PID CONTROL OF LINE FOLLOWERS

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Bachelor of Technology

In

Electronics and Instrumentation Engineering

By

SHEIKH FARHAN JIBRAIL (109EI0326)

RAKESH MAHARANA (109EI0328)

Department of Electronics & Communication Engineering
National Institute of Technology, Rourkela

2013

PID CONTROL OF LINE FOLLOWERS

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Bachelor of Technology

In

Electronics and Instrumentation Engineering

By

SHEIKH FARHAN JIBRAIL (109EI0326)

RAKESH MAHARANA (109EI0328)

Under the guidance

of

Prof .Tarun Kumar Dan

Department of Electronics & Communication Engineering
National Institute of Technology, Rourkela
2013

CERTIFICATE

This is to certify that the work in the thesis entitled "PID CONTROL OF LINE FOLLOWERS", submitted by Sheikh Farhan Jibrail (109EI0326) and Rakesh Maharana (109EI0328) has been carried out by them under my supervision for the award of Bachelor of Technology Degree in "ELECTRONICS & INSTRUMENTATION" Engineering during the session of 2009-2013 in the Department of Electronics & Communication Engineering at National Institute of Technology, Rourkela.

Place: Rourkela

Date:

Prof. TARUN KUMAR DAN

Department of Electronics and Communication Engineering

National Institute of Technology, Rourkela-769008

ACKNOWLEDGEMENT

We would like to take this opportunity to express our gratitude to our respected supervisor **Prof Tarun Kumar Dan** for the inspired guidance, insight, continuous encouragement, timely suggestions that he has provided throughout the duration of this work. The present work, being successfully completed due to his sincere monitoring and vital inputs.

We are grateful to **Prof. S Meher**, Head of the Department of Electronics and Communication Engineering, permitted us to make use of the available facilities in the department to carry out the project successfully. We would also thank all our friends, faculty and staff members of Department of Electronics & Communication Engineering for their support and all kinds of help to accomplish this work.

Sheikh Farhan Jibrail (109EI0326)

Rakesh Maharana (109EI0328)

ABSTRACT

Line Following is one of the most important aspects of robotics. A Line Follower Robot is an autonomous robot which is able to follow either a black or white line that is drawn on the surface consisting of a contrasting color. It is designed to move automatically and follow the made plot line. PID control of line follower is a method consisting of Proportional, Integral & Derivative functions to improve the movement of the robot. The robot uses several sensors to identify the line thus assisting the bot to stay on the track. The robot is driven by DC Motors to control the movement of the wheels. The Atmega Microcontroller will be used to perform and implement PID algorithms to control the speed of the motors steering the robot to travel along the line smoothly. This project aims to implement the PID algorithm and control the movement of the robot by proper tuning of the control parameters and thus achieve better performance. This project has various applications in the field of Medicine, Automation and Space Application.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction to Robotics:

1.1.1 What is a robot?

A Robot has been defined by the Robot Institute of America as:

"A robot is a reprogrammable, multifunction, manipulator designed for moving materials, tools, parts etc. through various programmed motions to perform a variety of tasks ".

A Robot has been defined by Webster dictionary as:

"A robot is an automatic device that performs functions normally ascribed to humans or a machine in the form of a human."

Generally robots have three main parts which include processor, sensor and motor control system. Robot sensors represent eyes while actuators act as legs and controller acts as the brain of a human.

1.1.2 Autonomous Robot

Autonomous robots are the robots that can perform desired tasks in any environment without continuous human guidance. In fields like space exploration high degree of autonomy is required where communication and delays are unavoidable.

In the real world a fully autonomous robot has the ability to gain information about the environments and to work for months without human intervention. It can travel from one location to the other without navigation assistance. It can avoid situations that are harmful to any property or itself and can repair without external assistance.

1.1.3 Applications of a Robot:

Robot can replace human's job because robots can perform faster than humans. Robots need not to drink, to be paid or rest as compared to humans. They can do repetitive work with high accuracy and will not stop or slow until the task is finished while humans get bored.

Robot can be applied in military to reducing the number of casualties which occur during military actions has been already been prioritized. The military also uses robots for locating and destroying mines on land and in water, spying on enemies and entering enemy bases for gathering information.

Nowadays a doctor can use robots while performing surgery. A human would not be able to make a hole exactly one 100^{th} of an inch wide and long. When making medicines, robots can perform their job much faster and much accurately and delicate in comparison to humans. Doctors and engineers sometimes are developing prosthetic limbs by using robotic mechanisms.

Individual stationary sensors have limited ranges and applications. Watchdogs and humans can lose their level of alertness during shift and be injured by intruders. Autonomous robot systems are the tools that combine the precision of sensors with the mobility and intelligence of humans. Robotic site of security sentries are able to work for long hours at consistency high level of vigilance and precision. People are interested in places full of danger like outer space or under seas. They cannot go themselves there so they use robots which are used for exploration. The robots can carry cameras and other instruments through which they can collect information and send back for processing to their human operators. The continuous development of autonomous robots increases our ability to explore universe.

1.1.4 Overview

Robots are widely used in industries due to their characteristics. Robots are able to work 24 hours continuously without feeling tired unlike human that confined to certain time. The cost required to setup the robot nowadays becomes more affordable and their long term prospectus is bright judging from their capabilities to perform. But in reality, there is no robot able to functions perfectly and are still making error. A better controller is required to allow the robot performs efficiently and make less error.

This project tries to implement a PID controller on autonomous robot to see whether the robot performs efficiently. This autonomous robot consist a line-tracking module, where it will follow the track made from black tape. This is the area where the PID is implemented, the robot will be able to follow the black tape effectively and moving along the track smoothly.

1. 2 Objectives:

The main objective of this project is to design a line follower robot with PID controller and compare the effect of PID to normal line follower. The objectives of this project are:

i. To design and develop an autonomous robot that follows a black line drawn on the floor while smoothing the tracking motion by using digital PID control.

ii. To study the concept of infrared sensors, LCD, bus and ADC interfacing, servo motors, PID controller tuning and AVR microcontroller.

iii. To design and construct the platform of including follower robot, the AVR microcontroller board, sensors and motor drive system.

iv. To design and implement the PID algorithms into AVR microcontroller, and to test and tune the PID control to achieve better performance.

1.3 Scope of the Project:

This project focuses at designing and building the line follower robot and implement PID controller. Therefore, this robot will cover the scopes as follow:

1. Design line follower robot using AVR Microcontroller.
2. PID controller implementation through programming in AVR studio by using basic language.
3. In the end of project, robots movement with PID and without PID controller by inspecting the pattern of movement of the robot while following the track.

1.4 Problem Statement:

Classical line following robot is slow response to the error occur will easily leave its track that drawn on the floor. This problem will result the motion of the robot to be unsmooth and sometimes robot tends to move out of the track. Although the line following robot can follow the black line, its motion still needs to be improved, so to overcome that problem, we need a better controller to make robot follow the line smoothly and make less error. The motion of line following robot can be improved by using feedback mechanism which forms an effective closed loop system. In this project, we are using PID controller because of easy implementation on autonomous robot.

1.5 Thesis Organization:

This thesis is a documentary delivering the ideas generated, concepts applied, activities done and finally the final year project itself. It contains four chapters. The following is a description of information in this thesis.

Chapter 1 provides a general overview of the project and the use and importance of autonomous robots in the world. The objectives, scope of project, problem statement are also described in this chapter.

Chapter 2 describes the hardware development unit in line following robot. This chapter describes about sensor arrays, microcontroller, motor driving system, PID control and tuning of PID. It also describes the project methodology and explains hardware development for the design of the robot. Controllers have also been covered in this chapter.

Chapter 3 contains all the results obtained from the software experiments that include the algorithm implemented in a program.

Finally, chapter 4 will summarize the final year project. The conclusion, suggestions or recommendations for improvements that can be implemented in future are discussed within this chapter.

CHAPTER 2

HARDWARE DEVELOPMENT

2.1 Introduction:

The line follower is a self-operating robot that detects and follows a line that is drawn on the floor .The line follower robot using PID controller is a self-operating system that detects and follows track drawn on the floor. The track consists of a black tape on white surface or vice versa. The control system senses the line and maneuvers the robot to stay on course while constantly correcting the wrong moves using PID control, thus might forming effective controlled system.

2.2 Basic Operation:

The basic operations of line follower are as follows:

1. Capture line position with optical sensors mounted at front end of the robot. For this a combination of IR LED'S and phototransistor called an optical coupler has been used. Requirement for line sensing process are high resolution and high robustness.

2. Steer robot requires steering mechanism for tracking. Two motors governing wheel motion are used for achieving this task.

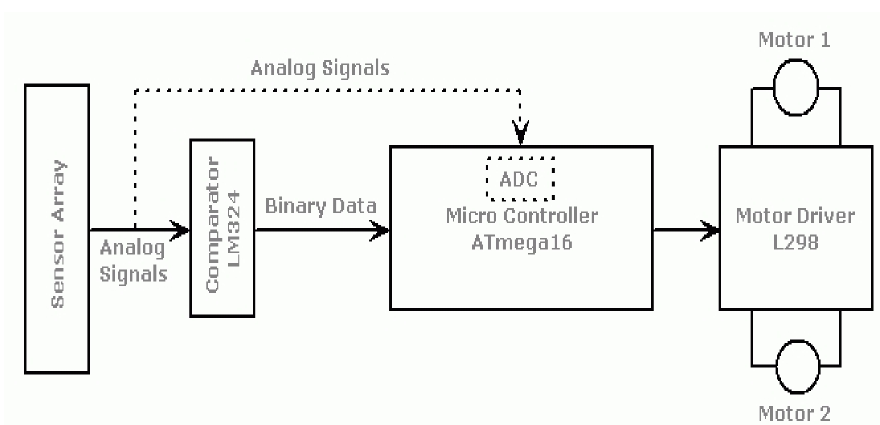


FIG 2.1 BASIC BLOCK DIAGRAM OF LINE FOLLOWER

This line following robot will easily leave its track from the black line drawn on the floor because it is an open loop system. This problem will make the motion of the robot to be rough. Line following robot even though follows the black line its motion still requires to be improved. Application of Digital PID algorithm can smooth the tracking motion. PID control is a closed loop system, which will provide feedback and correct the error that occurs with fast response.

2.3 INPUT SYSTEM

2.3.1 Sensors

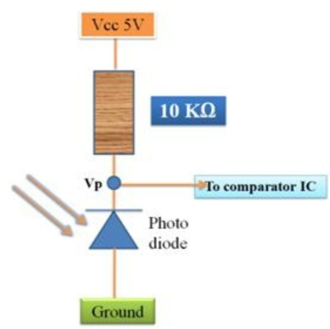


FIG 2.2 SENSOR CIRCUIT DIAGRAM

The robot uses IR sensors to sense the line; an array of five IR LEDs (TX) and sensors (Rx), facing the ground used in this setup. An analog signal is obtained in output, depends on the amount of light reflected back, which is provided to the comparator to produce 0s and 1s which are then fed to the microcontroller.

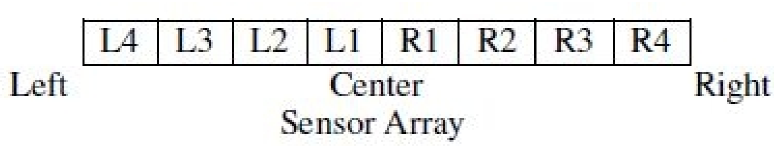


FIG 2.3 ARRAY OF SENSORS

The sensors on the left named as L1, L2, L3, L4 and R1, R2, R3, R4 on the right side.

Assumption should be considered that when a sensor is on the line it reads 0 and when it is off the line it reads 1.

The microcontroller correspondence to the algorithm given below decides the next movement, trying to position the robot such that L1 and R1 both read zero and the rest read one.

With sensors, robots can react and respond to changes in their environment in ways that appear intelligent or life-like. In general, the sensors are divided into two major types which are proximity sensor and distance measuring sensor.

2.3.1.1 Proximity Sensor

Proximity sensors only detect whether or not an object is within a predetermined range to the robot, whereas the distance measuring sensors determine the actual distance between an object and the robot. The robot processes the information received from sensors and reacts in a determined manner according to the design of the control system. For line follower to perform, the sensors required to sense the black line of the track and in order to keep track to center.

2.3.1.2 Infrared Sensor

The IR Sensor-Single is a general purpose proximity sensor. The module consist of a IR emitter and IR receiver pair. The high precision IR receiver always detects an IR signal.

The module consists of comparator IC. The output of sensor remains high if it is IR frequency and low otherwise. The on-board LED indicator helps the user for checking status of the sensor without using any additional hardware. The power consumption in this module is low. It’s output is digital in nature.

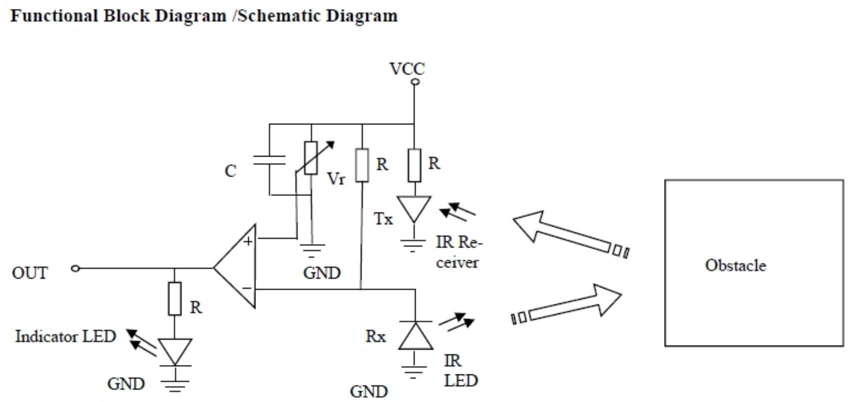


FIG 2.4 SCHEMATIC DIAGRAM OF IR SENSORS

The sensitivity of the IR Sensor is tuned using the potentiometer. The potentiometer is tunable in both the directions. Initially tune the potentiometer in clockwise direction such that the Indicator LED starts glowing. Once that is achieved, turn the potentiometer just enough in anti-clockwise direction to turn off the Indicator LED. At this point, the sensitivity of the receiver is at maximum. Hence, it has maximum sensing distance at this point. If the sensing distance (i.e., Sensitivity) of the receiver needed to be at minimum, then potentiometer tuned in the anti-clockwise direction from this point.

Further, if the orientation of both TX and Rx LED's is parallel to each other, such that both are facing outwards, then their sensitivity is at maximum. If they get moved away from each other, such that they are inclined to each other soldered at the end, then their sensitivity reduces.

Tuned sensitivity of the sensors is limited to the surroundings. For a particular surrounding, once tuned they will work perfectly until the IR illumination conditions of that region nearly constant. For example, if the potentiometer is tuned inside room/building for maximum sensitivity and then taken out in open sunlight, it will require retuning, since sun's rays also contain Infrared

(IR) frequencies, thus acting as a IR source. This disturbs the Receiver's sensing capacity. Hence it needs to be returned to work perfectly in the new surroundings.

The output of IR receiver goes low when it receives IR signal. This causes the output pin low since, though the IR LED is continuously transmitting, due to absence of obstacle, nothing is reflected back to the IR receiver. The indication LED is off. When an obstacle encountered, the output of IR receiver goes low; obstacle surface reflects IR signal, driving the output of the comparator low. The output connected to the cathode of the LED, turns it ON.

There are three common strategies for IR Sensor. They are:

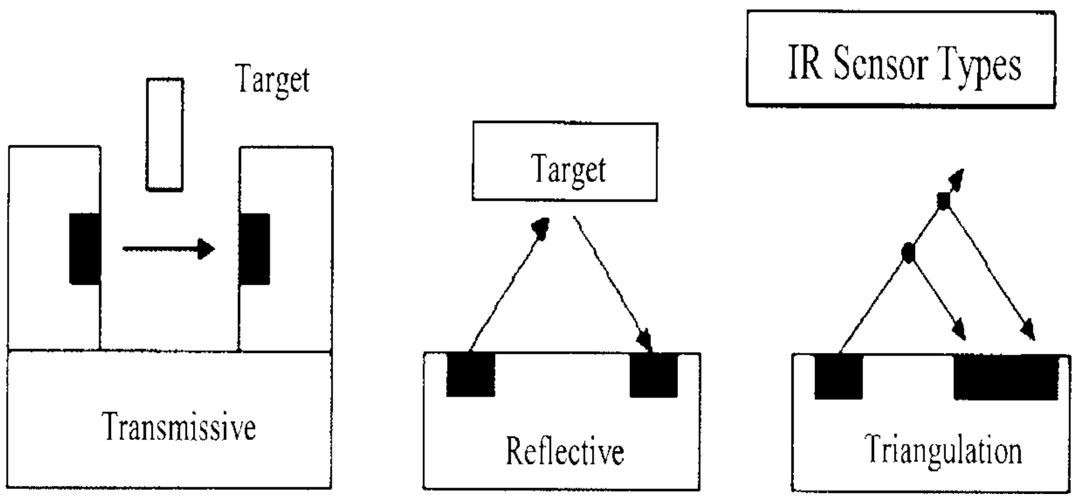


FIG 2.5 TRANSMISSION METHODS OF IR SENSORS

Transmit Configuration:

All instead of using the reflected light to indicate an object, this sensor uses the absence of light. The detector gets its input directly from infrared emitter. When the detector no longer senses any light, due to presence of object between the emitter and detector. Figure above shows the transmit configuration for infrared sensor.

Reflectance Configuration:

The reflective infrared sensor includes a source of light from the infrared emitter LED and a light detector from the photodiode or phototransistor arranged in package. Infrared emitter (LED) emits light thus acting as a source. When the light reaching on object, it is reflected back to the source. A detector at the source will receive the reflected light and indicate that an object is present.

Triangulation Configuration:

All these new rangers use triangulation and a small linear CCD array to compute the distance and presence of objects in the field of view. The basic idea is that the pulse of IR light is emitted by the emitter. The light travels out in the field of view and either hits an object or just keeps on going. In the case of absence of object, the light gets never reflected and the reading shows no object. If the light reflecting from an object, returning to the detector and creates a triangle between the points of reflection from the emitter to the detector.

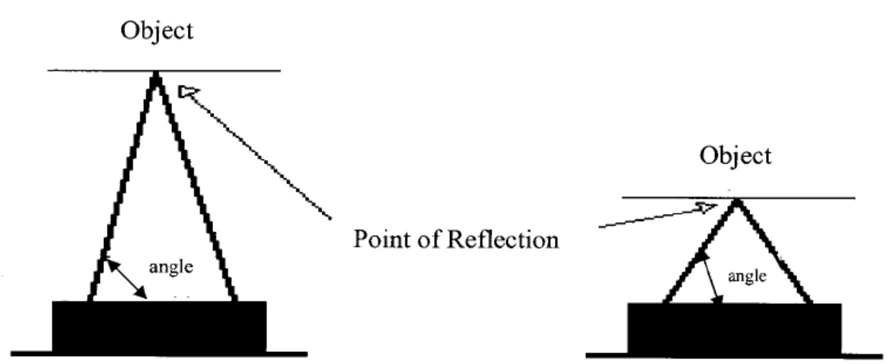


FIG 2.6 DIFFERENT ANGLES WITH DIFFERENT DISTANCES

The angle in this triangle varies based on the distance to the object. This receiver portion of these new detectors is a precision lens that transmits the reflected light onto various portions of the enclosed linear CCD array based on the angle of the triangle. The CCD array determines what angle the reflected light came back. Therefore, it calculates the distance to the object.

2.3.1.3 Arrangement of sensors

An array of sensors arranged in a straight row pattern is bolted under the front of the robot. It locates the position of line below the robot.

We can use any number of sensors. If we have low number then our robot movement is not smooth and it may face problems during sharp turns. If higher number of sensors were, used robot movement will become smooth and reliable for sharp turns; it requires complex programming for micro-controller and requires more hardware, which is its disadvantage. Thus, optimum number of sensors required.

The **distance between sensors** depend on

1. No of sensors used
2. Width of straight line (distance between sensors should be less than width of line).
3. Distance between sensors may not be constant (it depends on the logic).

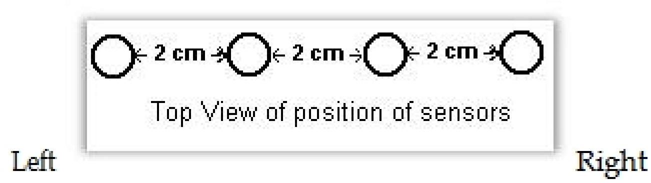


FIG 2.7 TOP VIEW OF POSITION OF SENSORS

2.3.2 Comparator:

Comparator is a device, which compares two input voltages and gives output high or low. In circuit diagram it is normally represented by a triangle having-Inverting (negative) Input, Non-Inverting (positive) Input (+), Vcc, Ground, Output.

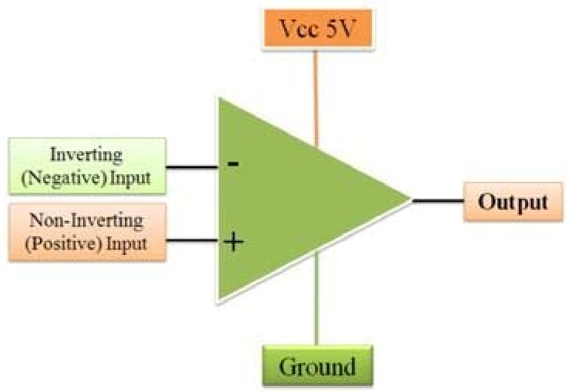


FIG 2.8 CIRCUIT DIAGRAM OF COMPARATOR IC

Properties of comparator:

If V+ > V- then V_O=Vcc (Digital High Output is 1)

If V+< V- then V_O=0 (Digital Low Output is 0)

Use of comparator in IR sensor

As above we see that two inputs are required for comparator. One input is from photo-receiver (like photo-diode),other is generated by potentiometer. The second voltage is reference voltage for that sensor.

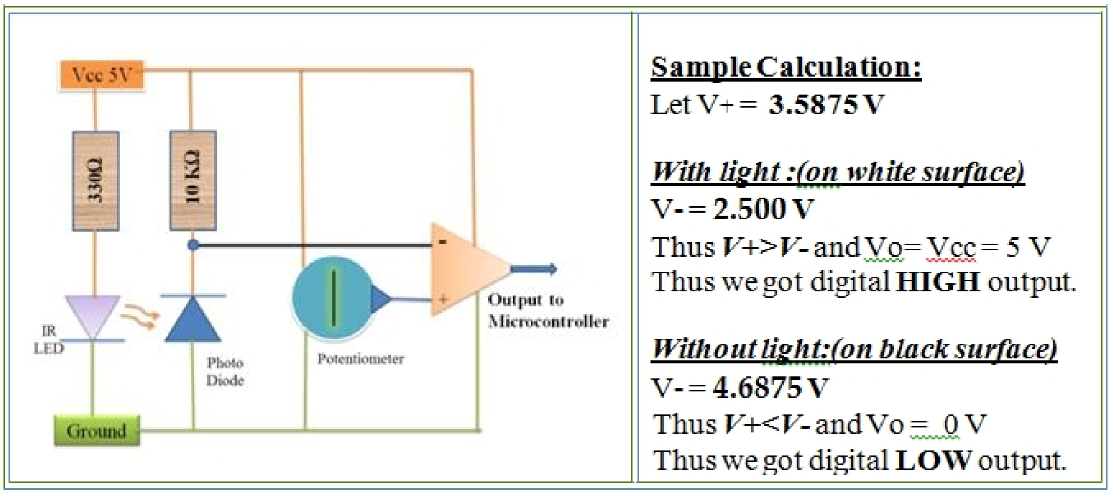
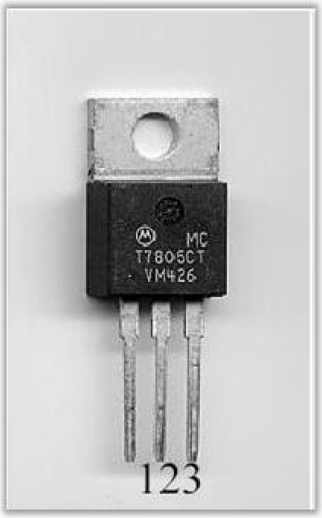


FIG 2.9 CIRCUIT DIAGRAM FOR IR SENSORS USING COMPARATOR

2.3.3 Voltage Regulator 78XX

Voltage regulators convert fixed DC output voltage from variable AC. 7805 and 7812 are commonly used. 7805 gives output fixed 5V DC voltage if input voltage is in range of 7.5V to 20V. They help to maintain a steady voltage level despite varying current demands and input voltage variations. If input voltage is <7.5 V then regulation will be improper i.e. if input is 6V then output is 5V or 4.8V, but there are parameters for the voltage regulators like maximum output current capability, and line regulation etc. that won't be proper.

To identify the leads of the 7805, you have to keep the lead downward (Fig a) and the writing to the side. The heat sink can be seen above the voltage regulator **(1-input, 2-gnd, 3-output).**



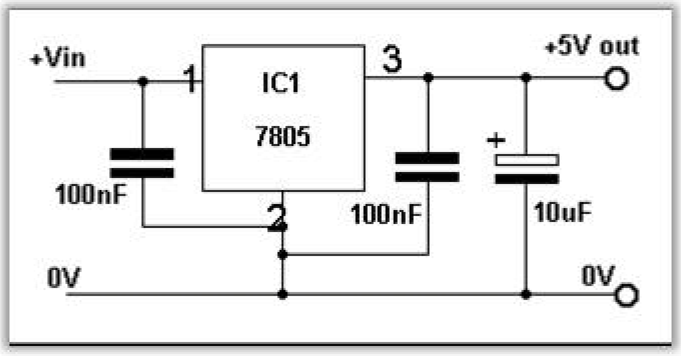


Fig 2.10 a **Fig 2.10 b**

FIG 2.10 CIRCUIT DIAGRAM FOR VOLTAGE REGULATOR

Fig b shows how to use 7805 voltage regulator. Coupling capacitors are good for regulation. There is no requirement for it in normal case. If 7805 used in analog circuit capacitor one should use, else the noise in the output voltage will be high. These are the mainly available 78xx IC's which are 7805, 7809,7812,7815,7824, high sensitive.

2.3.4 Processing System:

Processing system acts as the Brain of robot, generating desired output for corresponding inputs, in which microcontrollers are used. There are several companies nowadays that manufacture microcontrollers, for example ATMEL corporation, Microchip, Intel, Motorola etc. We will be using ATmega32L microcontroller in our robot. It is known as **AVR**.

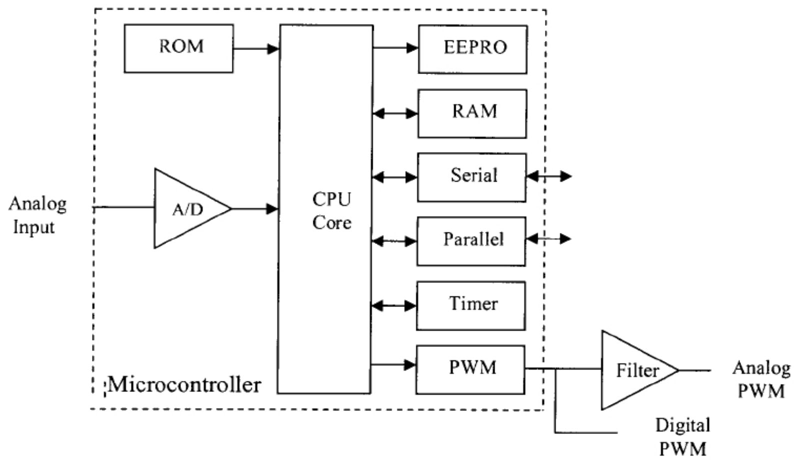


FIG 2.11 MICROCONTROLLER BASED SYSTEM

2.3.4.1 Microcontroller:

A microcontroller is an inexpensive single-chip computer. Single chip means that the entire computer system lies within the confines of the integrated circuit. The microcontrollers exist on the encapsulated sliver of silicon has features and similarities to our standard personal computers. The microcontrollers are capable of storing and running a program and interfacing to external devices that is most important feature.

The microcontroller consists a central processing unit (CPU), random-access memory (RAM), read-only memory (ROM), electrically erasable programmable read-only memory (EEPROM), input/output (I/O) lines, serial and parallel parts, timers, and other built-in peripherals, such as analog-to-digital (A/D) and digital-to-analog (D/A) converters.

2.3.4.2 Why Use Microcontroller?

Microcontroller provides inexpensive, programmable logic control and interfacing to external devices. The microcontroller has ability to store and run unique programs makes is highly versatile. For example, a microcontroller can be programmed to make decision (perform function) based on predetermined situations (I/O line logic) and selections. The microcontroller has ability to perform mathematic and a logic function allows it to mimic sophisticated logic and electronic circuit.

Features of Microcontroller:

1. Power on reset: Under most operating conditions, the microcontroller will go to a known RESET condition without the use of external circuitry. However, for ultra- reliable reset operation, a capacitor can be connected to the MCLR (reset) pin.

2. Watchdog timer: A technique which, when used, allows the processor to escape from an endless loop fault condition if the watchdog timer is not regularly reset.

3. Oscillator start-up timer: Internal circuitry holds the microcontroller RESET for approximately 18 ms after the master clear released.

4. Security EPROM fuses for code protection: When set, the security fuse prevents the internal program memory (the user program) from being read by external devices.

5. Power saving SLEEP mode: A software command that shuts the processor down until once again it is RESET.

6. EPROM selectable oscillator options: The microcontroller needs a frequency reference from which to operate; this can be chosen from RC components, quartz crystals or ceramic resonators. The oscillator section is hardware optimized to operate from 1 of 4 reference types. The following letters are the suffixes which are used:

Low cost RC oscillator : RC

Standard crystal/resonator : XT (200KHZ…4MHZ)

High speed crystal/resonator : HS(4……..20MHZ)

Power-saving low-frequency crystal: LP (32768Hz ... 200 kHz)

2.3.4.3 Why ATMEGA 32L?

Line follower robot requires simple microcontroller as it uses simple algorithms. Any microcontroller can use for that. But we use ATmega32L, because it has following extra features:

1. It is an **ISP (In System Programmable)** device. It means programming (Burning) of ATmega32 IC can be done without removing it from the system. **Note**: Programming (Burning) of a microcontroller means transferring the code from computer to microcontroller. We will explain burning afterwards.

2. It has on chip **PWM (Pulse Width Modulation)** circuit at three pins (Pin 15,16 and 17). We have explained PWM in another tutorial.

3. It has an inbuilt RC **oscillator**. (Oscillator can be clock generator circuit).

4. It consumes low power than other microcontrollers.

HARDWARE DETAILS

Basic hardware connections of ATmega32L

Pin 9 (Reset):

The use of reset pins to reset the ATmega8 microcontroller. Connecting the pin to ground solves the problem. In normal mode of execution, it should have a minimum of **2.7V**. Thus, it is connected to +5 volt through 10k ohm resistance .We should make sure it must be above 2.7 for proper execution of code.

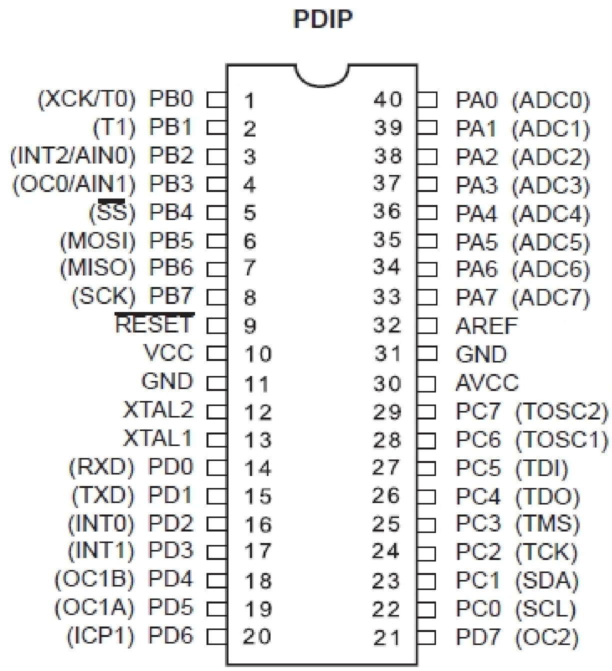


FIG 2.12 PIN CONFIGURATION OF ATMEGA 32L

Pin 10 and 30 (Vcc):

Pin 10 and 30 should be connected to Power supply. (2.7 to 5.5 volt for ATmega32L)

Pin 11 and 31 (Ground):

Pin 11 and 31 should be connected to Ground. The ground should be common to for the entire circuit.

Port A (PA7-PA0): Port A serves as the analog inputs to the A/D Converter. Port a can also serves as an 8-bit bi-directional I/O port, when the A/D Converter is not used. Port pins able to provide internal pull-up resistors (selected for each bit). The Port A output buffers having symmetrical drive characteristics with both high sink and source capability. When pins PA0 to PA7 being used as inputs and are externally pulled low, they have to source current if the internal pull-up resistors are activated. The Port A pins are usually tri-stated when a reset condition becomes active, even though the clock is not running.

Port B (PB7-PB0): Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers having symmetrical drive characteristics with both high sink and source capability. Port B pins as input are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are usually tri-stated when a reset condition becomes active, even though the clock is not running. Port B serving the functions of various special features of the ATmega32.

Port C (PC7-PC0): Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers having symmetrical drive characteristics with both high sink and source capability. Port C pins are externally pulled low will source current if the pull-up resistors are activated as can also be input. The Port C pins are usually tri-stated when a reset condition becomes active, even if still clock is not running. If the JTAG interface enables, the pull-up resistors on pins PC5 (TDI), PC3 (TMS) and PC2 (TCK) will be activated even if a reset occurs. The TD0 pin always tri-stated unless TAP states that shift out data entered. Port C serving the functions of the JTAG interface and other special features of the ATmega32.

Port D (PD7-PD0): Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers having symmetrical drive characteristics with both high sink and source capability. Port D pins are externally pulled low will source current if the pull-up resistors are activated as they can also be input. The Port D pins are usually tri-stated when a reset condition becomes active, even though the clock is not running. Port D serving the functions of various special features of the ATmega32.

XTAL1: Input is given to the inverting Oscillator amplifier and input to the internal clock operating circuit.

XTAL2: Output is taken from the inverting Oscillator amplifier.

AVCC: AVCC is the supply voltage pin for Port A and the A/D Converter. It's usually connected externally to VCC, even though the ADC is not used. If the ADC is used, it have to be connected usually to VCC through a low-pass filter.

AREF: AREF is analog reference pin used for the A/D Converter.

2.3.5. Analog to Digital Converter

The Atmega32 features a 10 bit successive approximation ADC. The ADC is usually connected to an 8 channel Analog Multiplexer which allows 8 single-ended voltage inputs constructed from the pins of Port A. The single-ended voltage inputs mostly referring to 0V (GND). The ADC contains a Sample and Hold circuit which ensures that the input voltage to the ADC is held at a constant level during conversion. It has a separate analog voltage supply, AVCC. The ADC has a separate analog supply voltage pin, AVCC. AVCC must not differ more than ±0.3V from VCC. Internal reference voltages of nominally 2.56V or AVCC are usually provided On-chip. The voltage reference is externally decoupled at the AREF pin by a capacitor for better noise performance.

The ADC converts an analog input voltage to a 10-bit digital value through successive approximation. The least value representing GND and the maximum value represents the voltage on the AREF pin minus one LSB. AVCC or an internal 2.56V reference voltage may be connected optionally to the AREF pin by writing to the REFSn bits in the ADMUX Register.

The internal voltage reference is decoupled by an external capacitor at the AREF pin to improve noise immunity.

The ADC is enabled by setting the ADC Enable bit, ADEN in ADCSRA. Voltage reference also the input channel selections will not go into effect until ADEN is set. The ADC usually does not consume power when ADEN cleared, so it is usually to switch off the ADC before entering power saving sleep modes.

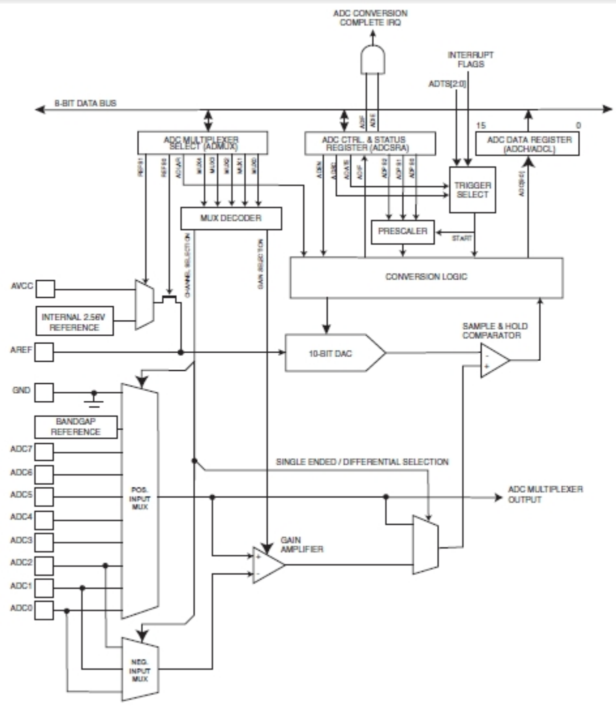


FIG 2.13 ANALOG TO DIGITAL CONVERTER SCHEMATIC

The ADC generates a 10-bit result that is presented in the ADC Data Registers, ADCH and ADCL. The result is by default, presented right adjusted, can optionally also be presented left, adjusted to set the ADLAR bit in ADMUX. If the result is as usual left adjusted and no more than 8-bit precision is required, it is enough to read

ADCH. Otherwise, ADCL must be read first, then ADCH, ensuring that the content of the Data

Registers usually belong to the same conversion.

The ADC has its own interrupt, which can be triggered when a conversion completes. When ADC Accesses to the Data Registers is prohibited between reading of ADCH and ADCL, the interrupt will trigger even though the result is lost.

2.4 Output System

2.4.1 Motor Driver

Motor driver is a current enhancing device; it can also be act as Switching Device. Thus, after inserting motor driver among the motor and microcontroller. Motor driver taking the input signals from microcontroller and generate corresponding output for motor.

IC L293D

This is a motor driver IC that can drive two motor simultaneously. Supply voltage (Vss) is the Voltage at which we wish to drive the motor. Generally, 6V for dc motor and 6 to 12V for gear motor are used, depending upon the rating of the motor. Logical Supply Voltage deciding what value of input voltage should be considered as high or low .So if we set Logical Supply Voltage equals to +5V, then -0.3V to 1.5V will be considered as Input Low Voltage and 2.3 V to 5V is taken into consider as Input High Voltage. L293D has 2 Channels .One channel is used for one motor.

•Channel 1-Pin1to 8

•Channel 2-Pin 9 to 16

Enable Pin is use to enable or to make a channel active .Enable pin is also called as Chip Inhibit Pin.

All Input (Pin No. 2,7,10 and 15) of L293D IC is the output from microcontroller (ATmeg32).

E.g.-We connected (Pin No. 2, 7, 10 and 15) of L293D IC to (Pin No. 14, 15, 16and 17) of ATmega32 respectively in our robots, because on pin 15 and 16 of ATmega32 we can generate PWM. All Output (Pin No. 3, 6,11and 14) of L293D IC goes to the input of Right and Left motor.

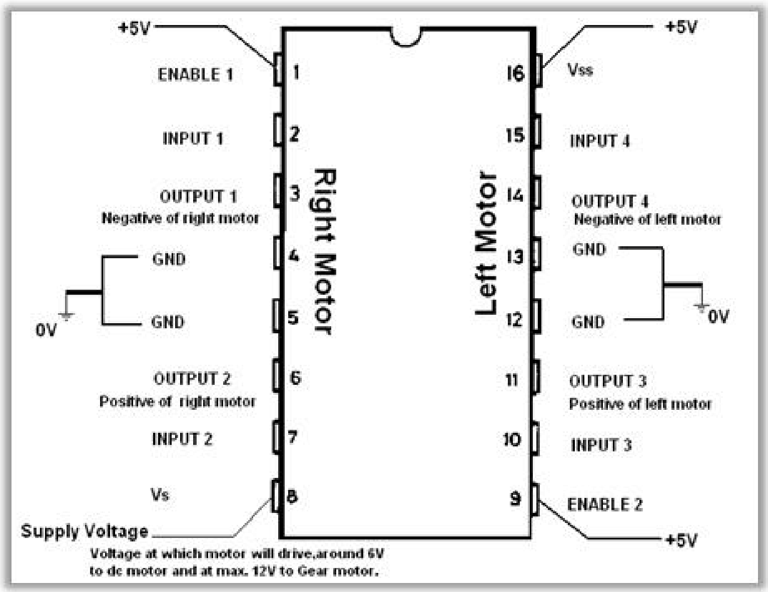


FIG 2.14 PIN DETAILS OF L293D

Output Connections

•OUTPUT 1 (Pin No 3) --- It is the negative Terminal of Right Motor

•OUTPUT 2 (Pin No 6) ---It is the positive Terminal of Right Motor

•OUTPUT 3 (Pin No 10) --- It is the positive Terminal of Left Motor

•OUTPUT 4 (Pin No 14) --- It is the negative Terminal of Left Motor

For one motor:

Positive Terminal (+ve)	**Negative Terminal (-ve)**	**Motor Output (O/P)**
0	0	No movement
Vss	0	Straight
0	Vss	Reverse
Vss	Vss	No Movement

TABLE 2.1 MOTOR MOVEMENTS WITH POWER SUPPLY

Input	***Input***	***Input***	***Input***	***Output***	***Output***	***Output***	***Output***	***Motors Output***		***Movement***
								Right	***Left***	
Low	High	High	Low	0	Vss	Vss	0	Straight	Straight	Straight
Low	High	Low	Low	0	Vss	0	0	Straight	No mov	Left Turn
Low	Low	High	Low	0	0	Vss	0	No mov	Straight	Right Turn
Low	High	Low	High	0	Vss	0	Vss	Straight	Reverse	Sharp Left
High	Low	High	Low	Vss	0	Vss	0	Reverse	Straight	Sharp Right
High	Low	Low	High	Vss	0	0	Vss	Reverse	Reverse	Backward

TABLE 2.2 OUTPUT FOR VARIATION IN INPUT

2.4.2 Electric Motor

Motor is a device that converts any form of energy into mechanical energy or imparts motion. In constructing a robot, 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 produces by the permanent magnet to make the motor rotate. There are generally three basic types of motor, DC motor, even servomotor and stepper motor, which are always being used in building a robot.

2.4.2.1 DC Motor

The DC motor is a device that converts 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.

Why DC motors?

DC motors are most easy for controlling. One dc motor has two signals for its operation. Reversing the polarity of the power supply across it can change the direction required. Speed can be varied by varying the voltage across motor.

2.4.2.2 Use of gears

The DC motors don't have enough torque to drive a robot directly by connecting wheels in it. Gears are generally used to increase the torque of dc motor on the expense of its speed.

Mathematical Interpretation:

Rotational power (Pr) is given by:

Pr= Torque (T) X Rotational Speed (ω)

$$T = \frac{\mathrm{Pr}}{\omega}$$

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

$$\boxed{T \,\alpha\, \frac{1}{\omega}}$$

For increase in the value of torque, we have to loose speed.

2.4.2.3 Why two motors?

By using two motors, we can move our robot in any direction. D**ifferential drive is the steering mechanism of robot**.

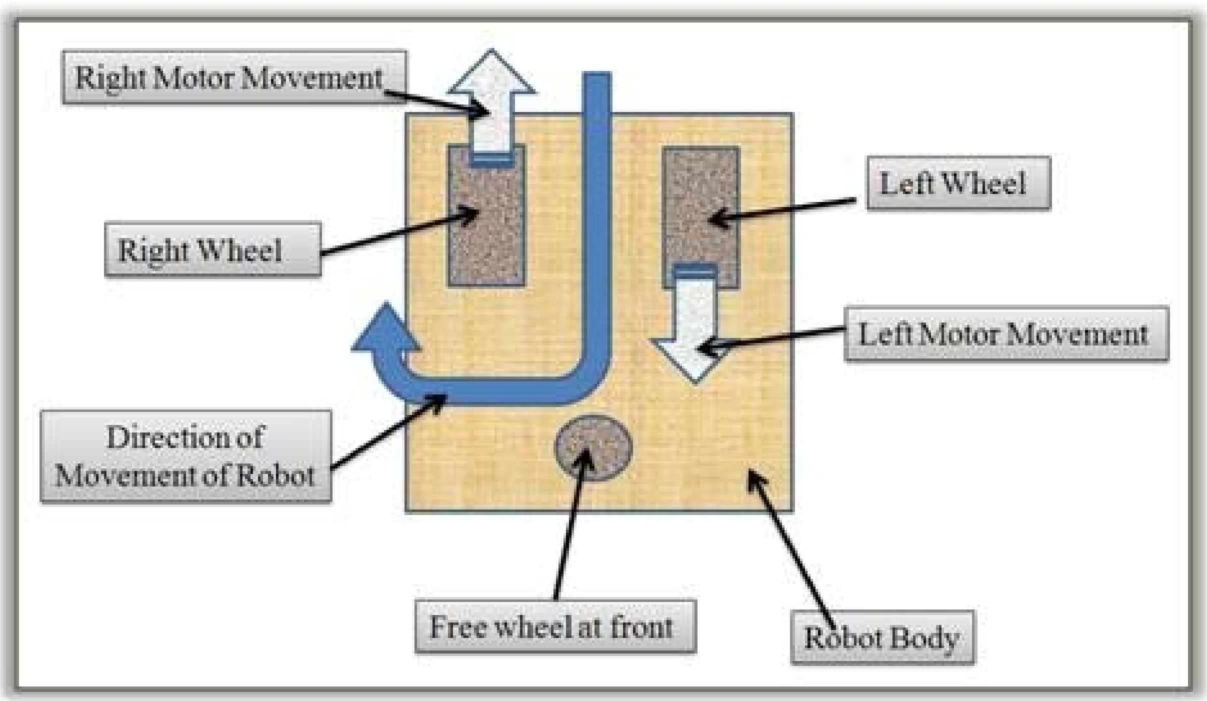


FIG 2.15 DESCRIPTION OF VARIOUS PARTS

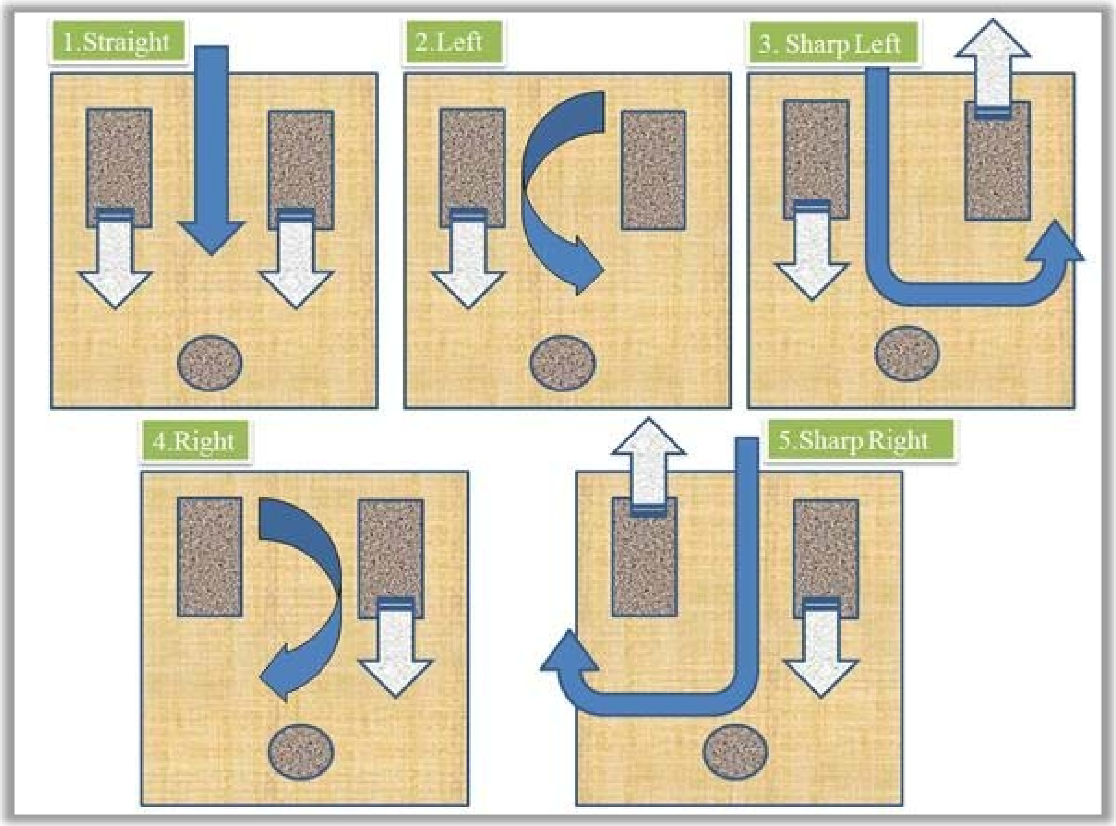


FIG 2.16 DIFFERENT TYPES OF MOVEMENT OF ROBOT

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

TABLE 2.3 MOVEMENT OF ROBOT WITH VARIATION IN MOTOR DIRECTION

2.5 CONTROLLERS

In a process control system controller's function is to influence the control system through control signals so that the value of the controlled variables equal to that of the value of the reference. Controller can be known as the "Brain" of process control system. Controller generates a control signal sending to final control element depending upon the deviation between the set point and the measured value of the controlled variable. controller mode is the way in which the controller responds to deviation. The sensor, the transmitter, and the control valves are located near the process itself, while the controller is located on the panel or is residing as a program inside the computer memory.

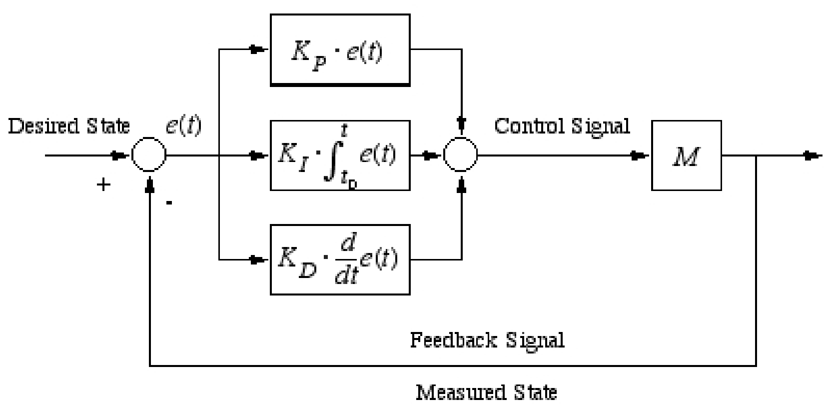


FIG 2.17 GENERAL CONTROLLER STRUCTURE

PID Controller Theory:

Proportional-Integral-Derivative (PID) control is the most common control algorithm used in industry and has been universally accepted. The PID controller attributes are partly to their robust performance in a wide range of operating conditions and partly to their functionality simple, allowing engineers to operate them in a simple manner. As the name suggests, a PID algorithm consists of three basic coefficients: proportional, integral and derivative. These gains are varied to achieve an optimal system response.

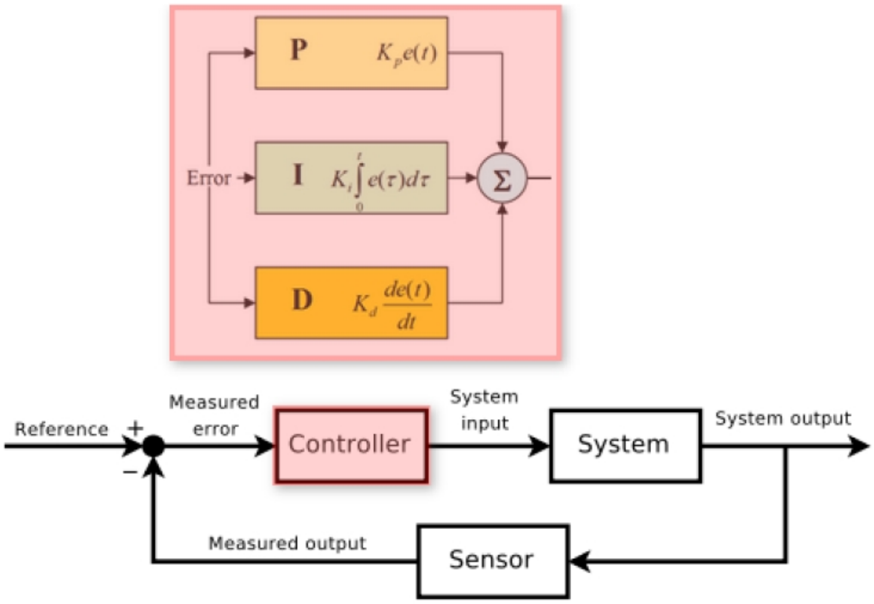


FIG 2.18 GENERAL STRUCTURE OF PID CONTROLLER

The basic structure of a system with PID control implemented is illustrated above. The system output (also called the process variable) with a sensor and compared to the reading to the reference value (also called the set point). Reference and the measured output are compared and the result is an error value which is used in calculating proportional, integral, and derivative responses. Summing the three responses to obtain the output of the controller. The output of the controller is used as an input to the system you wish to control, changing some aspect of the system. For example, if motor is controlled, the controller would provide more or less current. If controlling the flow of a fluid, the controller would cause a valve either to open or closed. The output of the system gets measured and the process continues. Iteration of the control loop can be known as completion of the process in single time.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t)$$

After Laplace transfer we get

$$\frac{u(s)}{e(s)} = K_p(1 + \frac{1}{\tau_i s} + \tau_d s)$$

Where

K_p = Proportional gain

K_I= Integral gain

K_D= Derivative gain

e: Error=Set point –Process Value.

t : Instantaneous time

t_I: Integral time

t_D: Derivative time

2.5.1 Proportional controller

The proportional term is produced, in which an output value that is proportional to the present error value. The proportional response is adjusted by multiplying the error with a constant *Kp*, called proportional gain constant.

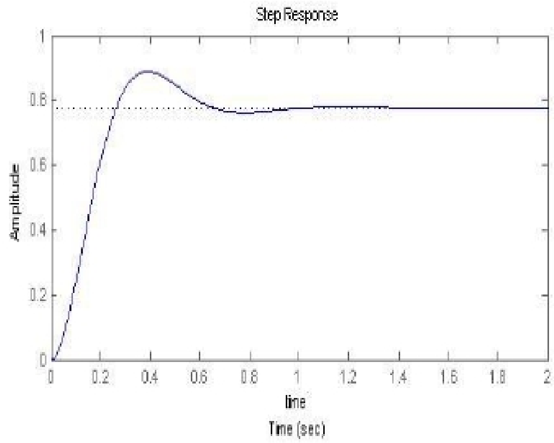
The proportional term is given as:

$$u(t) = K_p e(t)$$

Or

$$\frac{u(s)}{e(s)} = K_p$$

A high proportional gain resulting in a large change in output for a given change of the error. If the proportional gain is very high, the system becomes unstable. Whereas, a small gain results in a small output response to a large input error and a low responsive or less sensitive controller. If the proportional gain is much less, the control action may be too small when responding to the system disturbances. Tuning theory and industrial practice implies that, the proportional term must contribute to the bulk of the output



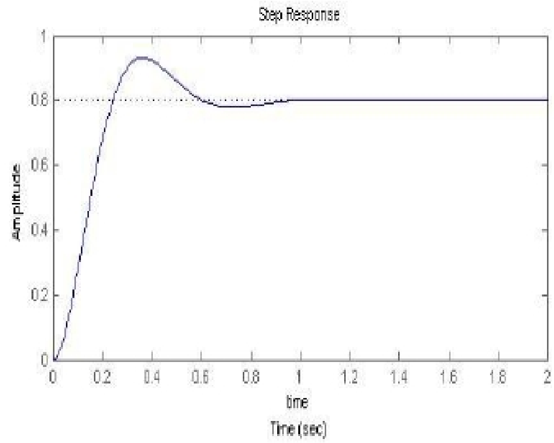


FIG 2.19 VARIATION OF PROPORTIONAL CONTROLLER WITH VARIATION OF Kp

2.5.2 Integral controller

The contribution of the integral term is proportional to both the magnitude of the error and the duration of error. In a PID controller the integral is the sum of the instantaneous error over time and gives the accumulated offset that should have been previously corrected. Accumulated error gets multiplied by the integral gain (Ki) and added to the controller output.

The integral term is given by:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau$$

Or

$$\frac{u(s)}{e(s)} = K_p(1 + \frac{1}{\tau_i s})$$

The integral term accelerates the movement of the process towards the set point and eliminates the residual steady-state error which occurs with a pure Proportional controller. Integral term responds to accumulated errors of the past, it can result the present value to overshoot the set point value.

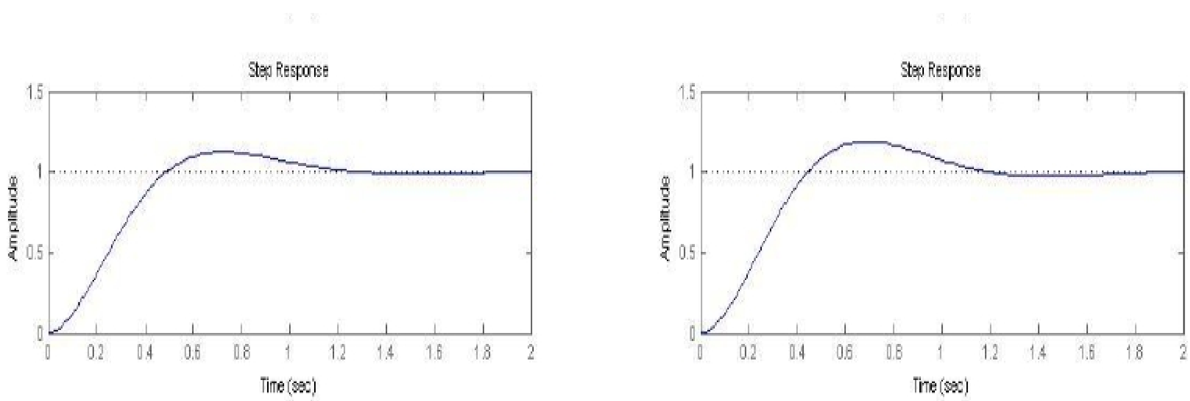


FIG 2.20 VARIATION OF INTEGRAL WITH KI

2.5.3 Proportional Derivative

PD controller is used in the process where offset value is acceptable, where some anticipation can be considered and should have low noise. The equation of PD Controller is:

$$\boxed{m(t) = \overline{m} + K_c e(t) + K_c \tau_D \frac{de(t)}{dt}}$$

The ideal transfer function of PD controller is given by:

$$G_c(s) = \frac{M(s)}{E(s)} = K_c(1 + \tau_D s)$$

Whereas the actual transfer function is given as :

$$\boxed{G_c(s) = \frac{M(s)}{E(s)} = K_c \left[\frac{(1 + \alpha)\tau_D s + 1}{\alpha \tau_D s + 1} \right]}$$

2.5.4 Proportional Integral Derivative Controller (PID)

PID control is derivative action in nature, also known as the rate action, or pre-act. Its purpose is foe anticipating where the process is headed through looking at the time rate change of the error, and its derivative. The equation is given as:

$$\boxed{m(t) = \bar{m} + K_c e(t) + \frac{K_c}{\tau_I} \int e(t)\, dt + K_c \tau_D \frac{de(t)}{dt}}$$

Where t_D.= derivative (or rate) time. .The PID controller contains three terms, Kc or PB, t_I and t_D, need to be adjusted, (tuned) to obtain satisfactory control. The derivative action provides controller the ability to anticipate the direction of the process that is, to through calculation of the derivative of the error. The tuning parameter decides the amount of anticipation required.

Assumption is taken that the inlet process temperature goes on decreasing by some amount and the outlet temperature starts decreasing correspondingly, at time ta, of the error quantity is positive and small. Hence the amount of control correction given by the proportional and integral modes is less.

However, the derivative of this error, due to high slope of curve is large and positive; make the control correction given by the derivative mode large. By seeing at the derivative of the error, the controller knows that the controlled variable traverses away from set point at a faster rate, and it uses this information to help in controlling. At time tb, the error is positive and is much larger than before.

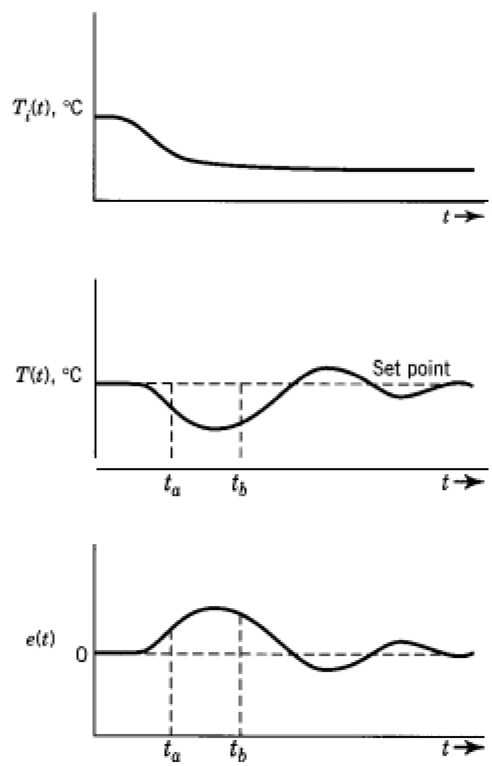


FIG 2.21 HEAT EXCHANGER CONTROL

The amount of control correction given by the proportional and integral modes is much larger than before and is still added to the output of the controller to open the steam valve more. However, the derivative of the error is negative at this time, signifies that the error is decreasing;

the controlled variable starts to come back to set point. From this fact, the derivative mode starts subtracting from the other two modes, since it recognizes that the error is decreasing.

This algorithm results in reduction of overshoot and decrease in oscillations around set point.

PID controllers are considered for use in slow processes (processes with many time constants or dead time) such as temperature loops, which are generally free of noise.

Fast processes (processes with short time constants) are more susceptible to process noise. The application of the derivative mode results in the amplification of the noise, due to the derivative of the fast-changing noise is a very large value. Processes with long time constants are mostly damped and, consequently, are less susceptible to noise. In the case of a slow process with a noisy transmitter, however, the transmitter should be fixed or the noise filtered before the PID controller is used.

As per the literature survey, Jim Remington (2007) had conducted studies relating to the tuning of PID controller parameters. It had been concluded that the initial tuning process of the controller should involve the tuning of the proportional parameter with the differential and integral parameters set to zero. The bot should be made to follow the line only with this parameter. After the bot has somewhat succeeded in following the line the differential parameters should be changed to control the oscillatory response. Perfect line following can be obtained by changing these two parameters only. The integral term should be introduced only in special cases.

CHAPTER 3

SOFTWARE DEVELOPMENT

3.1 **Programming and Simulating**

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 AVR Studio 4 and was burnt onto the flash memory of the microcontroller using Sinaprog Version 1.5. The program is coded using AVR studio and is then compiled to form a ".hex" file which can then be burnt into the microcontroller.

3.2 Program

```
#include <avr/io.h>

#include <avr/interrupt.h>

#define F_CPU 8000000UL

#include <util/delay.h>

#include <avr/adc.h>

#include "lcd.h"

void InitPWM()

{

TCCR0|=(1<<WGM00)|(1<<WGM01)|(1<<COM01)|(1<<CS00);

DDRB|=(1<<PB3);

TCCR2|=(1<<WGM20)|(1<<WGM21)|(1<<COM21)|(1<<CS20);

DDRD|=(1<<PD7);

}

void SetPWMOutput1(int duty)

{
```

```
OCR0=200+duty;
}
void SetPWMOutput2(int dut)
{
OCR2=140-dut;
}
 void InitMotor()
{
DDRB|=(1<<PB0)|(1<<PB2)|(1<<PB4)|(1<<PB6);
PORTB|=(1<<PB0)|(1<<PB2)|(0<<PB4)|(0<<PB6);
}

Int main()
{
InitPWM();
//uint8_t b=0;
InitMotor();
int e=0;
int pe=0;
int Kp,Ki,Kd,P,I,D,O;
Kp=50;
Ki=0;
```

```
Kd=4;
P=0;
I=0;
D=0;
O=0;

LCDInit(LS_BLINK|LS_ULINE);
LCDClear();
DDRA=0x00;

while(1)
{
if((PINA & 0x07)== 0b00000101)
{
e=0;
LCDWriteIntXY(0,0,101,3)
}
if((PINA & 0x07)== 0b00000110)
{
e=1;
LCDWriteIntXY(0,0,110,3);
}
if((PINA & 0x07)== 0b00000011)
```

```
{
e=-1;
LCDWriteIntXY(0,0,111,3);
}
if((PINA & 0x07)== 0b00000001)
{
e=-1;
LCDWriteIntXY(0,0,1111,4);
}
if((PINA & 0x07)== 0b00000100)
{
e=1;
LCDWriteIntXY(0,0,1110,4);
}
P=Kp*e;
I=I+e;
I=I*Ki;
D=(e-pe)*Kd;
O=P+I+D;
pe=e;
if(O<0)
{
LCDWriteIntXY(13,1,50,3);
```

```
}
Else
{
LCDWriteIntXY(13,1,O,3);
}
_delay_ms(100);
SetPWMOutput1(O);
SetPWMOutput2(O);

}

}
```

CHAPTER 4

OUTPUT AND CONCLUSION

Output:

The robot was successfully able to follow the line for Kp equal to 50 and Kd equal to 4. However the output was taken under a certain lighting condition and a similar response might not be obtained for a different ambient lighting condition. The sensors used would need to be calibrated to produce the perfect output.

Conclusion:

In the project, we studied the design of a line following bot and the implementation of PID controller. The parameters of the controller were designed using the manual tuning method as per the literature survey. Due to limitations in the hardware (Motor and Sensors), perfect control was not obtained. The method of tuning utilized was the manual tuning method as per the literature survey. However the response of the system was better as compared to that of a simple open loop controller. The amount of overshoot was also reduced using the above mentioned techniques.

Future Work:

Future Scope of this project involves the use of Ziegler Nichols method for tuning the PID controller. This can be done by finding out the transfer function of the process and tuning the value of the proportional gain up to the extent when the output starts oscillating. Noting down the gain and time period of oscillations, the value of the integral and derivative gain can also be determined.

REFERENCES

[1] Kanayama, Y., N. Miyake, "Trajectory Generation for Mobile Robots", Robotics Research, vol. 3, pp. 333-340, The MIT Press, 1986.

[2]Komonya, K., S. Tachi, K. Tanie, "A Method for Autonomous Locomotion of Mobile Robots," Journal of Robotics Society of Japan, vol. 2, pp.222-231, 1984.

[3]Jim (2005) PWM/PID/Servo Motor Control using Orangutan from Pololu. [Online]. Available at: http://www.uoxray.uoregon.edu/ (Accessed 5th May, 2013)

[4]PID for Line Following. (2007) *PID for Line following*. [Online]. Available at http://www.chibots.org/index.php?q=node/339 (Accessed on 3rd May, 2013)

[5]Pid Control Of Line Following Robot Cheu Kai Wee TJ223.P55.C43 2007.[Online]. Available on google as PDF.(Accessed on 3rd May, 2013)

[6]ATMEGA 32L DATASHEET [Online]. Available at <http://www.atmel.in/devices/ATMEGA32.aspx> (Accessed on 3rd May, 2013)