Journal Home Page: https://sites.google.com/a/univsul.edu.iq/sjes

Review Article:

Dc Motor Identification Based on Artificial Neural Networks Implementation in a Microcontroller

Shivan Shkur Mahmood[1]
Alaa M. Abdulrahman[1]

[1]University of Sulaimani, College of Engineering, Electrical Department

Article Inform

Article History:
Received 1 August 2019
Accepted 2 December 2019
Available online 1 April 2020

Keywords: Microcontroller, PWM, ANN, Identification, DC motor.

About the Authors:
Corresponding author:
Shivan Shkur Mahmood - MSc.
E-mail: shivan.sh.mahmood@gmail.com
Researcher Involved:
Dr. Alaa M. Abdulrahman - Lecturer

DOI Link: https://doi.org/10.17656/sjes.10124

Abstract

In the last few decades neural networks have been involved in many fields of researches and their application was restricted to be embedded on a computer. This paper describes implementing of artificial neural network (ANN) based on a low cost microcontroller to identify a DC motor model. The identifier will act as path for the output error feedback in Model Reference Adaptive Controller or to build an inverse controller. To reduce the size of the proposed system, the DC Motor is controlled conventionally using the method of Pulse Width Modulation technique on the same microcontroller. The proposed neural network is a multilayer feed-forward network which is of three layers (1-3-1) structure. The training technique used the back propagation algorithm by using C program. Identification process has been implemented successfully in the microcontroller and validated with different randomly picked data with least mean square error of 0.0007.

1. Introduction

DC motors have been commonly used in several industrial applications such as electric vehicles control, electric cranes, and robotic manipulators etc. [1]. It is essential to manage the speed of DC motor can give a high starting torque and most probably obtain speed control over varied range. With the development of technology; microcontrollers are becoming more suitable chips to control different electro-mechanical devices [2]. Over the last years, research about Artificial Neural Network (ANN) has grown in many fields. Until now most of the work are done in this field are based on software simulations. Ability of ANN to adaptive and learn any ambiguous system [3].

But to get the advantage of ANN, the hardware implementations are important and essential [4].ANNs, like human brain, learns by example. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons[3].

One researcher stated, "most embedded microprocessor cores lack the performance for running neural networks"[5]. Many people have repulsion about implementing neural networks, one being that it requires significant computing power and time to train them[5,6]. Some researchers have implemented neural networks using Field Programmable Gate Arrays (FPGAs) and Digital signal processors (DSPs) [4,7,8,9].

However, FPGA and DSP requires large computing power, therefore, they tend to be very expensive[6]. However, implementing neural network does not required this large amount of power and it can be done on an economic microcontroller[10,11].

In this paper, implementations of a multilayer feed forward network and the activation function of the hidden layer are sigmoid function with back propagation training algorithm for identify DC motor speed control system using AT89S52 microcontroller. In the sections that follow, speed control of DC motor using microcontroller (plant) with describes the hardware components, neural network algorithm for training to identify speed of DC motor, ANN implementation in microcontroller and experimental results will be presented.

2. DC Motor control Using Microcontroller

Dc motor is controlled using PWM (Pulse width Modulation) technique in microcontroller. PWM is a technique for binary signals generation, which has two signal periods (high and low). The width of each pulse varies between 0 and the period (T) [12].

The main principle is control of power by varying the duty cycle; a duty cycle is the fraction of one period in which a signal is active. Duty cycle is commonly expressed as a percentage or a ratio. This is defined as the ratio between the ON time and the period of the waveform as shown in equation below[12]

$$\text{Duty Cycle ratio} = \text{On time/ Time period} \qquad (1)$$

2.1. System Configuration

The microcontroller is an AT89S52 comes from the popular 8051 family Microcontrollers. It is an 8-bit microcontroller with four I/O ports, and 256 Bytes On-chip RAM [13].

The C compiler programs are up loaded to the microcontroller through USP programmer. To display the output speed and the input values. It is necessary to attach an LCD to the microcontroller. In order to enter input data to the microcontroller a numeric keypad is used. The rotational speed of the DC motor can be captured utilizing an Infrared Receiver (IR) sensor.

The sensor module works on the principle of reflection of Infrared Rays from the incident surface[14]. The IR led rays emits the continuous beam of the IR rays. An obstacle has been mounted on fan plate in order to cut the beams and count them as shown in Figure 1.

The motor which need to be identified is gearbox DC motor type with 12 V and 250 r.p.m rated values. However the microcontroller board is not enough to drive motors directly so we used some kind of drivers. Motor drivers act as current amplifiers since they take a Low current control signal and provide a higher current signal to drive the motors.

2.2. Operation and Result

DC motor converts electrical energy in the form of Direct Current into mechanical energy. The mechanical energy produced is in the form of rotational movement of the motor shaft. The motor can be rotated at a certain speed by applying a specific voltage to it and it can be varied by changing the applied voltage.

The average voltage applied to the DC Motor will depend on what is called as the Duty Cycle of the PWM Signal as represent in equation(2) :

$$V_{av} = \frac{t_{on}}{T} \times V_{in} \quad (2)$$

Where V_{av}: average voltage supply to DC motor (practically measured as V motor in Table 1)

V_{in} : DC Supply voltage (12V)

t_{on} : Time ON of pulse (ie.Duty Cycle), T = period of PWM signal.

A Variable called V-pwm value has been used which have a relation with duty cycle represented in equation below.

$$Duty\ cycle = 0.3032(\mathrm{V-pwm}) + 10.712 \quad (3)$$

The figure 2 represents different PWM Signals of a 12V supply with different duty cycles.

Figure 3 shows the complete circuit diagram has been drawn using Proteus software to control speed of dc motor.

The microcontroller creates pulses of varying width for pulse width modulation to controls the motor speed. To change the speed of the motor, whenever entered the value of V-pwm by using keypad, an interrupt is generated and the duty cycle of PWM signal varies.Timer-1 of the microcontroller has been used for generating PWM pulses, which is clocked using 11.0592MHz crystal oscillator.

The base frequency is kept constant at 3.9 kHz and the duty cycle of this wave is varied to change the analogue level at output pin P1.7 of the microcontroller.

The whole system is implemented in the workshop and Figure 4 shows schematic circuit with motor and all component combination.

The speed of Dc motor measured using IR sensor. But in LCD has been used to display both the speed of motor in RPM and V-pwm value. The need for enough data to train neural network 42 reading has been recorded in experiment. Few data of the records has been tabulated in Table (1).

The results have been indicated that the relationship of V-pwm value to the motor speed in r.p.m showed that V-pwm value is nonlinearly proportional to the DC motor speed as shows in Figure 5. Hence, the increasing of the V-pwm value, the duty cycle increases due to the increment of Time ON of pulse (t_{on}) which lead to increase , the motor voltage (terminal voltage) so that the motor will rotate faster.

In the figure (5), shows a zero speed at V-pwm value equal to zero. This is due to the DC motor used has a gearbox and the minimum voltage measured at this point is 0.524V and at this voltage the DC motor cannot be run.

3. Neural Network Implementation

In this section, neural networks have been implemented in AT89S52 Microcontroller for identification the DC motor system. Below are details of this implementation:

3.1. Artificial Neural Network Algorithm

The proposed three layer feed forward neural network used for identification is shown in Figure 6.The ANN is first trained in C program by Back-Propagation (BP) Algorithm using all the reading data from Table (1). Figure 7 shows the Representation of identification model where the input to ANN is V-pwm value and the output is the speed of dc motor (r.p.m).

Later, the difference between the output speed of the DC motor (target) and the output of ANN which is represented as error, then this error is fed-back through ANN to update all the weights of the connections.

The BP algorithm is a supervised learning method for training multilayer artificial neural networks[15]. At the beginning, the initial weights are chosen randomly and then the learning or training begins.

The neural network training algorithm can be decomposed in two steps applied on figure 6:

1) **Feed Forward:** The input (V-pwm value) feed forward though the neural network. Here's how we calculate the net input for each hidden neuron:

$$net_{hi} = \sum_{i=0}^{2} input * w1[i] + b * wb[i] \qquad (4)$$

Where: net_{hi} -Summation of the dot product of inputs with the corresponding weights to the hidden neuron (hi), b=1, wb is the Bias weight, and $w1$ is the input- hidden layer weight. Then use the sigmoid function to get the output of each hidden neuron (out_{hi}):

$$out_{hi} = \frac{1}{1+e^{-net_{hi}}} \qquad (5)$$

Then the calculation for the output layer neuron, using the output from the hidden layer neurons as inputs. Here's the output of neural network formula represents by symbol (out):

$$out = \sum_{i=0}^{2} out_{hi} * w2[i] + b * wb[3] \qquad (6)$$

Where: $w2$ is the hidden – output layer weight. Then, calculating the sum squared errors of the network outputs (E):

$$E = \sum \frac{1}{2}(target - out)^2 \qquad (7)$$

$target$ - Speed of the DC motor

2) **Back propagation:** Our goal with back propagation is to update each of the weights in the network so that they cause the actual output to be closer to the target, thereby minimizing the total error E.

$$\frac{\partial E}{\partial Out} = -(target - out) \qquad (8)$$

To update $w1[i]$ back propagation the error to hidden layer:

$$\frac{\partial E}{\partial w_{1[i]}} = \frac{\partial E}{\partial out_{hi}} * \frac{\partial out_{hi}}{\partial net_{hi}} * \frac{\partial net_{hi}}{\partial w_1[i]} \qquad (9)$$

$$\frac{\partial E}{\partial w_{1[i]}} = -(target - out) * w2[i] * out_{h1}(1 - out_{h1}) * Input \qquad (10)$$

$$w1\,[i]^{+} = w_1[i] - \eta * \frac{\partial E}{\partial w_{1[i]}} \qquad (11)$$

Where : $w1[i]^{+}$, $w2\,[i]^{+}$ - new update for w_1 and w_2 , Eta (η) - learning rate.

Then sub equation (10) into (11) to get new update for w1[i]:

$$w1\,[i]^{+} = w_1[i] + \eta * (target - out) * w2[i] * out_{h1}(1 - out_{h1}) * Input \qquad (12)$$

And to update the W2 [i] back propagation the error to the output layer:

$$\frac{\partial E}{\partial w2[i]} = \frac{\partial E}{\partial out} * \frac{\partial out}{\partial w2[i]} \qquad (13)$$

$$\frac{\partial E}{\partial w2[i]} = -(target - out) * out_{hi} \qquad (14)$$

$$w2\,[i]^{+} = w_2[i] - \eta * \frac{\partial E}{\partial w_{2[i]}} \qquad (15)$$

Equation (14) sub into (15) to get new update of $w_2[i]$

$$w2\,[i]^{+} = w_2[i] + \eta * (target - out) * out_{hi} \quad (16)$$

And repeated the process to update the weights of Bias $wb[i]$and $wb[3]$:

$$wb\,[i]^{+} = w_b[i] - \eta * \frac{\partial E}{\partial w_{b[i]}} \quad (17)$$

$$wb\,[i]^{+} = w_b[i] + \eta * (target - out) * \mathrm{w2[i]} * out_{h1}(1 - out_{h1}) * b \quad (18)$$

$$wb\,[3]^{+} = w_b[3] - \eta * \frac{\partial E}{\partial w_{b[3]}} \quad (19)$$

$$wb\,[3]^{+} = w_b[3] + \eta * (target - out) \quad (20)$$

Where : $wb[i]^{+}$, $wb\,[3]^{+}$ - new update for the weight bias.

Thus, the error at the output is back propagated through each layer in order to adjust each layer weights. First iteration of training of the output is observed in figure (8.a), where figure (8) represent responses of speed and also their corresponding identifier output with respect to V-pwm values. Weights update was continued for (100000 iteration) till convergence of total squared error E close to zero is satisfied as shown in figure (8.b) the small difference between the actual output and output of ANN can be neglected. Weights updating progress is represented in figure (9) for randomly selected weights update to their steady state values.

3.2. Microcontroller implementation

The main limitations of the implementation of neural network in AT89S52 microcontrollers are the memory size and the computing speed. However, they are inexpensive and their low power consumption features makes them ideal for any related applications. Furthermore, the average cost for single chip for FPGA is 50$ and it cost with development board is 300$, while the average cost for single chip for AT89S52 microcontroller is 3$ and it cost with development board is 19$.

The issue of the limitations has been solved by using the Feed forward part of the ANN with its trained weights. Hence, both computational time and storage space are reduced to improve the ANN implementation in the microcontroller for the purpose of identification the DC motor. The microcontroller has been programmed using Keil C compiler. Table (2) shows the experiment results.

It can be noticed from the results, in the last trial when the V-pwm value was 197 the least mean square error is more than 0.0007 which makes the difference between target speed and ANN outputs equals to 3.9 r.p.m. This difference in speed is due to the heat produced in the motor because of running it for a long time.

4. Conclusion

In conclusion, neural networks identification for the DC motor speed control system was proposed. Using ANN, the system is learning from the repeated historical results so there was no need to know neither the mathematical model nor the motor parameters of the motor when designing the identification system.

The back propagation technique was used to provide adaptive estimation of the motor speed and the training has been completed with a least mean squared error of 0.0007, so the difference between the actual output and the identifier output can be neglected. Due to restriction of AT89S52 microcontroller on code size and to reduce the code size of neural network identifier, it has been proposed to use only feed forward algorithm part.

The successful identification of DC motor will help to build an inverse identification (NN input will be speed of the DC motor, and its output will be the V-pwm value), hence the inverse identifier can be used as a controller.

References

1- Moleykutty George, "Speed Control of Separately Excited DC Motor", Proceedings of the American Journal of Applied Sciences , Faculty of Engineering and Technology, Multimedia University, Malaysia ,Vol.55, March 2008, pp.227-233.

2- M. H. Bhuyan, M. A. Rabby and M. M. G. Tarik, "Microcontroller based Automatic Traffic Light Control System Design," Proceedings of the National Conference on Electronics and Telecommunications for Digital Bangladesh, Bangladesh Electronics Society, Dhaka, June 2010, pp. 139-142.

3- Simon Haykin, "Neural Networks and Learning Machines", Copyright by Pearson Education, Inc. Printed in the United States of America, ,3rd Edition, Pearson, McMaster University, Canada 2009.

4- Dhirajkumar S. Jinde, Samrat S. Thorat, "Neural Network Implementation Using FPGAs" , International Journal of Computer Science and Information Technologies, Vol. 5 (3) , 2014, pp.3431-3433.

5- H. Esmaeilzadeh, P. Saeedi, B. N. Araabi, C. Lucas, and S. M. Fakhraie, "Neural network stream processing core (NnSP) for embedded systems," in Circuits and Systems. Proceedings.IEEE International Symposium on, 2006, p. 21–24.

6- Nicholas Jay Cotton, "A Neural Network Implementation on Embedded Systems", dissertation submitted to the Graduate Faculty of Auburn University in partial fulfillment of the requirements for the Degree of Doctor of Philosophy, August 9, 2010.

7- S. Himavathi, D. Anitha, and A. Muthuramalingam, "Feedforward Neural Network Implementation in FPGA Using Layer Multiplexing for Effective Resource Utilization," IEEE Transactions on Neural Networks, vol. 18, pp. 880-888, 2007.

8- D. Zhang and H. Li, "A Stochastic-Based FPGA Controller for an Induction Motor Drive with Integrated Neural Network Algorithms," IEEE Transactions on Industrial Electronics, Vol.55, 2008, pp. 551-561.

9- M. Mohamadian, E. P. Nowicki, A. Chu, F. Ashrafzadeh, and J. C. Salmon, "DSP implementation of an artificial neural network for induction motor control", IEEE Canadian Conference on Electrical and Computer Engineering , Engineering Innovation: Voyage of Discovery, 1997, pp. 435-437.

10- Mohamed Nasreldin M.A., Abdalaziz Abdalgfar A.A., Ahmed Alfatih A., Ahmed Abdelrahman A.A., "Implementation of a PID Control System on Microcontroller (DC Motor Case Study)", IEEE International Conference on Communication, Control, Computing and Electronics Engineering, Khartoum, Sudan, 2017.

11- Ben Joan, September2009,"Difference between FPGA and microcontroller", on-line, DifferenceBetween.net, http://www.differencebetween.net/technology/difference-between-fpga-andmicrocontroller/.

12- Barr Michael, September 2001, "Introduction to Pulse Width Modulation (PWM)", on-line, BARR group, pp.103-104. https://barrgroup.com/Embedded-Systems/How-To/PWM-Pulse-Width-Modulation.

13- Muhammad Ali Mazidi, Janice G. Mazidi and R.D. McKinlay, "The 8051 Microcontroller and Embedded Systems: Using Assembly and C", 2nd edition, PH publishing, 2006.

14- G. A. P. Raka Agung, S. Huda, I W. Arta Wijaya, "Speed Control for DC Motor with Pulse Width Modulation (PWM) Method Using Infrared Remote Control Based on ATmega16 Microcontroller", IEEE International Conference on Smart Green Technology in Electrical and Information Systems (ICSGTEIS), 2014,pp.109-111.

15- Toshinri Munakatat, "Fundamentals of the New Artificial Intelligence", 2nd edition, Springer-Verlog, London, 2008.

تمثيل نظام لمحرك التيار المستمر اعتمادا على تطبيق الشبكات العصبية الصناعية في وحدة التحكم الدقيق

شيڤان شكر محمود[1] - طالبة ماجستير

د . علاء محي الدين عبدالرحمن[1] - مدرس

[1] جامعة السليمانية ، كلية الهندسة ، قسم الهندسة الكهربائية

المستخلص

في السنوات الاخيرة تم استخدام الشيكات العصبية في عدة مجالات من الابحاث والتطبيقات المحددة المبنية على الحاسبات الالية. هذا البحث يوضح تطبيق الشبكات العصبية الصناعية في وحدة التحكم الدقيقة ذو التكلفة المنخفضة لتمثيل نظام مماثل لمحرك التيار المستمر . لتقليل حجم المنظومة المقترحة ، يتم التحكم بهذا المحرك بشكل تقليدي باستحدام طرىقة تعديل عرض النبض على نفس وحدة التحكم الدقيقة. الشبكات العصبية المقترحة عبارة عن شبكة ارتجاغ المسبق المتعدد الطبقة التي تتألف من ثلاثة (1 -3 -1) طبقات. تقنية التدريب المستخدمة هي طريقة الانتشار الخلفي والشبكة المتدربة هي من النوغ غير المتصل وبواسطة استعمال برامج سي. عملية النظام المتماثل طبقت بنجاح في وحدة التحكم الدقيقة وتم التحقيق باستخدام قيم عشوائية مختلفة وباقل قيمة خطأ المتوسط التربيعي 0.0007 .

الكلمات المفتاحية : وحدة التحكم الدقيقة ، تعديل عرض النبضة ، شبكات العصبية الصناعية ، التمثيل ، محرك التيار المستمر .

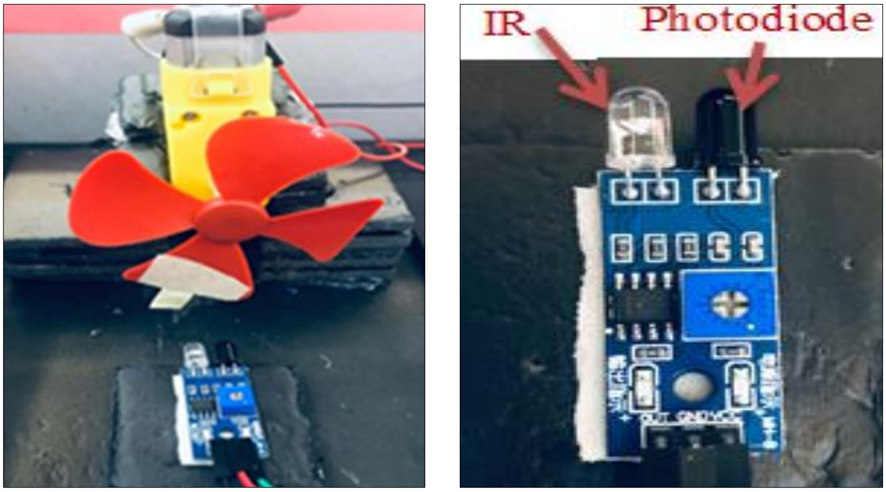


Fig. 1: Schematic diagram of IR sensor circuit and DC motor (Source: Researcher)

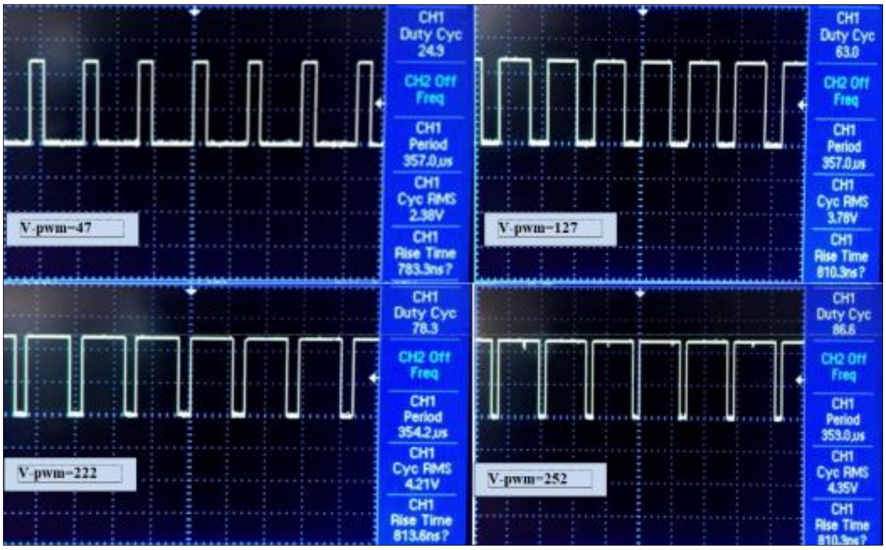


Fig. 2: Practical results of PWM waveform (Source: Researcher)

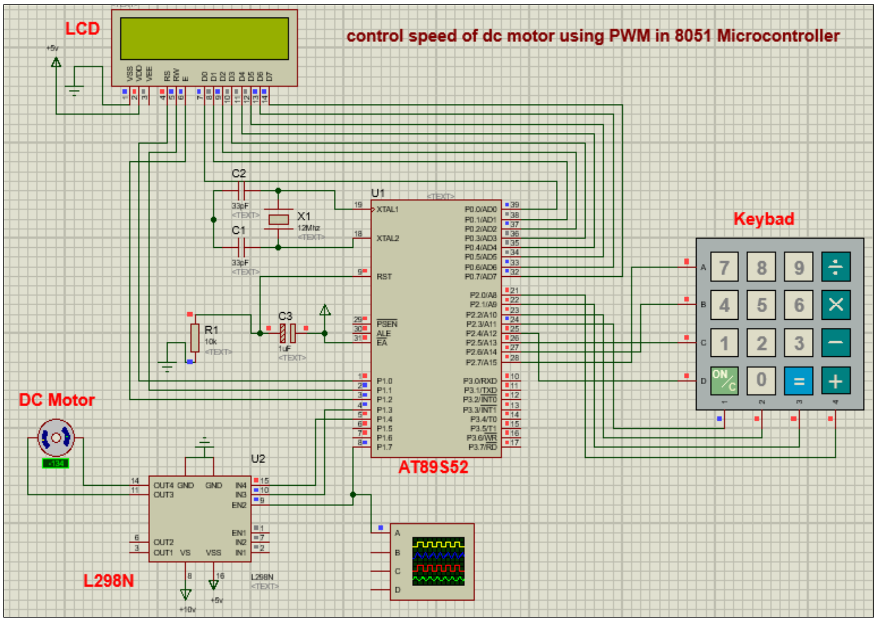


Fig. 3: Circuit diagram of microcontroller based DC motor speed controller (Source: Researcher).

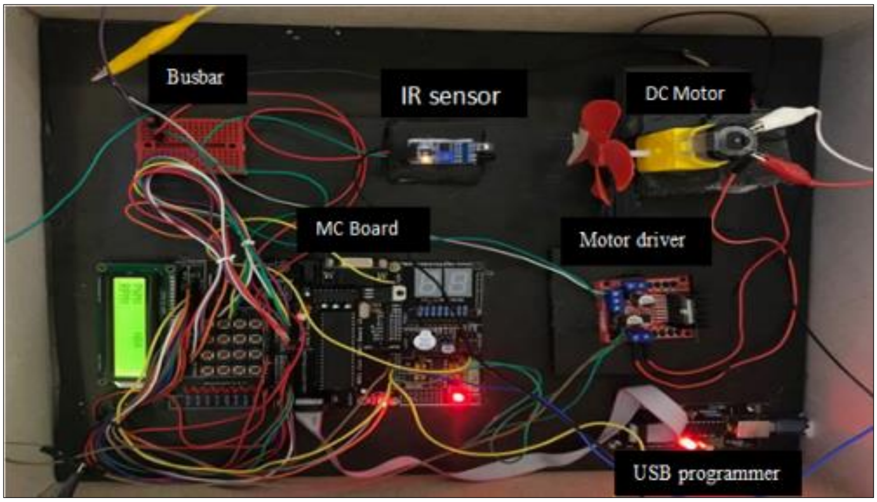


Fig. 4: Hardware setup (Source: Researcher)

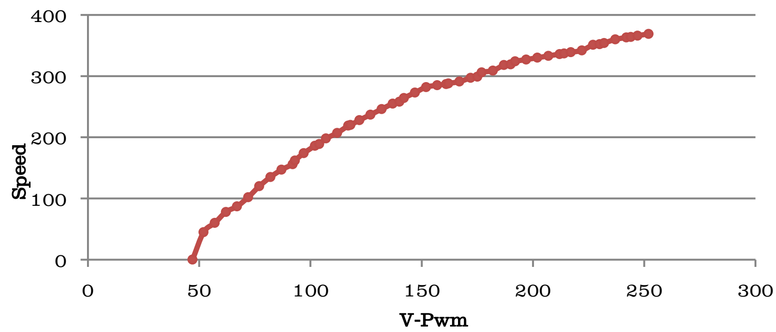


Fig. 5: Practical result of Speed vs. V-pwm value (Source: Researcher)

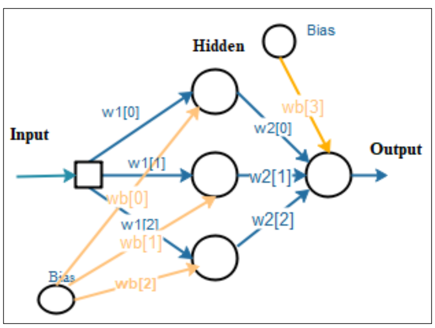


Fig. 6:1-3-1 multi-layer neural network structure (Source: Researcher)

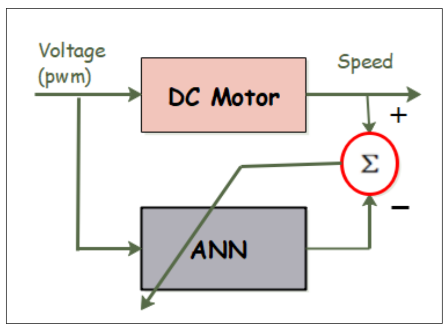
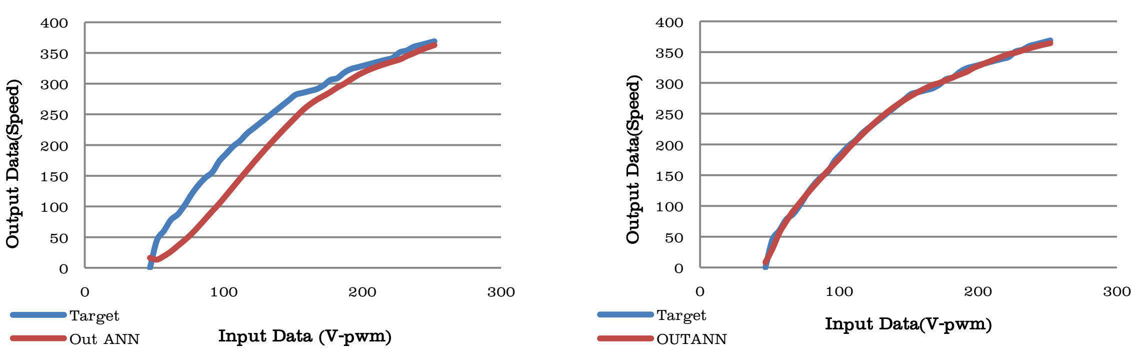


Fig. 7: Identification of Dc motor structure (Source: Researcher)



(a) At first iteration

(b) after 100000 iteration

Fig. 8 : Output speed of dc motor and ANN (identified) vs. V-pwm (Source: Researcher)

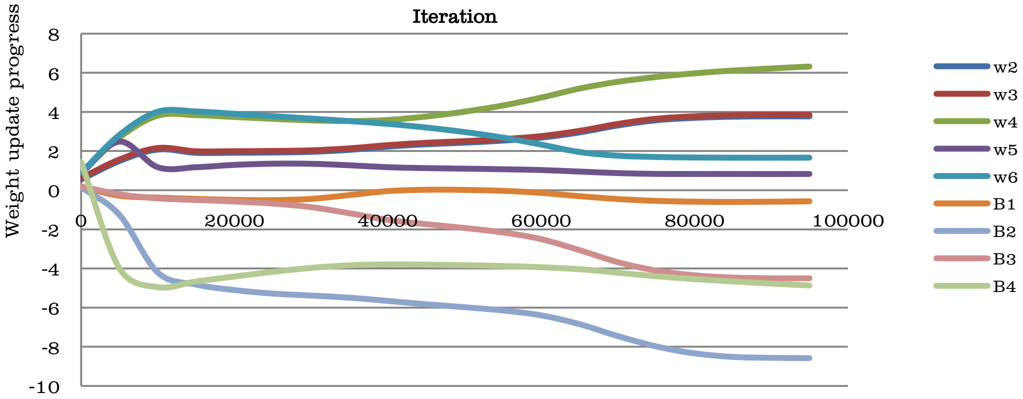


Fig. 9: Weights update progress vs. Iteration (Source: Researcher)

Table 1: Practical result of V-pwm value, duty cycle, motor Voltage and motor speed (Source: Researcher)

V-pwm	Duty cycle %	V Motor [volt]	Speed [r.p.m]
62	29.4	3.25	78
92	38.7	4.31	156
127	49.3	5.84	237
172	63	7.24	297
217	76.5	8.86	339
252	86.1	9.98	369

Table (2: Practical results of implemented ANN in the microcontroller (Source: Researcher)

V-pwm	Target Output (r.p.m)	ANN Output (r.p.m)
87	147	147.2
147	270	270.8
212	336	336.8
94	171	171.6
132	246	246.6
247	366	365.8
197	327	323.1