FPGA-Based Car-Like Robot Path Follower with Obstacle Avoidance

Aicha Moulay
University Mohamed Tahari Bechar
Bechar, Algeria
aichamoulay03@yahoo.com

Fatima Laoufi
Institute of Electrical and Electronic Engineering University M'Hamed BOUGARA (Boumerdes)
Bouira, Algeria
fatisem409@gmail.com

Tarak Benslimane
University Mohamed Tahari Bechar
Bechar, Algeria
bens082002@yahoo.fr

Othmane Abdelkhalek
University Mohamed Tahari Bechar
Bechar, Algeria
othmaneo8000@yahoo.fr

Abstract— **This project is about the design and implementation of an System on Programmable Chip (SoPC) based automated path following with obstacle avoidance using infrared sensors and ultrasonic sensor.**

The Line follower robot is a mobile machine that can detect and follow the line drawn on the floor using infrared sensors which are installed under the robot . The path is predefined as a black line on a white surface with a high contrasted color.

This robot is provided with ultrasonic sensor to avoid an obstacle either from the right or the and it will stop if an obstacle is detected on the path or if the path is ended.
The digital controller is designed and implemented using both the integrated circuits and the Field Programmable Gate Array (FPGA).

Keywords—line follower, obstacle avoidance, infrared sensor, ultrasonic sensor, DC motor, FPGA.

I. INTRODUCTION

A. Overview:

At present, the robots are widely used in industry capable of performing repetitive tasks with high speeds [1].

Line follower can be used to deliver mail with an office building and deliver medications in a hospital, it can be used also in guidance system for industrial robots moving on shop floor and in military as spy kids or in many other application [2].

A simple robot was designed by Jaseung Ku which is able to follow a black line on the ground without getting off the line. The robot consists of two sensors installed underneath the front part of the body, and two DC motors drive wheels moving forward. Mohammed Aidil B. Abdul Jalal developed an autonomous mobile robot which exhibits line tracking and obstacles avoidance behaviours using The PIC16F877A microcontroller as the brain to control the system with MPLAB IDE software to program this microcontroller in C language[3].

B. Motivation

Robotics is a part of today's communication .This later is a part of advancement of technology. Hence robotics is fast growing and interesting field [4]. So we decided to work on robotics field and design something which will make the life easier to human in today's aspect.

Path following and obstacle avoidance are two very important behaviors that must be considered in the process of Autonomous Mobile Robots development. Autonomous mobile robots have been widely used in many researches and applications. They can be programmed to do specific tasks such as collision avoidance and trajectory planning [5]. Our objective is to gather the two by designing and implementing a controller for a following path with obstacle avoidance car.

C. Project Objectives

The purpose of this project is to design and implement an FPGA-Based autonomous Following path with Obstacle avoidance car. The objectives to be achieved are:

1. Search for the black path using infrared sensors..
2. Adjust the car to follow this path using these sensors.
3. Stop the car if an obstacle is detected on the path by the ultrasonic sensor and move the car to the right to avoid it if no other obstacle is on this side.
4. Move the car to the left if an obstacle is detected on the right and no constraint is on the left side.
5. Search again for the predefined path after avoiding the obstacle.
6. Stop the car if there is no black path or if obstacles are detected on the path and on both sides.

D. Structure of the autonomous following path with obstacle avoidance

Due to the flexibility developing of the FPGA, we choose to implement our digital controller by integrating a heterogeneous computing platform made up of microprocessor based on the SoPC approach. The system is designed around the NIOS II/e soft-core processor instantiated in the Cyclone II EP2C35F672C6 FPGA system and some custom logic, all on a single chip. Our system consists of two main parts, hardware and software.

E. Organization of the report:

The remaining of the paper sections is organized as follows:

Hardware system design: it describes the design and implementation of both off-chip and on-chip subsystems. Software system design: presents the algorithms and flowcharts used to configure the FPGA . Conclusion: gathers the work presented in the paper, the remarks about

978-1-7281-2580-0/20/$31.00 ©2020 IEEE

the difficulties faced and the discussion of the possible direction for further enhancement of the system. Finally the paper terminates with a list of references.

II. HARDWARE SYSTEM DESIGN

The principle of this part is to define how the DC motors are controlled using motor driving unit and the sensors circuit and how these combined circuits are connected to the FPGA.

the system is made up of two main subsystems as shown in "Fig. 1" .

A. OFF-Chip Hardware Subsystem

This subsystem consists of the proto-board built circuits. It includes the the motor driving unit, the interface block (buffers), the sensors and data acquisition unit.

1. Motor driving unit (L298N H-bridge):

In order to make the robot move in all directions with different speeds, a motor driving is needed for the robot to operate properly.

The two DC motors (Direct Current) are controlled by the PWM generator (Pulse Width Modulation) implemented in the FPGA and L298N H-bridge circuit. The L298N is an IC(Integrated Circuit) with two parallel H-bridges. To control the movement of each DC motor, three pins of L298N are used. Two of them are used for controlling the direction by setting one input high and the other low, and the third one is used by setting it high to enable the DC motor.

2. Buffer Interface Block

All the components of the OFF-Chip circuitry implemented on the proto-board operate under a voltage of approximately 5V. However, the DE2 board supplies a voltage of 3.3V. In order to make the signals exchanged between the FPGA and proto-board's components in a compatible way, SN74LS244 buffers are integrated as an interface between them and the FPGA.

3. Sensors

- IR (InfraRed) Sensor

Our robot is designed to follow a black path on a white surface. In order to do that, optical sensors are used.

We have used the CNY70. It is a reflective optical sensor that includes an infrared emitter and phototransistor in a leaded package which blocks visible light [6]. “Fig. 2” illustrates the circuit of the CNY70 IR sensor.

Its principal of operation is based on the intensity of the IR light reflected back to the phototransistor.

We have used three IR sensors arranged in a ‘V’ shape for the line detection; based on the data which read from the three sensors the robot will move forward, right and left direction. “Fig. 3” represents the arrangement of the IR sensors. The width of the line here should be greater or equal the distance between the center sensor and the right or the left one (to make sure that if one sensor is out of the path another one is on it).

- Ultrasonic Sensor

In this part the ultrasonic sensor is used to detect and avoid any obstacle stands in the path of the robot. The sensor has two openings on its front. One opening transmits ultrasonic bursts and the other receives them. This module is interfaced with the FPGA through two pins, the Trig and the Echo pins through the expansion-header and SN74LS244 buffer.

The trigger pin receive from the FPGA a pulse of at least 10µS.If there is an obstacle in front of the robot, the transmitter module will reflect back the signal as an input through the Echo pin of the sensor to the FPGA; this signal will be high during the whole time from transmitting the ultrasonic burst until receiving it. After getting the time signal, the distance between the robot and the obstacle can be calculated.

4. Data Acquisition Unit

Since we need digital value from the sensors instead of analog, an analog-to-digital converter (ADC) is used. The ADC0808 is 8-channel, 8-bit analog-to-digital converter. It works on the principle of successive approximation. It requires a clock signal of 640 KHz to perform a conversion in 100µS. The ADC has a total of eight analog input channels which can be selected using the address lines (A, B and C). Since we have used three sensors, three channels are needed; in this case IN0, IN1 and IN2 are used.

The interfacing between the ADC and the DE2 is done through the expansion header. "Fig. 4" illustrates the interface between the ADC and the FPGA.

B. ON-Chip Hardware Subsystem

This subsystem represents the brain of the system. It mainly consists of the SoPC system which includes Nios II/e soft processor, memory, JTAG and I/O ports. In addition to some non-SoPC logic units developed in VHDL or LPMs, each has a specific function.

- *The SoPC subsystem*

The SoPC subsystem is built around the Nios-II 32-bit embedded soft-core processor, it includes the following items:

- NIOS II Processor: the base of the system is the NIOS II Soft-Core Processor under a clock of 50MHz.
- On-Chip memory: a memory of 32Kbytes is used to store the program.
- JTAG UART: a component that is used to allow the host computer to control the Nios II system.
- I/O Peripherals: to control the hardware part, several input/output peripherals are needed. The peripherals are described as following:
  - ADC_SOC: 1-bit output signal refer to Start-of-Conversion of the ADC, which triggers the conversion of analog data.
  - ADC_addr_lines: is a 2-bit output signal that connected to the address lines of the ADC; it is used to select between the infrared sensors.
  - ADC_EOC: is a 1-bit input signal that refer to the End-of-Conversion

978-1-7281-2580-0/20/$31.00 ©2020 IEEE

of the ADC, it is go high when the conversion is finished.

- Motor_direction: is a 4-bit output signal connected to the H-bridge chip.
- ADC_data: is an 8-bit input signal referring to the data coming out from the ADC.
- Ultrasonic_Trig: is a 1-bit output signal that is responsible of triggering the ultrasonic sensor.
- Ultrasonic_Echo: is a 1-bit input signal that receives the digital signal back from the ultrasonic sensor.
- Pwm_word: is a 32-bit output signal that is used to adjust the speed of the two dc motors depending on the requirements of the system.

Once all the peripherals are specified and assigned base addresses, the system is generated.

- *Non SoPC subsystem*

In addition to using the SoPC system, we have non-SoPC units. These units are the PWM generator for DC motors and the clock divider which are controlled by the SoPC system.

- PWM Generator

In order to control the speed of the DC motors a PWM is used by applied the PWM signal to the enable pin of the H-bridge. The PWM is implemented using a 5-bit binary counter connected to the select lines of a 32-to-1 multiplexer. The duty cycle of PWM will changed by changing the number of successive ones applied on the data inputs of the multiplexer, therefore, the more successive ones applied, the higher the duty cycle**.**

- The Clock Divider

As mentioned before the ADC0808 requires a clock signal of 640 KHz. To achieve this, we have implemented a counter using an LPM counter of modulus 79 provided in the Quartus II library to divide the system's clock frequency of 50 MHz and generate the required clock signal.

After specifying all the settings and assigning the base-addresses, the system is generated.

The compilation of the whole system was successful. The summary of the compilation shown in "Fig. 5", we can see that the entire system uses 5% of the total logic elements, 769 registers, 7% of the total pins and 56% of the total memory bits.

III. SOFTWARE SYSTEM DESIGN

After building the hardware design, the software takes its role to manipulate the data and take decision accordingly. This rule is ensured through a code written in assembly language or a high-level language and executed by a microprocessor.

Our project depends on the data read from the three infrared sensors and the ultrasonic one. The program receives data from these sensors and generates the output corresponding to the current situation. The generated output is used to control the rotation of motors such as forward, turn left and turn right.

- Data Acquisition Routine

The routine described by the flowchart in "Fig. 6" performs the acquisition of resulted data from three infrared sensors. The routine selects sequentially the channel to convert then sends the Start/ALE pulse to latch the address of the selected channel and starts the conversion. Next, it keeps polling the end of conversion output signal until the conversion is over. Once acquired, the data is temporarily stored into specific registers before it is processed by the Nios II system .

- The Obstacle Scanning Procedure

The aim of this procedure is to compute the duration for the Echo port to be high since this indicates the presence of an obstacle.

The procedure starts by triggering the sensor with high for at least 10 μs and waits for Echo port to be high. While this port is high (1) a counter is incremented. Once the Echo port turns low (0) the final count is translated into distance according to specific calculations.

- *The Main Code*

The complete code is divided into several subroutines; each one is responsible for achieving a desired task. The main code allows the connection between these subroutines to reach the objective of the system. The routine described by the flowchart in" Fig. 7".

- *Programming Language*

Nios II offers the possibility to work with assembly or high level language C.

The native language of any microprocessor is the assembly one which is the lower language. Therefore no need for the compiler to convert it like other higher languages. So it is very useful to communicate with the hardware level.

IV. CONCLUSION

An FPGA-based path following with obstacle avoidance controller was designed and implemented. The autonomous car successfully tracks the black line above the white surface whereas the obstacle avoidance mechanism didn't work well. The robot senses the line and follows the track in all directions.

The system is designed using SOPC approach. It is designed around the Nios II/e soft-core processor instantiated in control of the motors based on the data provided by the infrared sensors.

The SoPC approach has several benefits. It is flexible in both hardware and software. The designer can use the exact components that fit his application. Moreover the reconfiguration and modification of the system can be done at any time by adding a new component or deleting an exist one.

978-1-7281-2580-0/20/$31.00 ©2020 IEEE

During the realization of this project troubleshooting has taken a long period delaying our work. Some of these troubleshooting are:

- The circuits on the proto-board: they are so sensitive due the several connected wires. If the system does not work it is necessary to check the whole circuit to discover the problem.
- The motor driven unit: L298 has caused drooping of voltage several times due to its shape which is difficult to connect tightly.

The problems mentioned above can be solved by using printed circuit board PCB instead of proto-board circuit which is more efficient and reliable. L293 is a driven motor chip which can replace L298. it is easy to connect it. In addition, it is provided with the protective circuit so there is no need to add this latter to the whole driven circuit.

As future enhancement and further work:

- Providing the car with more infrared sensors (5 sensors) so the car may follow the path more accurately.
- Providing the two projects with an android application using Smart phone to select a mode that can be obstacle avoidance or a path follower.
- Using color sensors so the robot can sense different colors. It can be used in the robotic game competition and other fields.
- Optimizing the best path; the car tracks the different paths from a certain origin to a given destination with memorizing the shortest path to reach the destination so the robot will optimize its path from different paths in a short time as maze solver robot does.

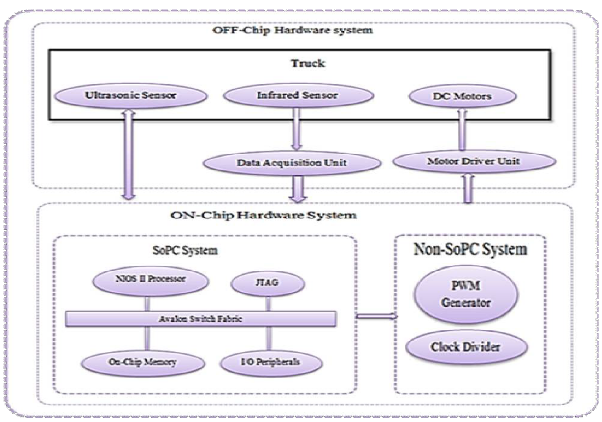

Fig. 1. Block diagram of the overall system

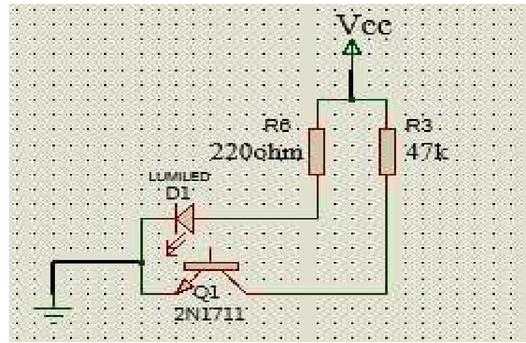

Fig. 2. The circuit of CNY70 Infrared sensor.

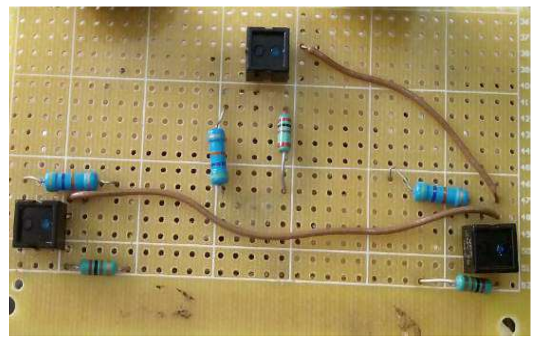

Fig. 3. Arrangements of Infrared sensors

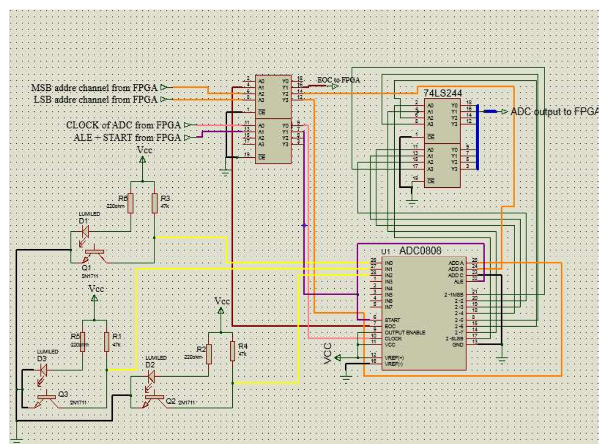

Fig. 4. Schematic diagram of interfacing IR sensors with ADC

978-1-7281-2580-0/20/$31.00 ©2020 IEEE

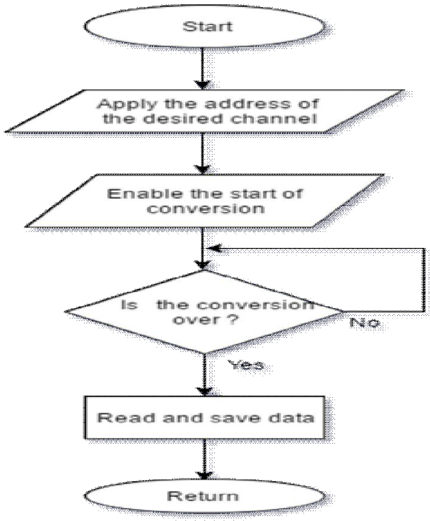

Fig. 5. Data Acquistion routine

978-1-7281-2580-0/20/$31.00 ©2020 IEEE

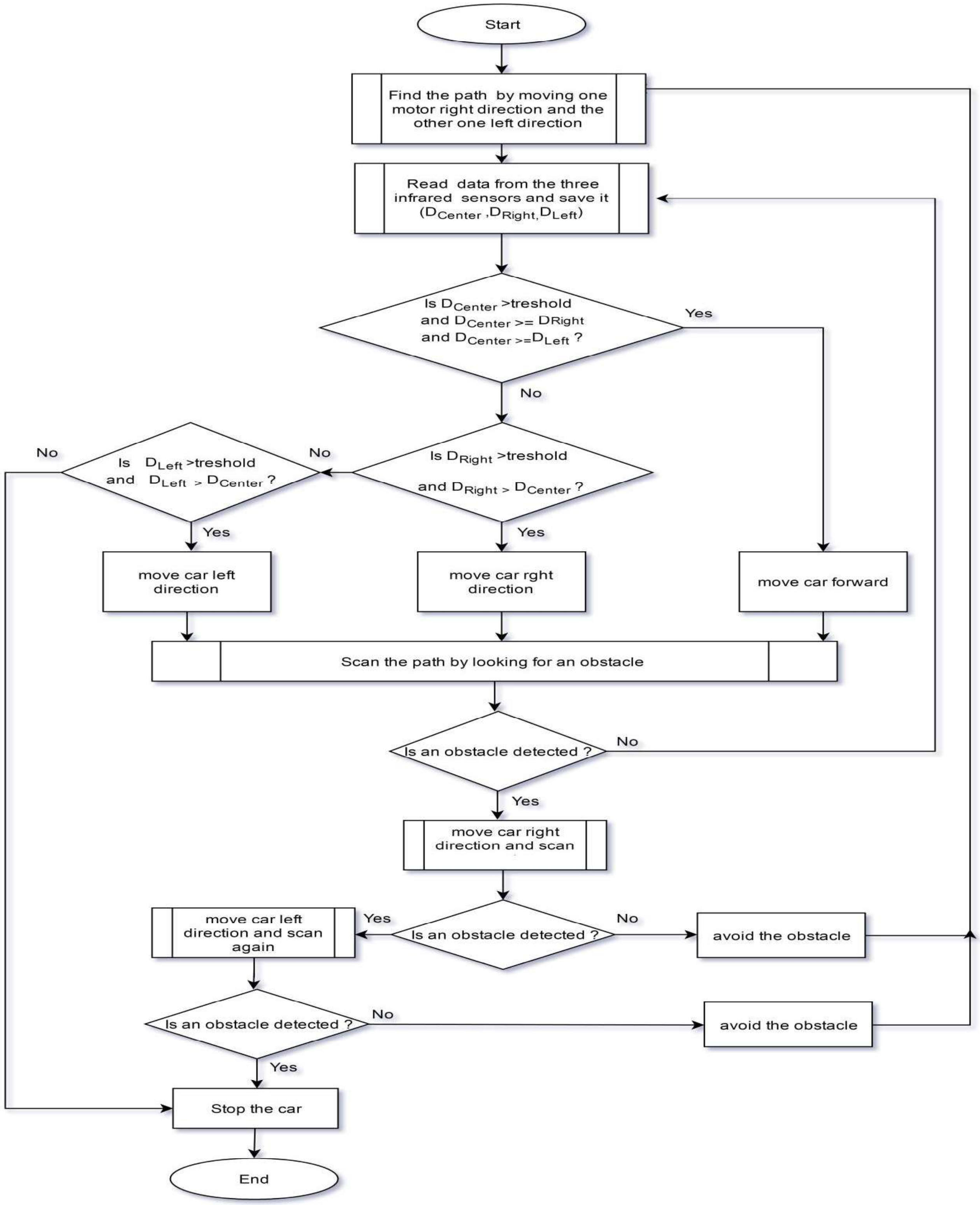

Fig. 6. The flowchart of the main program

.

978-1-7281-2580-0/20/$31.00 ©2020 IEEE

ACKNOWLEDGMENT

All praise and thanksgiving to Allah the most powerful and most merciful who give us the ability and patience to accomplish this humble work.

We would like to express our gratitude to our supervisor Dr.A.BENZEKRI for his guidance and support during our work. We are grateful to our teachers, academic staff and workers at the Institute of Electrical and Electronic Engineering who prepare for us the environment to work and offer to us their valuable help.

Special thanks to all people who supported and guided us in any respect during the completion of the project especially our colleagues.

REFERENCES

[1] KAUSHIK VELUSAMI, DEEPTHI VENKITA RAMANAN, SHRIRAM KUSUDEVAN, "*Introduction to AI-AN Intelligent Approach*." Lulu Publication,United States of American Edition 2013.

[2] M. S Islam, M. A. Rahman,"*Design and Fabrication of Line Follower Robot," Asian* Journal of Applied Science and Engineering vol. 2 no. 2 pp. 127-132, 2013 ISSN 2305-915X(p); 2307-9584(e)

[3] M K Islam, M T Islam and M R Islam "*Design and Fabrication of a simple line follower autonomous robot with magnetic gripper,*", International Conferencence on Mechanical Engineering and Renewable Energy 2013 (ICMERE2013), 1-3 May 2014, Chittagong, Bnagladesh DOI: 10.13140/2.1.13723.0408.

[4] Shireen Matar, Umera Rana , "*Line Following and Obstacle Avoidance Robot Using PIC MICROCONTROLLER*", Semester project.

[5] Marwah M Almasri, Khaled M Elleithy, Abrar M Aljlan "*Obstacle avoidance and line following mobile robot with the integration of fuzzy logic system in static and dynamic environments,*", IEEE Long Island Systems, Application and Technology Conference (LISAT), 1-6, April 2016 DOI:10.1109/LISAT.2016.7494139.

[6] Datasheet of CNY70:https://www.alldatasheet.com/datasheet-pdf/pdf/252297/VISHAY/CNY70.html.

.

.

978-1-7281-2580-0/20/$31.00 ©2020 IEEE