Microsystem Technologies (2023) 29:1069–1081
https://doi.org/10.1007/s00542-023-05442-9

TECHNICAL PAPER

Design and implementation for a high-efficiency hardware accelerator to realize the learning machine for predicting OLED degradation

I.-Feng Chang[1] · Hao-Ren Chen[1] · Paul C.-P. Chao[1]

Received: 10 August 2022 / Accepted: 20 March 2023 / Published online: 5 April 2023

Abstract

A new learning machine based on neural network (NN) and its hardware accelerator are successfully built in this study for predicting the luminance decay of Organic Light Emitting Diode (OLED) displays. It is known that although OLED displays has become the mainstream in the current high-end display market, OLEDs tend to degrade in emission as used extensively for a long time. The operable voltage also rises with the usage time increasing, which causes the operating point to drift. To compensate the OLED degradation, a NN model is successfully built with favorable accuracy. Furthermore, the built NN model is implemented in FPGA hardware platform with a high-performance computing architecture, which uses registers to access inputs, integrates the multiplication, addition operations for weights and activation function into the same combinational logic circuit, and a pipeline architecture to improve maximum operation per unit time. The hardware architecture is designed via Verilog, and further verified by Xilinx Artix-7. Its operating frequency can be as high as 55.6 MHz, while resource consumption is only 1.0k LUTs, favorable as opposed to all the other past, related studies. Experiment shows that the computation of the built NN by the proposed accelerator can be completed 55.6 million times per second. In addition, the degradation prediction errors by the accelerator are as small as 2.08%, 5.51% and 4.36% for red, green and blue OLEDs, respectively, while the figure of merit, the product of computation time and area is as low as 109.86 (Time*Area), the lowest compared to all the past reported works.

1 Introduction

Organic Light Emitting Diode displays (OLED displays) offer many advantages, such as active light emission, high reaction speed, low power consumption, wide viewing angle, wide color gamut, low operating voltage, thin panel thickness, and simple manufacturing that can be applied to flexible panels. Thus, it has become the mainstream of high-end display applications in recent years. However, there are common shortcomings with OLED displays, including inevitable drifts of threshold voltages of OLEDs, which lead to performance degradation of an OLED display panel, such as brightness non-uniformity, mura phenomenon and burn-in of caused by the long-term usage under high temperature stress. Since the degradation of OLED is difficult to eradicate from re-design and alternating manufacturing process, efforts by many studies were dedicated to establish a prediction model on OLED degradation, i.e., a Neural Network (NN) model. Based on the model, the strategy of emission compensation for OLEDs used over an extensive time period can be distilled to overcome the degradation effectively via adjusting drive current for OLEDs to required levels.

A few studies in recent years were dedicated to degradation modeling of OLEDs. Liu et al. (2017) employed an NN to model LED's photo-electro-thermal (PET) behavior, with temperature and current as the inputs of the NN to predict the luminance drop, efficiency and lifetime of LEDs. Liu et al. (2019) in 2019 proposed a two-stage NN to estimate the lifetime of LEDs, yet it can only be used in high-powered modules with 150 mA input present. Lu et al. (2017) proposed a different NN in 2017, which considered LED's current, temperature, lumens and the chromaticity coordinate to predict LED degradation, where the back propagation (BP) NN was employed for realizing the afore-mentioned NN. It requires a collection of training data with "input features" and "target results" to find a set of linked weights that allow input data to travel through

✉ Paul C.-P. Chao
pchao@nycu.edu.tw

[1] Department of Electronics and Electrical Engineering, National Yang Ming Chiao Tung University, Hsinchu 300, Taiwan

this set of weights to achieve the target value. The lifetime of the LED is then estimated by the inputting current, temperature, luminance and other data to the neural network built. Since the characteristics of LED and OLED are similar, the idea of this approach can also be used in this study to predict degradation OLED degradation. Note that all the aforementioned models are machine-learning models, while the method of compensating luminance degradation has not yet been suggested. In this study, not only is a BP-NN algorithm similar to that presented in Lu et al. (2017) built towards minimized errors, but also effective compensation schemes orchestrated successfully, and, most importantly, the degradation model is implemented in an hardware accelerator with minimized resource consumed. It is pertinent to note herein that none of all the above past studies on degradation modeling ever completed the realization of the model into a hardware accelerator, not to mention on OLED displays. In fact, the present work is dedicated to design and implement a hardware accelerator of the degradation model by the technique of field programmable gate array (FPGA) to drive and compensate an OLED display. The platform of Xilinx-XC7Z020-1CLG400C SOC with the capability of 50 million operations per second is utilized for this FPGA implementation. The performance of the built accelerator is compared to other most recent works on the hardware implementation of NN (Oliveira et al. 2017; Medus et al. 2019; Zhai et al. 2016; Nedjah et al. 2012).

This paper consist of five sections as follows. Section 1 gives the motivation and purposes. Section 2 introduces the method of building a NN model to predict OLED degradation. Section 3 designs the architecture of the built-in feedforward NN model in FPGA. Section 4 present the performance verification. Section 5 concludes this work.

2 Establishing NN models

To establish a NN model to predict OLED degradation, experiments are first conducted on an OLED panel lit upon different conditions and then observing its luminance decay over time. With varied current and temperature, three NN models are established via the software of Tensorflow, for red, green and blue OLEDs. These models consists mainly of multiple identical Multilayer Perceptron (MLP) units connected each other in a network, as shown in Fig. 1. The inputs of the models are operation time period t, drive current I, temperature T of OLEDs considered, while the output of the models is the predicted luminance. Measurements on OLED degradation are shown in Figs. 2 and 3. Figure 2 shows the degradation data of OLEDs at 26 °C, while Fig. 3 shows does 60 °C. The "relatively aging gray level = 1" refers to the case that, the lit OLEDs are at the maximum luminance. The initial luminances of red OLED at relatively aging gray level = 1 for measuring degradation as presented in Figs. 2 and 3 are 282, 928 and 99 nits for red, green and blue OLEDs, respectively.

For each of three models for red, green and blue OLEDs, 360 different combinations of (t, I, T) are randomly selected within their corresponding operations ranges for training while 840 combinations of (t, I, T) for testing. The selected combinations of (t, I, T) are normalized before being input to the NN model seen in Fig. 1 for training and testing. The normalization is carried out by

$$x_{\text{norm}} = \frac{(x - \mu)}{\sigma}, \tag{1}$$

where μ is the mean of all considered (t, I, T)'s while σ is the standard deviation. Figure 4 show the evolutions of loss during training on the OLED degradation model. It can be clearly seen from these figures that as the number of epochs rises, the losses stabilizes to very low values, and furthermore, the evolutions in losses of training and validation are consistent, indicating that the models do not overfit during the training process. Having the NN models successfully built, the prediction by the models are carried out with results shown in Fig. 5, while the accuracy of three degradation prediction models in values listed in Table 1, where mean absolute percentage error (MAPE) is adopted to evaluate the accuracy. It can be seen from this table that all the models lead to favorable accuracies, though the accuracies are slightly different among three models. The model for red OLED delivers the highest accuracy, while the model for green OLED renders the worst accuracy. It should be noted at this point that the prediction accuracy for the established models does drop over time, but being very limited. For the example of green OLED, the prediction accuracy of degradation over 400 h is 94.72%, while that of degradation over 1000 h is 93.45%, kept close to 94.72%. For red OLED, the prediction accuracy of degradation over 400 h is 97.83%, while that of degradation over 1000 h is 95.62%, kept close to 97.83%. As for blue OLED, the prediction accuracy of degradation over 400 h is 95.84%, while that of degradation over 1000 h is 95.43%, kept close to 95.84%. It is strongly shown that the decreases in accuracies over time up to 1000 h for three color LEDs are very limited.

Based on the accurate models successfully built on OLED degradation, effective schemes for compensating OLED emission to their originally-designated greys can easily be orchestrated. Seen from Table 2 are significant reductions in errors of displayed grey level by compensation at greys of 123, 168, 202 and 230. The reductions by compensation averaged over red, green and blue OLEDs in grey are 5.33 (= 7.58–2.25), 7.08 (= 9.49–2.41), 8.05

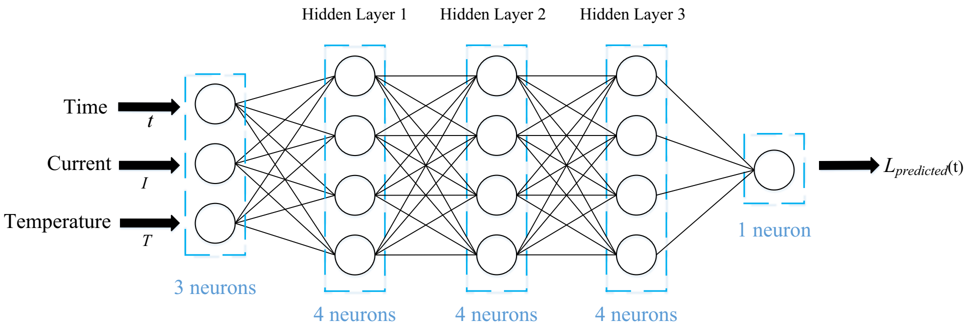


Fig. 1 The proposed NN model for predicting OLED degradation

(= 10.79–2.74) and 7.92 (= 11.89–3.97) at levels of 123, 168, 202 and 230, respectively, leading to an overall averaged reduction in error as low as 7.1 greys out of 255 (8 bits) over three colors of red, green and blue. Thus, the performance of the proposed three NN model for estimating OLED degradation is proven very effective for OLED luminance compensation.

3 Hardware design of NN model

The architecture of the feedforward NN used to predict OLED degradation is shown in Fig. 1. The input data, weights and biases of this NN are obtained off-line via training using the software of Tensorflow. The computation for training is conducted in the format of floating-point values, while fixed-point values are used for the hardware design (Aoyama et al. 2002). If the selected number of fractional bits is less than 12, it will cause a non-negligible error. Therefore, we decide to use 16 bits of data width (2 bytes) as the input data and weights of hardware computation, including one sign bit, two integer bits and 13 fractional bits. The conversion between floating- and fixed-point data is shown in Fig. 6. First, the decimal representation of the original data is multiplied by 2^{13}. Next, round down the decimal part of the data, and then convert it to binary representation in signed 2's complements. Finally, having divided it by 2^{13}, the data in fixed-point format is obtained.

The conventional process for implementing feedforward NN in hardware generally takes hardware consumption and computing speed into account. Thus, external memory of ROM (Read-only Memory) or RAM (Random Access Memory) is utilized to store the weights and biases (Hao 1711; Pearson et al. 2007). However, it takes too much time for data reading and writing. To solve the problem, a centralized controller is designed to control the order of computations. Also, to complete the computation of a feedforward NN in a fast speed in hardware, a register array is employed to store the weights and biases, which can be accessed immediately. Effort in the next is paid to optimize the architecture of NN models towards minimum computation, via reducing the numbers of neurons and hidden layers. With the register array, a new configuration of hardware including pipeline architectures is proposed to implement the feedforward NN in hardware, using its feature of real-time data access to achieve accelerated computation. The proposed new configuration is illustrated by Fig. 7 and elaborated in subsections below.

3.1 Finite state machine (FSM)

A new scheme of finite state machine (FSM) is first proposed to control the computation flow by hardware, as seen in Fig. 8. This FSM adopts a neuron counter or a layer counter to switch among various states (Oliveira et al. 2017; Medus et al. 2019), while the designed FSM has only one active state at any given time for computation. When the reset signal is active, the FSM enters the state of S0, which can be regarded as an idle state, waiting for valid signals of weights and biases to be pulled high to enter the S1 state. In the S1 state, the weights and biases of the model are set. Then, the signals of weights and biases are pulled down while entering the S2 state. The FSM remains S2 until the input valid signal is pulled high. Then, it enters the S3 state, which is the part of computation for NN. Since the built-in NN models are in a five-layer structure, the input data must go through four layers of operation to arrive at the final output. Therefore, once the input counter reaches four in the S3 state, the computation results are continuously out until computations of all input data are finished. Finally, the FSM returns to S2 to wait for new input data.

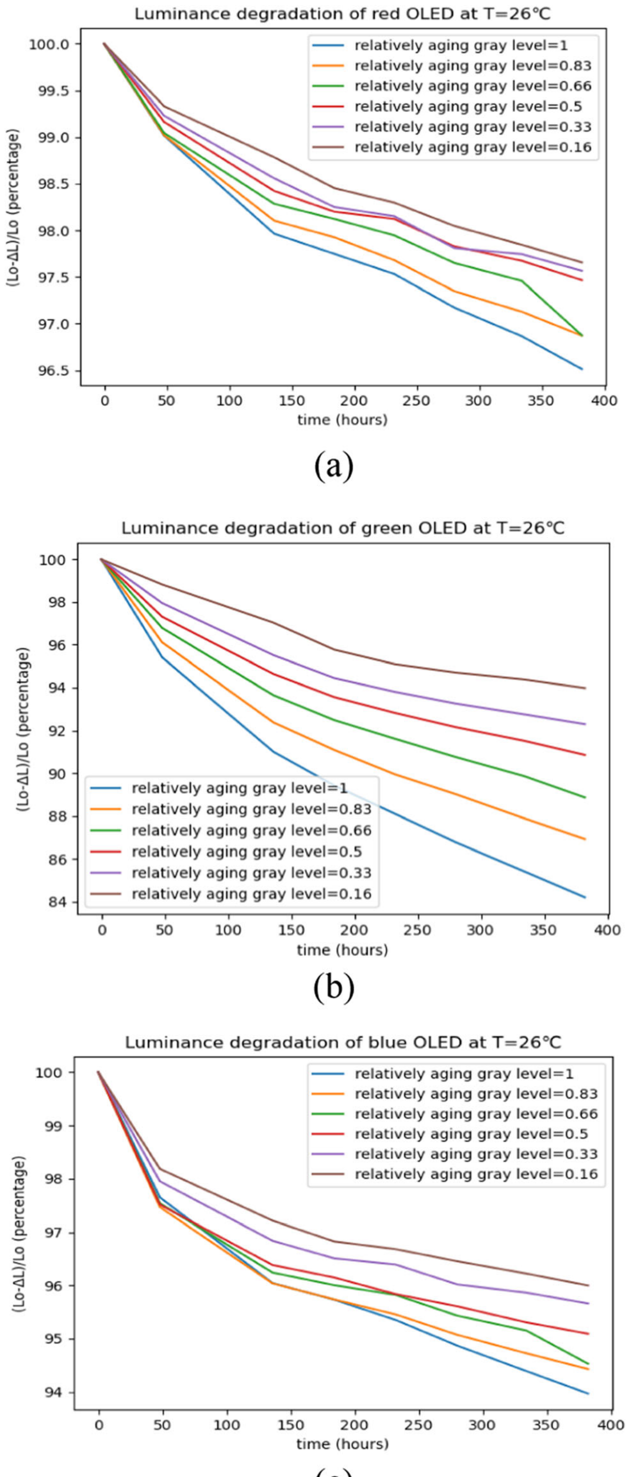


Fig. 2 Luminance degradation of OLEDs at $T = 26$ °C, **a** red; **b** green; **c** blue (color figure online)

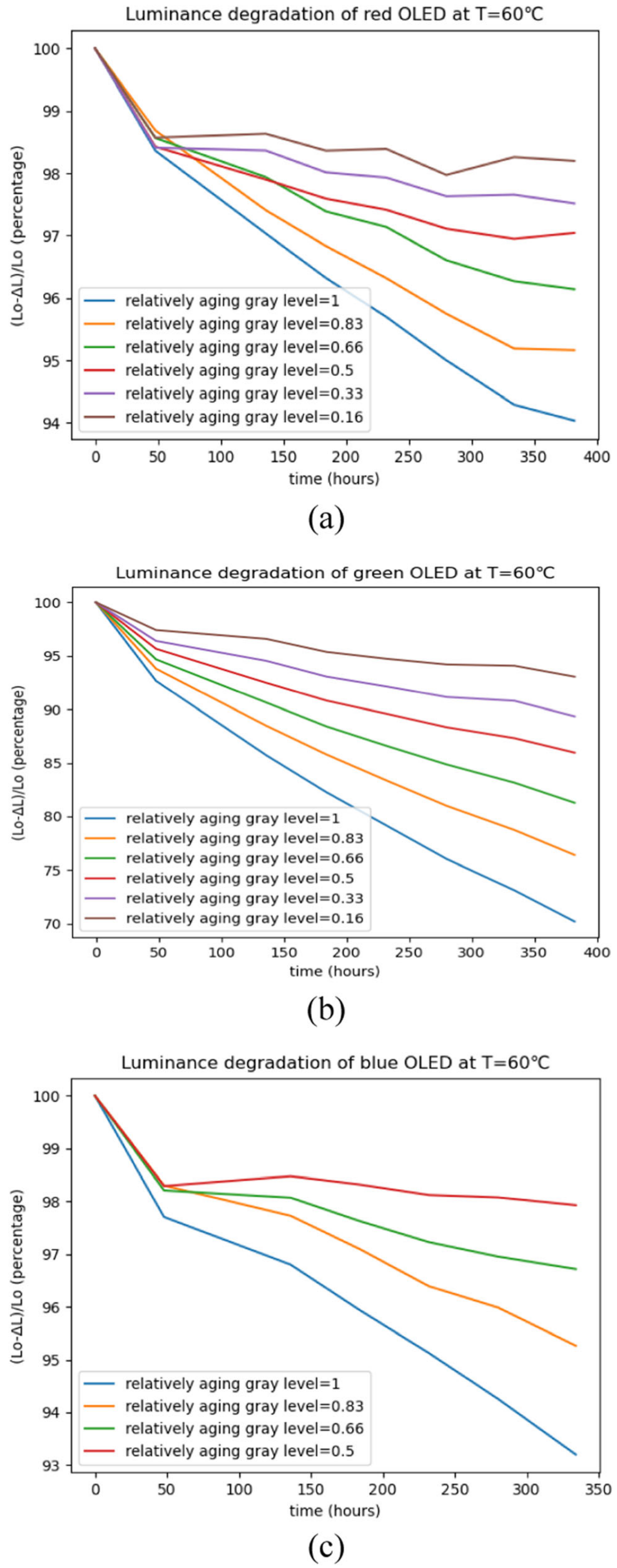
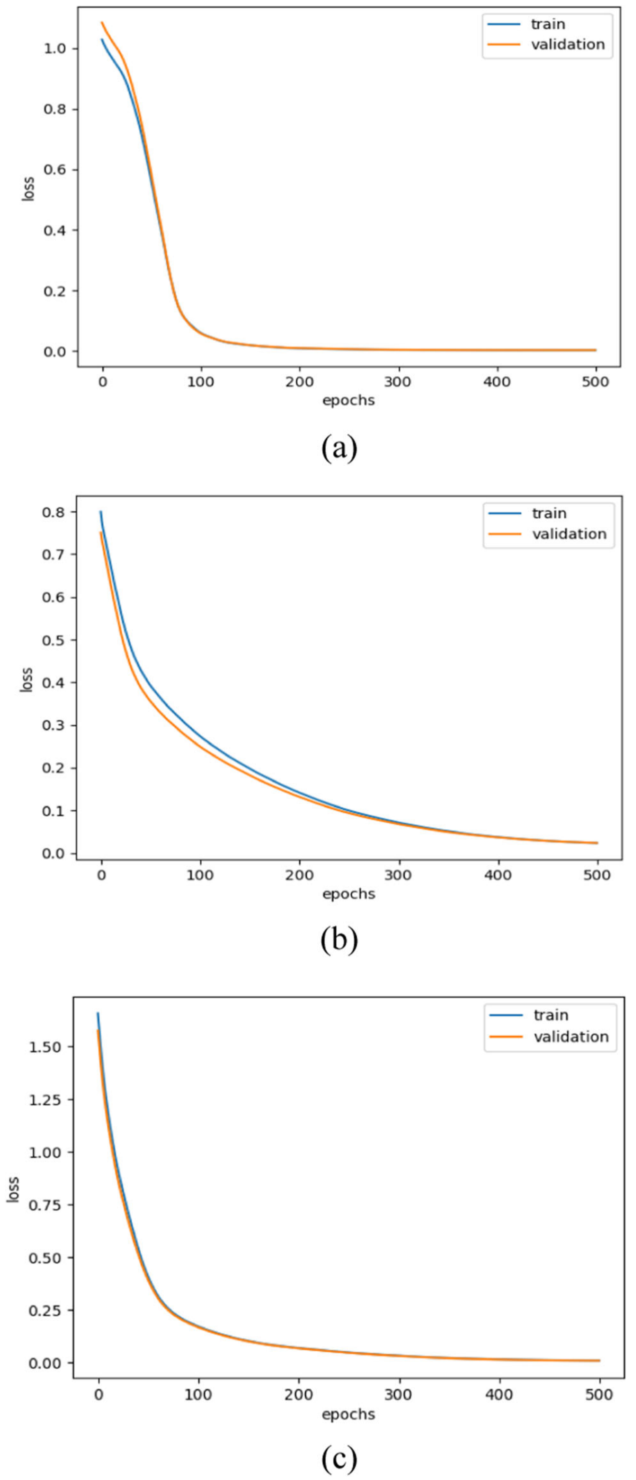


Fig. 3 The luminance degradations of OLEDs at $T = 60$ °C, **a** red; **b** green; **c** blue (color figure online)

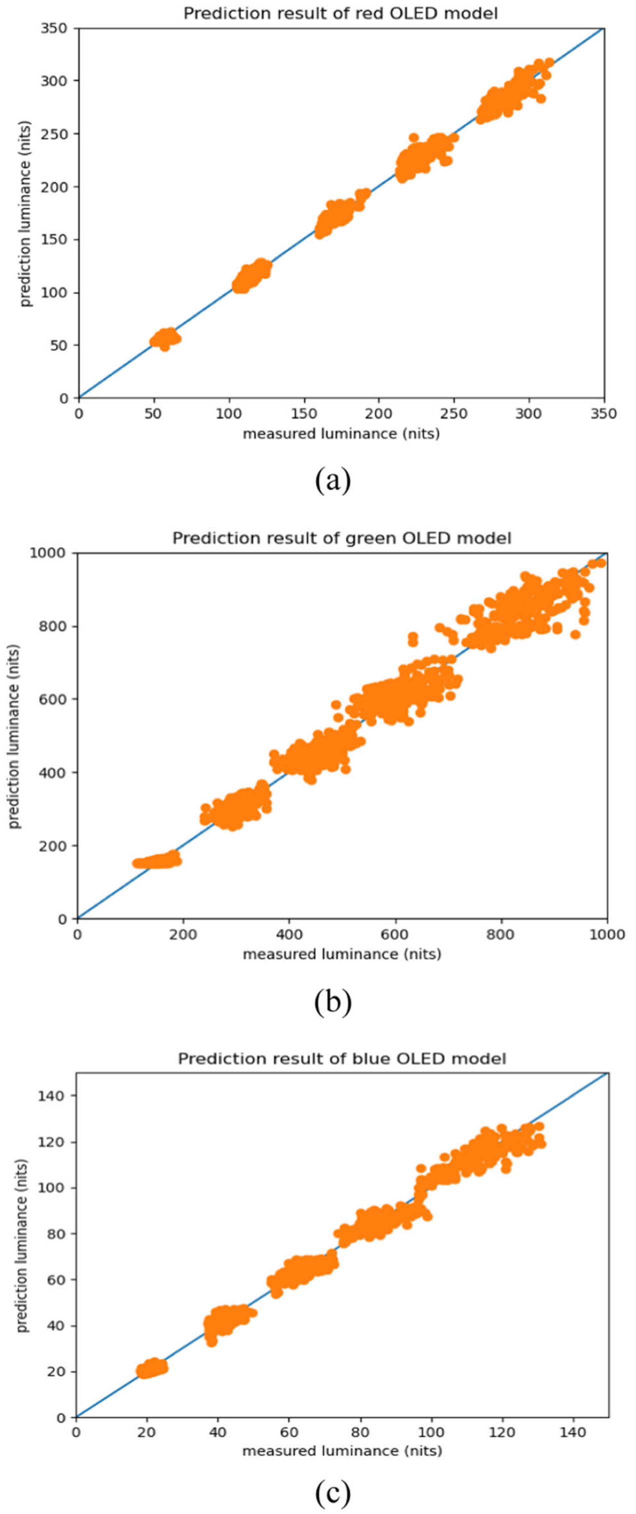


(a)

(b)

(c)

Fig. 4 The loss of training on OLED degradation models (MSE) for **a** red; **b** green; **c** blue (color figure online)



(a)

(b)

(c)

Fig. 5 The prediction results by the established OLED degradation models, **a** red; **b** green; **c** blue (color figure online)

3.2 Combinational logics of layer calculation module

The approach of improving computational efficiency adopted herein is to optimize the allocation of combinational logics. The Layer Calculation Blocks shown in Fig. 7 can be considered as an independent combinational module, which can adopted in multiples to complete a whole layer of neuron operations belonging to the hidden

Table 1 Resulted accuracies and errors by the established OLED degradation models

Degradation prediction model	Red OLED	Green OLED	Blue OLED
Mean absolute error (MAE in nits)	3.33	23.75	2.85
Mean absolute error (MAE in gray level)	1.85	7.28	2.56
Standard deviation of error (nits)	4.66	33.85	3.08
Standard deviation of error (gray level)	1.15	2.80	1.99
Accuracy in MAPE (nits)	97.83%	94.72%	95.84%

Table 2 Pre- and post-compensated degradations in gray levels

Gray level	Red OLED	Green OLED	Blue OLED	Averaged
123				
Pre-compensation	2.19	18.84	1.70	7.58
Post-compensation	1.42	4.18	1.15	2.25
168				
Pre-compensation	2.58	23.64	2.25	9.49
Post-compensation	1.06	3.88	2.30	2.41
202				
Pre-compensation	2.84	27.06	2.47	10.79
Post-compensation	1.06	5.36	1.80	2.74
230				
Pre-compensation	3.07	29.89	2.72	11.89
Post-compensation	1.75	7.91	2.26	3.97

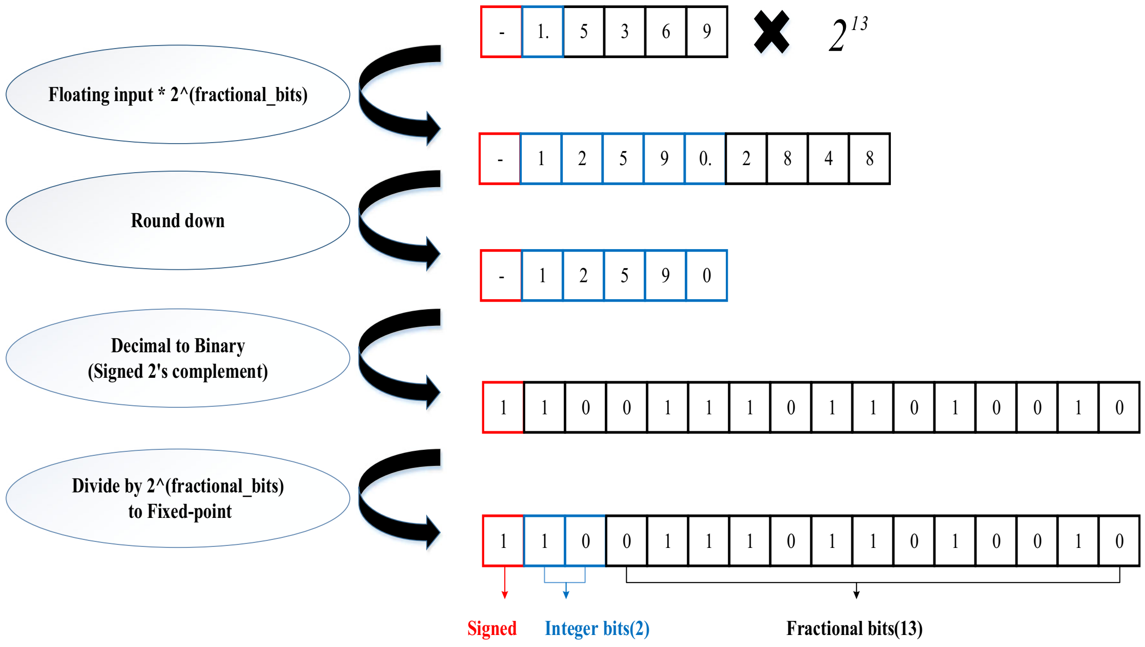


Fig. 6 The flow of data conversion from floating- to fixed-points

layer or output layer. As shown in Fig. 9, all the input data of the hidden layer are combined into a signal with a width of $j * 16$ bits. The width of the combined weights becomes $i * j * 16$ bits, while the width of the combined bias becomes $i * 16$ bits. Having multiplication and summation (sigma) conducted, the output width of each neuron becomes $(16 + 16 - 1) + j$ bits, which is obtained by multiplying one signed value with another and then accumulation. Finally, all neuron outputs are combined into a wider signal with a width of $i * [(16 + 16 - 1) + j]$ bits.

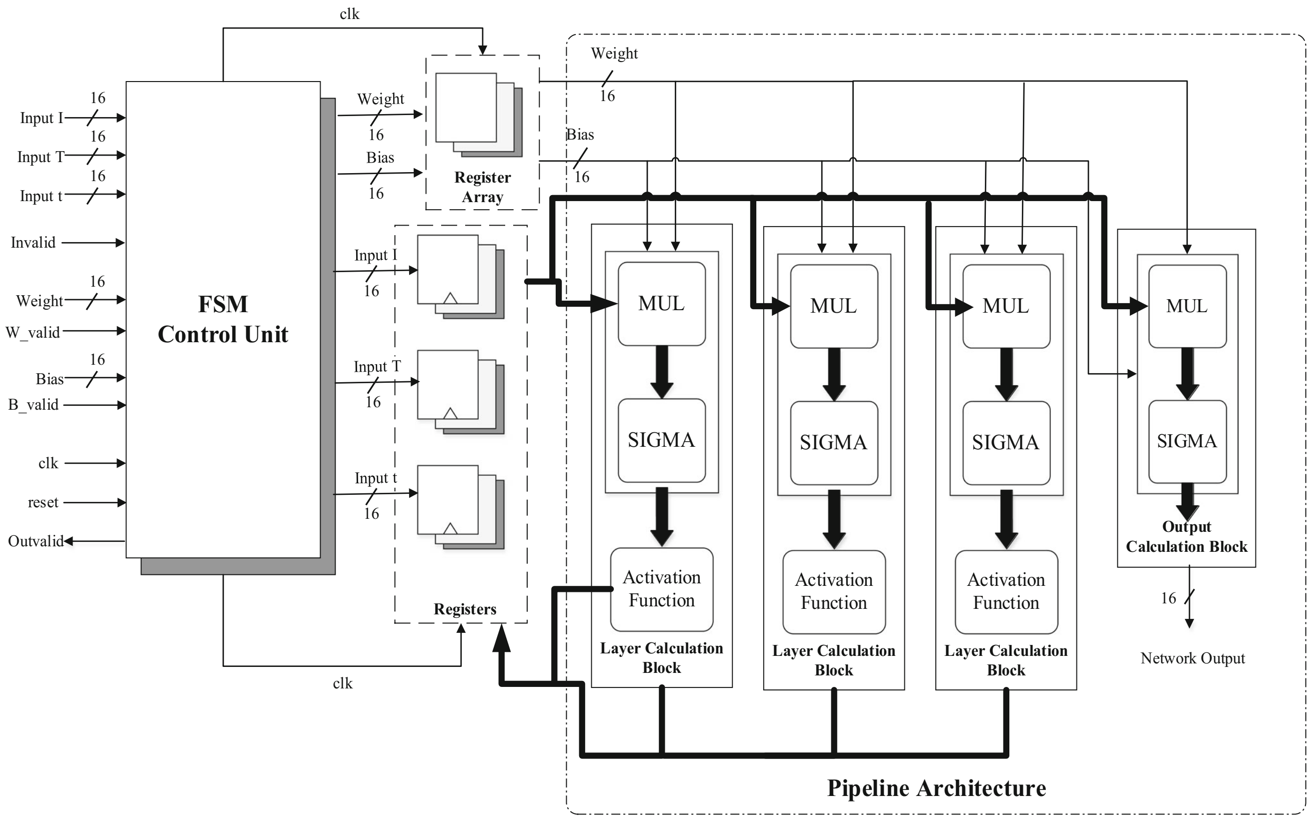


Fig. 7 The proposed hardware architecture for improving the efficiency of computing NN

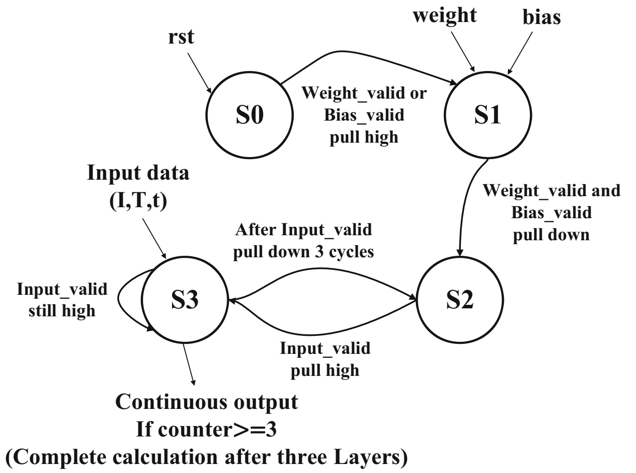


Fig. 8 The finite state machine (FSM) for FPGA hardware implementation

In the operation of the hidden layer, it contains multiplication, summation (sigma), and activation function for each neuron. Figure 10 shows the designed combinational logics of single neuron computation, the hardware description of which in a pseudo-code is given below.

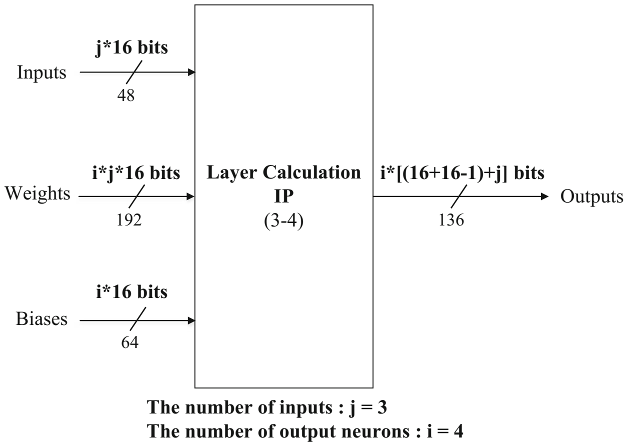


Fig. 9 Input and output formats of the layer calculation module

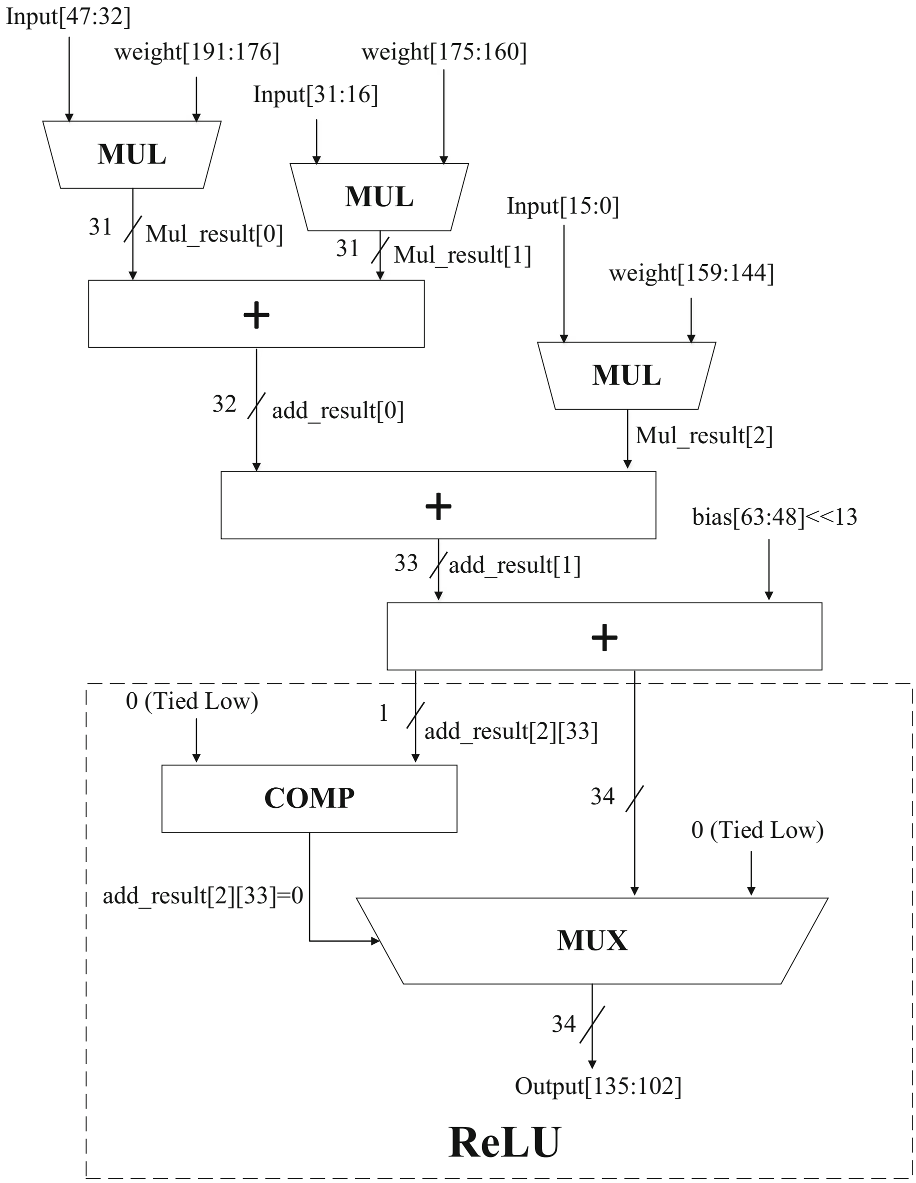


Fig. 10 The implementation logic of the calculation for a single neuron in the module of layer calculation

Hardware description: Layer Calculation

```
Input: [15:0] data[0:2], [15:0] weight[0:11]
Output: [15:0] network_result[0:3]
for each neuron i do:
  for each input data j do:
    mul_result[j] = data[j]*weight[i*3 + j];
    if (j != 2) then
      add_result[j] = mul_result[j] + mul_result[j+1];
    else
    add_result[j] = add_result[j-1] + (bias[i] << 13);
  end for
  output[i] = (add_result[2][33]=0) ? add_result[2] : 0;
end for
```

ReLU is selected as the activation function which is not only capable of accurate prediction results, but also reducing hardware consumption during the implementation of the feedforward NN (Medus et al. 2019). The traditional method for realizing the activation function in hardware is to build a large Look-up Table (LUT) in the circuit to reflect the output of the activation function accurately. It is replaced in this study by the combination of a comparator and a multiplexer to complete the operation of ReLU, which consumes much less resources.

3.3 Pipeline architecture

The aforementioned Layer Calculation module uses combinational logics to compute the output of each neuron in the hidden layer. Moreover, the pipeline architecture is adopted to improve the computational efficiency of the entire circuit, which can complete a NN computation in each clock cycle as shown in Fig. 11. At the positive edge of each clock, all registers send inputs to Layer Calculation modules. Each Layer Calculation module outputs the computational result to a register, reducing its width during transmission. Although each input needs four clock cycles to complete the computations of the entire NN, with the designed pipeline architecture, a piece of data can be calculated in every clock cycle, which leads to high computational efficiency.

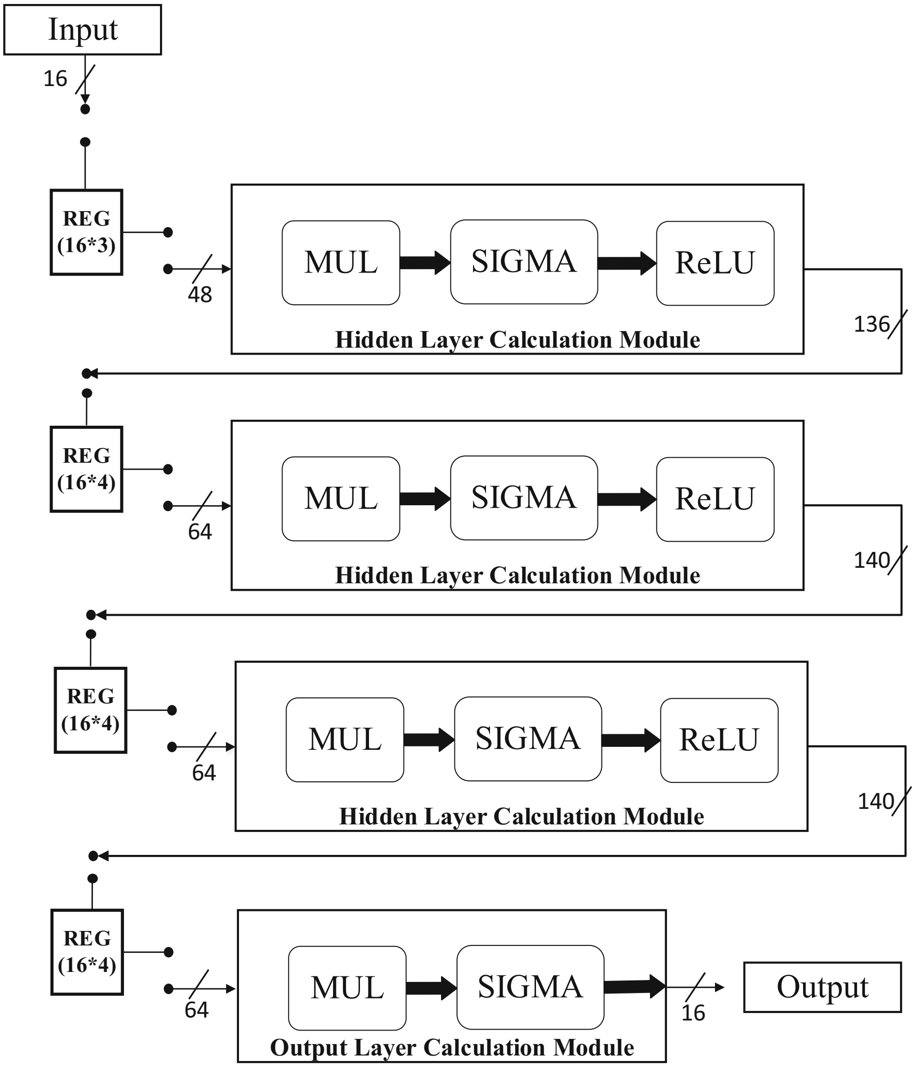
4 Experimental validation

Having finished the hardware implementation of the established models in an FPGA board, experiments are conducted to verify the accuracy of the established FPGA architectures calculating the built feedforward NN, as shown in Fig. 12. Prior to synthesizing Verilog code for FPGA, the Python-equivalent codes for realizing fixed- (C-model) and floating-point NN models were first built via the software of Tensorflow for performance assurance based on the comparison between the two models. In this way, the correctness of the computation by FPGA based on Verilog code can be ensured.

The design kit Xilinx Vivado was used to conduct the simulation of FPGA. The synthesized FPGA code is implemented in a circuit board Artix-7, with a core chip xc7a200tfbg676. Shown in Fig. 7 are the control pins of the user input interface, such as clock, reset, valid signals, current, temperature and time. On the other hand, the output interface includes the output of NN and the validity signal. In the designed FPGA code, the lock cycle is set up as 18 ns, prescribing the timing constraint of the code execution; that is, the delay time due to input and output are both 9 ns for each, which is half clock cycle. Having implemented the overall architecture in FPGA, another system of an Arduino board and its accompanying software is orchestrated to validate experimentally its correctness of predicting OLED degradation, as shown in Fig. 13. To this end, the input data is stored in ROM first and then read one by one. With inputs read, the FPGA code is executed to predict OLED degradations, which are next output to four pins based on the SPI protocols of MOSI, MISO, SCLK, SS, as seen in Fig. 14. Thus the data of predicted degradation was relayed to the SPI slave pin of the Arduino board, and further to a personal computer with both CPOL and CPHA set up as one for calculating the OLED prediction errors and showing results.

Having setting up experiment, the data of predicting OLED degradation was collected while degradation prediction errors by the hardware operation were obtained via the designed procedure seen in Fig. 12. For the three built-in OLED degradation models, 100 randomly selected combinations of operation time period t, drive current I, temperature T of OLEDs, (t, I, T)'s, are considered for evaluating the performance of compensation based on the built NN models. Mean absolute percentage error (MAPE) and Mean absolute error (MAE) are two indicators chosen to evaluate the performance of the models. Figures 15 and

Fig. 11 The pipeline architecture of the overall feedforward NN to be implemented by FPGA


Input
16
REG (16*3)
48
MUL
SIGMA
ReLU
Hidden Layer Calculation Module
136
REG (16*4)
64
MUL
SIGMA
ReLU
Hidden Layer Calculation Module
140
REG (16*4)
64
MUL
SIGMA
ReLU
Hidden Layer Calculation Module
140
REG (16*4)
64
MUL
SIGMA
Output Layer Calculation Module
16
Output

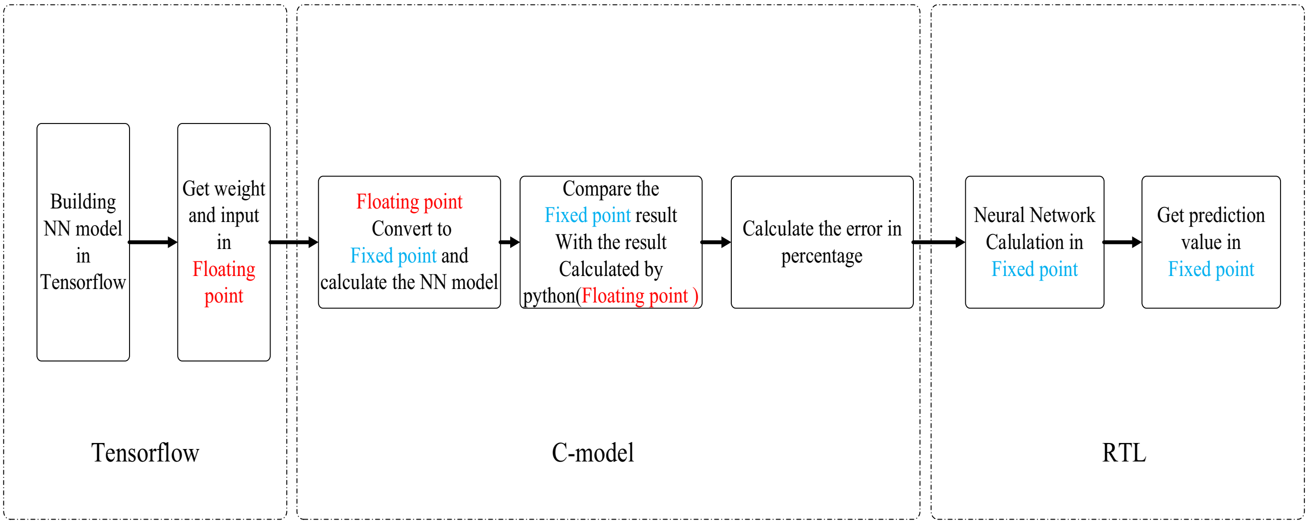

Building NN model in Tensorflow
Get weight and input in Floating point
Floating point Convert to Fixed point and calculate the NN model
Compare the Fixed point result With the result Calculated by python(Floating point)
Calculate the error in percentage
Neural Network Calulation in Fixed point
Get prediction value in Fixed point
Tensorflow
C-model
RTL

Fig. 12 The design and testing flow of the NN model implemented on the FPGA board

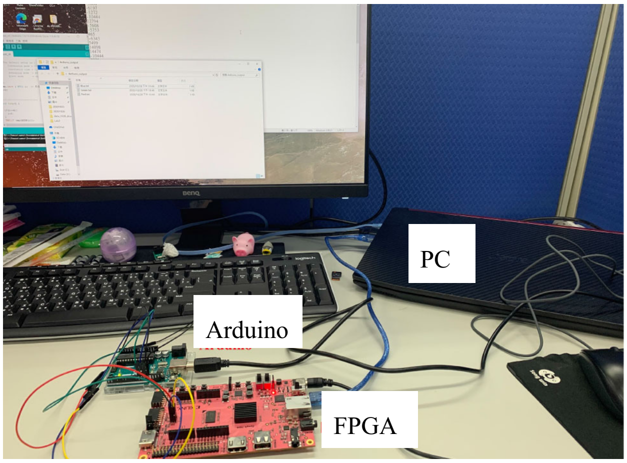


Fig. 13 Experimental setup for performance validation

16 show the prediction errors of degradation for red, green and blue OLEDs in 2 different scales, grey levels and nits, respectively. The abscissa represents the numbered combinations of different (t, I, T) as inputs to the neural network in Fig. 1, while the ordinate represents the error between the calculated degraded OLED luminance $L_{\text{predicted}}(t)$ based on the accelerator implemented in the FPGA accelerator and the Python-equivalent code via Tensorflow in percentages. It can be clearly seen from Figs. 15, 16 that the resulted errors are well within 0.488 grey and 0.013 nits, showing the effectiveness of the Verilog algorithm implemented in the FPGA board. The errors are also evaluated from the perspective of mean absolute errors (MAE), the result of which is shown in Table 3. It can be found from this table that the resulted errors for the degradation of the red OLED by FPGA are more accurate than green and blue OLEDs in MAPE. Of most importance are the errors by the FPGA accelerator as small as 2.08%, 5.51% and 4.36% for red, green and blue OLEDs, respectively. On the other hand, the OLED degradation models for green and blue result in very small errors, while the fixed-point truncation operation renders larger errors. Table 4 presents the performance achieved by the proposed FPGA hardware architecture implemented in the FPGA Artix-7 xc7a200tfbg676-2 board. The maximum frequency reaches 55.6 MHz, while the total number of LUTs is 1035.

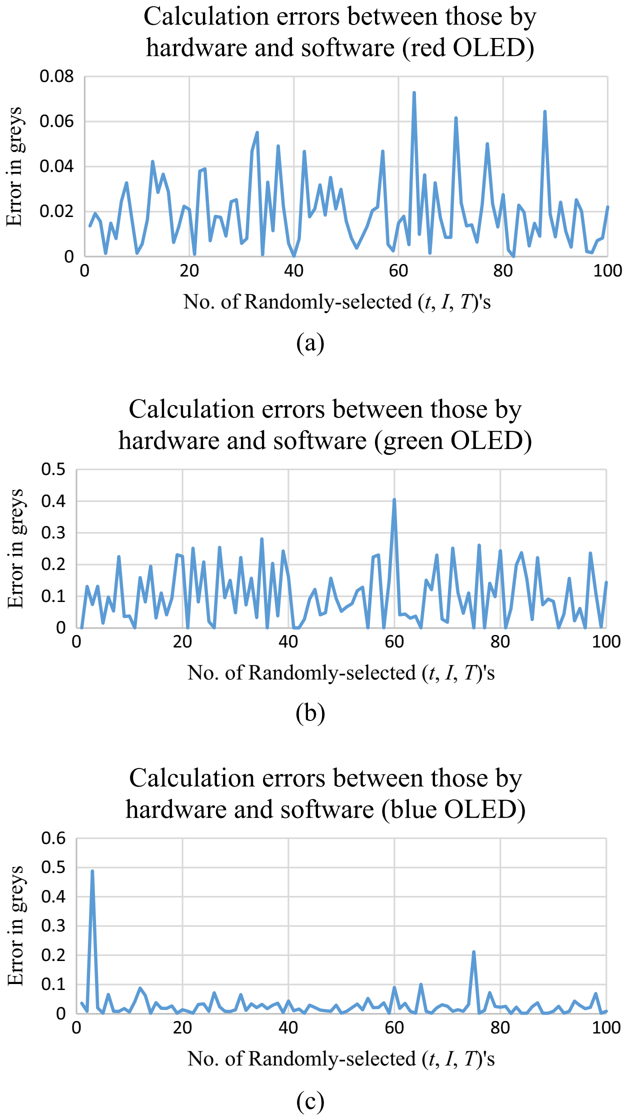
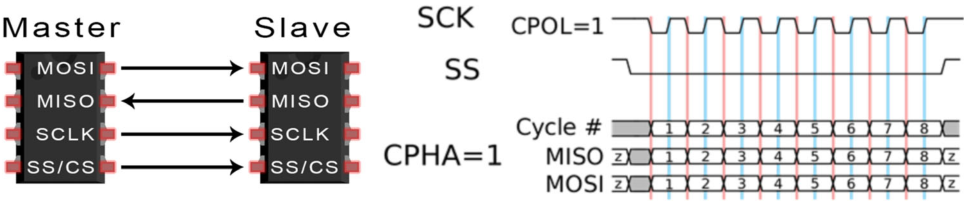


Fig. 15 Error in greys for predicting OLED degradation between those by FPGA (hardware) and fixed-point model by software (tensor flow). **a** Red; **b** green; **c** blue OLEDs (color figure online)

Table 5 shows the comparison among the architectures proposed by this effort and those in other past works (Oliveira et al. 2017; Medus et al. 2019; Zhai et al. 2016; Nedjah et al. 2012). Since the architectures of the implemented NN model for comparison are different, the calculation time per neuron and area consumption per neuron

Fig. 14 The protocol of serial peripheral interface (SPI)



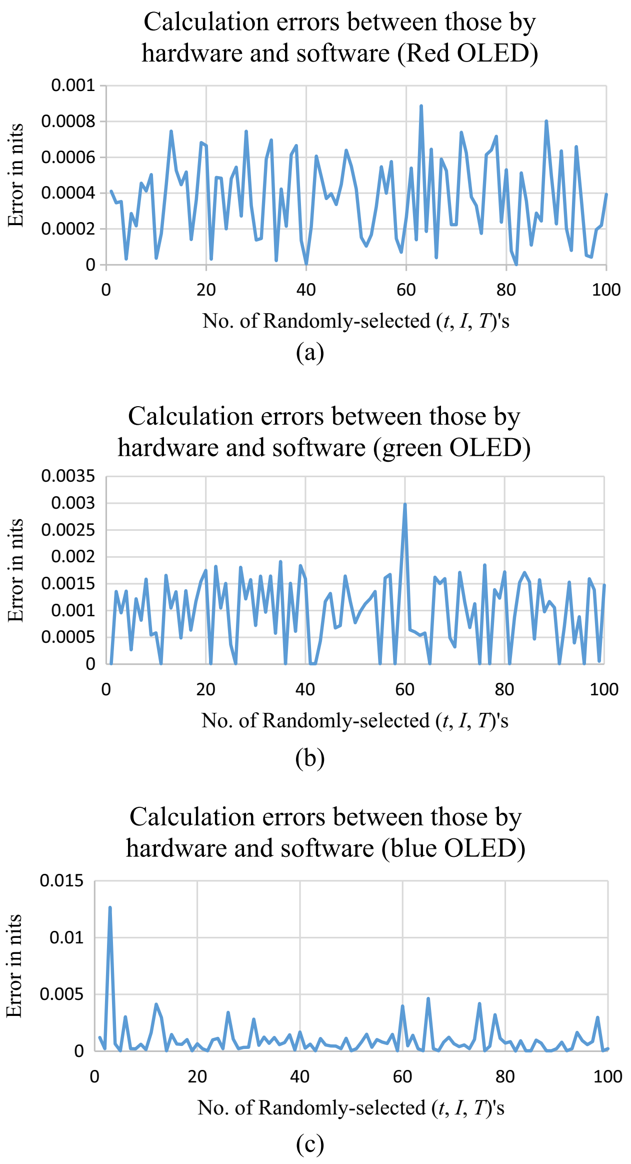


Fig. 16 Error in nits for predicting OLED degradation between those by FPGA (hardware) and fixed-point model by software (tensor flow). **a** Red; **b** green; **c** blue OLEDs (color figure online)

are considered as performance indices for evaluation. Note that there is always a tradeoff between computing time and the consumption of resources. Hence, the architecture with the smallest product of computation time and area can be considered for identifying the highest performance, therefore, serving as the figure of merit (FOM). It is clearly seen from Table 5 that the FOM of the product of computation tine and area achieved by the present work is as low as 109.86 (Time*Area), the lowest compared to all the past reported works; apparently, the proposed hardware architecture leads to the best performance.

Table 4 Achieved performance of the FPGA accelerator implemented

Device	f_{max} (MHz)	LUTs	Registers	DSP Blocks	MLP
Artix-7	55.6	1035	1178	48	3-4-4-4-1
Available		133,800	267,600	740	–
Utilization (%)		0.77%	0.44%	6.49%	

5 Conclusion

A machine learning model in the structure of neural network (NN) is established herein to predict well OLED degradation for compensation, with the currents and temperatures of OLEDs on each pixel sensed as references. To realize the NN, a new hardware architecture via FPGA is proposed and implemented successfully. This FPGA architecture can conduct a vast amount of calculations in a short time, with moderate consumption of hardware resources. With this architecture, the calculation by the NN can be executed efficiently based the built-in OLED degradation prediction models established. The proposed hardware architecture has been implemented successfully and verified on Xilinx's Pynq-z2 and Xilinx's Artix-7. In these FPGA implementations, the operating frequency in Artix-7 is 55.6 MHz, with data calculation time per neuron as 0.077 ns and LUTs consumption per neuron as 79.61. The errors of degradation prediction by the accelerator are as small as 2.08%, 5.51% and 4.36% for red, green and blue OLEDs, respectively, while the figure of merit, defined as the product of computation time and area, is as low as 109.86 (Time*Area), the lowest compared to all the all past reported works.

Table 3 The errors by the implemented hardware accelerators by FPGA

Degradation prediction model	Red OLED	Green OLED	Blue OLED
Mean absolute percentage error with software prediction	0.069%	0.292%	0.299%
Mean absolute error with software prediction (greys)	0.019403	0.10426	0.029277
Mean absolute error with software prediction (nits)	0.000372	0.001002	0.001010
Mean absolute percentage error with experimental data	2.083%	5.517%	4.363%

Table 5 Achieved specifications in comparison with other works

	Device	f_{max} (MHz)	Clock cycle (per neuron)	Calculation time per neuron(ns)	Area (per neuron)	DSP Blocks (per neuron)	MLP	Calculation time * area
Zhai et al. (2016)	Zynq-7	–	–	13.8	1008 LUT	7	12-3-1	13910.4
Nedjah et al. (2012)	Virtex-5	20.3	10.47	516.59	109 Slice	–	220-24-10	56038.31
Oliveira et al. (2017)	Cyclone IV	77.8	6.43	82.62	3035 Logic element	–	4-8-3-3	–
Medus et al. (2019)	Virtex-7	490.8	3	6.12	52.4 LUT	0.92	4-8-3-3	320.69
Proposed	Pynq-z2	50	0.077	2	79.61 LUT	3.69	3-4-4-4-1	159.22
Proposed	Artix-7	55.6	0.077	1.38	79.61 LUT	3.69	3-4-4-4-1	109.86

Acknowledgements This study is supported by Ministry of Science and Technology, Taiwan grant nos. MOST 111-2223-E-A49 -005-, 111-2221-E-A49-159-MY3 and 110-2223-E-A49 -001-. This work was also financially supported by the “Center for Intelligent Drug Systems and Smart Bio-devices (IDS2B)” from The Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE) in Taiwan. This research was also supported by the Hsinchu and Southern Taiwan Science Park Bureaus, Ministry of Science and Technology, Taiwan, R.O.C. under contracts 108A31B, 110CE-2-02 and 112A028B.

Data availability The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

References

Aoyama T, Wang Q, Suematsu R, Shimizu R, Nagashima U (2002) Learning algorithms for a neural network in FPGA. In: Proceedings of the 2002 international joint conference on neural networks. IJCNN’02 (Cat. No. 02CH37290), vol 1. IEEE, pp 1007–1012

Hao Y (2017) A general neural network hardware architecture on FPGA. arXiv preprint arXiv:1711.05860

Liu H et al (2017) High-power LED photoelectrothermal analysis based on backpropagation artificial neural networks. IEEE Trans Electron Devices 64(7):2867–2873

Liu H et al (2019) Lifetime prediction of a multi-chip high-power LED light source based on artificial neural networks. Results Phys 12:361–367

Lu K, Zhang W, Sun B (2017) Multidimensional data-driven life prediction method for white LEDs based on BP-NN and improved-adaboost algorithm. IEEE Access 5:21660–21668

Medus LD, Iakymchuk T, Frances-Villora JV, Bataller-Mompeán M, Rosado-Muñoz A (2019) A novel systolic parallel hardware architecture for the FPGA acceleration of feedforward neural networks. IEEE Access 7:76084–76103

Nedjah N, da Silva RM, de Macedo Mourelle L (2012) Compact yet efficient hardware implementation of artificial neural networks with customized topology. Expert Syst Appl 39(10):9191–9206

Oliveira JG, Moreno RL, de Oliveira Dutra O, Pimenta TC (2017) Implementation of a reconfigurable neural network in FPGA. In: 2017 International Caribbean conference on devices, circuits and systems (ICCDCS). IEEE, pp 41–44

Pearson MJ et al (2007) Implementing spiking neural networks for real-time signal-processing and control applications: a model-validated FPGA approach. IEEE Trans Neural Netw 18(5):1472–1487

Zhai X, Ali AAS, Amira A, Bensaali F (2016) MLP neural network based gas classification system on Zynq SoC. IEEE Access 4:8138–8146

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.