

ROBUST ANN CONTROL SYSTEM DESIGN AND BUILDING FOR AN AUTONOMOUS MOBILE ROBOT BASED ON FPAA⁺

Thair A. Salih *

Omar Ibrahim Yehya **

Abstract

It is difficult to find an autonomous robot capable of performing certain services in a real environment. One advantage of a robot is its ability to accomplish many of the tasks entrusted to him with high efficiency compared to that of humans. This paper aims to design and implement a robot system which has the ability to move along a predefined path and move with discretion through a place in so many forms. A field programmable analog array (FPAA) technology has been used in building the artificial neural network (ANN) controller system on the route of the robot which minimizes the tracking error to the goal point, while simultaneously maximizing the distance away from obstacles. The proposed robot system was tested, which showed the possibility of movement and it completed the required tasks with high accuracy according to the path planned.

Keywords- Robot, ANN, Controller, FPAA.

تصميم وبناء منظومة سيطرة موثوقة على إنسان آلي نقال مستقل ذاتيا باستخدام الشبكات العصبية
الصناعية استنادا الى FPAA

عمر ابراهيم يحيى

ثائر علي صالح

المستخلص:

من الصعب إيجاد إنسان آلي قادر على انجاز خدمات معينة في بيئة حقيقية. إن إحدى مزايا الإنسان الآلي هو قابليته على انجاز عدة واجبات منطوية به بكفاءة عالية مقارنة ذلك مع الإنسان. يهدف البحث إلى تصميم وتطبيق منظومة إنسان آلي ذات مقدرة على السير في مسار معرف مسبقا والتحرك بشكل عقلائي خلال أماكن وبعدة أشكال. استخدمت تقنيات المصفوفات التشابعية القابلة للبرمجة (FPAA) لبناء منظومة سيطرة باستخدام الخلايا العصبية الاصطناعية (ANN) وذلك لتسيير الإنسان الآلي. أظهرت نتائج الاختبار على المنظومة المقترحة إمكانية السير وفق المخطط المحدد مسبقا وتنفيذ المهام المطلوبة بكل دقة.

Introduction:

In recent years, wheel-based mobile robots have attracted considerable attentions in various industrial and service applications. These applications require mobile robots to have

⁺Received on 12/9/2011 , Accepted on 21/11/2012

^{*}Lecher / Technical College/ Mosul

^{**} Technical College/ Mosul

the ability to track specified path stably [1]. With the steady advancement of computers and software development tools, modern control techniques such as optimal control are becoming much easier to develop, simulate and implement. As a result, greater opportunity exists to exploit these advanced algorithms to tackle control problems that in the past have been approached using less sophisticated classical control techniques [2]. They are playing a more and more important role in industrial and service robotics especially when their motions are restricted on reasonably smooth grounds and surfaces in the absence of workspace obstacles. Due to the perfect rolling constraints where no longitudinal or lateral slipping of the wheels is presumed, most of the wheeled mobile robots (WMRs) are a typical example of a nonholonomic system. This has aroused enthusiasm among researchers in view of its theoretical challenges [3].

Field Programmable Analogue Arrays are based on configurable analogue blocks (CAB) consisting of operational amplifiers, passive elements and a programmable connecting network. The tolerances of the tested time discrete system, using switched capacitor technique to implement the desired analogue functions, only depend on the ratios of capacitors. So, tolerances down to 0.1% are achieved easily. CABs can be combined to configurable analogue modules (CAMs). CAMs are highly qualified circuits such as filters, integrators amplifiers etc. The configuration of the CAMs is provided in a library by the producer. Fig. 1 shows the block circuit of the FPAA, the Anadigm AN221E04 [4].

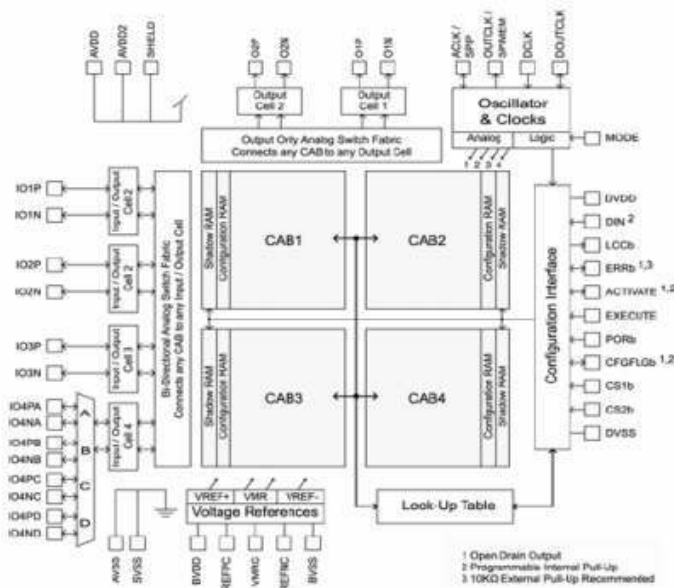


Fig. 1 Block circuit of an AN221E04

Artificial neural networks (ANNs) have proven to be an invaluable tool for complex control applications. Evolving ANNs has proven to be an effective technique for finding desirable solutions to difficult problems (sorting, control, pattern recognition, classification, etc.). However, efficiently implementing reconfigurable ANNs in hardware is still under investigation. Roggen et al report a successful implementation of spiking neural networks on an FPGA. While their work is promising, an analog solution is desirable because of the need to satisfy power and space constraints in space-exploration applications. Also, an analog implementation would virtually eliminate the need for interfacing the system

with the analog world of sensors, a task which requires extra resources. Implementing ANNs on microprocessor platforms presents many of the same problems[4]. An ANN is a group of interconnected artificial neurons interacting with one another in a concerted manner. The node receives weighted activation of other nodes through its incoming connections. First, these are added up (summation). The result is then passed through an activation function and the outcome is the activation of the node. The activation function can be a threshold function that passes information only if the combined activity level reaches a certain value, or it could be a continuous function of the combined input, for this purpose, the most common to use is the sigmoid function as a nonlinear activation function[6]. Recently, much research has been done on the applications of feedforward neural network (NN) for identification and control of dynamic systems. It is well known that a feedforward NN is capable of approximating any continuous functions closely. However the feedforward NN is a static mapping. Although much research has used the feedforward NN with tapped delays to deal with dynamical problems, the feedforward NN requires a large number of neurons to represent dynamical responses in the time domain. Moreover, the weight updates of the feedforward NN do not utilize the internal information of the NN and the function approximation is sensitive to the training data [7]. Previous research in rehabilitation robotics mostly employed either a traditional position based PID controller or an impedance controller. A PID controller requires continuous adjustment of control gains to accommodate a wide range of patients for a variety of tasks and conditions. In addition, a position based PID controller is not suitable for direct force control [8]. The goal of this paper is to design and implement a smart robot that tracks black line as fast, stable and accurate as possible used ANN. The paper is organized as follows: the system design is introduced in Section II, System input and output are addressed in Section III. Control system is demonstrated in Section IV. Section V includes the implementation of FPAA-based ANN controller. Finally, the conclusion and future work are given in Section VI.

System design:

The block diagram of carrying out the path tracking and stopping the robot is shown in Fig. 2. The hardware of the robot includes a wheeled mobile robot platform. The wheeled robot has 2 sensors at the front end of the wheeled robot body. This mobile robot is controlled by FPAA card. PID controller requires continuous adjustment of control gains to accommodate a wide range of tasks and conditions. The proposed controller implemented on a FPAA board to control the robot in real-time or the response of the ANN can be calculated offline and be reconstructed by controller using a lookup table. Error between the desired trajectory path and the path of the robot converges to zero rapidly and as the robot performs its tasks the controller learns the robot parameters and generates better control signal. The structure of the neural network implemented is the multi-layer feedforward network with single hidden layer and two hidden nodes for each layer. The input parameters for the ANN are the sensor 1 and sensor 2.

The robot is built with FPAA card, driver circuits, IR sensors and motors. The first task of the proposed system is to follow a black line on the ground without getting off the line too much. The black line is of 4 cm width (except at bends where a little variation may be occurred) on a white background. Fig. 3 shows courses to be used with the robot system. It could be any other suggested courses.

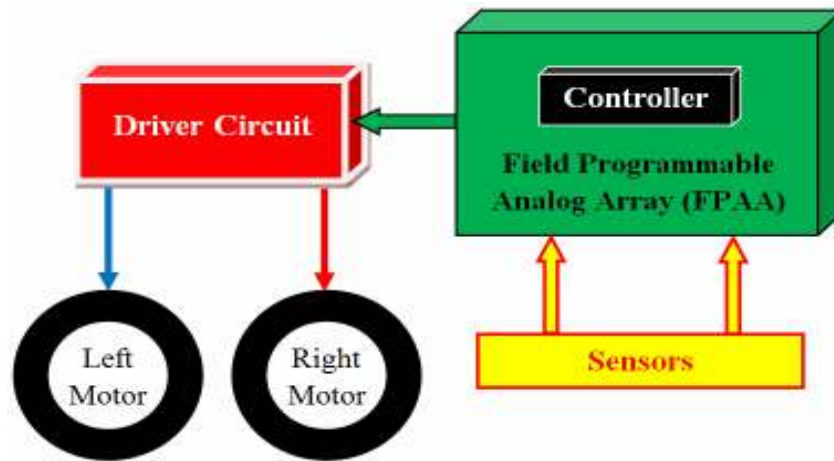


Fig. 2 System model for the robot path-tracking.

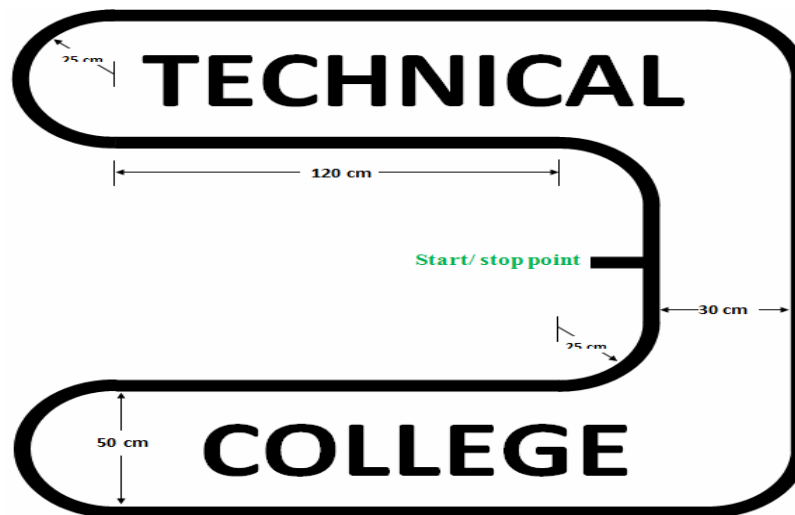


Fig. 3 Samples of path track shape

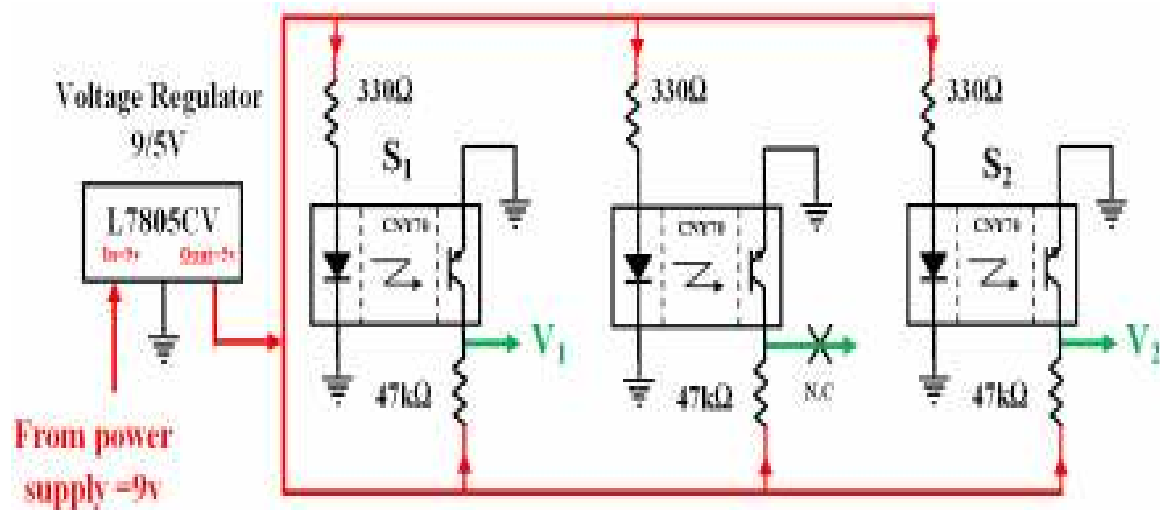
The basic operations of the line follower are as follows:

- 1- Capture line position with optical sensors mounted at the front end of the robot. For this, a combination of IR LED's and photo transistor, called reflective optical sensor with transistor output, is used. The line sensing process requires high resolution and high robustness.
- 2- Steer robot to track the line with any steering mechanism. To achieve this, two motors are used for governing wheels motion as shown in Fig. 2.

System input and output:

The system input of the path tracking is represented by IR reflective sensors, which have one emitter (IR LED) and one receiver (phototransistor). If the surface is white, it reflects the light and it will be sensed by the receiver. Similarly, if the surface is black it absorbs the light and the receiver will not be able to sense the light. When a light emitted from the diode is reflected off an object and back into the phototransistor, output current is produced, depending on the amount of infrared light, which triggers the base current of the

phototransistor. The amount of light reflected off a black line is much less than that of a white background, so the black line can be detected. In order to locate the position of black line below the robot and trying to keep the sensors on, sensors are fixed underneath the front end of the robot. The distance between the inner edges of sensors is chosen to be 2.4 cm, and that of the outer edges becomes 4 cm to minimize the IR interference between sensors. The width of the track is generally little more than the distance between the left edges of sensor-1 (S_1) and the right edge of sensor-2 (S_2). The practical connection circuit of the optical sensors is shown in Fig. 4, where CNY70 chip is used as a reflective optical sensor with transistor output.



$R_{CE}=188k\Omega$ without light (on black surface), $R_{CE}=11.7k\Omega$ with light (on white surface)

Fig. 4 Practical circuit of reflective optical sensors.

From Fig. 4, V_1 and V_2 are the output voltages of S_1 and S_2 respectively, which represent the input signals fed to the controller that implemented using FPAA card. When S_1 or S_2 is fully on the track, i.e. fully on the black line, there is no reflection of IR beam to the transistor base; hence the transistor works in the cut off region and its output V_1 or $V_2 \approx 4V$. When S_1 or S_2 is fully out of the track, i.e. fully on the white part, the reflection of the IR beam is fully to the transistor base, hence the transistor works in the saturation region and its output V_1 or $V_2 \approx 1V$.

The output voltage V_1 , for the circuit shown in Fig. 4 that is fed to controller, can be calculated as follows:

Without light: (on black surface)

$$V_1 = \frac{R_{CE}}{R_{CE} + R_C} * V_{CC} = \frac{188}{188 + 47} * 5V = 4V \dots \dots \dots (1)$$

With light: (on white surface)

$$V_1 = \frac{R_{CE}}{R_{CE} + R_C} * V_{CC} = \frac{11.73}{11.73 + 47} * 5V = 1V \dots \dots \dots (2)$$

Similarly V_2 can be calculated, obtaining the same values of V_1 .

The values measured of V_1 or V_2 are the same as the calculated values. When S_1 or S_2 is partially off the track, its output V_1 or V_2 is between the two mentioned conditions, which is proportional to the track area that the sensor is on it. To avoid practical problem, the power supply of 9V DC is used separately and independent on the main power supply. L7805CV voltage regulator is used to achieve 5V DC, which represents the voltage supply (V_{cc}) of the reflective optical sensor.

The path tracking system output is represented by two DC motors attached to wheel gears to move the robot. Its direction can be changed by just reversing the polarity of the power supply. The DC motors don't have enough torque to drive a robot directly by connecting wheels to it, so gears are used to increase the torque of a DC motor on the expense of its speed. Since the DC motor could not be attached directly to controller so driver circuit is used to pass enough current or voltage to spin a motor. The H-bridge driver circuit L293D is specifically designed to drive inductive loads (relays and DC motors) in the desired system, which allows forward and reverse motor rotation. Fig. 5 depicts driver circuit used as interface between the outputs of the FPAA and the two DC motors.

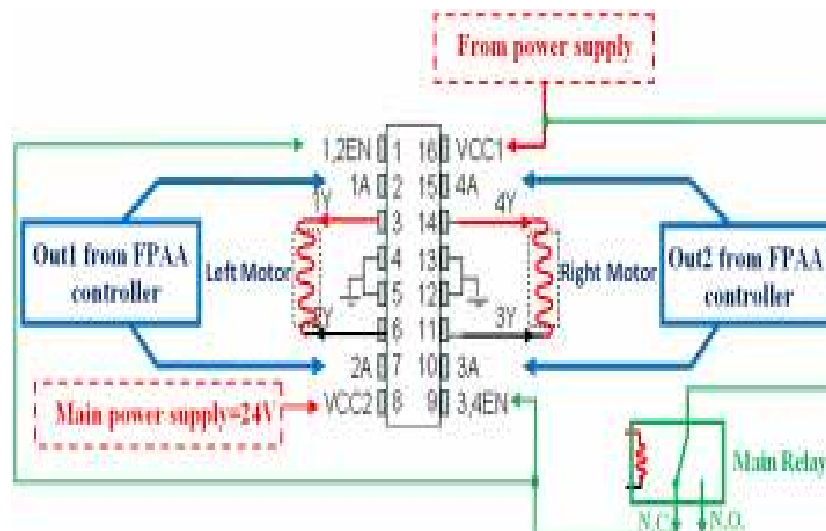


Fig. 5 Practical circuit connection of driver circuit.

Control system:

The control system which represented by FPAA card, acts as the robot brain which takes the decision of generating the desired output for corresponding inputs. The control system tries to keep the robot platform at the center of the track. It is certain that the robot at bent tracks will leave the center of the track, this causes one of the two sensors to be off the track (fully or partially). Here arises the importance of the control system to return the robot automatically back to the track center. Artificial neural network controller is implemented to carry out the FPAA-based path tracking control.

Implementation of FPAA-based ANN controller:

The ANN's hardware implementation using FPAA technology is realized in two steps: the first is training while the second is testing and used. The training step usually trains the

network for a certain target (output) with respect to the input. The most typical neural net setting, training is accomplished by presenting a sequence of training vectors, or patterns, each with an associated target output vector. The weights are then adjusted according to a learning algorithm. This process is known as supervised training [9], which is used in the present control system. Back Propagation Neural Network (BPNN) algorithm is used in this work because it is more accurate and matches with the required system. For training, it is required to specify the number of inputs, outputs, hidden layers and neurons in each layer as well as the input data and its target (output). The structure of the neural network implemented is the multi-layer (two layers) feedforward network with single hidden layer and two hidden neurons in each layer. The input parameters for the ANN are the sensor 1 and sensor 2. The transfer function for all the hidden nodes is pureline, and for the output network nodes pureline, the network structure configuration is shown in Fig. 6. The number of neurons in the network may affect the performance of the system and as the number of neurons increases the error of the system decreases however the network needs more training data. One idea is that at the first cycles of training the network is constructed with a few neurons and as the training continues newer networks with more neurons.

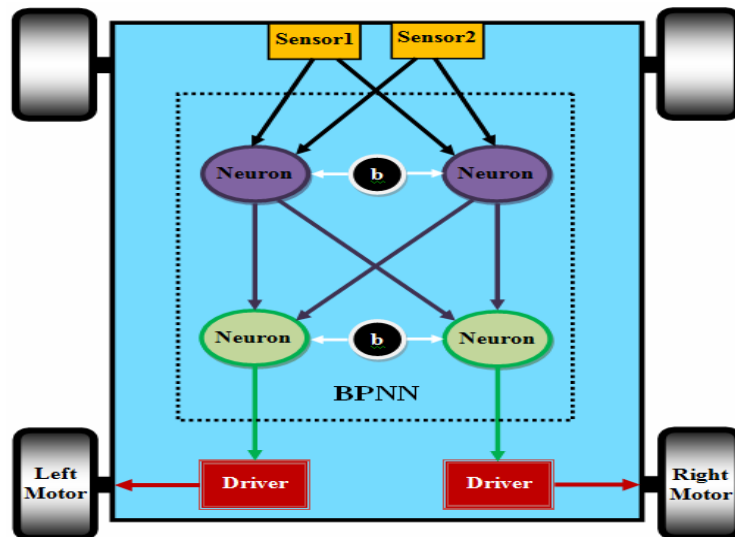


Fig.6 Architecture of implemented BPNNs

Training a back propagation neural network involves three stages:

- Feed forward of the input training pattern.
- Back propagation of the associated error.
- Adjustment of the weights.

Weights are obtained from an offline training procedure using MATLAB software to accurately simulate the network topology and to optimize the weights. The ANN training reached the goal with error of 0.0009978 through 664 epochs as shown in Fig. 7.

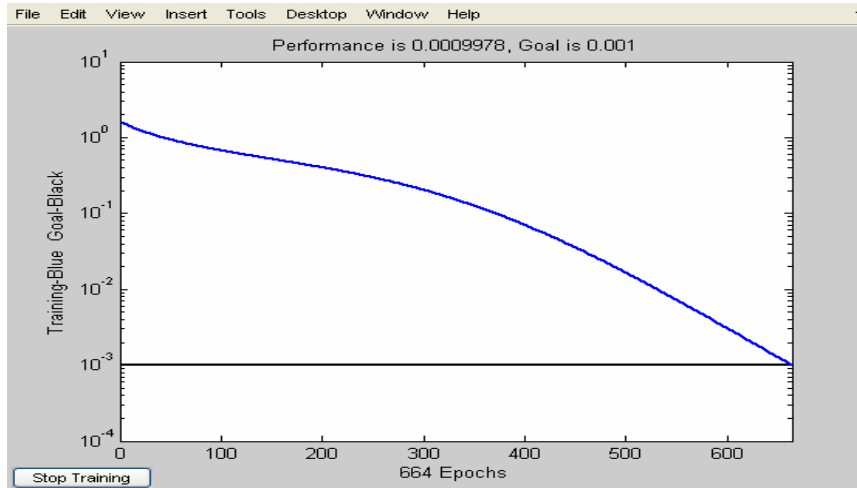


Fig. 7 Simulation results of ANN training using MATLAB program.

The optimal weights are downloaded to the FPAA chips for the corresponding real-time operation to control the mobile robot. The ANN controller circuits are implemented using FPAA card as shown in Fig. 8.

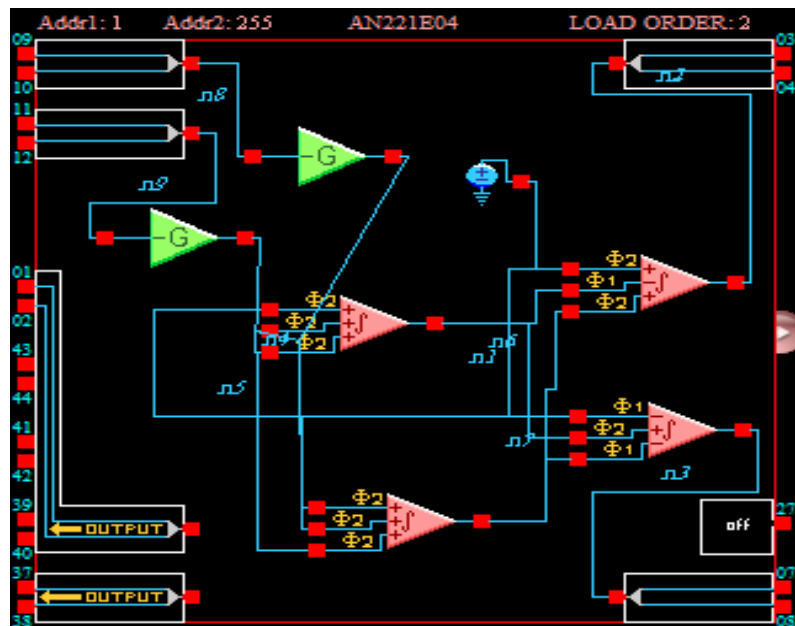


Fig. 8 FPAA-based ANN controllers.

It is designed to accept two inputs and two outputs; each output has two voltage values. Each neuron simply consists of a summer and an integrator. The simulation result of input and output of an ANN controller using AnadigmDesigner2 software are shown in Fig. 9. The practical values of the inputs and outputs of ANN controllers are shown in Fig. 10.

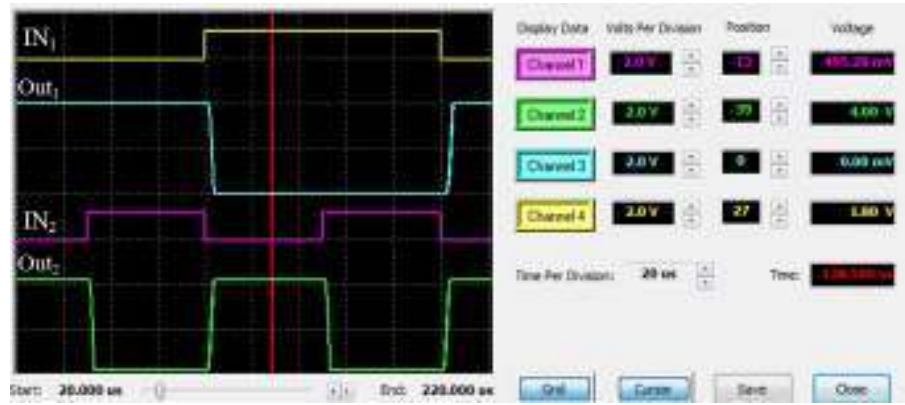


Fig. 9 The simulation results of ANN classifying two classes of data.

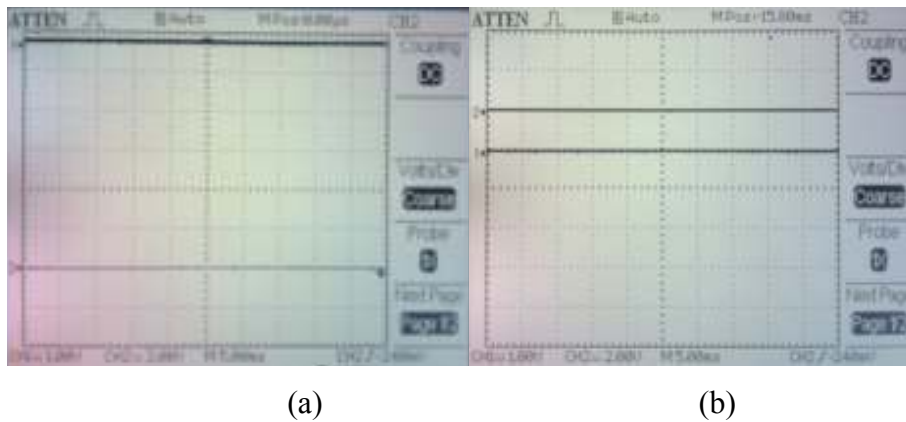


Fig. 10 Practical value (a) when sensor is at white portion, (b)when sensor is at black portion.

The whole practical implementation work; including designing and building all the circuits as well as the container of the robot body, flying wheels, DC motors ...etc; shown in Fig. 11 and Fig.12.

Conclusion:

This paper generally shows the application of neural networks in tracking control of wheeled mobile robots. The proposed controller can be implemented on a FPAA card to control the robot in real-time or the response of the ANN can be calculated offline and be reconstructed by controller using a lookup table. Error between the desired trajectory path and the path of the robot converges to zero rapidly and as the robot performs its tasks the controller learns the robot parameters and generates better control signal. The performance of controller is tested in simulation and on a real manipulator with satisfactory results.



Fig. 11 View of robot internal structure.



Fig. 12 Tracking the path.

References

- 1- Naoji Shiroma, Ryo Miyauchi and Fumitoshi Matsuno, " Mobile Robot Teleoperation through Virtual Robot ", *Proceedings of the 17th IEEE International Symposium on Robot and Human Interactive Communication, Technische, Universität München, Munich, Germany*, p.p.477-482,2008.
- 2- Micho Radovnikovich, Ka C. Cheok and Pavan Vempaty , " Comparison of Optimal Path Planning Algorithms for an Autonomous Mobile Robot", *IEEE Conference on technologies for practical robot applications* ,p.p. 3□39,2011.

- 3- Le QiA, Baoli Ma , and Wei Li , "Novel Path Following Controller for an uncertain wheeled mobile robot", *IEEE International Conference on Electrical and Control Engineering*, p.p. 20□209.
- 4- Kerstin Eckstein, Peter Möhringer, " Dynamically Reconfiguring of the Field Programmable Analogue Array AN221E04",
www.fh-sw.de/sw/fachb/et/pp/vuv.html,200□
□ Dmitry Berenson and Hod Lipson, " Hardware Evolution of Analog Circuits for In-situ Robotic Fault-Recovery", *IEEE NACA/DoD Conference*, p.p. 12-19,200□
- 6- Ali T. Hasan, N. Ismail, A.M.S. Hamouda, Ishak Aris, M.H. Marhaban, H.M.A.A. Al-Assadi," Artificial neural network-based kinematics Jacobian solution for serial manipulator passing through singular configurations", *Advances in Engineering Software* No.4,p.p. 3□9-367, 2002.
- 7- Rong-Jong Wai, and Chia-Ming Liu, "Design of Dynamic Petri Recurrent-Fuzzy-Neural- Network for Robust Path Tracking Control of Mobile Robot", *IEEE International joint conference on neural networks*, p.p.1-8, 2010.
- 8- Duygun Erol, Vishnu Mallapragada, Nilanjan Sarkar, Gitendra Uswatte, Edward Taub, "A New Control Approach to Robot Assisted Rehabilitation", *Proceedings of the IEEE, 9th International Conference on Rehabilitation Robotics*, p.p.323-328,200□
- 9- Laurene Fausett, *Fundamentals of neural networks, architectures, algorithms and applications*.