Wafer Automatic Defect Classification *Using FPGA accelerator*

Dongbin Shin, SeungJu Byun, KangHyeon Cha and EunSeong Lee
Department of Electronic Engineering, Kwangwoon University, South Korea
E-mail: dongbin4013@kw.ac.kr
Advisor: Prof. Seoung-Jun Oh

Abstract— **Wafer bin map has a lot of data. In order to process this efficiently, deep learning technology is being introduced. Previously, VGG-16 was used as a feature extractor on the GPU and the output was processed by post-processing. However, GPU requires high power consumption since its architecture is not optimized for convolution operation. More over GPU prices are getting expensive these days. In this study, we propose designing FPGA accelerator with purposes that improve 50 times of the power efficiency and more than 2 times cheaper by using FPGA.**

***Index Terms*—Wafer Bin Map, FPGA, CNN, VGG-16, Kibana**

I. Introduction

In the semiconductor production process, a lot of time and resources are invested in one wafer. Therefore, semiconductor yield is a key factor for a company's competitiveness. It is very important to maintain a high yield through quick cause identification and improvement work for defects.[1][2] In order to increase the semiconductor yield, the defective pattern is analyzed. A defect forming a specific pattern is likely to indicate a problem of a specific process, and a semiconductor chip having the same defective pattern is also likely to have the same cause. At present, in most companies, engineers perform post-analysis on the Wafer Bin Map (WBM) defect pattern classification, so differences arise depending on the capabilities between engineers. In addition, there is a problem of cost and time to inject many engineers to meet the high semiconductor demand.[3]

To overcome this, research using deep learning is actively underway. Park et al. (2018)[4] proposed WBM classification using CNN. However, this study has a disadvantage that it is not possible to classify a new pattern of WBM. If a new pattern cannot be classified and extracted, various causes cannot be analyzed and resolved. Cheon et al. (2019)[5] proposed a method of classifying WBM by combining CNN and distance measurement algorithm. We classified existing defective patterns through CNN and proposed clustering using the Euclidean Distance measurement method, which is a distance measurement method for unclassified defective patterns. This method ensures stable ADC performance.

In the existing GPU deep learning, most of the propagation processing time is consumed in the convolutional layer. Accordingly, the amount of power consumption increases. FPGAs show superior performance and power consumption compared to GPUs in data parallelization and cumulative multiplication operations. Therefore, if the 2D convolution acceleration layer part is solved by designing an FPGA accelerator, it is possible to gain benefits in computation time and power consumption.[6] Therefore, in this study, we will design an FPGA accelerator by extracting the weight after learning Wafer Automatic Defect Classification in advance. After that, post-processing will be performed on the unclassified WBM pattern in a distance measurement method.

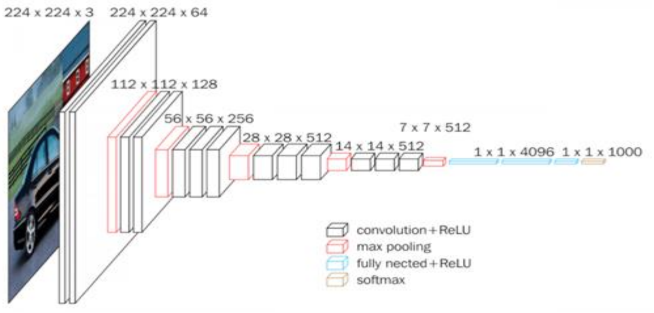
II. Proposed Design

A. Dataset

The data used in this study is open as a binary map, and the WM-811K dataset, which is used most, is used. It is a wafer map of 811,457 sheets, collected from actual fabrication of 46,393 lots. Each image is composed of 26 * 26.

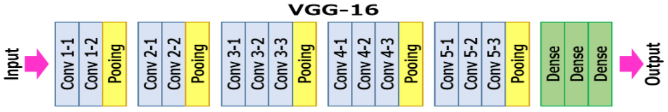
B. VGG-16(Visual Geometry Group from Oxford)

VGG-16 Overview



VGG-16 is a convolutional neural network composed of 16 layers. Multiple layers can be stacked using a 3*3 filter on all convolutional layers by reducing the number of parameters to increase nonlinearity. In general, as the depth of the layer increases, it consists of 16 layers to detect the features of the object in detail.

[VGG-16 Flow]



The VGG-16 structure consists of 13 convolution layers (feature extraction) + 3 fully-connected layers (classification process), 3*3 convolution filters, stride: 1& padding: 1, 2*2 max pooling (stride:2), ReLU.

In the case of WBM, it was confirmed in previous studies that the VGG-16 structure is most specialized for feature map detection. (S.cheon 2019)

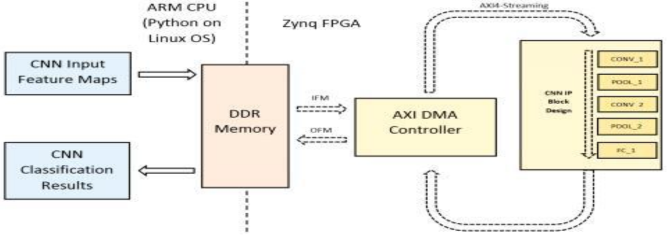
C. *Post processing*

This is for the operation of sorting wafers that have not yet been sorted after sorting through VGG-16. Therefore, it goes through the process of classifying into unclassified ones once more through the method of determining a new class for the ones that exist above the set distance (threshold point) for which the method of measuring the distance is set. For the distance, the Euclidean distance measurement method was used, which is a method of obtaining the shortest distance between a point and a point. The Euclidean distance measurement equation is as follows.

$$\sqrt{(Ax - Bx)^2 + (Ay - By)^2}$$

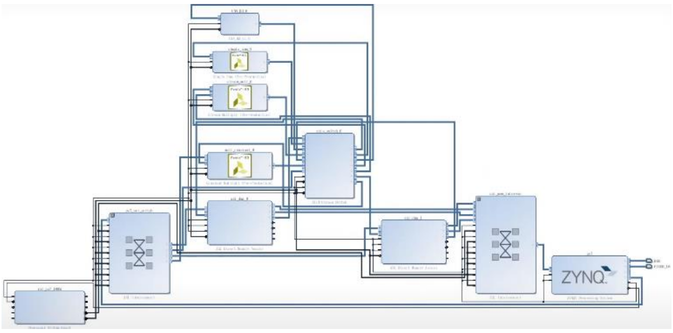
D. *FPGA*

Proposed FPGA Structure

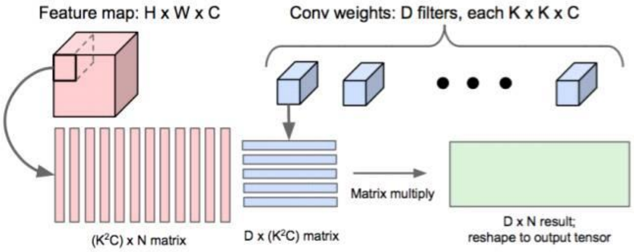


It is divided into two parts: ARM Linux and Zynq FPGA implementation in host PC. In Linux, the part controlled by python and deep learning framework is set at a high level, and the framework loads the input features from the DDR memory of the board and outputs the classification result again. After using DMA controller API, design CNN IP written in Verilog and perform high speed forward propagation. This is an accelerator structure aimed at constructing a high-performance CNN [7]

[IP Design Diagram] (in Vivado 2020.02)



1. im2col[8]



2. Pooling Layer

3. Fully-connected Layer

ID	Batch	Convolution (커널)	input feature map (채널)	input feature map dimension	output feature map (채널)	output feature map dimension	padding dimension

4. data(parameter) flow

[API driver to use]

Data transfer request from DDR under the framework moves from DDR to FPGA embedded BRAM.

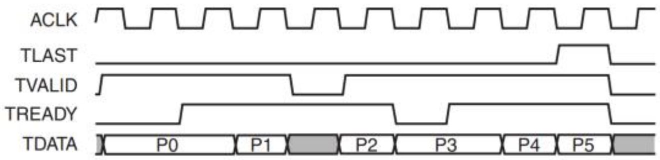
1. MMIO (memory-mapped input/output)

With the memory access protocol, the pointer initializes the pointer of the control/status register of the FPGA, and then the MMIO loads and stores it through the pointer in the FPGA.

2. DMA (Direct Memory Access)

With direct memory access, data transfer is independently controlled and reserved through the FPGA's built-in controller. It is transmitted at maximum throughput. The provided AXI DMA transports data streams in the AXI4-Streaming protocol.

[Waveform of AXI4-streaming interface]

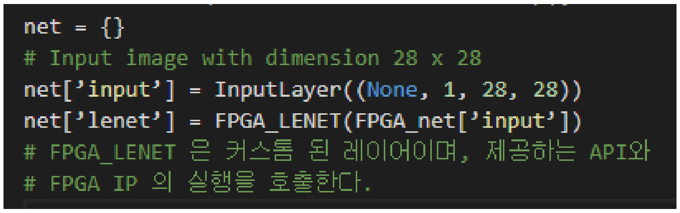


Supports streaming rate of one word per clock cycle (1 word per clock cycle is used in 1 layer IP, and every new word is output.)[9]

[Framework]

Among Caffe, TensorFlow, and Theano, we plan to use Theano and Caffe frameworks, which are said to be the most compatible cross-platform when built on PYNQ linux OS. (Wei Dai. 2019)

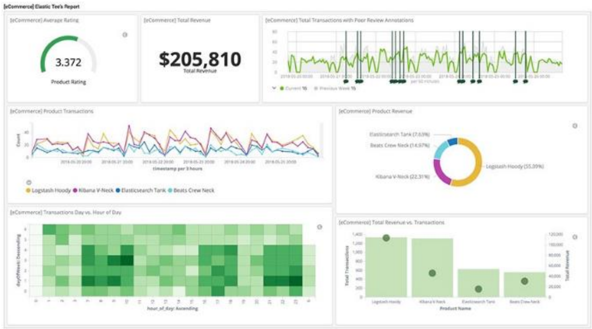
[FPGA-CNN Accelerator 실행 예제][10]



```
net = {}
# Input image with dimension 28 x 28
net['input'] = InputLayer((None, 1, 28, 28))
net['lenet'] = FPGA_LENET(FPGA_net['input'])
# FPGA_LENET 은 커스톰 된 레이어이며, 제공하는 API와
# FPGA IP 의 실행을 호출한다.
```

E. Visualize(Kibana)

Kibana is an open source front-end application built on the Elastic Stack. It searches, visualizes, and analyzes indexed data in Elastic Search. It uses various visualization tools such as histogram, GEO map, and chart to improve data readability by combining it with the dashboard specified by the user.



(https://www.elastic.co/kr/kibana/features)

As shown in the figure above, big data can be provided to users in the form of a highly readable service through various visualization tools. In this study, the results of the analyzed wafer data will be provided in Kibana format rather than a simple console window. This can reduce the difference in interpretation between engineers who check the classified data and increase convenience.

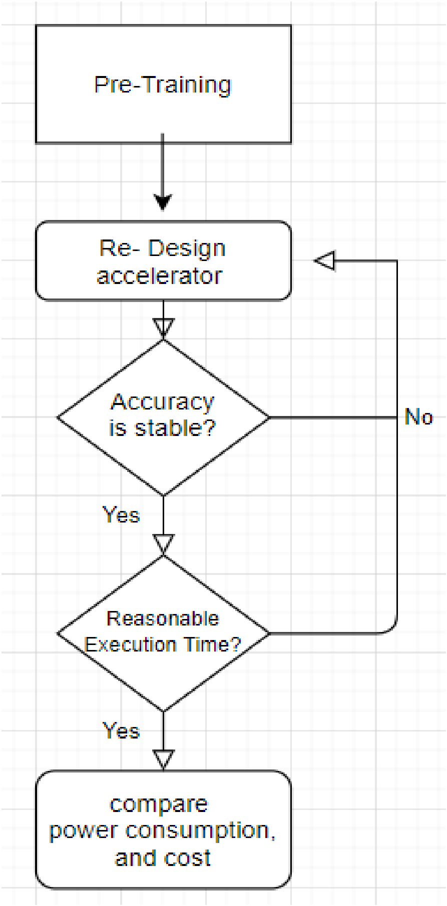
III. Performance Evaluation

System	i5-10400F GTX 1060 3gb	i5-10400F RTX 2070	pynq z2
cost (msrp)	$157 / $249	$157 / $599	$119

The FPGA used in this study is PYNQ-Z2, with an MSRP of $199. Also, for the GPU, the GTX1060-3GB and RTX-2070 cost $199 and $599, respectively. The reason for the use of a GPU that has a three-fold difference in price was used to confirm that there is no difference in price and power/efficiency through comparison between an FPGA and a GPU with a similar price.

When the FPGA accelerator derives similar run time and accuracy compared to the conventional GPU accelerator, we will measure how much price, average power consumption, and maximum power consumption differ from each other.

<Project Flow Chart>


Pre-Training
Re- Design accelerator
Accuracy is stable?
No
Yes
Reasonable Execution Time?
Yes
compare power consumption, and cost

IV. Conclusion

In this study, when using an FPGA instead of a GPU as an accelerator, it is expected to remain similar in terms of accuracy and time, and at the same time show an advantage in efficiency and cost versus power. By comparing the FPGA and GPU with the RTX-2070, which is a GPU that is three times the price of an FPGA and GPU with the same price as possible, the FPGA will dominate in terms of power efficiency and price. Therefore, the resulting advantage is that in terms of power efficiency, FPGAs differ by more than 50 times compared to GPUs, and in the case of GPU (RTX-2070), they will show a difference of more than two times in terms of price.

V. References

[1] Han Young Shin, Hwang Mi Young, Lee Chil Gee, (2003) Automatic classification of failure patterns in semiconductor

EDS Test using pattern recognition, The Institute of Electronics and Information Engineers 2003.7,703-706

[2] Lee Chang Hyun, Ki Seoung Bum, (2019) Identifying Wafer Defect Patterns by Variational Autoencoder and SegNet, Journal of the Korean Institute of Industrial Engineers 45(2), 117-124

[3] Liu, C.-W. and Chien, C.-F. (2013), An intelligent system for wafer bin map defect diagnosis: An empirical study for semiconductor manufacturing, Engineering Applications of Artificial Intelligence, 26(5),1479-1486.

[4] Jaesun Park, Junhong Kim, Hyungseok Kim, Kyounghyun Mo, Pilsung Kang, (2018) Wafer Map-based Defect Detection Using Convolutional Neural Networks, Journal of the Korean Institute of Industrial Engineers 44(4), 2018.8, 249-258(10 pages)

[5] Sejune Cheon, Hankang Lee, Chang Ouk Kim, Seok Hyung Lee, (2019) Convolutional Neural Network for Wafer Surface Defect Classification and the Detection of Unknown Defect Class, IEEE TRANSACTIONS ON SEMICONDUCTOR MANUFACTURING, VOL. 32, NO. 2, MAY 2019

[6] Chao Huang, Siyu Ni, Gengsheng Chen, (2017) A Layer-based Structured Design of CNN on FPGA, 2017 IEEE 12th International Conference on ASIC (ASICON),10.1109/ASICON.2017.8252656

[7] Xilinx. Vivado high-level synthesis.

[8] Ben Cope Implementation of 2d convolution on fpga, gpu and cpu.

[9] Xilinx. Axi reference guide. 2011.

[10]Hongxiang Fan, Martin Ferianc, Shuanglong Liu, Zhiqiang Que, Xinyu Niu, Wayne Luk, (2020) Optimizing FPGA-Based CNN Accelerator Using Differentiable Neural Architecture Search, 2020 IEEE 38th International Conference on Computer Design(ICCD),10.1109/ICCD50377.2020.00085