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

An Effective Optimization of FPGA Based Scalable Deep Learning Accelerator Unit (DLAU) for Deep Convolution Neural Networks.

V.SATHISH[1],Dr.B.NAGESHWARRAO[2] ,Dr.T.VAMSHI[3]

[1]M.Tech Student, Talla Padmavathi College of Engineering, Somidi, Kazipet, Telangana, 506003
[2]Assoc Professor, Talla Padmavathi College of Engineering Student, ,Somidi, Kazipet, Telangana, 506003
[3]Assoc Professor, Talla Padmavathi College of Engineering,Somidi, Kazipet, Telangana, 506003

sathishreddyvelmajala@gmail.com,nagesh.south@gmail.com, vamshi22g@gmail.com

Abstract

It is clear that deep learning, as a new branch of machine learning, is capable of addressing complex learning issues with ease. However, due to the needs of real applications, the complexity of the networks grows significantly, making it difficult to implement high-performance deep convolution neural network neural networks. DLAU is a hardware prototype for a scalable acceleration design for large-scale deep learning networks that uses a field-programmable gate array (FPGA) as a hardware prototype to boost performance and preserve low power consumption. Deep learning applications benefit from the DLAU accelerator's exploration of locality using tile techniques and three pipelined processing units. The latest Xilinx FPGA board shows that the DLAU accelerators can outperform Intel Core2 processors in terms of speed.

Key words: Deep learning, field-programmable gate array (FPGA), hardware accelerator, neural network.

1. Introduction

FPGA (Field-Programmable Gate Array) is a primary tool for accelerating deep learning algorithms because of its improved performance and reduced power consumption. Data centers need a lot of power because of the enormous amount of data they process. Data centers in the United States alone are expected to consume 140 billion kilowatt-hours of electricity per year by 2020 [4]. Implementing highly efficient deep learning with low power costs, particularly for large-scale based on neural models, is therefore a major issue. This technology has so far been limited to FPGA, ASIC, and GPUs for deep learning algorithms, but this is expected to change in the near future (GPU). FPGA and ASIC are examples of hardware accelerators that, when compared to GPU

acceleration, can provide at least reasonable performance while using less power. Complex and huge deep neural networks are difficult to create with hardware accelerators because of their limited computational resources and memory. For ASIC, the development cycle is longer, and the versatility is less than desirable. When Chen et al. [6] introduced the DianNao machine learning hardware accelerator, the field of computational intelligence processors was born. It introduces a new paradigm for neural network-based machine learning hardware accelerators. Due to the lack of reconfigurable hardware, DianNao is unable to meet the needs of varied applications. The limited Boltzmann machine can be accelerated using FPGA-based techniques developed by Ly and Chow [5]. RBM-optimized hardware processing cores have been developed by the company. Another accelerators for the RBM was created using FPGA technology by Kim et al. [7]. For each node, they use numerous RBM processing modules that are individually accountable for a relative handful of nodes in total. FPGA-based neural accelerators have also been presented in comparable studies [9]. A FPGA-based accelerator was introduced by Yu et al. [8], however it is unable to adapt to changes in network size and topology. There is still a lot of work to be done in the area of scalable and adaptable hardware architectures for neural networks, which are the focus of the current studies.

1.1 Introduction to Neural Networks

Neural networks are computing systems made up of multiple of simple, densely interconnected processing components, which process information through their dynamic state reaction to external inputs but at considerably smaller sizes. It is not uncommon for an ANN to contain hundreds of processor units, while the mamal nervous system contains billions of neurons, greatly increasing the magnitude of interconnectivity and emergent behavior. Whereas most ANN researchers don't give a hoot about how closely their networks match biological systems, there are a few that have. Researchers, for example, have been able to accurately recreate the retina's function and model the eye quite effectively. Although neural networks is not a trivial subject, a user can easily get an includes of its structure and function.

1.1.1 The Basics of Neural Networks

It is not uncommon for neural networks to be constructed in a layered structure. There are several 'nodes' in each layer that have a 'activation function'. The 'input layer' connects with one or much more 'hidden layers', where the real data are processed via a weighted 'connections' system. When all of these layers are linked together, the result is displayed as illustrated in the image below.

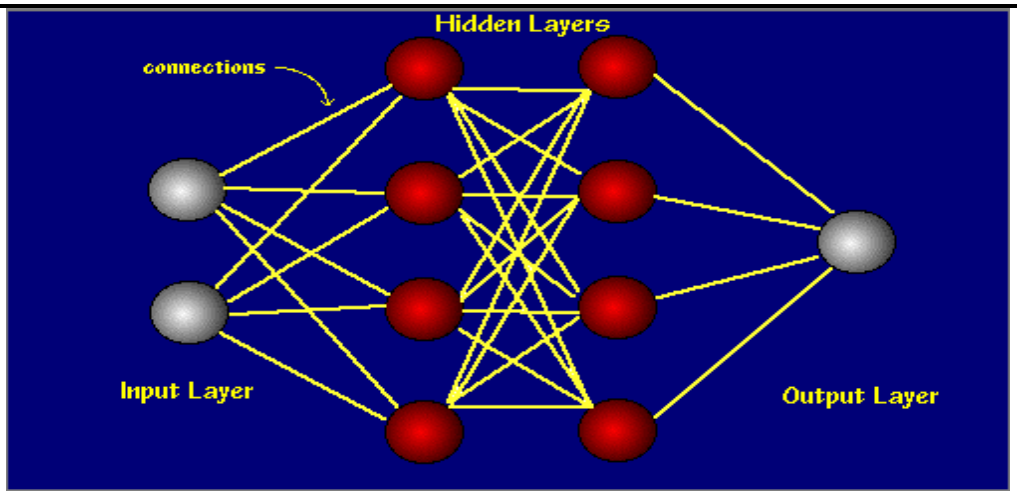


Fig.1 Neural Network

.

2. Literature survey

Using a convolutional neural network (CNN), which mimics the function of the optic nerves in biological beings, is a frequently used method for image identification since it can reach high accuracy. Research and implementations have recently improved due to the rapid expansion of application areas deep learning algorithms. FPGA-based accelerators for DCNN have been recommended as a result of their high performance, reconfiguration, and speed of development. Although contemporary FPGA accelerators have shown outstanding quality over generic computers, the design space for accelerators has not been fully investigated. It is possible that the computing speed given by an FPGA platform does not match the memory bandwidth. This is a major issue. As a result, current techniques are unable to reach optimal performance since logic resource and memory bandwidths are being underutilized. The intricacy and flexibility of practical applications exacerbate this difficulty. We offer an analytical design strategy based on a roofline model to address this issue. We use several optimization approaches, such like loop tiling and transformation, to objectively analyze the compute speed and needed memory bandwidth for every CNN design solution. This will allow us to quickly determine which solution uses the least amount of FPGA resources and performs at its peak. This case study compares a CNN accelerate on an FPGA board with earlier techniques. In comparison to earlier implementations, ours reaches best performances of 61.62 GFLOPS under a 100MHz working frequency.

3. Deep neural network elements

We call "stacked neural networks," or networks with multiple layers, "deep learning" to describe this type of learning. Nodes are the building blocks of the layers. As with neurons in the brain, a node is essentially a location where processing takes place, and it fires when it receives enough stimulation. By combining data from the data set with a set of parameters, or weights, the input is amplified or dampened depending on the algorithm's goal. To identify data without error, for instance, which information is most helpful? When the input-weight products are added up, the total is then run through a node's so-called activation function to see if and to what extent a signal can move on to effect the final result, such as a classification act.

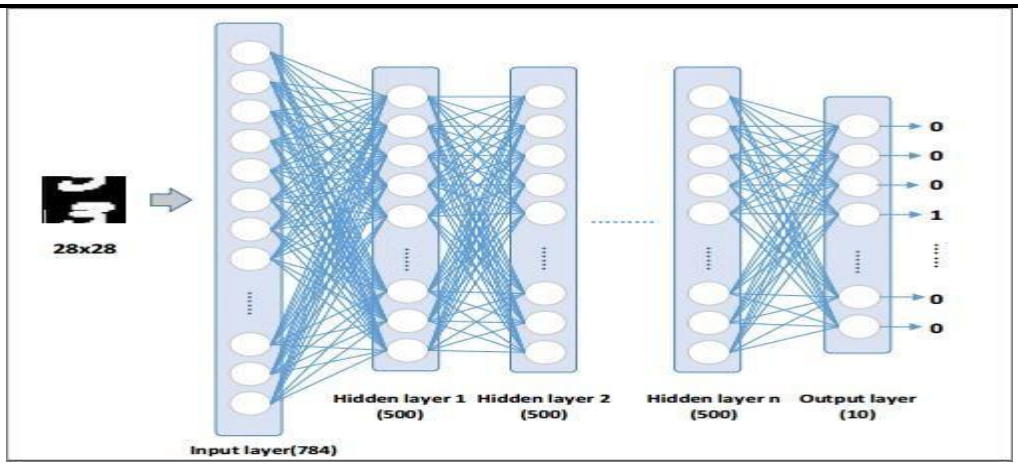


Figure 2. The schematic diagram of DNNs for Mnist

There are numerous hidden layers and one output layer in the deep neural network depicted in Fig.2 for handwritten digits recognition (Mnist is a dataset of handwritten digits). DNNs are used as an example in this paper. In a DNN, there are two types of computation: prediction and training. Feed-forward computing is used in the prediction process to figure out the output based on each input's weight coefficients. Prior to global training with Back Propagation method, pre-training is used to modify the weights of the connections between units within adjacent layers using the training datasets (BP algorithm). We use a prediction technique rather than a training procedure because of technical and business factors.

4. Proposed system

Our solution is a scaled deep learning accelerators unit called DLAU, which can speed it up the kernel computations of deep learning techniques by a factor of 10 to 100. Specifically we use tile algorithms, FIFO caches, and pipelines to reduce memory transfer operations and reuse computational units to create huge neural networks. The following contributions set this technique apart from earlier works in the field.We use tiling approaches to split the large-scale input data in order to investigate the localization of the deep pedagogical approach. Different tile data sizes can be used with the DLAU design to take advantage of the tradeoffs of speed and hardware cost. Therefore, the FPGA-based acceleration is more adaptable to various machine learning applications. the part sum accumulate unit (PSAU) and the nonlinear activation acceleration unit (AFAU) comprise the DLAU accelerator's three completely pipelined processing components (AFAU). Distinctive system

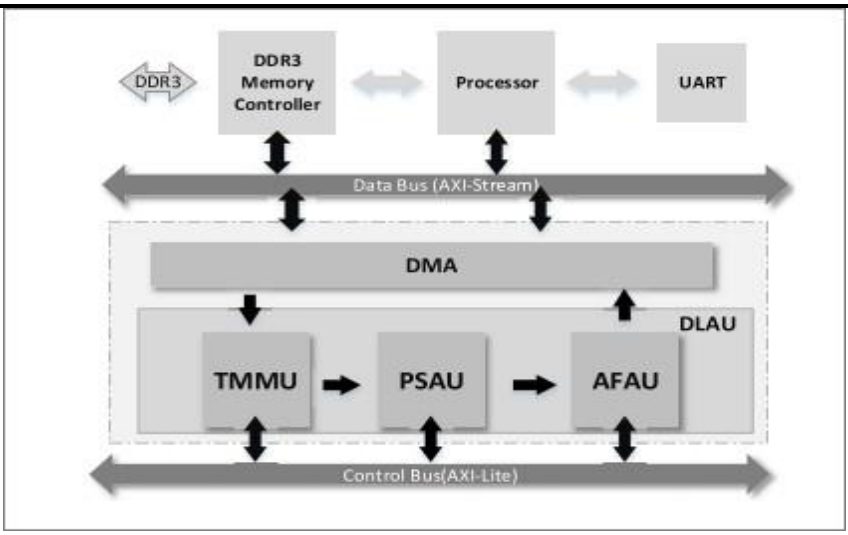


Fig. 3. DLAU accelerator architecture.

Figure 3 shows the design of the DLAU accelerator. Basic modules like these can be used to build complex topologies like CNN, DNN, and even evolving neural networks. As a result, FPGA-based accelerators offer greater scalability than ASIC-based ones.

4.1 DLAU architecture and execution model.

FIG. 4 shows the overall system architecture, which includes an embedded processor, DDR3 memory contr oller (DMA module), and the DMA accelerator (DLAU). JTAG-UART communications between embedded processor and DLAU are handled by the embedded processor. The input data and weight matrix are transferred to inner BRAM blocks, the DLAU accelerator is activated, and the results are returned to the user following execution. The DLAU is a stand-alone machine that may be configured to work with a variety of various applications and setups. The DLAU comprises of three processing units: 1) TMMU; 2) PSAU; and 3) AFAU. The DLAU is organized in a pipeline way.DLAU takes the tiled instruction from memory using DMA, performs computations on all three processing units, and afterwards sends the data back to the memory. Key features of DLAU's accelerator architecture are these.

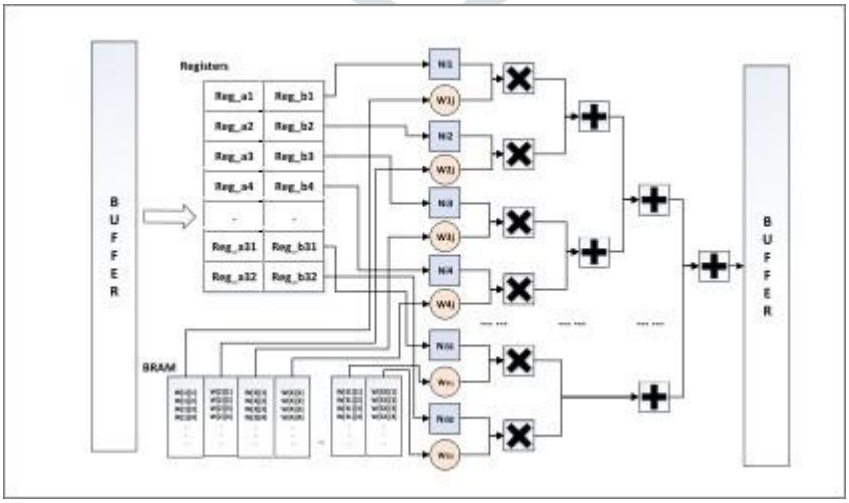


Fig. 4. TMMU schematic.

4.2 Input/Output Flow Control (FIFO) Buffer: An input and output buffer are built into all of the processing units in DLAU. To avoid data loss as a result of processing units with irregular throughput, these buffers are used. The size of a neural network may vary depending on the machine learning application. A neural network accelerator can be scaled to any size by using the tile technique to break up a vast amount of data into little tiles that are stored on the chip. Therefore, the FPGA-based accelerators is more adaptable to various machine learning applications.

5. Tools

5.1 Introduction:

To simplify things, we'll divide the key tools needed for this project into two categories.
The use of hardware is a prerequisite.
Software is a requirement.

5.2 Hardware Requirements:

The FPGA KIT
Xilinx ISE 10.1i coding can be easily run on a PC with a Pentium III processor, 1 GB of RAM, or a 20 GB hard drive, as long as the hardware is adequate.

5.3 Software Requirements:

MODELSIM 6.4b
XILINX 10.1 is now available.
Plan execution is possible only with the use of source code from the Virion project.

6. Results&Discussion

Fig. 4. Entity diagram:

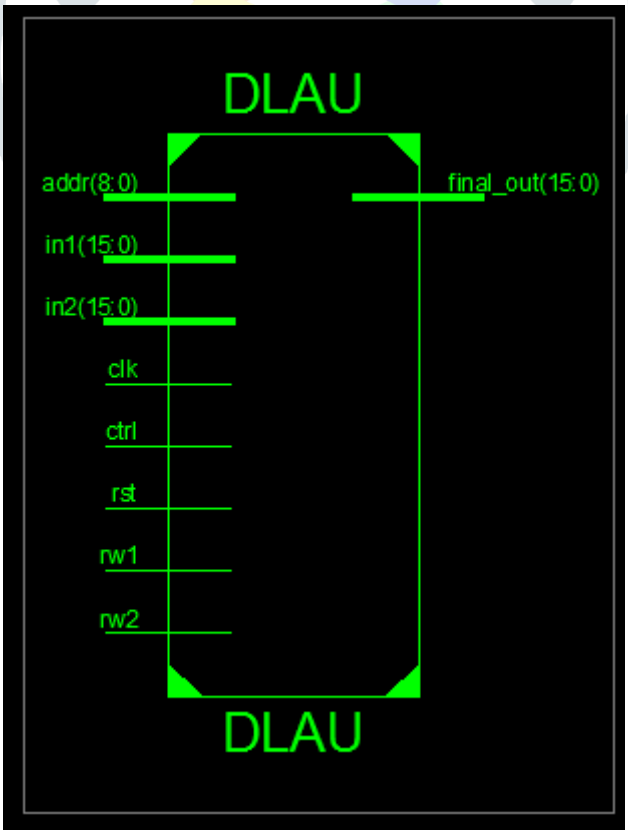


Fig. 5. RTL schematic:

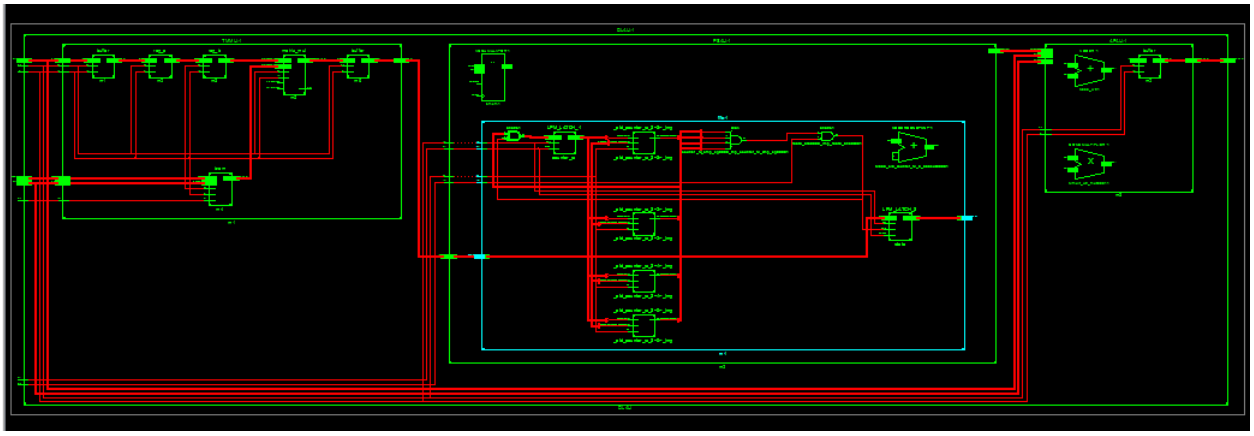
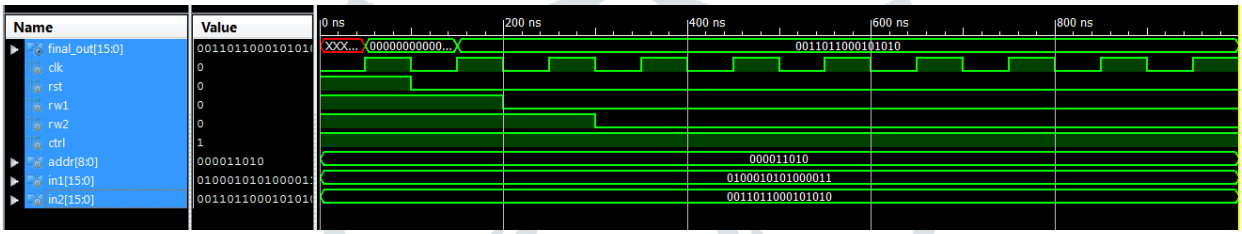


Fig. 6. Simulation results:



7. Conclusion

DLAU, a scalable and adaptable computational intelligence accelerator based on FPGA, was presented in this research. For large-scale neural networks, the DLAU's three pipelined hardware resources can be recycled. Tile techniques are employed by DLAU in order to divide up the input network data into smaller sets, which are then repeatedly computed using time-sharing logic. Prototype tests suggest that with reasonable hardware costs and minimal power consumption, the 36.1x speedup achieved by the DLUX algorithm can be achieved by DLAU.

8.References

[1] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 436–444, 2015.

[2] J. Hauswald et al., “DjiNN and Tonic: DNN as a service and its implications for future warehouse scale computers,” in Proc. ISCA, Portland, OR, USA, 2015, pp. 27–40.

[3] C. Zhang et al., “Optimizing FPGA-based accelerator design for deep convolutional neural networks,” in Proc. FPGA, Monterey, CA, USA, 2015, pp. 161–170.

[4] P. Thibodeau. Data Centers are the New Polluters. Accessed on Apr. 4, 2016. [Online]. Available: http://www.computerworld.com/ article/2598562/data-center/data-centers-are-the-new-polluters.html.

[5] D. L. Ly and P. Chow, “A high-performance FPGA architecture for restricted Boltzmann machines,” in Proc. FPGA, Monterey, CA, USA, 2009, pp. 73–82.

[6] T. Chen et al., “DianNao: A small-footprint high-throughput accelerator for ubiquitous machine-learning,” in Proc. ASPLOS, Salt Lake City, UT, USA, 2014, pp. 269–284.

[7] S. K. Kim, L. C. McAfee, P. L. McMahon, and K. Olukotun, "A highly scalable restricted Boltzmann machine FPGA implementation," in Proc. FPL, Prague, Czech Republic, 2009, pp. 367–372.

[8] Q. Yu, C. Wang, X. Ma, X. Li, and X. Zhou, "A deep learning prediction process accelerator based FPGA," in Proc. CCGRID, Shenzhen, China, 2015, pp. 1159–1162.

[9] J. Qiu et al., "Going deeper with embedded FPGA platform for convolutional neural network," in Proc. FPGA, Monterey, CA, USA, 2016, pp. 26–35.