operation by arranging these PEs in an 8x8 array format. After image convolution, it creates a feature map. For this feature map, maxima and minima pooling are performed and then provided to the activation function, which passes some values and adds constants. Relu (Rectified Linear Unit layer) is a technique where only positive values are taken into account for the overall activation function.

IV CONVOLUTIONAL NEURAL NETWORK (CNN)

A Convolution Neural Network is a Deep Learning algorithm that can take in an input image, assign various objects and elements in the image importance (learnable weights and biases), and distinguish among them. In contrast to certain other classification techniques, a ConvNet has significantly less pre-processing. ConvNets can comprehend these filters and their characteristics, in contrast to primitive methods where filters are hand-engineered.



convolution

Fig 3: Convolution neural network image classification

CONVOLUTION PROCESS OF AN IMAGE

Two types of data can be combined using the mathematical process of convolution. In the case of CNN, convolution is employed in order filter the data and to generate a feature map from the input data. The size of this filter also termed a kernel or feature detector can be, for instance, 3x3. The kernel repeats over the input image performing element-by-element matrix multiplication in order to perform convolution. The feature map tracks the outcome for each receptive field or the region where convolution occurs.

MAX POOLING IN IMAGE CONVOLUTION PROCESS

The pooling layer, which incorporates the convolution layer, is responsible for minimizing the spatial size of the convolved feature. By reducing dimensionality, minimal computational power will be utilized to process the information. Max pooling is being used in this case, as illustrated in fig 4. This enables us to estimate a pixel's highest value from a region of the image that the kernel has obscured. Moreover, Max Pooling also performs as a noise-canceling technique. Along with totally eradicating noisy activations, it also performs de-noising and dimensionality reduction.

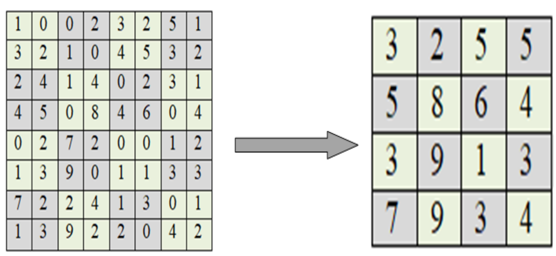


Fig 4: Max pooling

Each layer's neuron connects with another layer's neuron via fully connected layers. The architecture is precisely the same as a conventional multilayer perceptron neural network (MLP). The flattened matrix passes through a layer that is fully connected to categorize the images

V SYSTOLIC ARRAY

A systolic array shown in fig 4, comprised of rows of cells, called data processing units that resemble a matrix. The data is immediately transmitted to every cell's neighbors after processing. As data is transmitted between neighboring DPUs across the systolic array, it is often rectangular and flows in a number of different directions. ASMs (auto-sequencing memory units) generate the data streams that enter and exit the ports of the array.

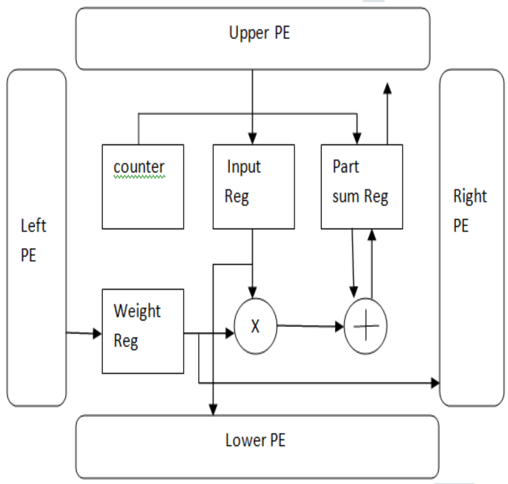


Fig 5 : Systolic Array Architecture

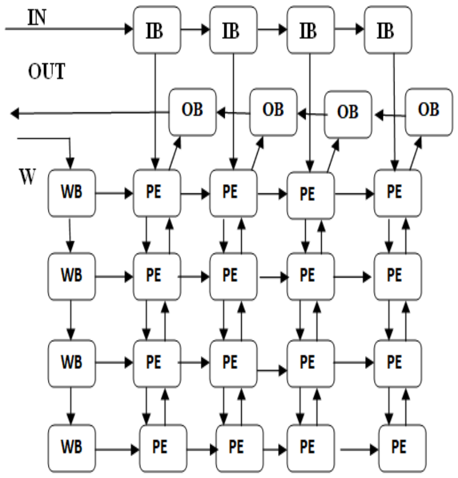


Fig 6 : Systolic Array Architecture for CNN

For matrix multiplication, a systolic algorithm is used. Each of the two matrices is given simultaneously, one from the top of the array passing down the array, the other from the left side of the array passing from left to right. Dummy values are then sent to each processor unless they have seen an entire row and an entire column. A single row or column at a time could be used to display the multiplication result, which had been saved in the array.

VI RESULTS

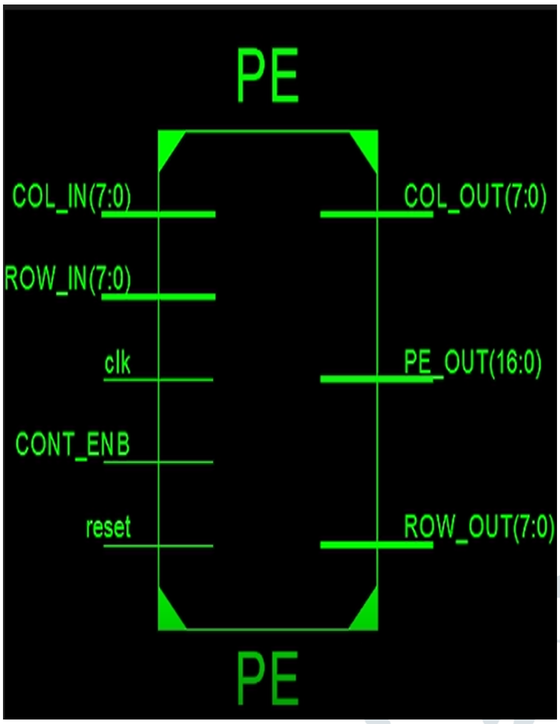


Fig. 7: Systolic array processing element

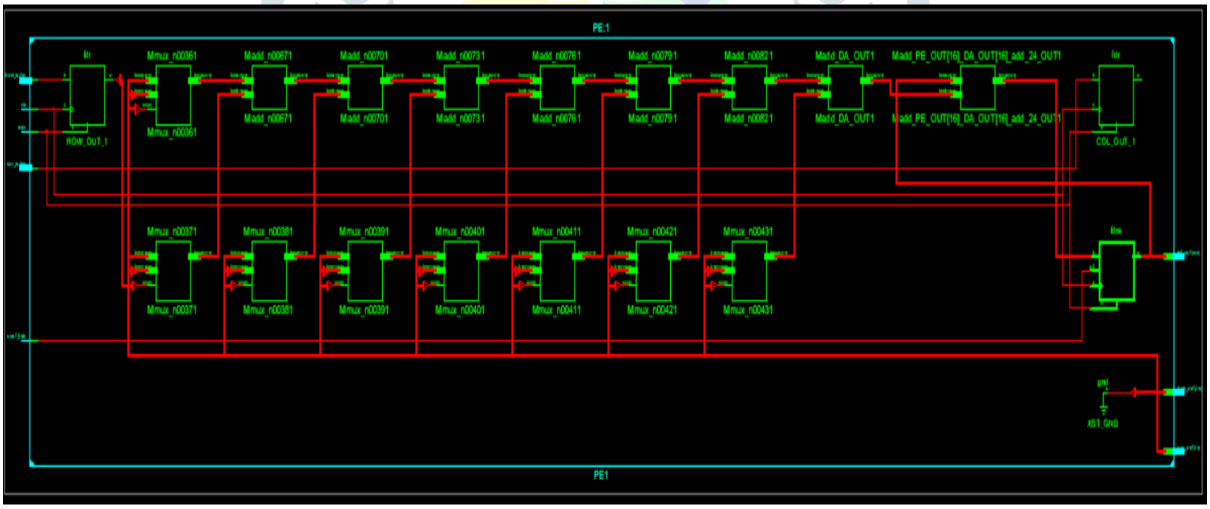


Fig 8 : RTL Schematic

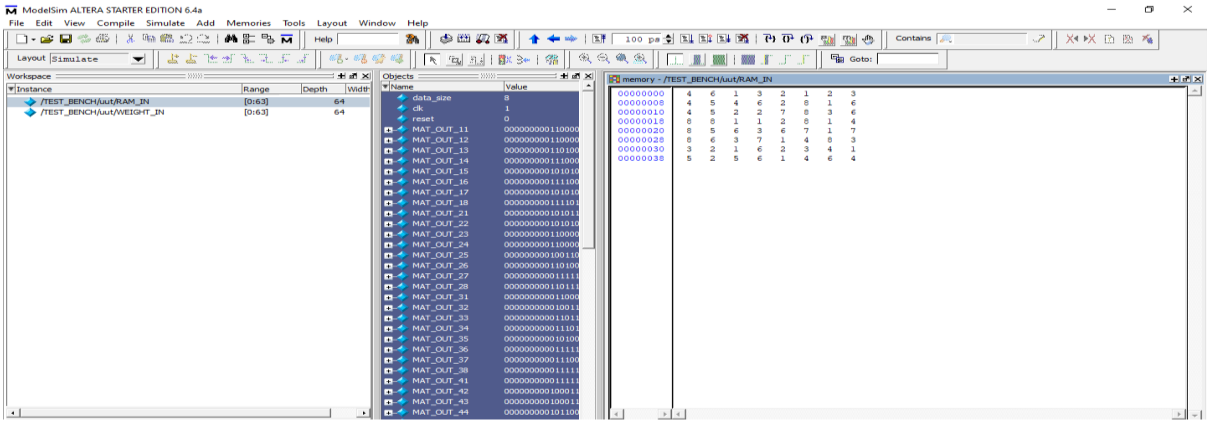


Fig 9 (a) : Input matrix

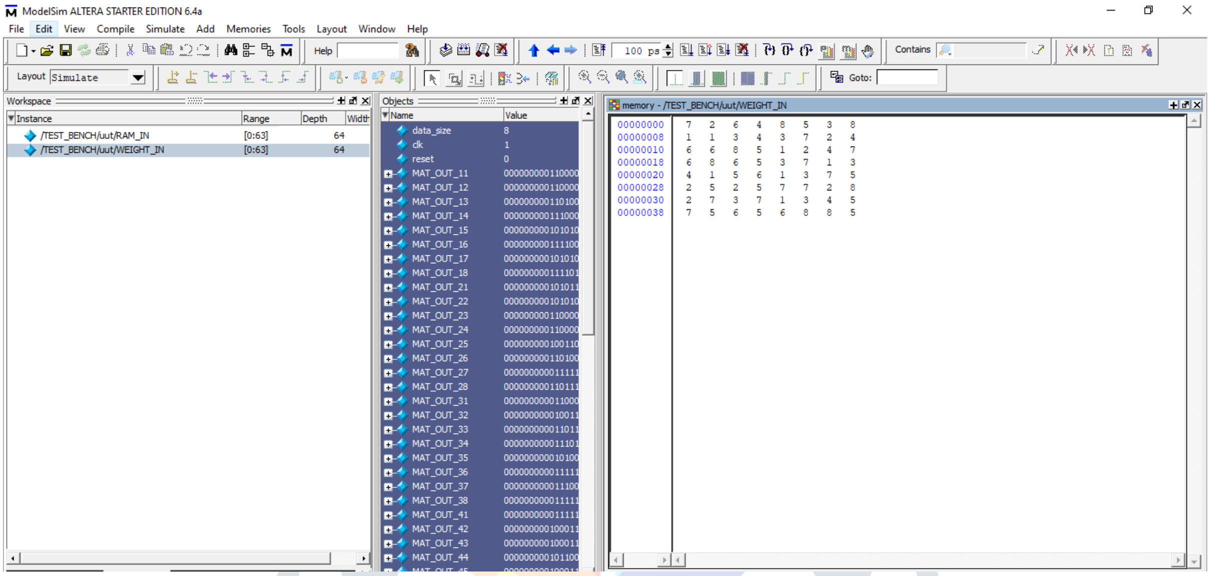


Fig 9 (b) : Weight matrix

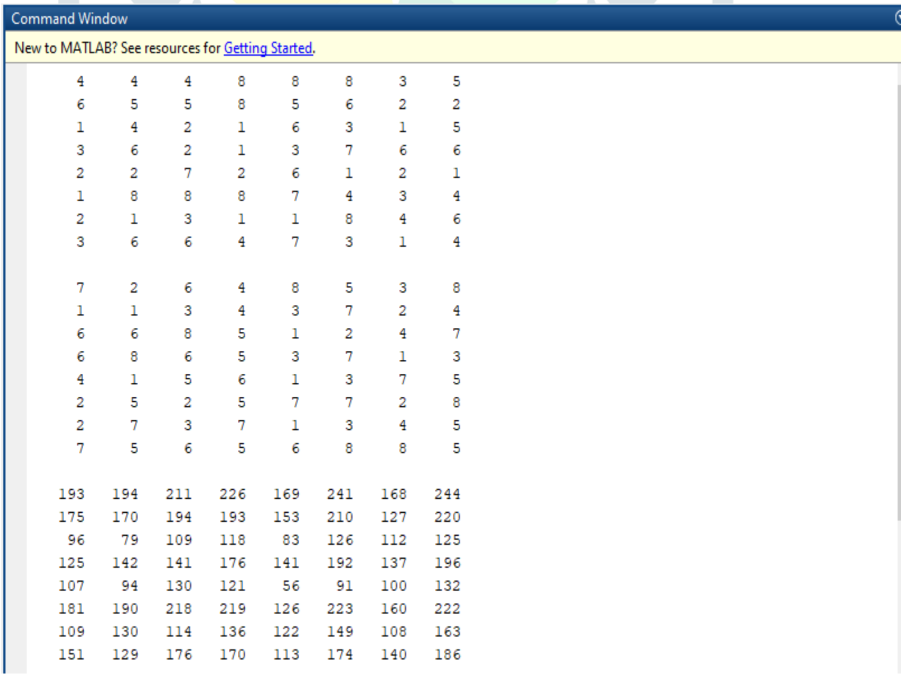


Fig 9(c) : Matrix Multiplication results and end results

Figure 9(a) shows the 8x8 input matrix, while Figure 9(b) shows the 8x8 weight matrix (b). Fig. 9(c) displays the matrix multiplication of the input matrix and weight matrix.

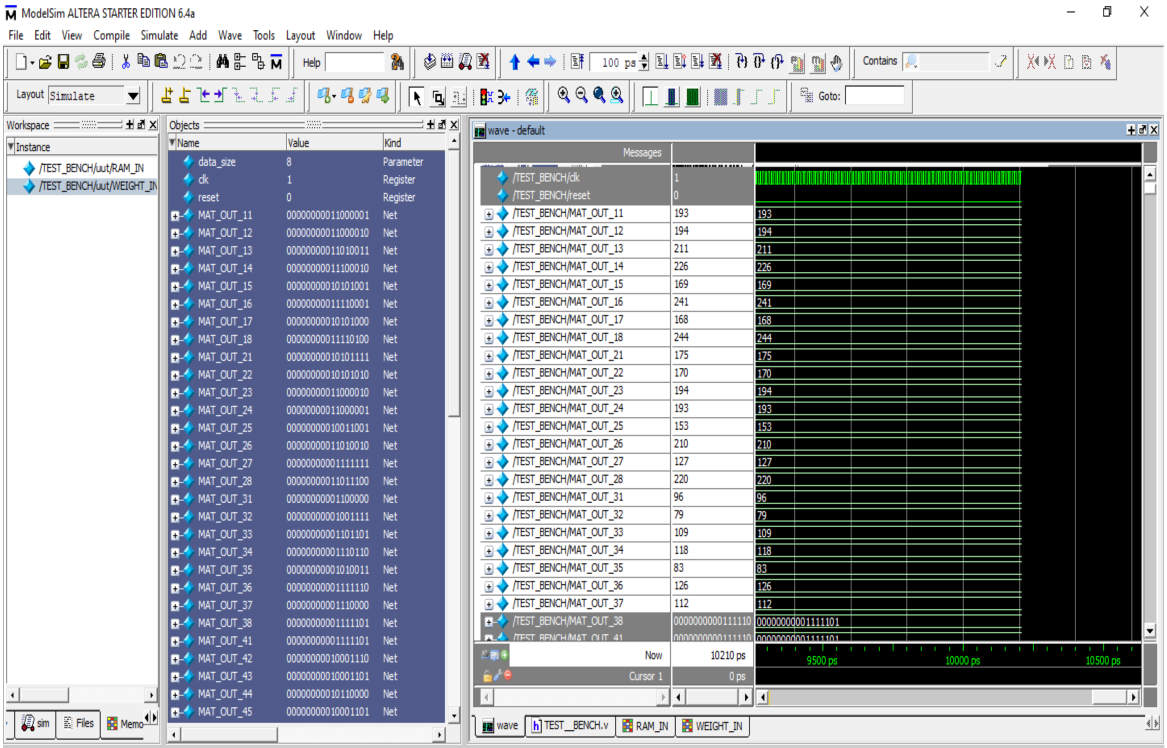


Fig10: Simulated Output of above matrix multiplication

COMPARISION

Table 1 Comparission of DA(MAC) and Conventional (MAC)

Model type	Area	Max Freq	Delay	No of slice registers	No of slice LUT's	No of fully used LUT-FF Pairs
Systolic Array using conventional (MAC)	83/200	209.996 MHz	4.762 ns	83	200	35
Systolic Array using DA (MAC)	33/162	242.430 MHz	3.257 ns	33	162	19

Table 1 shows the comparison of DA MAC and Conventional MAC. Here Conventional MAC area is 83/200, which is reduced to 33/162 in DA MAC. In proposed method the frequency has been increased from 209.996 MHz to 242.430 MHz. Delay has been reduced from 4.762ns to 3.257ns.

VII CONCLUSION

In this paper, designing a reconfigurable systolic array that handles both signed and unsigned image convolution operations and changes the accuracy level (input-output bit width) of the deep learning architecture. The configuration level and improved distributed arithmetic (DA) models can accelerate computing. Employing shifters and adders in place of all multiplier units in DA (multiplier less) arithmetic for low complexity. Using DA model occupies less area and delay.

REFERENCES

[1] Shamik Kundu, Suvadeep Banerjee, Arnab Rah, Suriyaprakash Natarajan, and Kanad Basu,“Toward functional safety of systolic array – based deeplearing hardware accelerators” IEEETransactions on very large scale integration (VLSI) systems, vol. 29, no. 3, March 2021.

[2] J. J. Zhang, K. Basu, and S. Garg, “Fault-tolerant systolic array based accelerators for deep neural network execution,”IEEE Des. Test, vol. 36, no. 5, pp. 44–53, Oct. 2019.

[3] G. Li et al., “Understanding error propagation in deep learning neural network (DNN) accelerators and applications,” in Proc. Int. Conf. High Perform. Comput., Netw., Storage Anal.,Nov. 2017, pp. 1–12.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in Proc. Adv. Neural Inf. Process. Syst., 2012, pp. 1097–1105.

[5] J. J. Zhang, T. Gu, K. Basu, and S. Garg, “Analyzing and mitigating the impact of permanent faults on a systolic array based neural network accelerator,” in Proc. IEEE 36th VLSI Test Symp. (VTS), Apr. 2018, pp. 1–6.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in Proc. Adv. Neural Inf. Process. Syst., 2012, pp. 1097–1105.

[7] A. Parashar etal., “SCNN: An accelerator for compressed-sparse convolutional neural nets,” in Int. Symp. on Comp. Arch. (ISCA), 2017.

[8] P. N. Whatmough et al., “FixyNN: Efficient Hardware for Mobile Computer Vision via Transfer Learning,” in 2nd Conference on Machine Learning and Systems (SysML), 2019.

[9] H. Kung, B. McDanel, and S. Q. Zhang, “Packing sparse convolutional neural networks for efficient systolic array implementations: Column combining under joint optimization,” in Int. Conf. on Arch. Support for Programming Languages and Operating Systems (ASPLOS), 2019.

[10] W. Wen et al., “Learning Structured Sparsity in Deep Neural Networks,” in Adv. in Neural Information Processing Systems (NIPS 2006), 2016