

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/377605450>

DESIGN AND FABRICATION OF AN AUTONOMOUS LINE FOLLOWER ROBOT CAPABLE OF PICKING AND DROPPING OBJECTS FROM ONE POINT TO ANOTHER SERONEY DANIEL KIPTOO

Preprint · December 2020

DOI: 10.13140/RG.2.2.35328.81925

CITATIONS

0

READS

96

2 authors, including:



Seroney Daniel

Moi University

47 PUBLICATIONS 1,410 CITATIONS

SEE PROFILE

**TDESIGN AND FABRICATION OF AN AUTONOMOUS LINE
FOLLOWER ROBOT CAPABLE OF PICKING AND DROPPING OBJECTS
FROM ONE POINT TO ANOTHER**

SERONEY DANIEL KIPTOO

**A project report submitted in partial fulfilment of the requirements for the
award of Bachelors in Engineering (Electrical and Electronics) in the School of
Engineering Department of Electrical and Communication Engineering.**

December 2020

DECLARATION

I hereby declare that this my original work and has not been presented for the award of a degree or other awards in any other University

Seroney Daniel Kiptoo

EC/25/16

Department of Electrical and Communication

Moi University.

Signed..... Date.....

This project report has been submitted with our approval as university supervisors.

MR KALENDA

Department of Electrical and Communication

Moi University

Signed.....Date.....

MR MUIGAI

Department of Electrical and Communication

Moi University

Signed.....Date.....

DEDICATION

This research is dedicated to my parents, Mr. and Mrs. Reynolds Chepng'abit Seroney, my brothers and sisters who really accorded me financial and moral support during my journey with the project.

ACKNOWLEDGEMENTS

I wish to thank all my lecturers in the Electrical and Communications department and especially so to my supervisors Dr. Kalenda and Dr. Muigai for guiding me through this research. I would also wish to thank the Moi University Electrical Laboratory Staff and in particular Mr. P. Mutai for assisting me in the fabrication of the robot line sensing model from the Portable Circuit Board, resistors and connecting wires. I would also not forget my colleague, Mr. Mike Rotich for guiding me in the software development of the system and also the other colleagues namely Mr. Omoso and Mr. Kongere Kingsley just to mention a few for being there always to provide peer review to my research. The Moi University technical staff also assisted me a great deal during the hardware characterization of component hence, I wish also to thank them all.

Finally, I wish to thank my relatives for their moral and financial support and above all thank God for the gift of life all through my research.

TABLE OF CONTENTS

DECLARATION.....	ii
DEDICATION	iii
ACKNOWLEDGEMENTS.....	iv
TABLE OF CONTENTS	v
LIST OF TABLES	vii
LIST OF FIGURES	viii
ABSTRACT	ix
CHAPTER ONE: INTRODUCTION	1
1.1 Background	1
1.2 Major parts of a line follower robot	3
1.2.1 The mechanical part (manipulator).....	3
1.2.2 The power supply	4
1.2.3 The Control Unit	4
1.2.4 Problem Statement	4
1.3 Research Objective.....	5
1.4 Rationale of the Study	5
CHAPTER TWO: LITERATURE REVIEW	8
CHAPTER THREE: THEORETICAL CONSIDERATION OF	
COMPONENTS	11
3.1 Analogue Robot Control Systems	11
3.2 Digital Robot Control Systems	12
3.3 The Sensor	17
3.4 Direct Current Motor.....	18
3.5 Microcontroller	21
3.5.1 The Organization of Microcontroller-Based System	21
3.5.2 Microcontroller Applications	22
3.7 Control theory	24
3.7.1 The Transfer Function Concept	25
3.7.2 Types of Control Systems	27
3.7.3 Fundamentals of Automatic Control	29

CHAPTER FOUR: RESEARCH METHODOLOGY	30
4.1 Selection of hardware components used	30
4.3 Robot construction	32
4.4 System construction, Testing and Troubleshooting	32
4.5 Design specification	33
4.6 The system hardware design	34
4.6.1 The mechanical hardware	34
4.7 Other hardware of the control system	35
4.7.1 Sensors	35
4.7.2 Motor Drive	36
4.8 Input Circuit	39
4.9 Output system	44
4.9.1 Electric Motor	44
4.9.2 Purpose of gears	45
4.10 Robot body	46
CHAPTER FIVE: RESULTS AND DISCUSSION	47
5.1 Base motors	47
5.2 The arm and gripper	48
5.4 The sensors	50
5.5 Overall System Performance.....	51
CHAPTER SIX: CONCLUSIONS AND RECOMMENDATION FOR FURTHER STUDY	56
6.1 Conclusions	56
6.2 Recommendations	56
REFERENCES	57
APPENDICES	59
APPENDIX I: The Software	59
APPENDIX II: Pictures of the Constructed Robot	68
 LIST OF TABLES	
Table 5.1: Motor movements with power supply	48
Table 5.2: Robot moving according to motor direction	48
 LIST OF FIGURES	
Figure 3.1: Block diagram of digital control technique of a robot motor	14

Figure 3.2: Block diagram of closed-loop digital computer robot control system...	14
Figures (a) 3.3, and (b) 3.3: Cut-away diagram and side view of a two-phase PM motor.	19
Figure 3.4: Rotor magnetization (https; / www. en; org ; D.C motor)	19
Figure 3.5: The organization of a micro controller-based system	22
Figure 3.6: (a & b): Block diagram of DC motor used as a control component with negative feedback	26
Figure 3.7: Block diagram of open-loop control system	28
Figure 3.8: Block of closed-loop system	28
Figure 4.1: Robot design block diagram	31
Figure 4.2: Designed circuit of gripper sensor	34
Figure 4.3: Pin details of L298N	37
Figure 4.4: ATmega 328 board	38
Figure 4.5: Input circuit.....	39
Figure 4.6 Arrangement of line tracking sensors	40
Figure 4.7: Line tracking sensor circuit.	40
Figure 4.8: Range sensor circuit.....	41
Figure 4.9: Sensor strategies	42
Figure 4.10: 7805 Regulator	43
Figure 4.11: circuit diagram for voltage regulator	43
Figure 4.12: Description of various parts	45
Figure 4.13: Different types of movement of robot	46
Figure 5.1: DC base motors wheels interfaced to motor drivers.....	47
Figure 5.2: Motor Driver Truth	48
Figure 5.3: Connector circuit between arm and gripper	49
Figure 5.4 :The end- effector (gripper)	50
Figure 5.5: Fabricated proximity and line tracking sensors	50
Figure 5.7: The System Software Flow Chart	53
Figures 5.8: Robot operations	55

ABSTRACT

Robot becomes widely used in industries due to their characteristics. Robot is able to work in 24 hours continuously without feeling tired unlike human that confine to certain time. The cost to setup the robot nowadays becomes more affordable and their long term prospect is bright judging from their capacity to perform. But in reality, there is no robot able to function perfectly and without making errors. A better controller is needed, to allow the robot performs efficiently and make less error. This research try to implement arduino duemilanove ATmega 328 controller on mobile robot to establish whether the robot perform efficiently. This mobile robot has a line tracking module, arm and gripper, where it will follow the track made from white line, pick and drop object. This is an area where the arduino duemilanove ATmega algorithms is implemented, the robot has been able to follow the white line effectively and moving along the track smoothly while at the end of the track, picked, carried and dropped object to destination. All the robot objectives were achieved. The objectives included, constructing a program and uploading it to the microcontroller that was used to control the whole functionality of the robot. However, the recommendation is that, to avoid malfunctioning, steering mechanism should be well managed and more functionality of the system added in order to allow other operations like, sensing colour, counting and tracking curving tracks.

CHAPTER ONE INTRODUCTION

1.1 Background

The history of line follower automation is characterized by periods of rapid change in automation techniques. Either as a cause or, perhaps, an effect, such periods of change in automation techniques seem closely tied to the world economics. The use of a line follower robot became an identifiable and a unique device in the 1960s due to the efforts of Joseph Engelberger, George Devol who formed the robotic company called “unimation” characterized the latest trend in the automation of the manufacturing process.

The concept of automation refers to the ability of a machine to perform a given sequence of tasks and meet certain specifications automatically in 1985 by Asfahl C.R who had constructed automatic robots aided in manufacturing industries. The concept also involves the ability of the machine to control its performance enabling the system to monitor and adjust its performance through a feedback system to ensure that the given specifications are met. Within the range of automated manufacturing equipment, British automation and robot association defines a robot as a reprogrammable multifunctional manipulator designed to move part of items, materials, tools or other specialized devices through variable programmed motions for the performance of a variable tasks. Machines which are for the most part relegated to one class of tasks are considered fixed automation.

This thesis focuses on the control of the most important form of the line robot, the mechanical manipulator with two degrees of freedom using an Arduino duamilanove ATmega 328 microcontroller as the “heart “of the control system. The manipulator is controlled by four 12-V D.C motors for wheels, arm and the gripper movements. The

base motors controls the rotation of the wheels, arm motor controls the raising of the arm in the vertical plane and the gripper motor controls the opening and closing of the gripper. The gripper D.C motor used is coupled to the gripper via gear designed from aluminum sheets.

By and large, the study of mechanics and control mechanical engineering contributes knowledge of manipulators is not a new science but a collection of concepts taken from “classical” fields. Mathematics supplies the tools for describing actual directions and other attributes of manipulators. Control theory provides the tools for designing and evaluating algorithms to realize desired directions or force applications. Electrical engineering techniques are brought to bear in the design of sensors and interfaces for line follower robots and computer science concentrated on repetitive and hazardous applications.

The robot are well suited for doing repetitive jobs that must be done in manufacturing plants, which require a person to act like a machine. The job may be is to pick and place an item for a number of times to complete the manufacture of products. A robot placed at such a situation can perform the job at the same rate without experiencing fatigue and boredom normally associated with such jobs. Jobs are considered hazardous because of the toxic fumes, the weight of the material being handled or the danger of working in an environment containing high levels of radiation.

Line follower robots are being used throughout the America, Asia, Australia, Europe and other cited by National robot Associations (2017). Approximate distribution by location is as follows by 2016/2017.

Country	Number of robots
Japan	38000
N. America	58000
Germany	33000
France	21000
Thailand	3000
Italy	7200
China	90000
India	2600
Other	1800

1.2 Major parts of a line follower robot

The major parts of a line follower robot are: the manipulator, the processor, the sensor and the motor. The major parts also called components combined are used to classify robots and define the capabilities of a robot.

1.2.1 The mechanical part (manipulator)

This is the physical structure of the Robot. The manipulator consist of the base, which carries the processor, the sensors and the wheel motors. Figure 1.1 which was a research on a path following autonomous robot by Andrew J. T., (2001) shown base part of a line following robot which constituted the manipulator.

Microcontroller performs the work of brain, sensors acts as eyes and actuators work as legs of human.

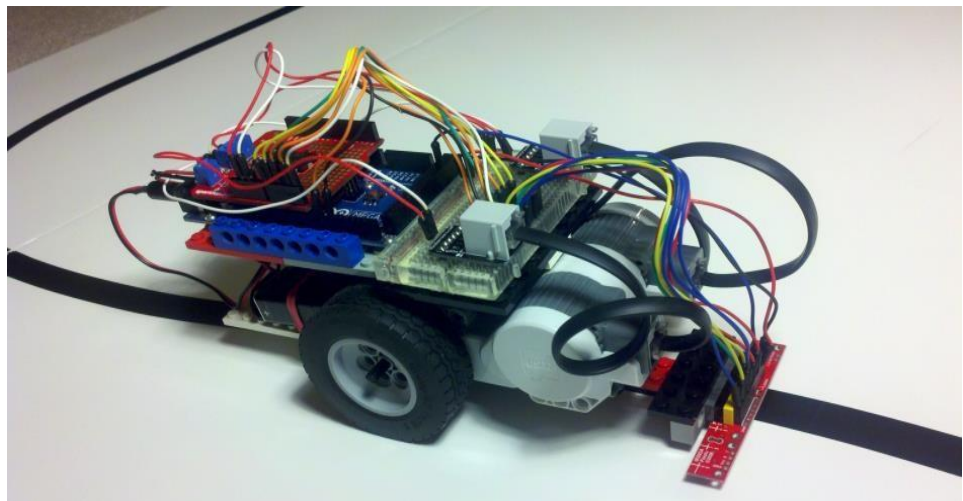


Figure 1.1: Base part of a robot.

1.2.2 The power supply

This supplies power to the manipulator so that it can be trained to move through its programmed direction. It is the muscle power of the robot. There are three types of power supplies commonly used, that is hydraulic, pneumatic and electric power supplies.

Hydraulic power supply provides the manipulator movement by pumping oil or some other hydraulic fluid through pipes or hoses, hydraulic cylinders or hydraulic motors.

Pneumatic power supply provides the power to move the robot's manipulator through the use of compressed air. A pneumatic power supply directs air through pipes or tubing to cylinders. The compressed air enters the end of cylinder and as the air pressure in the cylinder goes up, the piston in the cylinder moves, thus moving the manipulator. Electric power supply uses electrical motors to move the manipulator.

1.2.3 The Control Unit

This controls the power supply so that the manipulator can perform its tasks. There are quite a number of robot control systems, which vary from the classical analogue robot control to the modern computer/microcontroller control systems. All these control systems have the same purpose: to direct the motion of the robot's manipulator.

1.2.4 Problem Statement

People are unwilling to work, because the kind of job to be done is repetitive.

Although, a variety of line follower robots that principally are analogue/digital exist, there is a need to develop a robot that is more accurate, efficient and less expensive

in terms of components used in designing and fabrication. Where, while designing and fabricating, a new knowledge on how to make robot will be enhanced.

1.3 Research Objective

1.3.1 General Objectives

To design and fabricate a line follower robot that is able to pick and drop objects within its working environment.

1.3.2 Specific Objectives

- i. To develop the program of microcontroller.
- ii. To build and test a proto- type of the line follower autonomous robot that is able to;
 - Coordinate and follow the line.
 - Proximity sense and stop when there is an obstacle at the front.
 - Lift the arm and open the gripper to grip the object.
 - Lift an object, carry it and drop it to destined position.

1.4 Rationale of the Study

In view of the rapid depletion of existing hazard areas and unwillingness to work of human beings, it is apparently clear that there will be no manpower for work, therefore, unless scientific measures are taken a problem will be encountered in the field of work. Due, then, there is a need to explore a machine that will work like a human being.

As per international and British association, Robot is a machine that is, economical, saves time, can work in any type of environment and is faster. However, the major challenge is how to harness this artificial human being, the material to use and how

to develop them. For efficiency, microcontroller with desirable properties is needed. A drive motor to provide an arm with two degrees of freedom is required. Among the materials, aluminum sheet has been used to make body part and for upward movement. Downward movement of the arm will be enabled by use of a manual switch and bolts for fixing various part of the arm. These materials will easily and cheaply fabricate autonomous robot and Since there is very little information available on these materials, the arduino ATmega328 microcontroller and dual L298N motor driver carrier which switches clockwise or anticlockwise to enable forward and backward movement has been investigated and optimized for autonomous robot applications.

A microcontroller offers greater flexibility since bits in memories replace wired connections hence any modification is done only by reprogramming chips. Microcontroller also has other good features as compared to other controller components such as Versatility, compact size, high speed of operation, low cost and high reliability as seen from ATmega 328 Datasheet (2012), thus it is adapted for this research. A 16-bit microcontroller was used in the design application because a microcontroller is an inexpensive single-chip computer. Single chip means that the entire computer system lies within the confines of the integrated circuit. Microcontrollers are capable of storing and running a program and interfacing to external devices that are most important features.

The research undertaken is a design and fabrication of a robot prototype that has been improved as compared to existing manipulator control systems by the other researchers, for instance, JICA report (2007), PID for line following, Techbitor (2012), Robot built by use of arduino, uno, adafruit motor shield, pololu's, QTR-8R line sensors, Islam and Rayman (2013), Built robot by use of op-amps and transistors, Walaa, Sheba, Elnemr and Gamal (2014), Consists of webcam mounted

on a vehicle with PID and a PIC16F877 Microcontroller and a D.C motor and Sheikh and Rakesh (2013), Constructed robot by several sensors fabricated on other arduino duamilanove ATmega by incorporating sensors in its design for the provision of real-time feedback for affecting adaptive correction of the D.C motor's switching sequence and motions and hence, the line follower robot's manipulator trajectories. The beauty of this system is that it is reprogrammable hence any improvement on it is done simply by updating the software and is also suitable for innumerable other applications. Once the autonomous robot has been adapted it will relief people off their routine or repetitive work and hence, enhancement of the knowledge on how to make robot.

CHAPTER TWO

LITERATURE REVIEW

Humans use control systems to extend their physical capabilities, to compensate for their physical limitations, to relieve them of routine or tedious tasks or to save money. In order for a line follower robot manipulator to traverse through its programmed motions, a reliable and efficient control system is required. As a consequence, refinement in robotic control systems has been a subject of many researchers such as;

Russell, (2001), International Journal of Robotics and Waurzyniak, (1999), Manufacturing Engineering Journal, Robotics Evolution, in this field with a view of helping to boost robot's accuracy and also to design and construct a reliable line follower control system with high degree of freedom.

Thilakshan (2010) used an inertia measuring unit of an accelerometer and gyroscope for measuring acceleration and angular velocity. The output from the sensor was sent to a microcontroller. An algorithm was developed and programmed to the microcontroller to translate the sensor data into information on orientation and position movements of attitude and displacement. The project needed improvement because it could perform in three –dimensional space due to the missing attitude angles required in the rotation matrix computation. Also time integration of inertia sensor data leads to errors and increased position uncertainty

Techbitar (2012) used arduino uno, adafruit motor shield, pololu's QTR-8RC line sensors. He said one could build a cheaper and lighter version of robot using the Atmel atmega 328 and the L293D h-bridge.

Sheikh and Rakesh (2013) the wheel movement of robot was controlled by the use of several sensors and D.C motors. For robot, steered by the motors to move along the line smoothly, ATmega microcontroller performed and implemented Pulse Integrated Derivative (PID) algorithms to control the speed of the motors. Response was better than open loop controller. The tuning utilized was the manual tuning method. Due to limitations in the hardware (motors and sensors) perfect control was not obtained.

Sushil (2013) the robot used arduino duemilanove ATmega 328 which received information from the sensors and converted them into digital values using ADC of the microcontroller. He compared the result and generated output to the motor to keep it in track. The robot was Two-wheeled driven by a motor IC circuit. The line track was determined by sending an infrared signal to the track and photo-transistor used to sense the infrared signals. Thus, the robot was able to solve a maze of which it had no more information than that the track was in black and the background was white.

Islam and Rayman (2013), robot was made by use of op-amps and transistors, due to the motor speed of rotation the speed was ON and OFF using the output signal from comparator. The robot used two line sensors, so the line could not be tracked due to fluctuation of the line. Therefore they recommended that the techniques of using comparator could be replaced by Pulse Width modulation (PWM) using more sensors, microcontroller and H-Bridge motor controller IC(L293D), also, instead of Light Dependent Resistor (LDR), Phototransistor could be used. Five sensor array may be used to detect the black/white line. However, robot could track the black line and carry some load likely 500g.

Walaa, Sheba, Elnemr and Gamal (2014), the robot consists of webcam mounted on vehicle. A Pulse Integrated Derivative control algorithm was used to adjust the robot on the line. Microcontroller PIC16F877 and DC motor were used to provide control signal and steer the robot wheels. They used the camera to take the surrounding environment image which was to be processed through the MATLAB environment to produce an output signal informing the microcontroller the location of the line with respect to the robot. Digital image processing techniques was used which was robust against environmental factors such as darkness, lighting, camera distortion as well as a line color.

CHAPTER THREE

THEORETICAL CONSIDERATION OF COMPONENTS

3.1 Analogue Robot Control Systems

The Analogue Robot generally utilizes the switching devices such as relays, thyristors (SCR), triacs, diacs, as control elements. These systems are basically of two types according to a publisher John (2000) on introduction to robotics, mechanics and controls: (a) The air logic controller

(b) The drum controller.

The air logic controller is made up of small pneumatic valves and timers connected together with small pieces of tubing. The sequence of hoses hooking the valves and timers together controls the opening and closing of the robot's main valves connected to the motors thus moving the manipulator. This controller is exclusive to pneumatic robots.

The drum controller is the classical type of controllers used in robotics. Though it is simple and reliable, it is, however, limited only to controlling motions of the pick-and-place robots but cannot be used for controlling point-to-point, controlled path or continuous path robots. It is similar in design to a music box and has hundreds of holes in it. Small pegs are inserted into the holes and as the drum rotates, the small pegs close switches wired to hydraulic valves, pneumatic valves or electric power drivers. When a switch closes, the valves are opened or the contacts of the electric power drives are made and the robot's manipulator moves. After one movement is completed, the drum is advanced one step, another peg closes another switch and the manipulator moves again. Though the use of electromechanical

devices such as relays in this control system and offer many of the desirable characteristics of an 'ideal' switching device, they, however, have several shortcomings which limit their use in control application.

Notably, amongst the disadvantages of simple electromechanical relays is their inherently low switching speed coupled with the 'contact bounce', which occurs during the transitory state existing between the 'on' and 'off' conditions. Furthermore, electromechanical relays are by virtue of their moving parts and open contacts prone to failure when compared with their modern digital counterparts.

There is also the arcing, which may form between the contacts when they break resulting in the generation of heat and radio frequency interference (RFI), hence affecting the control system's performance. To minimize such effects, analogue control systems normally require the use of heat sinks and L-C filters. Due to these, therefore, analogue robot control systems have been observed to be costly, lack reprogrammable memory and have low noise immunity.

3.2 Digital Robot Control Systems

This control scheme is generally implemented in two ways:

- (a) The programmable robot controller;
- (b) The microcomputer robot controller.

The programmable robot controller is an electronic version of the drum and the air logic controllers. The memory of the programmable controller is stored electronically rather than in the pegs of a drum controller or the airlines of an air logic controller as a result of Nnaji (1993) who wrote a theory of automatic robot assembly and programming. Figure 3.1 shows a block diagram of digital control technique of a robot motor, which governs the positional movements of a

manipulator. Positional sensors with wheels and light arrangements coupled to the motor of the robot give electrical pulses when the motor rotates which are counted when a transistor switch is made. These pulses are compared with the control inputs and when the count is equal to the control inputs, the motor rotation stops and the counter resets automatically. The AND gate which gets one input from the transistor switch and another from the control command inputs ensures that the counting starts at the instant the manipulator moves. The interface used ensures that the manipulator through the motor drivers communicate with the control circuit as explained by Owade (1998) in his design and development of a programmable laboratory interface system research.

Though this control scheme is better than the drum or the air logic controller due to its possibility of achieving a high degree of accuracy, it is limited to simple pick-and-place robots and lack reprogrammable memory thus system's flexibility is compromised. Microcomputers on the other hand are available for performing the computations necessary in a complex control system. Researchers and industrial robot designers like Andrew (2001), Pascal (2005), researched on path following system and intelligent line following robot respectively have been adapting digital computers to control robots. An industrial robot can be controlled on-line by a digital computer as shown in Figure 3.2.

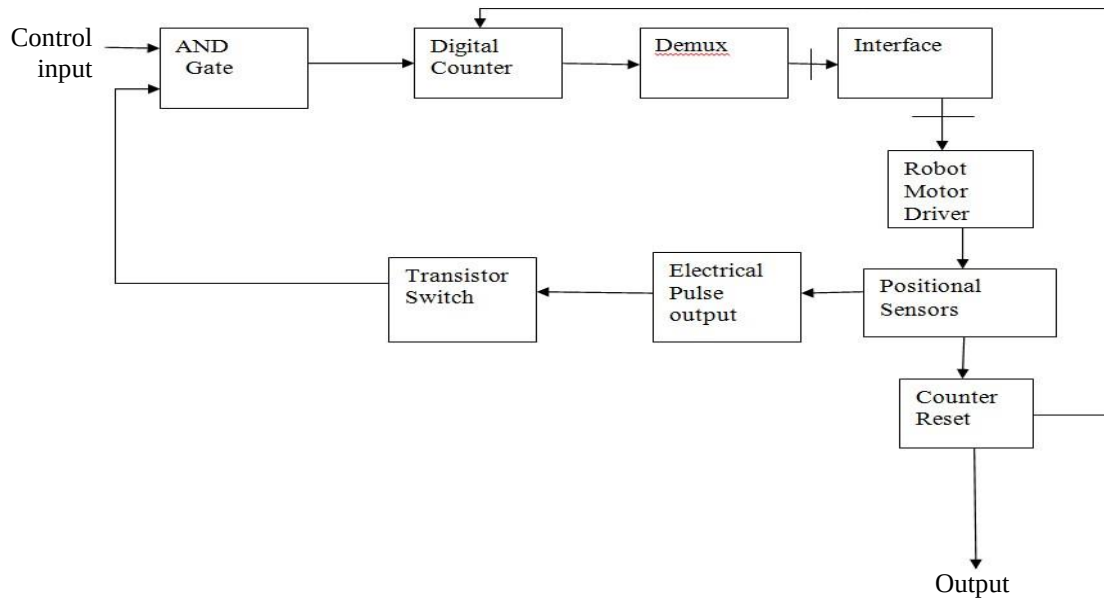


Figure 3.1: Block diagram of digital control technique of a robot motor

The ADC device samples the device's input signal at some sampling instants and the process is completed with the analogue signal value being converted to a discrete digital value and fed into a digital computer. The output of the microcomputer is fed into a specialized electronic control circuit to control the robot's manipulator.

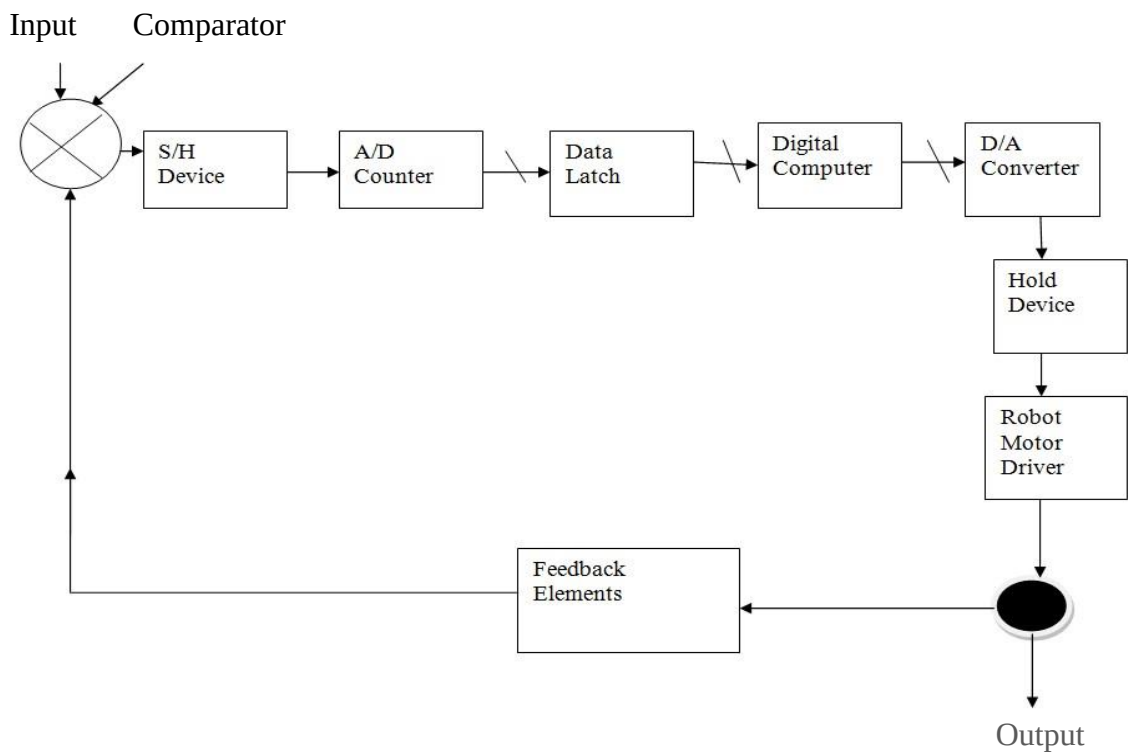


Figure 3 .2: Block diagram of closed-loop digital computer robot control system.

The DAC performs the reverse conversion process generating an analogue control signal. This signal opens and closes the valves or switches of the manipulator power drivers, hence moving the manipulator accordingly. Whereas the use of a digital computer as a control element permits control that is more accurate in general, it has also been observed to have the following weaknesses in some real-time control applications.

- (a) Digital control systems are costly and generally more complex to design due to the large number of circuitry involved.
- (b) Normally, the best-reconstructed analogue signal from the digital signal using the hold device is only an approximation of the input analogue signal hence a loss of signal information occurs affecting the system's performance.
- (c) The Analogue to Digital converter (ADC), Digital to Analogue converter (DAC), Sample and Hold (S/H) device, the digital computer and other circuitry components involved generally delay the control signal output; hence the system's speed of operation is constrained.

The limitations, of analogue and digital robot control systems suggested that a more reliable system capable of performing various computing functions and making decisions to change the sequence of program execution needed to be developed. Line follower robots such as those used by General Motors in vehicle assembly line which are not yet essentially different from mechanized mills require a cost-effective control system for its revolution, which also facilitates increased productivity, flexible automation, accuracy and repeatability. Most recent line follower robots come with an array of sensors such as heat sensors, infrared range finder, touch sensors, acceleration and speed sensors and thus again suggesting a dire need to have modern control system.

A microcontroller is the most suitable candidate, capable of catering a multiplicity of input command signals, receiving a multiplicity of sensor-based feedback signals and largely processing these signals digitally and therefore offers solutions to problems inherent in analogue / digitally robot control systems. The invention and wide applications of microcontroller(s) have changed the philosophy of instrumentation, signal processing and control engineering fields. Its application is limited more by the imagination of their users than by their technology. Microcontroller-based systems have replaced the conventional ones based on standard analogue and digital computing equipment because of the following advantages.

- (a) Increased flexibility of the control programs.
- (b) Decision-making and logic compatibility of the system's components involved.
- (c) Program can be modified to accommodate design changes or adaptive performance without any variations on the hardware used.

Though already have embedded systems based on purely analogue / digital components, while others are based on other microcontroller for control of line follower robot manipulators, additional performance improvements in my design are as follows:

- (a) Employing D.C motors which respond to discrete signals to drive manipulator as opposed to servo motors hence avoiding the use of ADC and DAC in my system hardware design and consequently minimizing the hardware count used hence the system's cost.
- (b) Employing work piece and light feedback sensors to identify the right type of material to be picked and determine the proper phases of the

D.C motors to switch at proper timings using work piece motor driver and light sensing system respectively.

3.3 The Sensor

A sensor is a device that measures a physical quantity like speed or pressure and converts it into a signal that can be measured electrically. There are many sensors that can be used for a simple application like line following such as IR LED and a photodiode, LED and LDR. For instance,

3.3.1 Image sensors - these are in digital cameras, camera modules and other imaging devices based on charge coupled device (CCD) or crystal metal oxide semiconductor (CMOS) technology.

3.3.2 Light sensors - can be included in the proximity sensor category and it's a simple sensor that changes the voltage of photo resistor or photovoltaic cells in concordance with the amount of light detected. A light sensor is used in very popular applications for autonomous robots that track a line-marked path.

3.3.3 Color sensor - Different colors are reflected with different intensity for example the orange color reflects red light in an amount greater than the green color and this is the color sensor. This sensor is in the same range with light sensor, but with a few extra features that can be useful for applications where the robot has to detect the presence of an object with a certain color, or to detect the types of objects or the surfaces.

3.3.4 The touch sensor - can be included in the range sensors category and are designed to sense objects at a small distance with or without direct contact. This

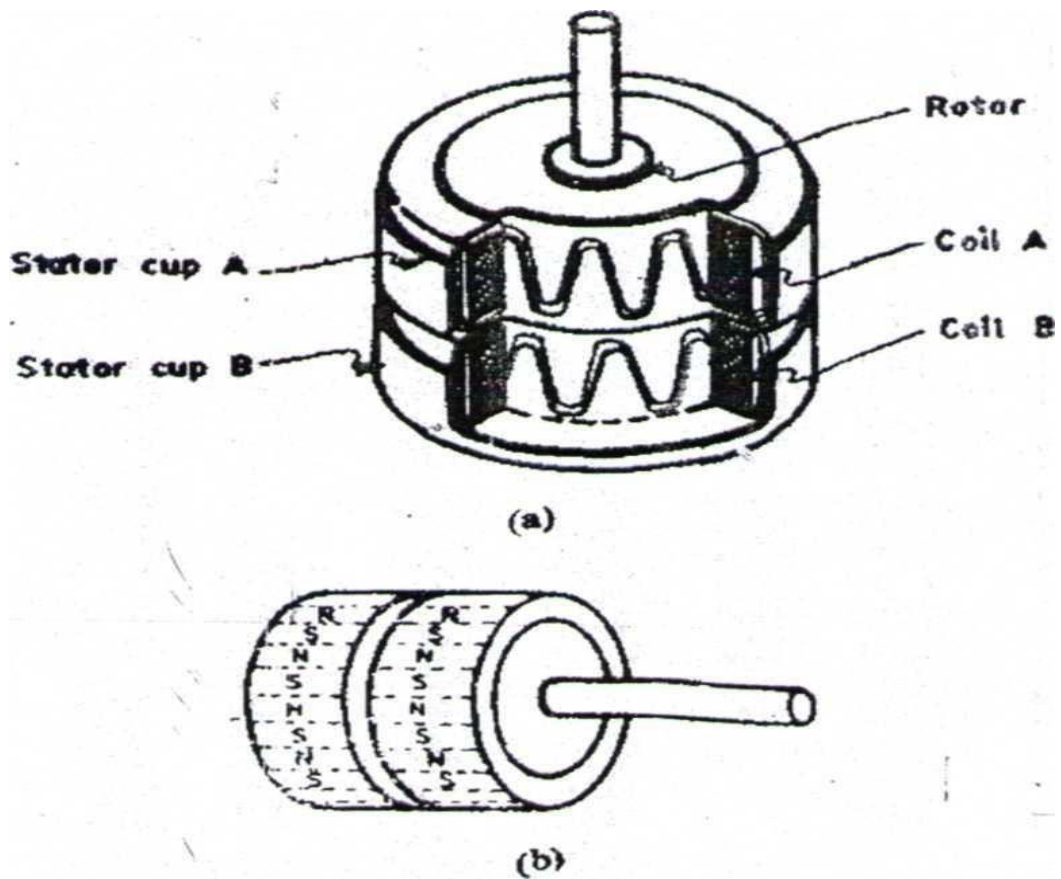
sensor is designed to detect the changes in the capacitance between the on-board electrodes and the object making contact.

3.3.5 The ultrasonic sensors - are designed to generate high frequency sound waves and receive the echo reflected by the target. These sensors are used in a wide range of applications and are very useful when it is not important in the detection of colors, surfaces texture, or transparency.

3.3.6 The infrared sensor (IR) - Measure the IR light that is transmitted in the environment to find objects by an IR. This type of sensor is very popular in navigation for object avoidance, distance measured or line following applications. This Sensor is very sensitive to IR lights and sunlight and this is the main reason that an IR sensor is used with great precision in spaces with low light.

3.4 Direct Current Motor

This has a stator with a number of teeth on it, which carry the excitation current, and a rotor in the form of a permanent magnet having a number of poles equal to the stator teeth. A 2-stack (2-phase) PM motor has two stator cups having two stator windings each of which produces a number of poles as shown in Figure 3.3(a), 3.3(b) and 3.4 below as explained by a researcher Peregrinus P.A (1998) on a guide to modern theory and practice of D.C motors. A feature of the PM motor is that even when the motor is unpowered a torque has to be applied to the shaft to displace it from the rest position. This is known as the detent torque of the motor and is due to the attraction between the permanent magnet rotor poles and the residual magnetic poles on the stator.



Figures (a) 3.3, and (b) 3.3: Cut-away diagram and side view of a two-phase PM motor.

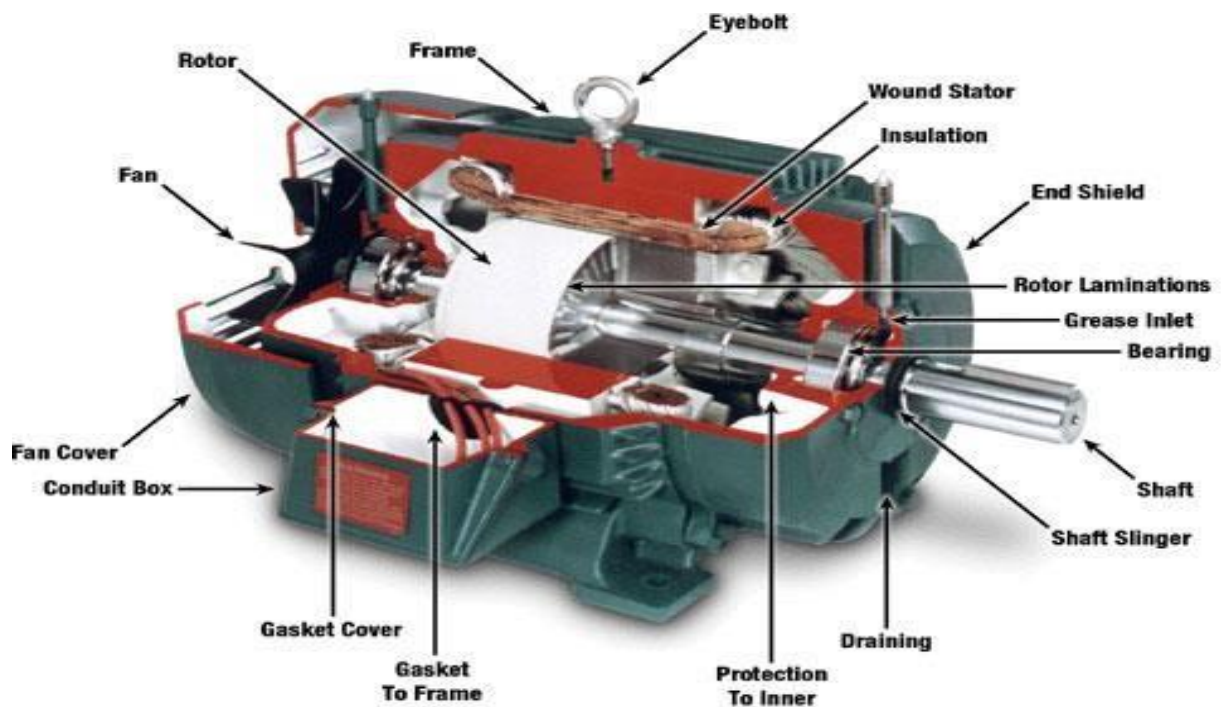


Figure 3 .4: Rotor magnetization (<https://www.en.org/>; D.C motor)

When the coil is powered, a magnetic field is generated around the armature. First, the left side of the armature is pushed away from the left magnet and drawn toward the right, causing rotation. Second the armature continues to rotate. Third, when the armature becomes horizontally aligned, the commutator reverses the direction of current through the coil, reversing the magnetic field. The process then repeats. If the shaft of a DC motor is turned by an external force, the motor will act like a generator and produce an Electromotive force (EMF).

During normal operation, the spinning of the motor produces a voltage, known as the counter-EMF (CEMF) or back EMF, because it opposes the applied voltage on the motor. This is the same EMF that is produced when the motor is used as a generator (for example when an electrical load (resistance) is placed across the terminals of the motor and the motor shaft is driven with an external torque). The CEMF is proportional to motor speed. When an electric motor is first started or is completely stalled, there is zero CEMF. Therefore the current through the armature is much higher. As the motor spins, the CEMF increases until it is equal to the applied voltage, minus the parasitic voltage drop.

For this research D.C motor has been used because the rotational speed of a DC motor is proportional to the voltage applied to it, and the torque is proportional to the current. Speed control can be achieved by variable battery tapping, variable supply voltage, resistors or electronic controls. The direction of a wound field DC motor can be changed by reversing either the field or armature connections but not both.

3.5 Microcontroller

A microcontroller is a microcomputer on a single chip and is a multipurpose programmable logic device that reads binary instructions from a storage device called memory, accepts binary data as input and process data according to those instructions and provides results as output. The invention and wide use of microcontroller have changed the philosophy of instrumentation, signal processing and control engineering fields.

The microcontroller is capable of performing various computing functions and making decisions to change the sequence of program execution and is therefore suitable for highly specialized applications. It acts as a control center for all operations and executes instructions contained in memory. The basic operations of the microcontroller include the transfer of data and instructions between itself and memory, the manipulation of data in the memory and the transfer of data between itself and the input/ output devices

3.5.1 The Organization of Microcontroller-Based System

Microcontroller-based system is a product or machine that uses a microcontroller to run or execute its operations. It is represented by the following components: input / output, memory and microprocessor devices as in Figure 3.5. These components are arranged in a bus, which are wires that carry bits.

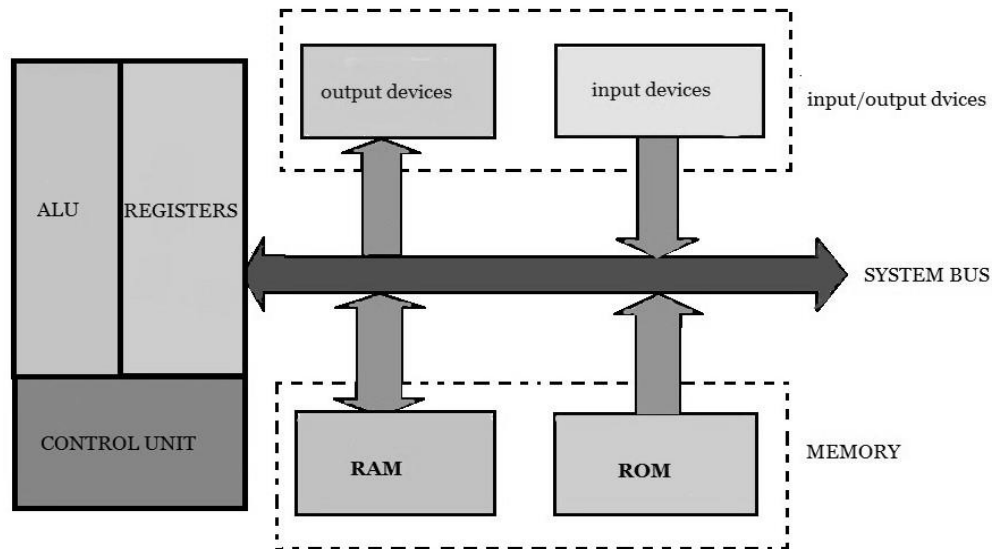


Figure 3.5: The organization of a micro controller-based system

3.5.2 Microcontroller Applications

Microcontrollers are applied in three principal ranges of operation namely: control, calculation and administration. Its primary function is to fetch, decode and execute instructions resident in memory pictured clearly by Atmega 328 datasheet (2012).

Many applications that previously employed hardware logic have been made feasible by the use of microcontroller. Specific applications among others range from TV games, process control, ATM banking systems, automobile industry in controlling fuel delivery and medical operations control. In process industry, microcontroller is now used for implementing a number of process-control algorithms as economic alternatives to the classical analogue controllers. In the automobile industry, a few recent car models include microcontroller for ignition timing, sensing and carburetor control to improve efficiency.

Some microcontroller are more suited for particular applications than others, for instance, 8-bit microcontroller such as, the Intel 8051 or Motorola MCB 6801 are suited for low cost process control applications. On the other hand, 32 -bit microcontroller such as 1280, Pentium III / IV and microcontroller Arduino ATmega 328/168, are more suited for complex applications.

In this research work, Arduino duamilanove ATmega 328 microcontroller was used to control Two 12V direct current motors for controlling the motions of a line follower robot's manipulator model with two degrees of freedom designed from aluminum sheets.

3.5.3 The Arduino duamianove AT mega 328 Microcontroller

The ATmega microcontroller" Duemilanove" 2009 in Italian and is named after the year of its release. The Duemilanove is the latest in a series of USB Arduino boards; for a comparison with previous versions.

It has been used in this research study because of the following features. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button.

It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The ATmega328 has 32 KB of flash memory for storing code, (of which 2 KB used for the boot loader). The ATmega328 has 2 KB of SRAM and 1 KB of EEPROM. The microcontroller stores, executes, processes and outputs the data

signals in digital form for control of line follower robot. The digital bits of the control signals are stored in RAM cells, which serve as input signals for automatic operations.

3.6 Computer programming language

There are four language levels that can be used to write a program for a microprocessor; machine language, assembly language, high-level language and Arduino C language in wiring. The Arduino Duemilanove can be programmed with the Arduino software.

The ATmega168 or ATmega328 on the Arduino Duemilanove come preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol. You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header.

3.7 Control theory

According to Waurzyniak (1999) during Robotics Evolution Journal, control theory deals with the dynamic response of a system to commands or disturbances. If a system is stable, it will eventually settle down to an equilibrium position. The behavior of the system prior to this settling down is called the transient response.

An unstable system on the other hand will never settle down. Mathematically, its transient, response will continue indefinitely, physically, it will continue until constrained by some physical limitation such as motor saturation, or by damage to the system itself. The application of control theory has two phases: dynamic analysis and control system design. The analysis phase is concerned with the determination of

the response of a plant to commands, disturbances and changes in the plant parameters.

If the dynamic response is satisfactory, there need be no second phase, if unsatisfactory and modification of the plant is unacceptable, a design phase is necessary to select the control elements needed to improve the dynamic performance to acceptable levels.

Control theory itself has two categories, namely: classical control techniques and modern control techniques. Classical control techniques are characterized by the transfer function concept with analysis and design principally in the Laplace and frequency domains. Frequency domain is a method of analysis particularly useful for fixed linear systems in which one does not deal with functions of time explicitly but their Laplace or Fourier transforms which are functions of frequency. Modern control techniques have emerged with the advent of digital computers and the highspeed microcontroller. They are characterized by the state variable of concept with emphasis on matrix algebra, analysis and design principally in the time domain.

3.7.1 The Transfer Function Concept

Classical control theory is based on the input-output relationship principally the transfer function concept. If the input-output relationship of a linear system is known, then the characteristic of the system itself is known.

Most motors have internal damping that is viscous in nature and arises from the induced currents or back electromotive force generated by the rotation of the

armature in the presence of magnetic fields. This damping is represented as a negative feedback voltage proportional to the angular velocity of the output shaft ($s\theta_{out}$) in the Laplace domain. The block diagram simulating this control scheme is shown in figures 3.6 (a) and 3.6(b)

Where: $K_m = 1/B_m$, $r_m = 1/K_m B_m$

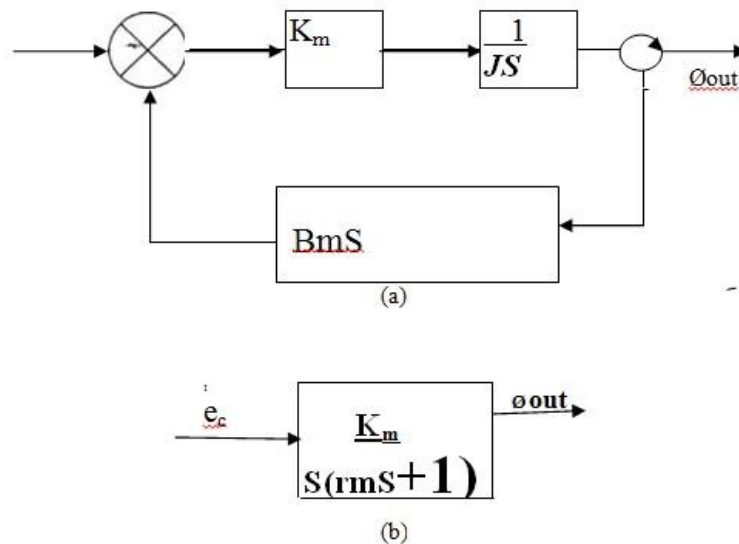


Figure 3.6: (a & b): Block diagram of DC motor used as a control component with negative feedback

Here, (a) is reduced to (b) with the same transfer function.

Thus:

$$\frac{\theta_{out}}{s e_c} = \frac{K_m}{s(r_m s + 1)} \dots\dots\dots (3.1)$$

Where K_m is the effective motor gain, r_m is the time constant.

Equation (1.1) shows that the motor now acts as an integrator with a first-order time

lag. Since the motor speed is $\frac{d\theta(t)}{dt}$ equation (1.4) can be written as

$$d\theta(s)/e_c = K_m/(r_m s + 1) \dots\dots\dots (3.2)$$

If a constant voltage is applied, the motor will settle down to a constant speed of time period of the order of the motor time constant (r_m). The expression for (r_m) shows

that it is inversely proportional to the internal damping (B_m). When the internal damping is too small to provide the desired output, it can be augmented using a tachometer to feedback voltage proportional to the motor speed.

Although the techniques of classical control theory are powerful and relatively simple, they do have limitations and shortcomings that multiply as plants and control systems become more complex. For instance, as the number of inputs and outputs increases, the number of transfer functions needed to describe the system increases drastically. A system with ten inputs and ten outputs requires one hundred transfer functions with the classical theory and only a single matrix vector equation with the modern theory.

Initial conditions must be added and treated separately in the classical approach whereas they are automatically included in the modern approach. Modern control theory makes provisions for both the inclusion of detailed and varied performance and the direct design of controllers by synthesis in contrast to the trial and error design and performance evaluation techniques of classical control theory. In principle, modern control theory has no inherent limitations.

3.7.2 Types of Control Systems

Control systems are classified in terms that describe the system itself or its variables. They are therefore either open-loop or closed-loop control systems. An open-loop control system shown in Figures 3.7 and 3.8 is characterized by the input entering directly into the control elements unaffected by the output; the output is related to the input solely by the characteristics of the plant and the control elements.

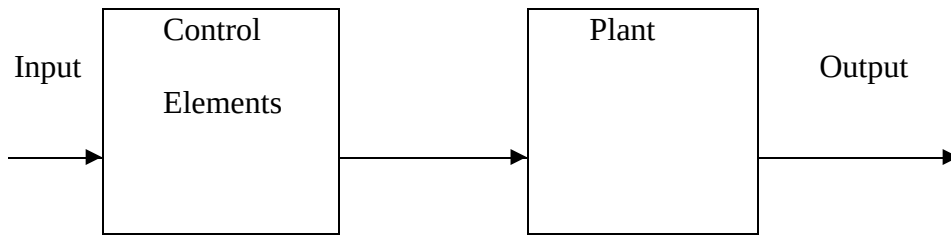
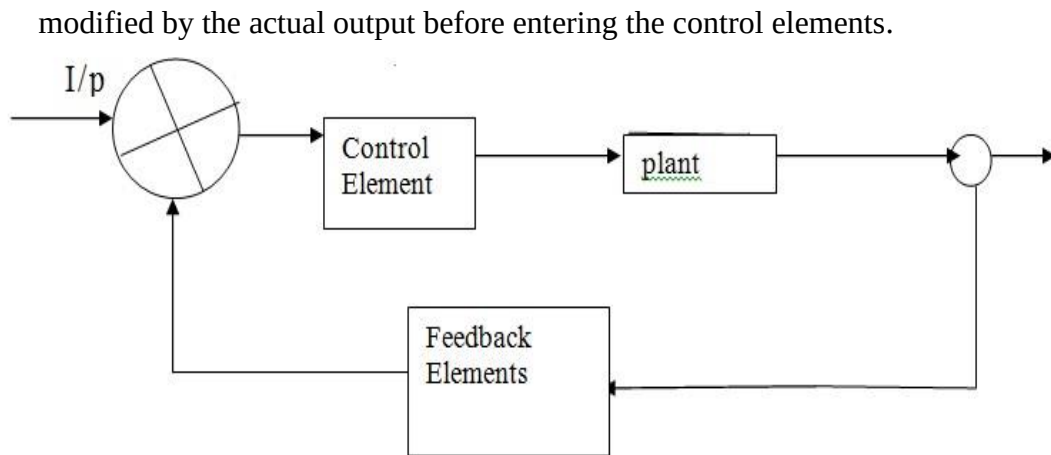


Figure 3.7:
Block
diagram of
open-loop
control
system



I
n

t
h
e

c
l

osed-loop control system, shown in Figure 3.7, however, the input is

Figure 3.8: Block of closed-loop system

A Line follower robot can be controlled using either of the two methods. However, this discussion focuses on the closed-loop to control a line follower robot with two major degree of freedom realized by the use of four 12-V direct current motors. In this system, suitable types of motor drivers are coupled to the output motion to generate a feedback signal exactly proportional to this movement. This signal is in

opposite sense to the input signal and connected to a signal comparator in the micro controller.

The microcontroller then receives the difference between the input and the output signals, which decreases in strength as the output approaches its command position. When the input and the feedback signal are equal and opposite, there is no further movement of the output movement. A decreasing drive signal achieved in this way as the motor motion slows down as it approaches its final position reduces the tendency of the robot overrunning and hitting its final position.

3.7.3 Fundamentals of Automatic Control

Automatic is a technique that measures a variable value and provides a counter response to limit its deviation. The automatic control is understood by the advantages it provides on the controlled energy. Closed-loop control is a good example of an automatic control in that, the production achieved is more economical. This is why some processes cannot take place without this kind of control.

Line follower robot applications provide a clear specification of problem area with various pragmatic requirements of automatic control systems that help to constrain and hopefully simplify the eventual solution. A line follower robot arm is normally employed in production lines for assembling parts, radiopharmaceuticals handling during radio-emission experiments, mixing combustible chemicals, lifting objects, inspecting components in an austere environment.

Automation of line follower robot using microcontroller and in particular using an 16-bit microprocessor as the "heart" of the control system thus leads to economical production achieved in several ways which include among other things lowering

labor cost, eliminating or reducing human errors, improving process quality, reducing the size of the process equipment and the amount of space it requires and minimizing energy consumption.

CHAPTER FOUR

METHODOLOGY

RESEARCH METHODOLOGY

The methodology adopted in this research work is based on some of the system designs sequence presented and is illustrated by means of a block diagram shown in Figure 4.1. The autonomous robot system designed consists of two parts, namely: the system hardware and the system software designs.

This chapter will describe the various components, operation and performance of each of designs. In particular, the interfacing of DC motors, sensors, the operation of the various components used and the overall performance of the microcontroller system will be discussed.

In this discussion, the D.C motors utilized together with aluminum sheets to fabricate the mechanical body obtained from Nairobi, Kenya whose specifications will be given later in the chapter.

4.1 Selection of hardware components used

The appropriate hardware components for the system design were selected on the basis of the following factors:

- (i) Logic family compatibility
- (ii) Operating temperature
- (iii) Current and voltage ratings.

For the construction of an autonomous line follower robot system, the following hardware components were used: Arduino Duamilanove AT mega 328 microcontroller, project board, H-bridge (Dual H-bridge IC L298N), 12 V 7 Ah D.C battery and D.C motors.

The range of the Object and Obstacle is estimated by the variable resistor(0-100 k Ω) .Sharp infrared range sensors , line tracking sensors, a 5 V regulator (7805), and IN 4002 diode .For gripper, one normally open switch was used, one 10 k Ω resistor, 3 gears , a D.C motor, rubber materials and some aluminum sheets. For mechanical arm, beam of aluminum rods, screws and nuts and a D.C motor were used. The motion was facilitated by two wheels each coupled to a D.C motor and a castle wheel for a differential drive.

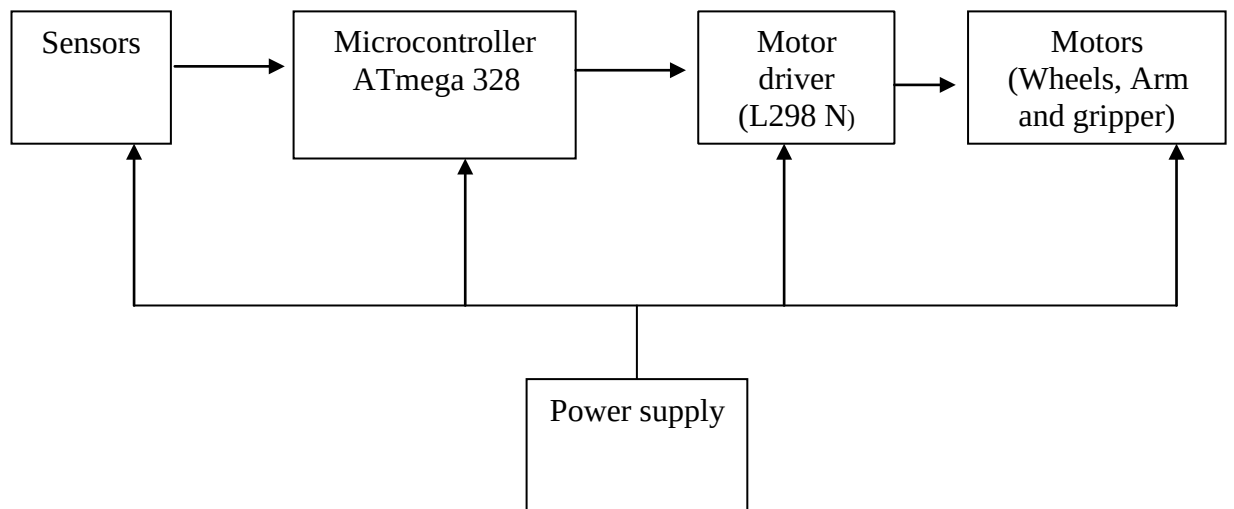


Figure 4.1: Robot design block diagram

4.1.1 Hardware (Physical Components)

The aluminum sheets and plywood formed the casing of the robots. The sensor provides the input signal to the microcontroller. Microcontroller processed the input signal and output a digital signal to the motor driver which output an analog signal to the inputs signal of the motors. The motor drives the base wheels to the desired directions. Gears and rubber bands support the gripper in terms of increasing the torque and friction respectively when gripping and holding the object.

4.2 Circuit design and Simulation

The system circuit was designed with the help of MCS-86™ system design kit software, and cable interface, simulated completely where possible using the

electronic workbench and circuit maker software. This helped greatly in minimizing errors, which could have otherwise been transferred to the constructed system.

To develop the software the requirement of the problem was clearly defined which included defining the pins of the programmable input/output devices and finally writing subroutines in ATmega 328 Arduino C language for moving the robot in a user defined direction. The subroutines were then combined with the main program for the overall control of the manipulator.

4.3 Robot construction

The Autonomous line follower Robot Construction involved fabrication of mechanical body, arm, gripper, gearing system, electrical and electronic control system.

4.4 System construction, Testing and Troubleshooting

The autonomous robot designed circuit was constructed on the project board and analyzed for performance. The troubleshooting of the hardware system was carried out with the help of multi-meter and the cathode ray oscilloscope. A small test program shown in (4.5 Design Specification) was written and used to test the operation of the whole constructed hardware system. Any error encountered in this stage was rectified.

The software errors arising during testing were debugged until the whole process was error-free and then the system hardware and software were integrated and analyzed. The autonomous system constructed was used to train 12 V D.C motors coupled to the wheels, arm and gripper via gears fabricated from aluminum sheets for the purpose of controlling its movements on the track. The movement of these motors in user-defined directions and speed automatically successfully controlled the

constructed robot model on the defined direction. The robot's end – effector for the purpose of picking objects components (for this case the gripper) suggests by extension the command of a robot by the user to do so in the same sequence and interval predetermined. The automatic and repetitive operation of our system is simply done by changing the mode control switch.

4.5 Design specification

The design specification for autonomous line follower robot involved use of algorithms as a design tools.

Algorithm: line tracking, picking and dropping

L= leftmost sensor which reads 0; R= rightmost sensor which reads 0. If no sensor on left (or Right) is 0 then L (or R) equals 0;

If all sensors read 1 trace line,

Else,

If L=1 Move right

If R=1 move Right

If L&R=0 Move Forward

Move clockwise if line was last seen on Right Move

counterclockwise if line was last seen on Left

Repeat trace the line till line is found.

Else,

Stop.

4.6 The system hardware design

4.6.1 The mechanical hardware

The mechanical manipulator robot chassis model for control was constructed using aluminum sheets. Figure 4.2 shows gripper circuit sensor. When the switch of the

gripper sensor is open the output is high and if the switch is closed the output is low. The motors control the robot model in three different movements, namely; the robot following the track, lifting of the arm, opening and closing the gripper. The arm was interfaced to the gripper and gripper coupled to the DC motor using gears both fabricated from the same type of sheets.

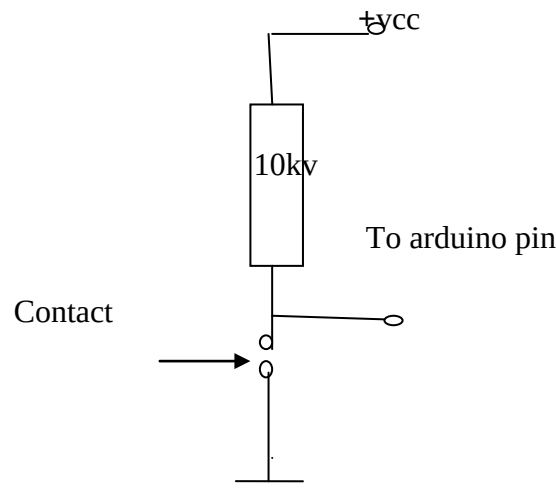


Figure 4.2: Designed circuit of gripper sensor

The specifications of the D.C motors used in our interface design used in this research are;

Base motors (2) – MOTOR AP 68 manufactured by Alps Electric co. Ltd. Japan.

Input: 12VDC

Resistance: 33Ω

Arm motor– DC MOTOR BP-485725 manufactured by Brother Industries LTD

Input: 12VDC

Resistance: 36Ω

Gripper Motor- PULSE MOTOR KHL-55 MOI manufactured by Oki Electric

Industry Co Ltd.

Input: 12VDC

Resistance: 36Ω

The drive wheels are driven using two D.C motors and powered from 12V 7Ah DC battery. The motors are connected by a dual H-Bridge module (L298N) which will

switch clockwise or anticlockwise. The rotations and switching of the motor are controlled using a microcontroller. The base motor used in our system, is used to drive the robot wheels to the defined directions on the track which are interfaced to the microcontroller via motor driver as shown in figure (4.3)

4.7 Other hardware of the control system

The system comprises of sensors, motor drive control, microcontroller and power control. Modules have been soldered carefully to avoid cold joints and damaging of the circuit components.

4.7.1 Sensors

In the design two types of sensors have been used: The proximity (range) sensor and line tracking sensors.

The line tracking sensors are based on natural light, that read the track enhancing navigation of the robot. The proximity sensor used is sharp infrared range sensor (GP 2Y OA 21YK). The sensor aids in determining the closeness to the object. The line sensors (photodiode) will intelligently note position of the line drawn on the ground underneath the robot, and assist the robot to move along the line and intersections. In order for the microcontroller to perform efficiently the in build sensor circuit should give outmost number of information of that track and of the object.

4.7.2 Motor Drive

The performance of electric motor is decided by a motor controller device. The motor driver is always meant to guide and control the motor in various ways such as in the manner of starting either manual or automatic, selection of rotation direction, controlling and regulating speed or torque and protecting against faults and overloads.

The motor driver L298N is a dual H-bridge and therefore can be used to control two D.C motors to rotate clockwise and anticlockwise directions. The L298 N can output 600 mA and 1.2 A peak value. The drive wheels are driven using two D.C motor and powered from 12 V 7 AH D.C battery. The motors are connected by a dual Hbridge module (dual L298N motor driver carrier an I.C) as in figure 4.4, which shows pin details for connections. The rotations and switching of the motor are controlled by the Arduino ATmega microcontroller.

Motor driver take the input signals from the microcontroller and generate corresponding output for motor, since it's a current enhancing device.

In this case, the motor driver acts as a switch device. For this research L298N motor driver is utilized. This motor driver is used to drive two motors simultaneously. DC motor uses 6V and 6V to 12V drives the gears which is decided by the ratings of the motor Supply voltage and logically, determine what input voltage is, either, high or low. Typically, if the supply voltage is set to be +5V, then 0.2V to 1.4 will be considered as low input and 2.4V to 5V considered high voltage.

L298N has 2 Channels .One channel is used for one motor.

Channel 1-Pin1to 8

Channel 2-Pin 9 to 16

Enable Pin (or inhibit pin) make a channel active.

All Input (Pin No.2, 3, 13 and 14) of L298N IC is the output from microcontroller (ATmega328). For instance.-I connected (Pin No. 2,3,13 and14) of L298N IC to (Pin No. 5,6,7,10,11 and12) of ATmega328 respectively in our robots, because on pin 10 and 11 of ATmega328 PWM can be produced. All Output (Pin No. 2, 3,13and 14) of L298N IC goes to the input of Right and Left motor.

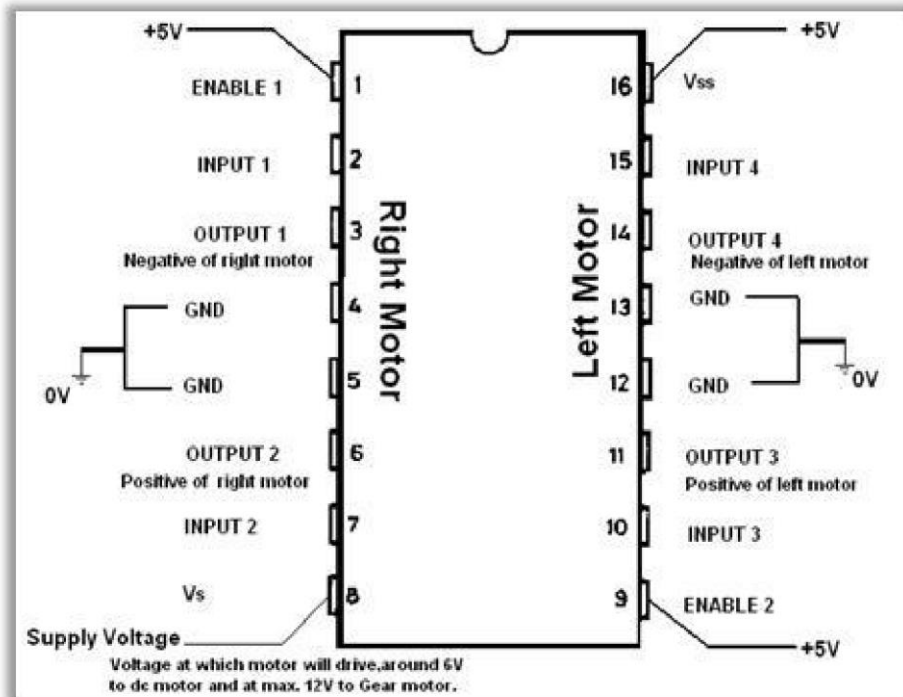


Figure 4.3: Pin details of L298N Output Connections

- OUTPUT 1 (Pin No 2) --- It is the negative Terminal of Right Motor
- OUTPUT 2 (Pin No 3) ---It is the positive Terminal of Right Motor
- OUTPUT 3 (Pin No 13) --- I t i s t h e positive Terminal of Left Motor
- OUTPUT 4 (Pin No 14) --- It is the negative Terminal of Left Motor

4.7.3 Microcontroller

Microcontroller is the brain of the line follower robot model. The microcontroller board used here is arduino duamilanove ATmega 328. The arduino ATmega is powered via the USB connection or with an external power supply. The power source is selected automatically. The board can operate on an external power supply of 6 to 20 volts. The ATmega 328 used in our work, has 126 KB of flash memory for storing code (of which 2KB is used for the boot loader). The ATmega 328 has 2KB of SRAM and 1KB of EEPROM (which can be read and written with the EEPROM library). Figure 4.5 shows the ATmega 328 board.

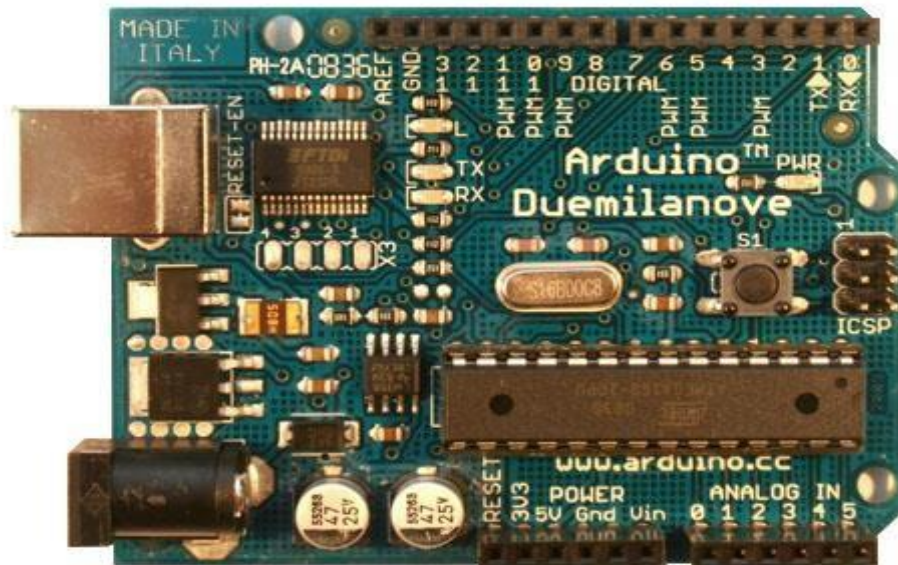


Figure 4.4: ATmega 328 board

4.7.4 System Software Design

The program code is decision -maker in the micro-controller which decides about the outputs for particular set of inputs. The line follower program is written using Arduino C language and uploaded by USB to the microcontroller. The program is then compiled to form a “.hex” by java file which is then uploaded into the microcontroller.

Note, the automatic operation cannot be executed if the robot does not come to its initial position at the end of the task.

4.7.5 Software Solutions

The system’s software solution writ-tern in Arduino C language using ATmega 328 instructions set is given in Appendix A. The system’s control monitor program stored in EPROM is straight forward and may be understood easily with the help of a flow chart given in figure 5.7.

4.7.6 Power control

Different circuitry uses different values of voltages hence a need for a power control to reduce and regulate the power from 12V7Ah D.C battery. The power regulator board used has eight 5V D.C power points that cater for the whole electrical and electronic control system

4.8 Input Circuit

A Dual IR sensor are used by robot to sense the line that is Right (RX), Center (C) and Left (LX) as shown in figure 4.7 and 4.8. The three IR sensors face the ground as per the setup. Depending on the amount of light reflected back, an analog signal is received at the output. The bits (1, 0), which are produced by the comparator are forwarded to microcontroller.

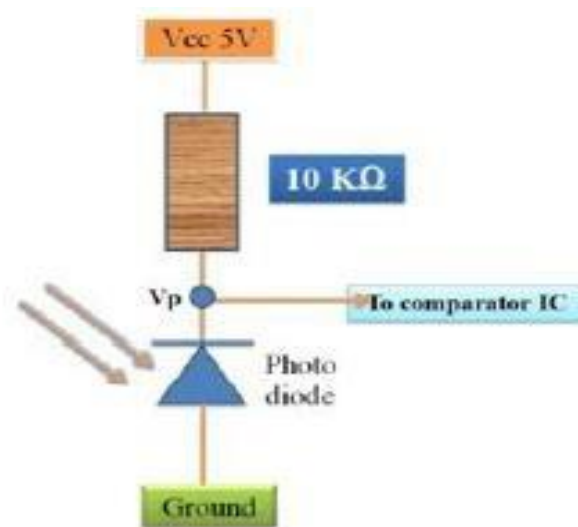


Figure 4.5: Input circuit.

For line follower robot to function, the sensors should sense white line of the track, so as to keep robot to center (C). In this research, there are two type of sensors used; that is line tracking and proximity sensors. Sensors are used to guide the robot on what kind of environment to adopt. Line tracking sensor is meant to sense the path while proximity sensor is used to locate the obstacle and object to be picked.

LX	C	RX
----	---	----

. Right central Left

Figure 4.6 Arrangement of line tracking sensors

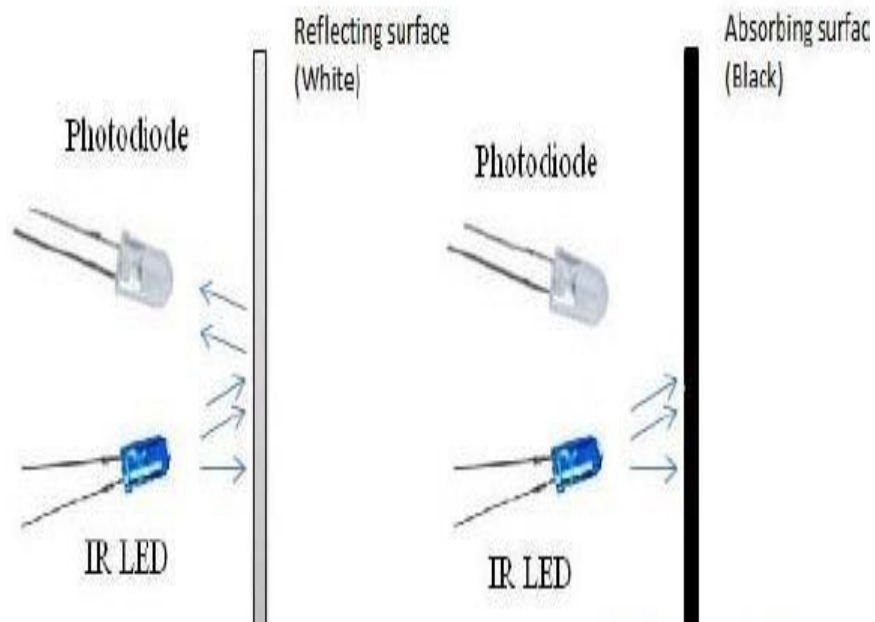


Figure 4.7: Line tracking sensor circuit.

4.8.1 The proximity/Range sensor Circuit

Potentiometer which can be tuned in either direction is used to tune the sensitivity of the IR sensor. The sensitivity of the receiver is at maximum, when the potentiometer is tuned clockwise. Thus, at this point it has maximum sensing distance.

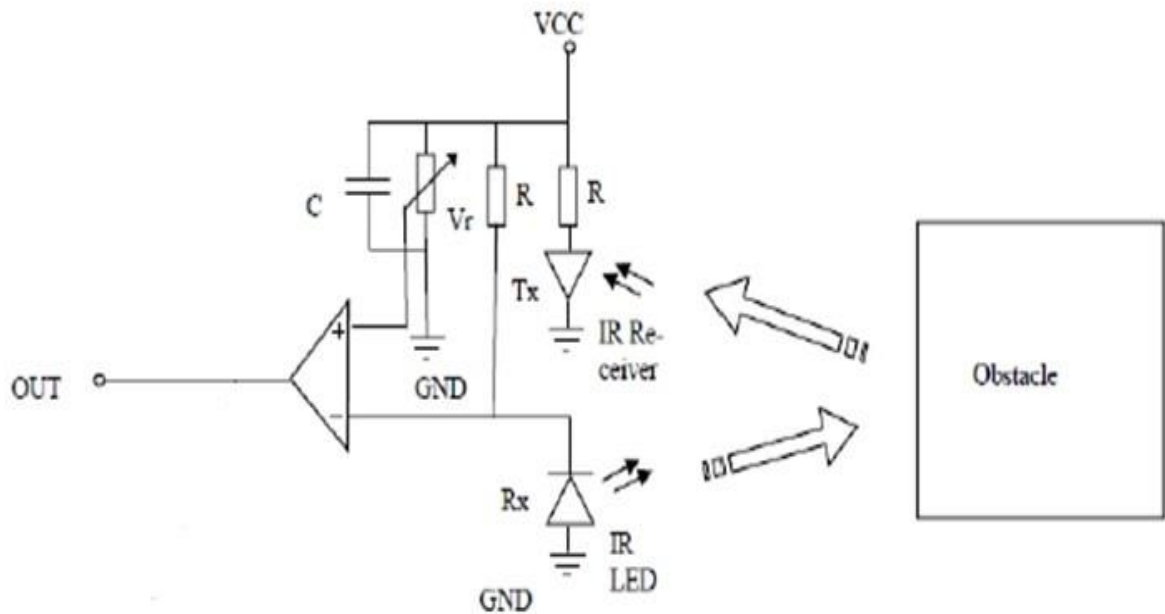


Figure 4.8: Range sensor circuit

For the sensitivity to be minimum, then, the potentiometer has to be turned anticlockwise. If LX and RX is in parallel to each other, such that faces one another outwardly, then, they are at maximum sensitivity, but limited to the surroundings. The LX and RX sensors work perfectly up to when the infrared illumination to that area is almost constant.

Therefore, once the potentiometer is switched for maximum sensitivity and is taken out of room/building, requires retuning, because, rays from the sun have infrared frequencies. This is why the receiver's sensing ability is affected. Due to this, the sensor needs to be tuned in a new environment for its perfection. Once, the IR signal is received, there is no reflection to the IR receiver because the obstacle is absent. If an obstacle is encountered as shown in figure 4.9, the output of the comparator is driven low, due to the IR signal reflected by obstacle surfaces. Transmissive, Reflective and Triangulation are three common strategies for IR Sensor.

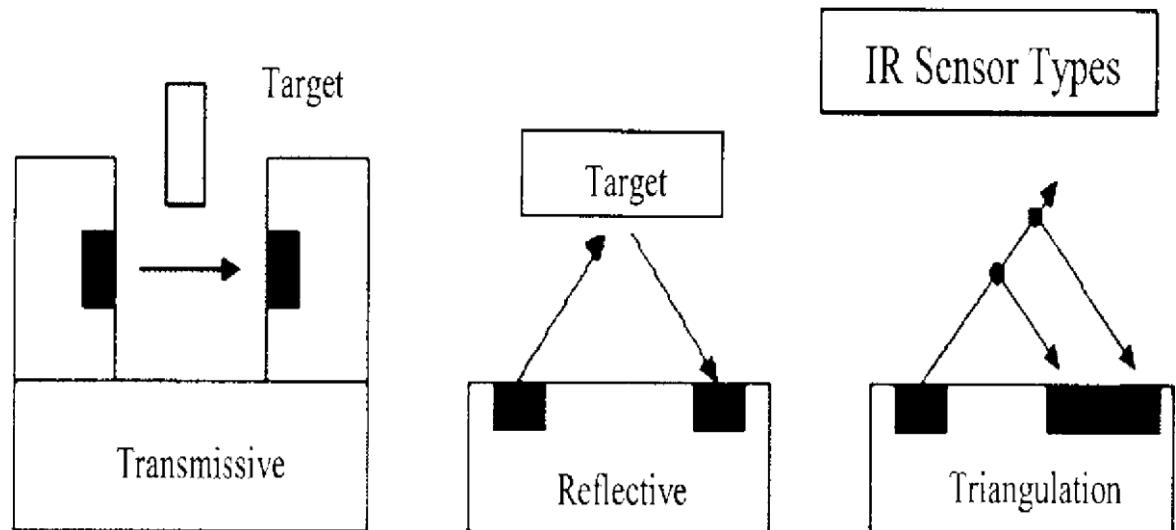


Figure 4.9: Sensor strategies

4.8.2 Transmit strategy

Here, the sensor uses the absence of light to indicate object instead of reflection of light. The input of the detector is received from the IR emitter directly as long as no light is sensed by the detector as shown in figure 4.10.

4.8.3 Reflectance strategy

The infrared emitter LED and a light detector are made from photodiode or phototransistor and are included in arrangement package which acts as a light source to infrared sensor. Infrared acts as a source by emitting light, once light reaches to the object, light is reflected back to the source. Detector receives reflected light from the source and indicates the presence of object.

4.8.4 Triangulation strategy

In this strategy, Ranger uses triangulation and potentiometer device array to estimate the distance and presence of objects. Basically the pulse of IR light is emitted by the emitter. In the field of view, the light travels out and either hits an object or just keeps on going. The light will not be reflected in the case of absence of object, and reading shows no object. A triangle is formed between the points of reflection from

the emitter to the detector, if the light reflects from an object and returns to the detector.

4.8.5 Voltage Regulator 78XX

If range of the input voltage is between 7.5 V to 20V, the 7805 regulator is used to give the output fixed of 5V DC voltage. The voltage regulators are used to convert fixed DC voltage from varying AC. Despite varying currents demands and input voltage variations, the regulator maintains a steady voltage level. The leads of 7805 are identified by keeping the leads downwards as shown in the figure 4.11 and the writing to the side. Above the voltage regulator (1-input, 2-gnd, 3- output), the heat sink can be Seen.

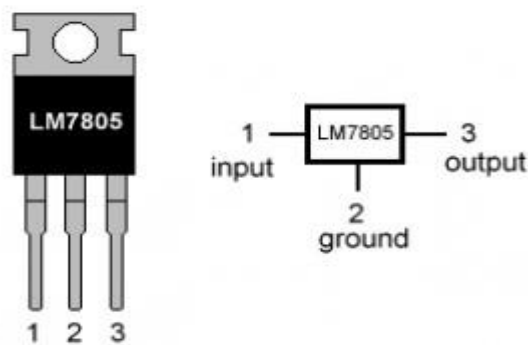


Figure 4.10: 7805 Regulator

Retrieved 17th May 2014 (<https://www.google.com>)

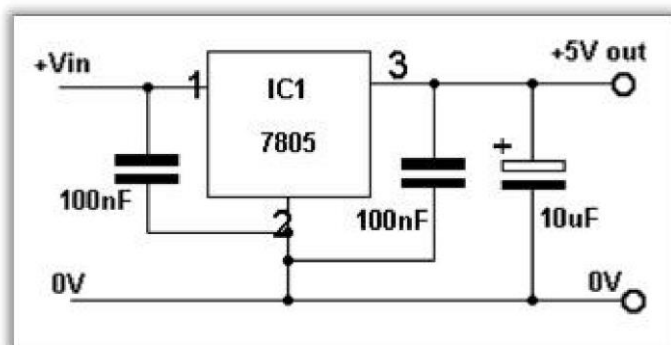


Figure 4.11: circuit diagram for voltage regulator

Figure 4.11 shows the use of 7805 voltage regulator and figure 4.12 the voltage regulator circuit. Capacitors in the circuit are used for reducing the noise at the output voltage.

4.9 Output system

4.9.1 Electric Motor

Motor is a device that converts any form of energy into mechanical energy or imparts motion. In constructing a robot, a motor usually plays an important role by giving movement to the robot. In general, motor operating with the effect of conductor with current and the permanent magnetic field. The conductor with current usually producing magnetic field that will react with the magnetic field produced by the permanent magnet to make the motor rotate. There are generally three basic types of motors, DC motor, servomotor and stepper motor, which are devices that convert electrical energy into mechanical energy.

The DC motor consist a rotating armature in the form of an electromagnet. A rotary switch known as commutator reversing the direction of the electric current twice every cycle to flow through the armature so that the poles of the electromagnet push and pull against the permanent magnets on the outside of the motor. The armature electromagnet passes the poles of the permanent magnets, since using the poles, the commutator reversing the polarity of the armature electromagnet. During that instant switch of polarity, inertia actuates the classical motor going in the proper direction.

Controlling DC motors is the easiest and DC motors have two signals of operation. If the polarity of the power supply is reversed, the direction of the movement of the motor will change. Varying voltage across motor will cause variation in speed of the motor.

4.9.2 Purpose of gears

At the expenses of speed, gears are used to increase the torque of the D.C motor, since, the D.C motors don't have enough torque to drive a robot directly by connecting wheels in it.

4.9.3 Mathematical Interpretation Rotational

power (P_r) is given by:

$P_r = \text{Torque (T)} \times \text{Rotational Speed } (\omega)$

$$T = \frac{P_r}{\omega} \dots \dots \dots (4.1)$$

P_r is constant for DC motor for a constant input electrical power. Thus torque (T) is inversely proportional to speed (ω).

$$T \propto \frac{1}{\omega} \dots \dots \dots (4.2)$$

For increase in the value of torque, speed has to be lost. Robot can be moved into any direction, if two motors are used. By use of two motors, the steering mechanism of robot is a differential drive.

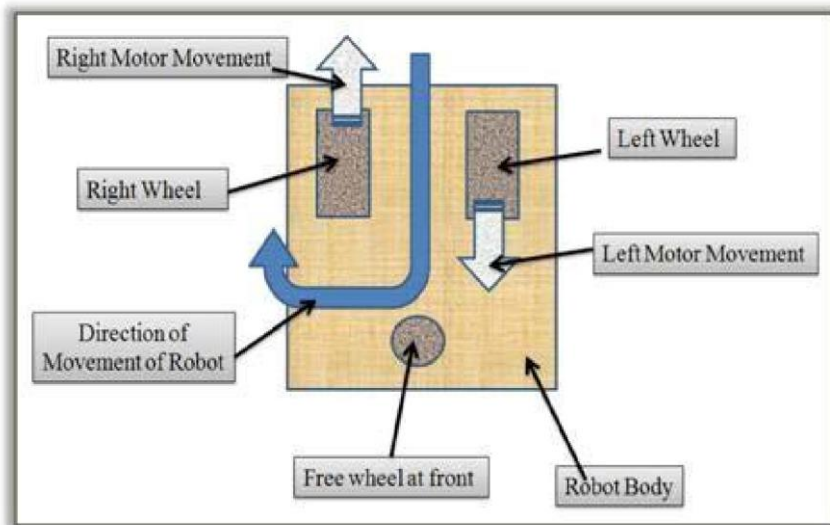


Figure 4.12: Description of various parts

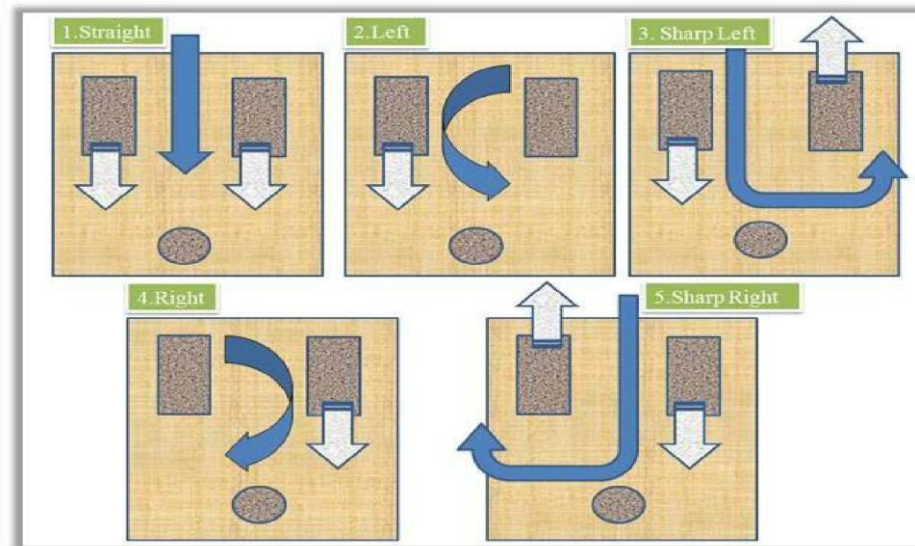


Figure 4.13: Different types of movement of robot

4.10 Robot body

The body of line follower robot is fabricated using aluminum and plywood. The materials cut in different sizes and fixed by use of self-tapping screws, nuts and bolts. It measures 360mm by 400mm by 300 mm car shape and a mass of 4 kilogram. Underneath the body there is one fixed caster wheel and two drive wheels (each driven by a motor).

CHAPTER FIVE

RESULTS AND DISCUSSION

5.1 Base motors

Figure 5.1 shows two base wheel motors interfaced with motor drivers (L298N). The motor drivers are used to provide speed and power to the motors. The robot turned clockwise and anticlockwise that is caused by the motor driver as shown by the truth table of Figure 5.2. Table 5.1 shows the truth table of the motor movement after the power supply was switch ON during testing. Due to the differential drive incorporated in this robot system, the robot moved to desired directions, as shown in Table 5.2.

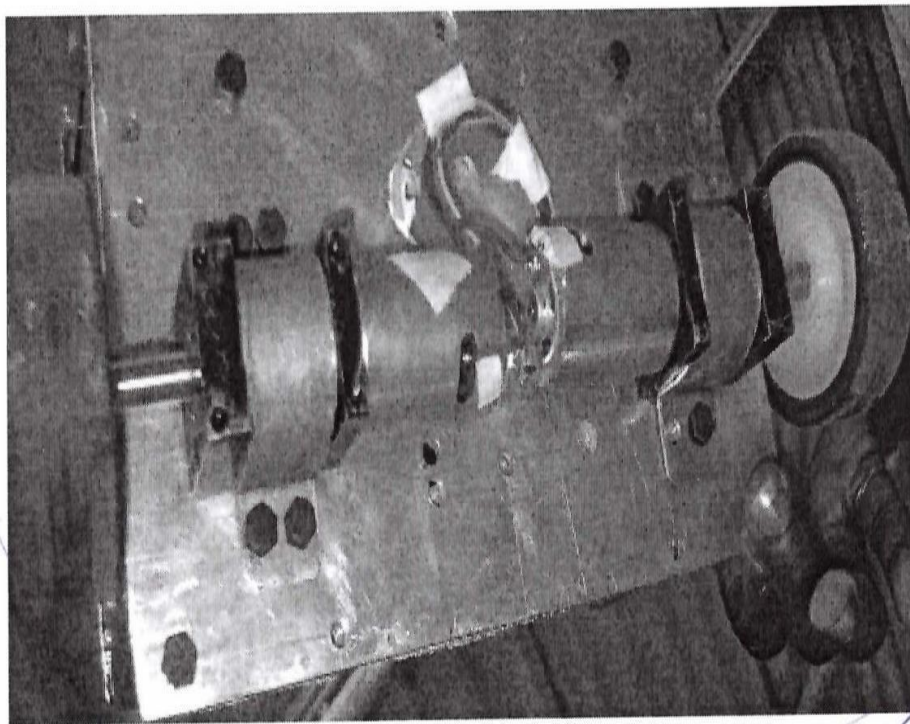


Figure 5.1: DC base motors wheels interfaced to motor drivers

TABLE 5.1: Motor movements with power supply

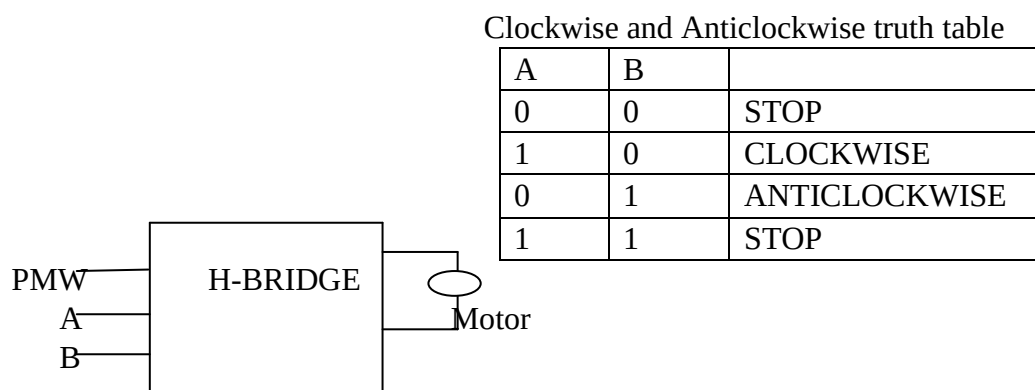
Positive Terminal (+ ve) Negative Terminal (- ve) Motor Output

0	0	NO movement
1	0	Straight
1	1	Reverse
1	1	No movement

TABLE 5.2: Robot moving according to motor direction

Left Motor **Right Motor** **Robot Movement**

Straight	Straight	Straight
Stop	Straight	Left
Reverse	Straight	Sharp left
Straight	Stop	Right
Straight	Reverse	Sharp right
Reverse	Reverse	Reverse

**Figure 5.2: Motor Driver Truth Table**

5.2 The arm and gripper

Figure 5.3 shows a connector circuit that is used to link the arm and the gripper. The body of the arm and gripper was designed using aluminum sheets and chipboard.

The arm is interfaced to the gripper and gripper coupled to the DC motor using gears both fabricated from the same type of sheets. Figure 5.4 shows the end –effect or (gripper), that is used to open and pick objects from the ground which is constructed by use of aluminum sheets and reinforced with rubber band to reduce friction or sliding of object once held. The gripper is fixed to the arm using bolts.

The motor that is fitted at the arm is connected to pin 13 and pin 14 of the driver circuit (L298 N). This motor is used for rising the end-effect or (gripper). Lowering (or setting) of the arm is done by a switch manual ling. The motor connected to the gripper is connected to pin 2 and 3 of the driver circuit. This motor is meant for opening, picking, holding and dropping of an object.



Figure 5.3: Connector circuit between arm and gripper

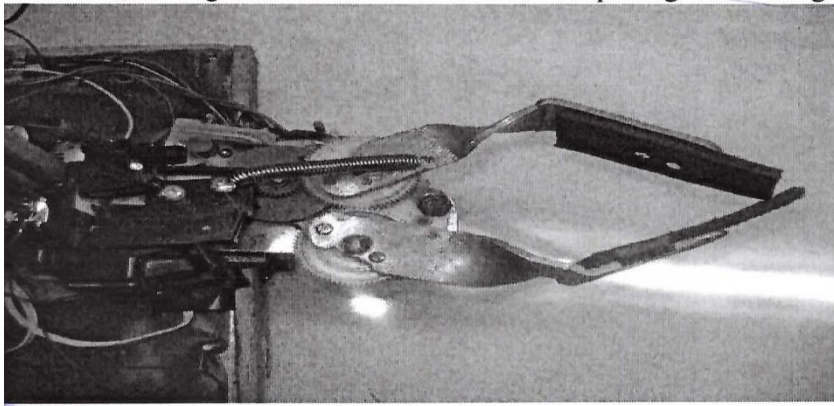


Figure 5.4 :The end- effector (gripper)

5.3 The sensors

The line tracking sensors were mounted to the underside of the robot chassis towards the front of the robot. The sensor operates in an extremely wide range from about 12mm from the floor to almost touching the surface. Proximity sensor is mounted on a wooden angle bar placed on the bumper of the robot. Figure 5.5 shows fabricated proximity and line tracking sensors. The sensors are connected to the microcontroller module which coordinates their senses.

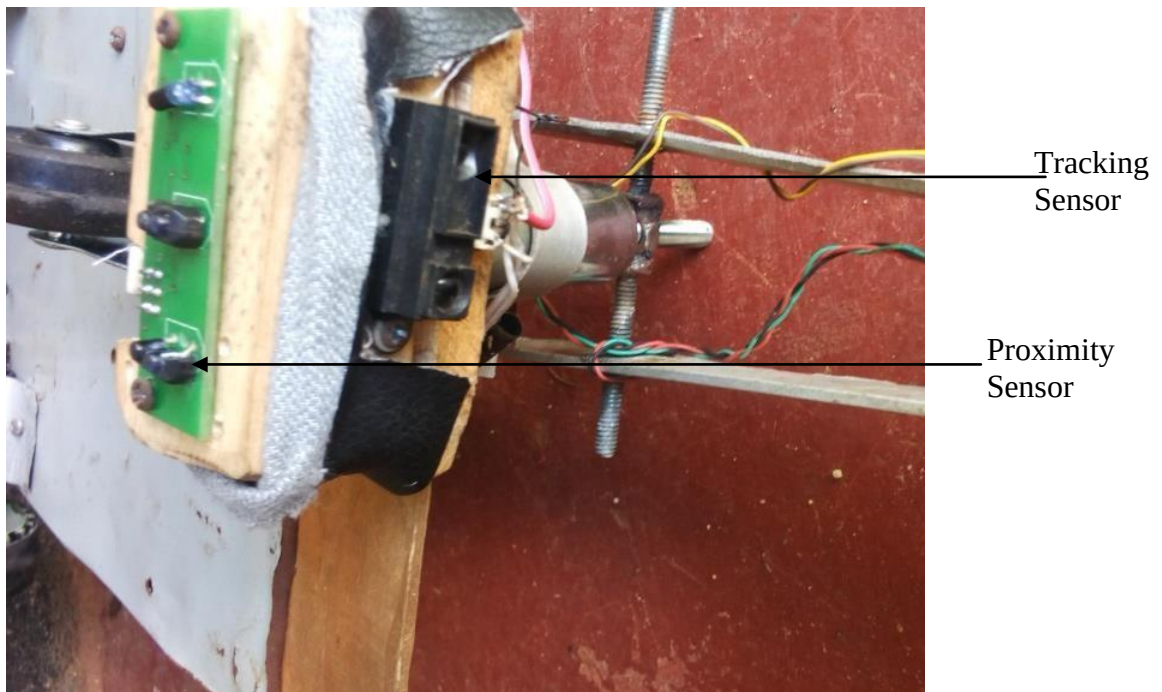


Figure 5.5: Fabricated proximity and line tracking sensors

5.4 Overall System Performance

The complete block circuit diagram of the autonomous line follower robot system is shown in Figure 5.6. This system designed has two modes of operations performed by two different software routines: the training session and automatic operation. Before the system is set into automatic operation, it is first trained by applying appropriate commands through analog (manual) switch, interfaced to the microcontroller. An open switch produces a “high level” signal at the port line and a closed switch produces a “low level” signal.

The „training session“ produces a sequential record of all the operations performed. This record serves as the input for the automatic operation routine. When the switch is open (OFF), then there is no movement of the robot but when the switch is closed (ON), the robot start to move and continues automatically hence forth until a command from the controller signals to stop .The microcontroller system detects and identifies the commands given by the sensors. Microcontroller applies appropriate signals to the motors so that the robot manipulator body connected to them obeys the commands. Motor driver provide the necessary buffering between the motor windings and the output pin of the microcontroller.

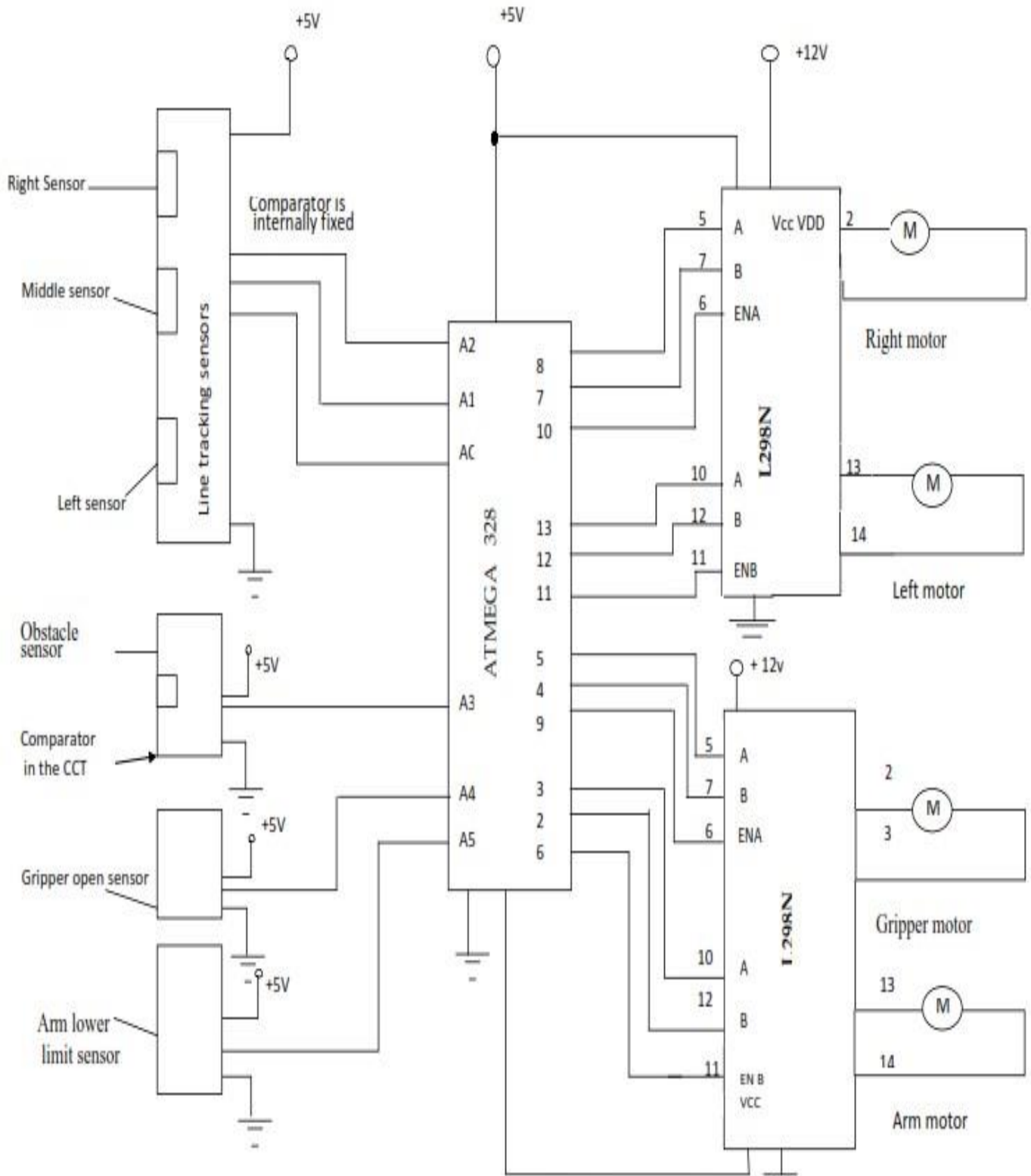


Figure 5.6: Block diagram circuit of an autonomous line follower robot system

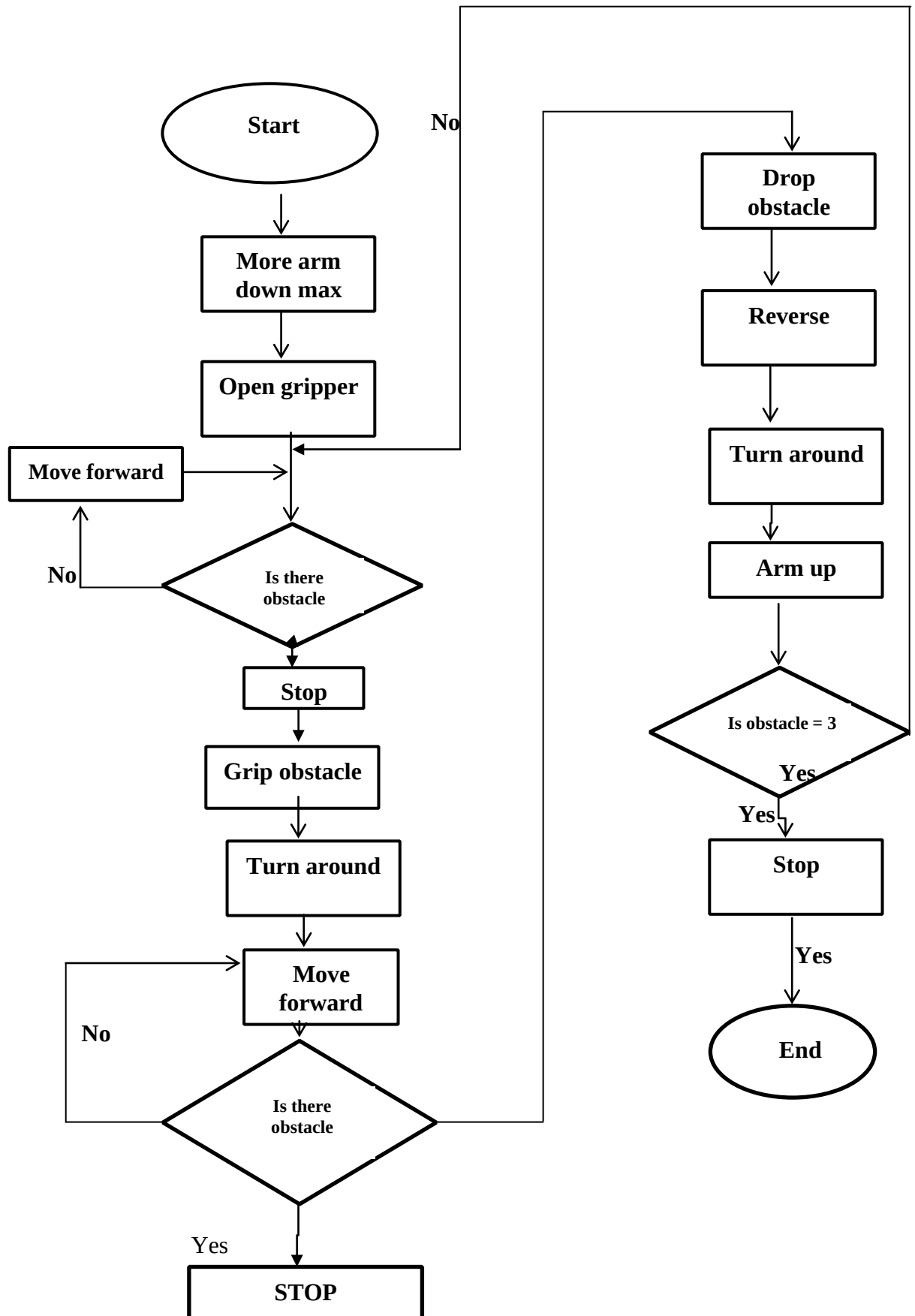
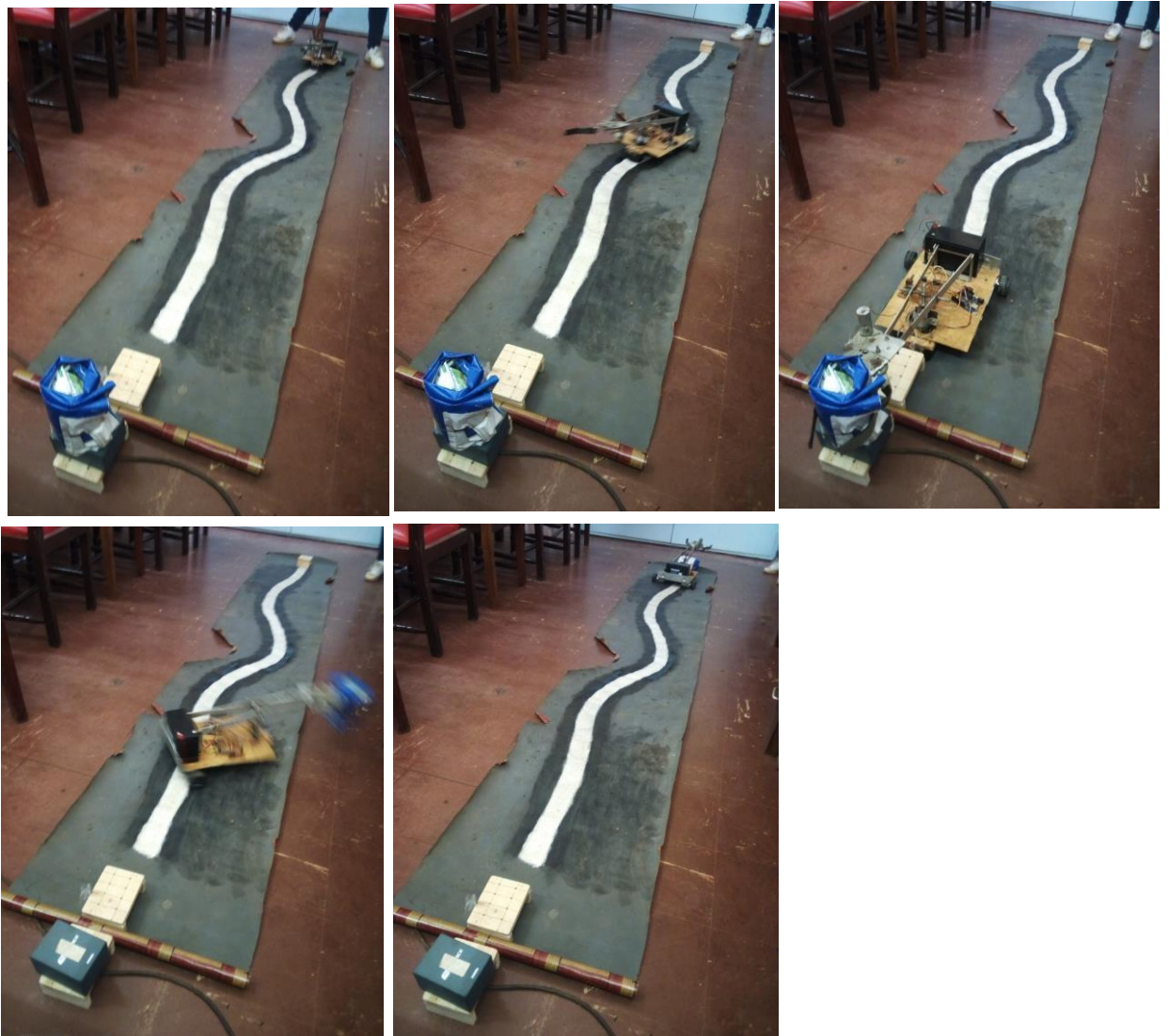


Figure 5.7: The System Software Flow Chart

The system software program is written in the Arduino C language programming and compiled by java compiler. During programming, Four things were carried out; first, flow chart was made to show a complete program as shown in figure 5.7, pins for each component were identified, the subroutines program for every components were written and operation program constructed and then the whole program compiled together and uploaded to the microcontroller which controls the robot functionality. The performance evaluation of the integration of the system's software and hardware with reference to the research objectives presented. To verify the effectiveness of the system's hardware and software, the line tracking of the robot was tested and then gripping, carrying and dropping the object.

The robot tracked the line while the gripper, were open and using three line tracking sensors moved from one end of the line to the other end. The robot stopped when the object was placed in front of it. The arm of the robot moved up, opened the gripper and picked the object. Once the robot had held the object, the robot made a 180 degree clockwise turn and carried the object back along the path to the destination. After dropping the object the robot turned again 180 degree and stopped the movement. During this time of testing, two methods of testing were used that is unit testing and integration testing. Unit test, involved testing each unit module individually while integration testing was done to ensure proper interfacing and compatibility of various module. . However, during testing trials the following failures were observed. When tracking the line the robot could lose the track of the line. This was due to misalignment of the caster wheels, and the distance between the ground and the sensors. Also, due to several codes that were tried, sometimes, the robot behaved unpredictable. It could make several turns where unnecessary. During

the robot operation, after the gripper had released the object to the destination, it could stop a little and then continue moving after a short while because the delay a location time was 5000 seconds instead of 1000000 seconds. The faults were rectified after testing and then the robot operation was smooth and good as shown in figures 5.8.



Figures 5.8: Robot operations

CHAPTER SIX

CONCLUSIONS AND RECOMMENDATION FOR FURTHER STUDY

6.1 Conclusions

The microcontroller program has been developed using arduino C programming language and was able to control the robot functionality.

The robot has been designed and fabricated such that, it coordinated and followed the line that was drawn on a black resin carpet, the robot opened the gripper, sensed and stopped when there was an Obstacle in front of it, lifted the arm ready to grip an object. It picked the object, carried it and dropped the object to another point as required. Designing and Fabrication of the robot prototype was achieved and worked according to the specifications set.

6.2 Recommendations

The steering mechanism of the robot should be well managed during fabrication to prevent malfunctioning.

More functionality of the system can be added that allow implementing of algorithms for sensing color, counting and tracking curving tracks

REFERENCES

- Acemogly.R and Restrepo.P (2017).** *Automation robots and jobs research work*, M.sc, thesis work, Boston University and MIT Economic.
- Andrew J. T (2001).** *A path following system for autonomous Robot with Minimal Computing Power*. B. SC, Thesis work, University of Aukland.
- Asfah, C.R (1985).** *Robots and Manufacturing Automation* .Wiley, New York.
- ATMEGA328 Datasheet (2012).** Retrieved 30 may 2014 from <Http://www.atmel.in/devices/ATMEGA328.aspx>
- Hale F.J (1973.,** *Introduction to Control Systems Analysis and Design*, Prentice Hall Inc., Englewood Cliffs, New Jersey U.S.A.
- Hale F.J (1973).** *Introduction to Control systems Analysis and Design*. Prentice – Hall Inc., Englewood Cliffs, Jeff
- Islasm A and Rayman F (2013).** *Robot operational amplifier research work*, B.Sc. Boston University.
- JICA report (2007).** *PID for Line Following*. Retrieved from 17th may 2014. www.chibots.org/Index.php?node/339.
- John I.C (2000).** *Introduction to Robotics; mechanics and Control*, Second Edition, Addison Wesley publishing Co., Longman Inc.
- Jossie M (2005).** *PWM/ PID/servo Motor Control using Orangutan* from Pololu . Retrieved 20 May 2014,from [Http://www.uoxray.uoregon.edu](http://www.uoxray.uoregon.edu)
- Kanayama Y.N (1986).** *Trajectory Generation for Mobile Robot*, *Robotics Research* Vol.3, pages 333-340, the MIT Press.
- Karimi P.M (2000).** *Microprocessor Controlled Multifunctional Signal Generator*, M.Sc. Thesis work, Kenyatta University.
- Lee M.H (1989),** *Intelligent Robotics*, Open University press, Great Britain.
- Nnaji B.O (1993).** *Theory of Automatic Robot Assembly and programming*, Chapman and Hall, Great Britain.
- Pascal D. O (2008).** *Intelligent Line following for vision enabled mobile robots*. MSc, Thesis work, Technical University of Denmark, DTU Elektro.
- Peregrius P.A (1998).** *D.C Motors: a Guide to Modern Theory and Practice*, Third Edition IEE Publication.

- Russell A.R (2001).** *International Journal of Robotics Research, Survey of Robotic Applications for Odor-sensing Technology*, Vol.20,2 Sage Publications Inc., Thousand Oaks.
- Sheikh F. J., and Rakesh M, (2001).** *PID Control of Line Followers*, Thesis work, National Institute of Technology. Rourkela 2013
- Sushil, P, (2013).** *Autonomous Robots*, M.Sc. Thesis work, University of Windsor.
- Techbitar (2012).** Robot built by use of arduino,uno, adafruit motor shield, pololu's QTR-8R Line sensors and motors, *Instructables technology*, Vol: 3 pages 334- 350.
- Thilakshan (2010).** An inertia measuring unit of an accelerometer and gyroscope for measuring acceleration and angular velocity. B Sc, Thesis work, University of Auckland
- Thomas M.E. and Louw, A (1984).** Scientific Instruments, *Journal of Physics*, Vol. 17 No. 8, Pages 631-714, The Institute of Physics, Great Britain.
- Thomas M.E. and Louw, A(1984).** Scientific Instruments, *Journal of Physics*, Vol. 17 No. Pages 631-714. The Institute of Physics, Great Britain.
- Thorisson G (2005).** *Motion Control for Mobile Robots*, MSc., Thesis work, Institute Oersted, Dan marks Tekniske University.
- Tooley M (1996).** *Electronic Circuits Handbook, Design, Testing and Construction*, Second Edition, Newness, Great Britain.
- Walaa M, Sheba G, Elnemr H.F and Gamal T.J,(2014).** *Robot with webcam and PID control, Algorithm*. Retrieved June 2014 from [https:// W.W.W. research gate .ne> publications](https://www.researchgate.net/publications).
- Warring R.H (1984).** *Robots and Robotology*, TAB BOOKS Inc.,U.S.A.
- Waurzyniak P, (1999).** *Manufacturing Engineering Journal, Robotics Evolution*, Vol. 1222 Manufacturing Engineers Society, Dearborn.

APPENDICES

APPENDIX I: THE SOFTWARE

The complete software for the microcontroller is listed as follows.

USER NAME: SERONEY DANIEL KIPTOO

Microcontroller with two degree of freedom implemented using 12VDC motors assembled with aluminum sheets to form robot prototype.

ABSTRACT

The program code acts as the decision-maker embedded in the micro-controller deciding about the outputs for particular set of inputs. The following program was written using Arduino C language and was uploaded to the memory of the microcontroller using USB. The program is compiled by java to form a “.hex” file which is uploaded into the microcontroller. The user has to start the control of the robot by the switch.

PROGRAM

```
int leftmotor_A=13; int
leftmotor_B=12; int
leftmotor_pwm=11; int
rightmotor_A=8; int
rightmotor_B=7; int
rightmotor_pwm=10; int
left_sens=A0; int
centr_sens=A1; int
right_sens=A2; int
obstacle_sens=A3; int
gripper_A=5; int
gripper_B=4; int
gripper_pwm=9; int
gripperOpen_sens=A5; int
arm_A=3; int arm_B=2;
int arm_pwm=6; int
armlower_sens=1; int
state1; int state2; int
obstacle; int
obstacleState; int
sensStateL;
int sensStateC; int
sensStateR; int
```

```

gripper_openLimit; int
arm_lowerLimit; int
objects=0;
//Variables for all the speed constants are defined here
int max_speed=240; int
valx,valy,val, val2; int
val1=max_speed; void
setup()
//Declare the pins as either inputs or outputs.
{pinMode(leftmotor_A,OUTPUT);
pinMode(leftmotor_B,OUTPUT);
pinMode(leftmotor_pwm,OUTPUT);
pinMode(rightmotor_A,OUTPUT);
pinMode(rightmotor_B,OUTPUT);
pinMode(rightmotor_pwm,OUTPUT);
pinMode(left_sens,INPUT);
pinMode(centr_sens,INPUT);
pinMode(right_sens,INPUT);
pinMode(gripper_A,OUTPUT);
pinMode(gripper_B,OUTPUT);
pinMode(gripper_pwm,OUTPUT);
pinMode(gripperOpen_sens,INPUT);
pinMode(arm_A,OUTPUT);
pinMode(arm_B,OUTPUT);
pinMode(arm_pwm,OUTPUT);
pinMode(armlower_sens,INPUT);
}
int
read_sensors()
{ obstacleState= digitalRead(obstacle_sens);
sensStateL= digitalRead(left_sens); sensStateC=
digitalRead(centr_sens); sensStateR=
digitalRead(right_sens); gripper_openLimit=
digitalRead(gripperOpen_sens); arm_lowerLimit =
digitalRead(armlower_sens);
} void move_forward(){
analogWrite(leftmotor_pwm,110);
analogWrite(rightmotor_pwm,110);
digitalWrite(leftmotor_A,HIGH);
digitalWrite(leftmotor_B,LOW);
digitalWrite(rightmotor_A,HIGH);
digitalWrite(rightmotor_B,LOW);
}
void TURNRIGHT()
{
analogWrite(leftmotor_pwm,180);
analogWrite(rightmotor_pwm,20);
digitalWrite(leftmotor_A,HIGH);
digitalWrite(leftmotor_B,LOW);
digitalWrite(rightmotor_A,HIGH);
digitalWrite(rightmotor_B,LOW);
}

```

```

    }
    void TURNLEFT()
    {analogWrite(leftmotor_pwm,0);
    analogWrite(rightmotor_pwm,250);
    digitalWrite(leftmotor_A,HIGH);
    digitalWrite(leftmotor_B,LOW);
    digitalWrite(rightmotor_A,HIGH);
    digitalWrite(rightmotor_B,LOW);
    }

    void align_robot(){
read_sensors();
    if((sensStateL ==LOW)&& (sensStateR ==LOW)&&(sensStateC == LOW))
    { stop_moving();//don't move.
    }
    else
    {
        if((sensStateL==HIGH) && (sensStateR==LOW || sensStateC == LOW))//if the
robot gets off the track towards the right side. {
        TURNLEFT();//turn it to the left.
    }
        if((sensStateL==LOW) && (sensStateR==HIGH ||sensStateC == HIGH))//if the
robot gets off the track towards the left side. {
        TURNRIGHT();//turn it to the right.
    }
        if((sensStateL==HIGH) && (sensStateR==HIGH || sensStateC == HIGH))//if the
robot is on track.
        {
            move_forward(); //continue moving forward.
        }
    }
    }
    void
stop_moving()
    {
        analogWrite(leftmotor_pwm,10);
        analogWrite(rightmotor_pwm,10);
        digitalWrite(leftmotor_A,LOW);
        digitalWrite(leftmotor_B,LOW);
        digitalWrite(rightmotor_A,LOW);
        digitalWrite(rightmotor_B,LOW);
    }
    void reverse(unsigned long duration)
    {
        unsigned long start = millis();
        while (millis() - start <= duration)
        {
            analogWrite(leftmotor_pwm,150);
            analogWrite(rightmotor_pwm,150);
            digitalWrite(leftmotor_A,LOW);

```

```

digitalWrite(leftmotor_B,HIGH);
digitalWrite(rightmotor_A,LOW);
    digitalWrite(rightmotor_B,HIGH);
    }
    }
    void left_correction()
    {
        analogWrite(leftmotor_pwm,130);
        analogWrite(rightmotor_pwm,10);
        digitalWrite(leftmotor_A,LOW);
        digitalWrite(leftmotor_B,HIGH);
        digitalWrite(rightmotor_A,HIGH);
        digitalWrite(rightmotor_B,LOW);
    }
    void right_correction()
    {
        analogWrite(leftmotor_pwm,10);
        analogWrite(rightmotor_pwm,130);
        digitalWrite(leftmotor_A,HIGH);
        digitalWrite(leftmotor_B,LOW);
        digitalWrite(rightmotor_A,LOW);
        digitalWrite(rightmotor_B,HIGH);
    }
    void
left_Uturn()
    {
        analogWrite(leftmotor_pwm,120);
        analogWrite(rightmotor_pwm,120);
        digitalWrite(leftmotor_A,LOW);
        digitalWrite(leftmotor_B,HIGH);
        digitalWrite(rightmotor_A,HIGH);
        digitalWrite(rightmotor_B,LOW);
    }
    void right_Uturn()
    {
        analogWrite(leftmotor_pwm,120);
        analogWrite(rightmotor_pwm,150);
        digitalWrite(leftmotor_A,HIGH);
        digitalWrite(leftmotor_B,LOW);
        digitalWrite(rightmotor_A,LOW);
        digitalWrite(rightmotor_B,HIGH);
    }
    void find_trackL()
{read_sensors();
    while ((sensStateR && sensStateC)== LOW)
    {left_Uturn();
read_sensors(); if ((sensStateR
&& sensStateC) == HIGH){
    stop_moving();
delay(200);    break;

```

```

    }
    }
}
void                correct_trackL()
{read_sensors();
  while ((sensStateR && sensStateC)== LOW)
    {left_correction(); read_sensors(); if
((sensStateR && sensStateC) == HIGH){
      stop_moving();
delay(200);      break;
    }
  }
} void
find_trackR()
{read_sensors();
  while ((sensStateL && sensStateC)== LOW)
    {right_Uturn(); read_sensors(); if
((sensStateL && sensStateC) == HIGH){
      stop_moving();
delay(200);      break;
    }
  }
}
void                correct_trackR()
{read_sensors();
  while ((sensStateL && sensStateC)== LOW)
    {right_correction(); read_sensors(); if
((sensStateL && sensStateC) == HIGH){
      stop_moving();
delay(200);      break;
    }
  }
}
void armUp(unsigned long duration)
{
  unsigned long start = millis();

  while (millis() - start <= duration)
  {
    digitalWrite(arm_A, LOW);      digitalWrite(arm_B, HIGH);
    analogWrite(arm_pwm, val1);
  }
}
void armDown(unsigned long duration)
{
  unsigned long start = millis();
  while (millis() - start <= duration)
  {

```

```

        digitalWrite(arm_A,                HIGH);
digitalWrite(arm_B, LOW);
        analogWrite(arm_pwm, val1);
    }
}
void armDown_max(){ while
(arm_lowerLimit==HIGH)
{
    digitalWrite(arm_A, LOW);
digitalWrite(arm_B, HIGH);
analogWrite(arm_pwm, val1);
read_sensors();
    if (arm_lowerLimit== LOW){
stopArm(100);    break;
    }
}
}
void stopArm(unsigned long duration)
{
    unsigned long start = millis();
    while (millis() - start <= duration)
    {
        digitalWrite(arm_A,                LOW);
digitalWrite(arm_B, LOW);
    }
}
void openGripper(){
read_sensors();
    while (gripper_openLimit==HIGH)
    {
        digitalWrite(gripper_A, LOW);
digitalWrite(gripper_B, HIGH);
analogWrite(gripper_pwm, val1);
read_sensors();    if
(gripper_openLimit == LOW){
stopGripper();    delay(100);
break;
    }
}
}
void closeGripper(unsigned long
duration)
{
    unsigned long start = millis();
    while (millis() - start <= duration)
    {
        digitalWrite(gripper_A,                HIGH);
digitalWrite(gripper_B, LOW);
        analogWrite(gripper_pwm, val1);

```

```

    }
}

void stopGripper() {
digitalWrite(gripper_A, LOW);
digitalWrite(gripper_B, LOW);
}

void loop(){
read_sensors();
if(arm_lowerLimit==0)
{
digitalWrite(arm_A, HIGH);
digitalWrite(arm_B, LOW);
analogWrite(arm_pwm, 250);
read_sensors();
}
else
{
stopArm(100);
//armDown(50000);
//align_robot();
//openGripper();
//closeGripper(2000);
//stopGripper();
//delay(5000);
//armDown(5000);
//stopArm(100);
openGripper();
read_sensors();
while(obstacleState != LOW){
align_robot();
read_sensors(); if
(obstacleState == LOW){
stop_moving(); delay(200);
break;
}
}
closeGripper(3000);
stopGripper(); delay(100);
armUp(5000);
stopArm(100);
reverse(800);
read_sensors();
left_Uturn(); delay(700);
stop_moving();
delay(200);
openGripper();
armDown(5000);
stopArm(100);

```

```

//reverse(500);
read_sensors();
find_trackR();
correct_trackL();
/*
    read_sensors();
    //armDown_max();
    while(obstacleState != LOW){
        align_robot();    read_sensors();
        if (obstacleState == LOW){
            stop_moving();    delay(200);
            break;    }
        }
        closeGripper(3000);
    stopGripper();
    delay(100);
    armUp(5000);
    stopArm(100);
    reverse(1000);
    stop_moving();
    read_sensors();
    right_Uturn();
    delay(500);
    stop_moving();
    delay(200);
    openGripper();
    armDown(5000);
    stopArm(100);
    reverse(500);
    read_sensors();
    find_trackL();
    stop_moving();
    delay(1000); */
    //armDown(5000);
    //stopArm(100);
    read_sensors();
    openGripper();
    read_sensors();
    while(obstacleState != LOW){
        align_robot();
        read_sensors();    if
        (obstacleState == LOW){
            stop_moving();
            delay(200);    break;
        }
    }
    closeGripper(3000);
    stopGripper(); delay(100);
    armUp(5000); stopArm(100);
    reverse(500); read_sensors();

```

```

left_Uturn(); delay(500);
find_trackL(); correct_trackR();
read_sensors();
//armDown_max();
while(obstacleState != LOW){
    align_robot();
    read_sensors();    if
    (obstacleState == LOW){
        stop_moving();
        delay(200);    break;
    }
}
    openGripper();
    stopGripper();
    delay(100); reverse(500);
    read_sensors();
    left_Uturn();
    delay(500);    find_trackL();
    correct_trackR();
    armDown(5000);
    stopArm(100);
    delay(1000000);
}
}

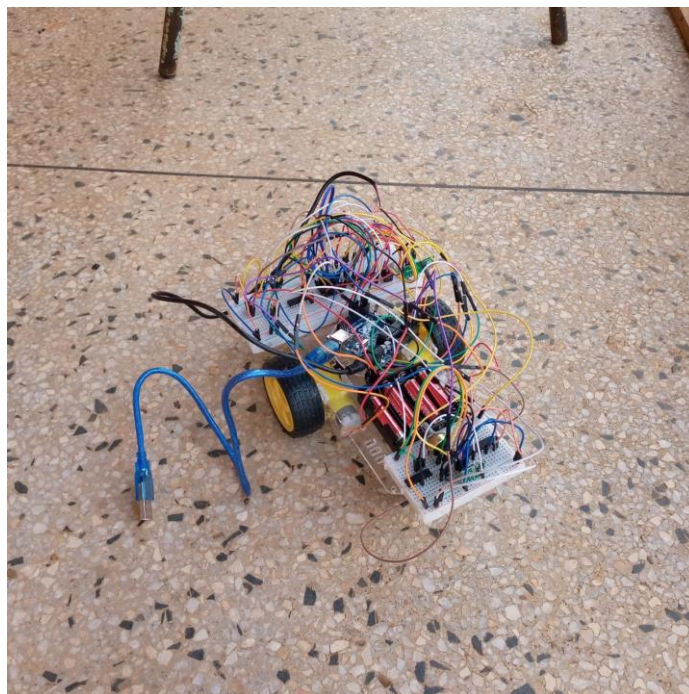
```

APPENDIX II: Pictures of the Constructed Robot

The system side view



The system top view



Robot on the Track

