UNIVERSITI MALAYSIA PAHANG

BORANG PENGESAHAN STATUS TESIS◆

JUDUL: **MOBILE NAVIGATION SYSTEM USING INFRARED DETECTOR**

SESI PENGAJIAN: 2009/2010

Saya **MOHD HASNAN BIN NORDIN (860204-30-5245)**
(HURUF BESAR)

mengaku membenarkan tesis (Sarjana Muda/~~Sarjana~~ /~~Doktor Falsafah~~)* ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hakmilik Universiti Malaysia Pahang (UMP).
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. **Sila tandakan (√)

[ ]	SULIT	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
[ ]	TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
[√]	TIDAK TERHAD	

Disahkan oleh:

(TANDATANGAN PENULIS) (TANDATANGAN PENYELIA)

Alamat Tetap:

NO.15, JALAN 5
TAMAN MAWAR 72200 BATU KIKIR
NEGERI SEMBILAN

MOHD FADHILBIN ABAS
(Nama Penyelia)

Tarikh: **23 NOVEMBER 2009**

Tarikh: : **23 NOVEMBER 2009**

CATATAN:
* Potong yang tidak berkenaan.
** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali tempoh tesis ini perlu dikelaskan sebagai atau TERHAD.
◆ Tesis dimaksudkan sebagai tesis bagi Ijazah doktor Falsafah dan Sarjana secara Penyelidikan, atau disertasi bagi pengajian secara kerja kursus dan penyelidikan, atau Laporan Projek Sarjana Muda (PSM).

"I hereby acknowledge that the scope and quality of this thesis is qualified for the award of the Bachelor Degree of Electrical Engineering (Control & Insturmentation)"

Signature : ______________________________

Name : MOHD FADHIL BIN ABAS

Date : 23 NOVEMBER 2009

MOBILE NAVIGATION SYSTEM USING PASSIVE INFRARED DETECTOR

MOHD HASNAN BIN NORDIN

This thesis is submitted as partial fulfillment of the requirements for the award of the Bachelor of Electrical & Electronic Engineering (Hons.) (Control & Instrumentation)

Faculty of Electrical & Electronics Engineering
Universiti Malaysia Pahang

23 NOVEMBER, 2009

"Hereby, I declare that this thesis entitled "Mobile Navigation System Using Passive Infrared Detector" is a result of my own research and idea except for works that have been cited clearly in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any degree."

Signature : ____________________

Author : MOHD HASNAN BIN NORDIN

Date : 23 NOVEMBER 2009

To my beloved mother and father

ACKNOWLEDGEMENT

In preparing this thesis, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thoughts. In particular, I wish to express my sincere appreciation to my supervisor, Mr. Mohd Fadhil Bin Abas, for encouragement,guidance, critics and friendship. He has given me important advices, suggestion, motivation and guidance throughout the whole project.

I would also like to thank my beloved parents that have always given me inspiration to go forward and do my best through various obstacles and difficulties.

I would also like to express my gratitude to all the technician who have guided and taught me during my PSM project especially to Mr. Mohd . Azlan Bin Sayuti, Mr. Abd Latip Bin Hj Idris ,and Mr. Mohd Salmizan Bin Mohd Zain for their guidance, advices and motivation. They have taught me invaluable knowledge which is prerequisite in completing this project. Without their continued support and interest, this thesis would not have been the same as presented here.

Furthermore, I would like to express my heartfelt appreciation to my fellow friends and course mates for their help, ideas and suggestion.

ABSTRACT

The objectives of this project is to make the maze mapper robot to move in the obstacle area. This robot are divided into three part that is microcontroller, DC motor and sensor. This robot use PIC microcontroller technology as the brain of the robot to function, DC motor for the motion or movement so that the robot can move and the infrared sensor as the detector for the robot to detect the obstacle wall which stop the robot movement and avoid it from crash those wall. It is important by knowing the background knowledge of microcontroller in order to understand a complete control system of the robot function. The schematic circuits for connection for connection of sensors and motors through PIC microcontroller are designed. Next, interfacing between PIC (microcontroller chip) and input/output hardware are drawn and integrated. Soldering process is carried out after the circuit of the electronic components on the strip board are designed. A program in PIC PBasic language source code to control the maze mapper robot is then developed. Finally, maze is constructed for the robot to move in and for the robot to map/ recognize the maze. Sensors that used for the robot are Infrared transmitters, which will be placed at the front, left, and right side of the robot. These sensors emit the infrared light, which reflected when it close to the wall. When a wall is detected, a signal sent back to the phototransistor. The microcontroller reads the signal from the sensors, and determines the appropriate DC motors. This robot will able to move in and out of maze automatically with minimum wall contact upon completion.

ABSTRAK

Objektif projek tahun akhir (PSM) ini ialah mereka bentuk *maze mapper robot* dan berfungsi mencari jalan keluar serta mengurangkan kadar pelanggaran robot dengan dinding maze tersebut apabila ianya masuk ke dalam litar yang mempunyai laluan berhalangan. Robot ini terbahagi kepada 3 bahagian iaitu mikrokawalan, DC motor dan peranti penderia (sensor). Robot ini menggunakan technology mikrokawalan PIC sebagai otak pada robot itu supaya berfungsi, DC motor sebagai pergerakan robot dan juga peranti penderia infra merah sebagai pengesan supaya robot itu dapat mengesan kehadiran dinding maze tersebut dimana ianya akan memberhentikan pergerakan robot dan mengurangkan pelanggaran robot dengan dinding maze. Adalah penting untuk mengetahui latar belakang ilmu pengetahuan mikrokawalan dalam memahami system kawalan yang lengkap pada robot tersebut. Litar skematik untuk sambungan antara DC motor dan peranti penderia infra merah melalui mikrokawalan PIC telah direka. Seterusnya sambungan antara PIC dan masukan/keluaran *hardware* telah dilukis dan disambungkan. Process memateri dilakukan selepas component elektronik pada strip board direka, dan dilukiskan. Program arahan dalam bahasa PIC PBasic untuk mengawal pergerakan robot didalam litar telah diprogramkan. Process akhir ialah pembinaan litar maze dibina untuk pergerakan robot masuk dan keluar disamping mengingati laluan pada litar tersebut. Peranti penderia infra merah yang digunakan ialah infra merah transmitter dimana ketiga-tiga penderia tersebut dipasang didepan, kiri dan kanan robot tersebut. Penderia tersebut mengesan kehadiran dinding maze dan menghantar isyarat ke mikrokawalan. Mikrokawalan akan membaca isyarat diberikan oleh peranti infra merah tersebut dan menghantar arahan ke motor. Dengan arahan yang diberikan oleh mikrokawalan robot ini dapat bergerak masuk dan keluar semasa didalam litar secara automatic serta mengurangkan perlanggaran dinding maze tersebt.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1. Background

Recently, there have been many breakthroughs in artificial intelligence. Robots are able to do increasingly complex behaviors without human intervention. One of the classic examples of testing intelligence is putting a rat in a maze. Our objective was similar. The robot that I used was a Maze Mapper Robot PIC programmed to navigate and map an unknown maze.

To do this, I had to make the Maze Mapper Robot detect the walls of the maze and align itself to them. This allowed it to avoid drifting into the walls when moving through the maze. Another algorithm allowed the Maze Mapper Robot to map what it had seen, remember where it was in the maze and choose its next direction. With these two processes working together within a finite state automation, the Maze Mapper Robot was able to explore, map, and complete a maze.

The Mobile Navigation using passive infrared detector or Maze Mapper Robot is a small robot that can find its way through and out of a constructed maze map using hardware and software interface. This robot similarly like micromouse robot which many final year student had designed this project by using HC11 microcontroller as the brain of the robot which is only based on the instruction programming to find its way out of the map. The problems that always occur in the micromouse robot are this robot cannot memorize the map in the mapping area and cannot find its way out in the maze map. For this project, the Maze Mapper Robot are designed by using PIC18F4550 microcontroller. By using PIC microcontroller on the robot, it is not only just be able to find its way out in the maze map area based on the instruction programming but it can memorize the maze map so that when the robot get into the maze map for the second time, it can memorize which the pathway that it had went when it figured out how to exit the maze map previously.

Basically, the Maze Mapper robot has to follow the track without any external help, and has to carry its' own power source. Almost without exception, these robot use battery-powered electric motors and light sensitive sensors. This project are divided into 3 major part generally that are same as the micromouse robot part which are the microcontroller, IR sensor and DC motor. In analogy of this project, the sensor will be the eyes of the robot to detect the maze wall, the dc motor as the legs for the robot's movement and the microcontroller as the brain of the robot so that the robot can move functionally and avoid the mobile robot from crash the maze map wall.

1.2. Objective

The objective of this project is to developed Maze Mapper Robot hardware. The maze map algorithm written as a computer program will be embedded in a hardware device which is the microcontroller. This firmware is intended to be used for dc motor and infrared sensor that needs to operate under Maze Mapper Robot control system and avoid the robot from crash the maze map wall.

1.3. Scope

The project consists of 4 scopes. The first scope is architecture of the mechanical hardware robot. The architecture of the mechanical robot is defined as the mechanical design of the robot that suitable for the maze map. The second scope is implemented the IR sensor, DC motor on the robot. For this project, three IR sensors and two DC motors are used as the detector and the movement of the robot. Sensors will act as the detector when they detect the signal of absence wall and send the signal to the PIC microcontroller to read the signal that received. The PIC microcontroller will make the map's image for the robot based on the type of the map during execution by using algorithm. DC motors will make movement for the robot as they receive the signal or information from the microcontroller.

The third scope is develop of the robot electronic hardware. The electronic circuit of the robot are designed by using the PIC18F4550 microcontroller that acts as the brain of the robot. The microcontroller will control the sensors and motors based on the program that made. The fourth scope is develop the firmware of the robot by using PBasic language. This language will make the robot perform based on the instruction that had made. The language were made from the map algorithm and other instruction to control the electronic device such as sensors and motors based on the control system.

The result of the project is the robot will memorize the map when it move into the maze map. Usually for the first time the robot will try searching the way out of the map but when it gets into the map for the second time, it can find its way out faster than the first time. The performance of the robot which similarly like the human. The analogy of the similarity for example if a human gets into a new house and he want to locate where is the bathroom. It will takes time for him to find the bathroom at the first time. When he find the bathroom, he will memorize the location/coordinate the place. After that he will find the bathroom for the second time with less time than the first time he gets into the house. This execution of the robot is called intelligent control which it memorize the location when it gets into the maze map. The time will be taken from first and second time to get the different time when execute.

1.4. Application

Although the designs of this Maze Mapper Robot seem simple at first, its application can be applied in a search and rescue situation where this small robot can navigate and search for victim trapped inside a collapsed building or any situation which the use of rescue dogs in not advisable. With a few sophisticated gadgets added on the robot, it can transmit a clear digital picture to the search party above so that they can locate and save the survivors. The design can be upgraded to be an automatic vacuum cleaner with the abilities to sense and avoid obstacles when cleaning the floor. Whenever the battery level drop to a minimum state, it can recharge itself by plugging its' socket to the power supply without any external help.

1.5. Main Project

The main idea of this design is to design a two wheel using DC motor robot which can interact with its surrounding through three IR sensors and a PIC microcontroller.

CHAPTER 2

LITERATURE REVIEW

2.1. Introduction

Before any development can take place, a research must be carried out on all the possible components that will go into the Maze Mapper Robot. Most of the research was done through the websites, books, journals, and articles. Much can be learned by questioning other candidate which make the similarity of the project such as micromouse, obstacle robot, and navigation system robot about the features of the robot about issues they experienced while developing the Maze Mapper Robot. Only then, issues such as microcontroller, electronic circuit design, wheel & DC motor implemented, and sensor placement are to be decided. This is perhaps the most critical design phase, which will affect the rest of the project.

2.2 Microcontroller and features

The Microcontroller is the core of the system. All the other functions where built around this function and connected to this function. The analogy of the microcontroller as the brain of the device to function based on the application which the programmed had applied in the microcontroller. In short, it controls all the circuitry system. Based on the thesis that taken, it state that *any processors are well suited to driving a device such as robotics especially micromouse provided as long as it is user friendly, has low power consumption, and a EEPROM memory on chip to drive the robot* [1]. In particular, the microprocessor must be able to control the motors with encoder, drive the sensor emitters, drive the speed of driver motors and read the sensor results. In this case, the small, cheap, easy to use which Microchip's PICs or Motorola's 68HC11 is the best choice to drive the robot. For this project, the Microchip PIC18f4550 is selected as the microcontroller. This 40-Pin High-Performance has 256 bytes of data EEPROM memory on the chip. All code that was kept in this EEPROM will stay put even though the power supply is cut off. With it, the robot can be switched off until it is ready to go. Erasing EEPROM can be done instantly by using PICkit. As there are 5 ports available on the chip, the number of I/O pins is no longer a constraint. There is abundantly I/O ports for controlling the 5 movement on the mouse in the maze. 3 sensor circuit will take up to 3 output pins from Port E while 8 output pins from Port D is used for controlling dc motors and motor driver. The using of PIC microcontroller than the MC68HC11 microcontroller have some advantages and disadvantages which is:

Advantages of Pics:

- Many on-board periferals such as: system clock, ADC, EEPROM, timers, PWM and comparators.
- VERY cheap
- Easily available
- Easy to use
- Very small models available (Less than 1mm height, although the DIPs that used to practice are normal DIP size)

Disadvantages:

- No inbuilt FPU (but they can do IEEE floating point, all the C compilers know the algorithms)
- Not much RAM (can use external though)
- Slow (40MHz for the 18 range, 20MHz for the 16, so more than enough speed for basic control applications though)

2.3 Motor Controller

Based on the thesis that taken, *a motor controller is selected based on its simplicity and accuracy in controlling the acceleration, speed and position. The motor must be light, small and uses the energy as minimum as possible* [1]. There are 2 methods most popular ways of propelling a maze mapper robot by using DC motors or stepper motors. Each package has its own advantages and disadvantages. From the thesis about micromouse project, the micromouse use stepper motors as the its movement. This kind of motor are heavy and use lot of power, not always good for speed and will need relatively heavy duty driving electronics. *Compared with DC motors , the motors can be small and light, the electronics are simple and moderate in power requirements. Most small DC motors spin at high speeds and low torque, but this situation can be corrected by using some kind of gear system* [1]. Some DC motors use a motor driver to control the speed. For this project, the robot used the PD3046 DC motor with encoder and MD30A motor driver which act as the movement of the robot. Encoder of the DC motor is act as a sensor of the motor. It function to detect the speed of the rotation motor based on the data signal that received from PIC microcontroller so that the motor will rotate exactly same with the data from PIC. The motor driver act as the guidance for the motor to run how much speed that needed based on the data programming from PIC. The DC motor run with the various speed from high speed 5:1 gear ratio to lower speed 939:1 gear ratio. The 12V DC motor can run at full speed 5950rpm with below 900 mA rated current. A common way of controlling DC motors is through an H-Bridge. These can be made out of discrete components or can be bought in the form of an IC.

2.4 Infrared Sensor

The two most common ways of detecting the walls within the maze are infrared sensors and wall feelers [1]. The type of the sensor that commonly used is infrared sensor. This sensor is being used to detect obstacles or maze map in three directions that is the front, right and left side of the robot. *These sensors return true/false readings indicating the presence or absence of walls* [1]. Below is the flowchart on how the maze mapper robot works.

31

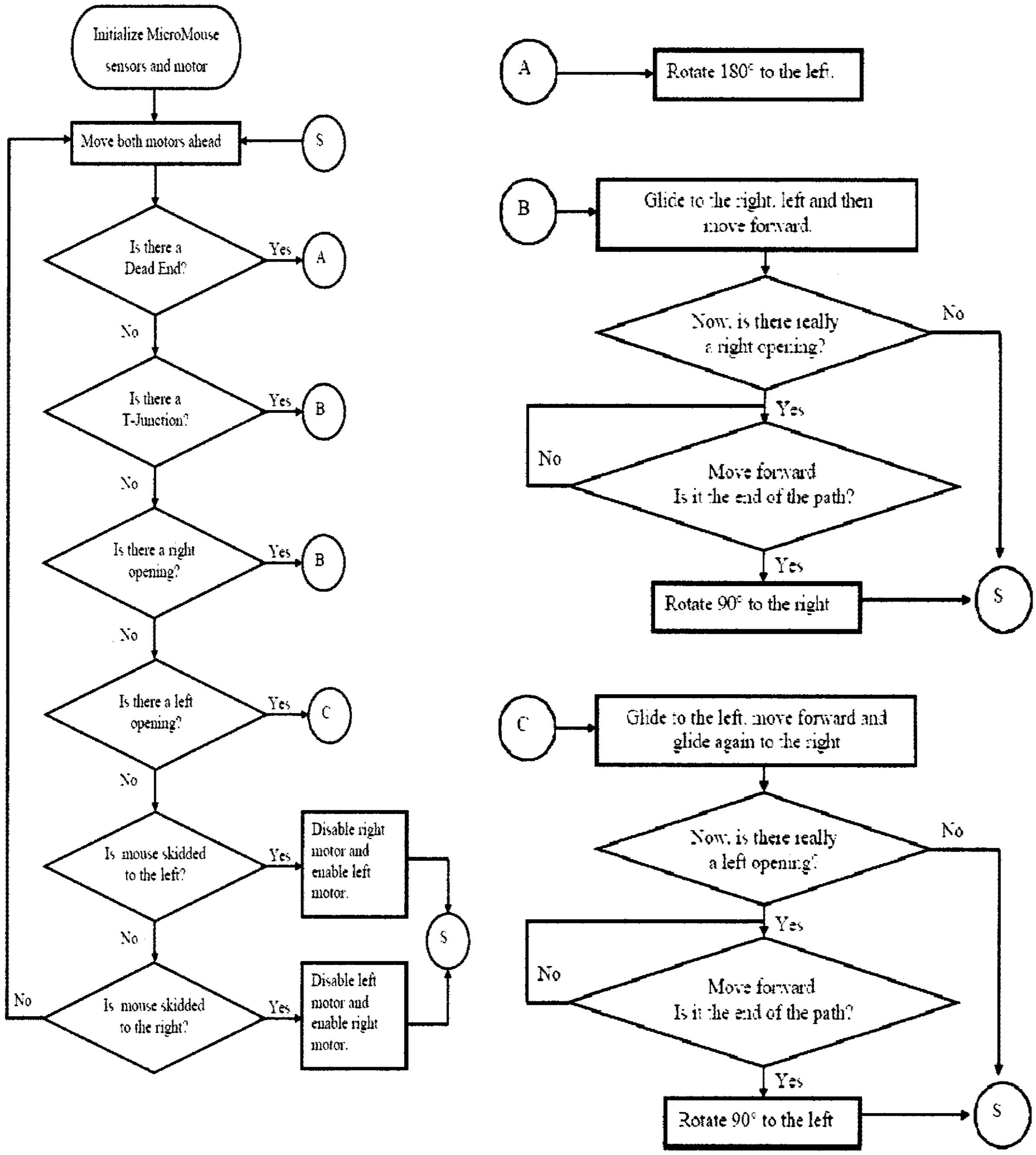


Figure 2.4: Alignment Corrective Flowchart

Other article that had been taken from website states that *this rat robot which has been optimized on simulated autonomous agents and implemented on a mobile robot which learns to navigate within its environment through exploration using vision such as camera as its main sensor* [2]. This project use the visual detector to detect the object that it want to find with the maze wall surround it.

This robot is able to locate itself accurately and simultaneously, to build a map of the environment. Algorithms in the article are based on the usage of laser range finders or sonar sensor. *This robot use vision-based mobile robot localization and mapping algorithm, which uses scale-invariant image features as natural landmarks in unmodified environments* [3]. All the statement about the sensor [2] [3] are using vision system detector as the sensor in the navigation robot. For this project of maze mapper robot, infrared range detector are is used as the sensor of the robot. It will detect the presence of the maze wall and transmit the signal to the microcontroller so that microcontroller will alert the signal and send the information to the dc motor to change the direction and avoid the robot to crash the maze wall. The type of the infrared sensor that used is GP2D120XJ00F SHARP analog output type distance measuring sensor. This sensor have measuring distance range about 4 to 30 cm

Infrared (IR) sensors are used for obstacle avoidance in mobile robots. *IR sensors have lower cost and faster response time than ultrasonic (US) sensors* [4]. The characteristic of the IR sensor and US sensor are shown below:

IR sensor	US sensor
non-linear	able to measure the robot-obstacle distance with better precision from 1 or 2cm to 5-6 meters. (longer range)
reflected light quality depends on the surface quality and environment light intensity	Small objects can be detected over longer distances
have influences on the precision of obstacle avoidance	Resistance to external disturbances such as vibration, infrared radiation, ambient noise, and EMI radiation
good as proximity detectors in robotics	can be used outside in bright sunlight
the angular resolution is better	less affected by target materials and surfaces, and not affected by color
response time is faster	use sound instead of light for ranging

Table 2.4: Characteristic between IR sensor and US sensor

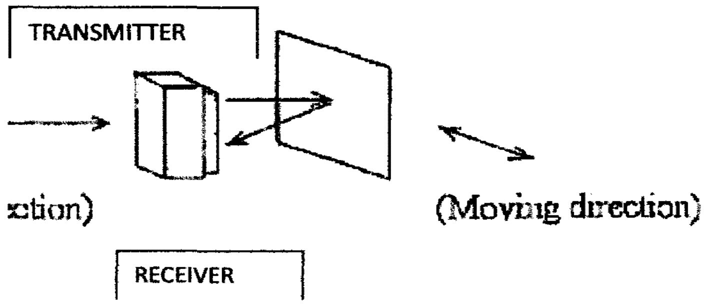


Figure 2.4.1: The direction of the sensor detection

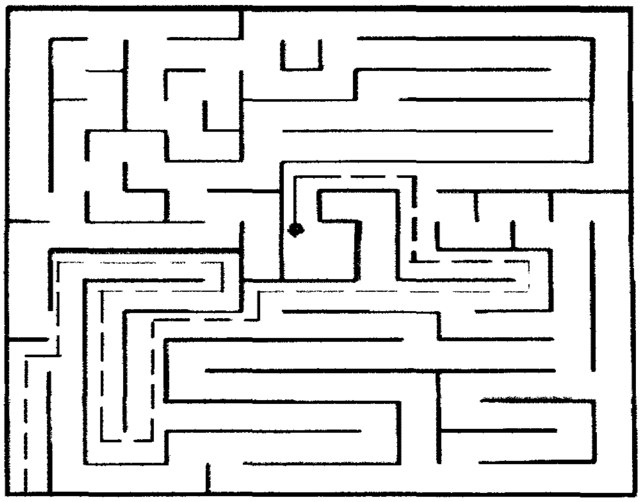


Figure 2.4.2: Maze map location

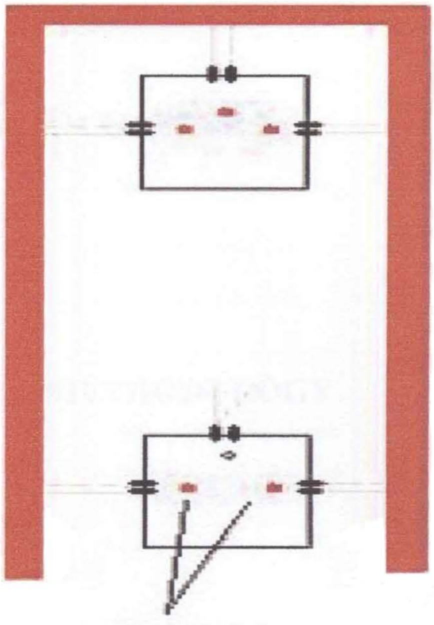


Figure 2.4.3: Maze mapper robot with infrared range detector

2.5 Mechanical and Maze Construction

In order for a robot to operate reliably, its parts have to be mounted to some type of chassis. This does not have to be an elaborate design. My chassis is screwed to 2 circular plastic chassis where there will be 1 piece of PIC circuit strip board mounted on first layer and Motor Driver circuit on the second layer. This design uses two wheels, one on either side of the robot plus a ball caster wheel. By slowing down or reversing one of the motors, the robot is able to steer. Balance is achieved by placing casters front of the chassis. The 5 x 5 maze is designed which have 2 outer opening; the entry and the exit, and a lot obstacle.

CHAPTER 3

METHODOLOGY

3.1 Introduction

In this project, there are 3 methodology to complete the Maze Mapper Robot which is hardware design, and firmware design.

3.1.1 Hardware Design

In hardware design, it will be separate into 2 step that are:

i. Mechanical Design

ii. Electronic Circuit Design

3.1.2 Mechanical Design

For the mechanical design, the robot must be design depends on the creativity, and stability. The robot must be have the DC motor placement, infrared sensor placement and the electronic circuit placement. For this mechanical design, an AutoCAD 2007 software is used to create the shape of the robot. This figure is shown of design the mechanical maze mapper robot.

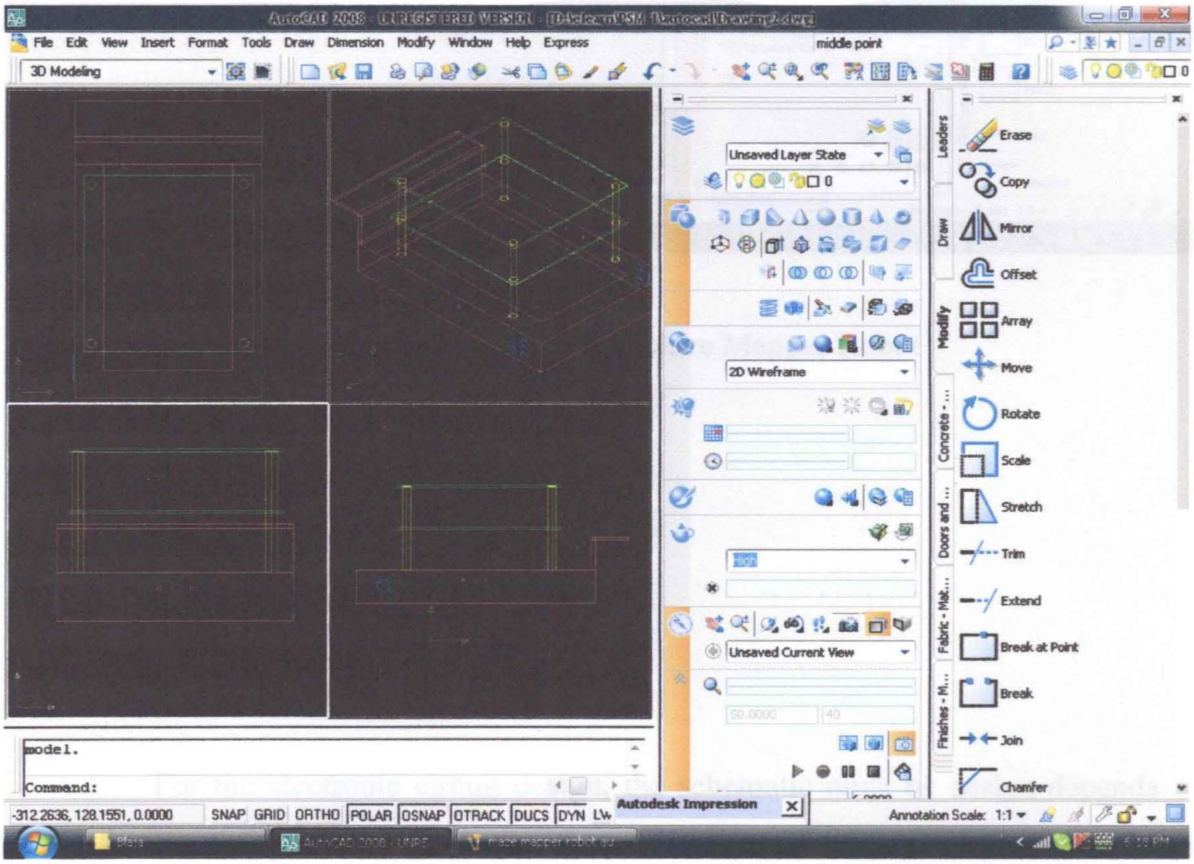


Figure 3.1.2 (a): 2D Wireframe Maze Mapper Robot

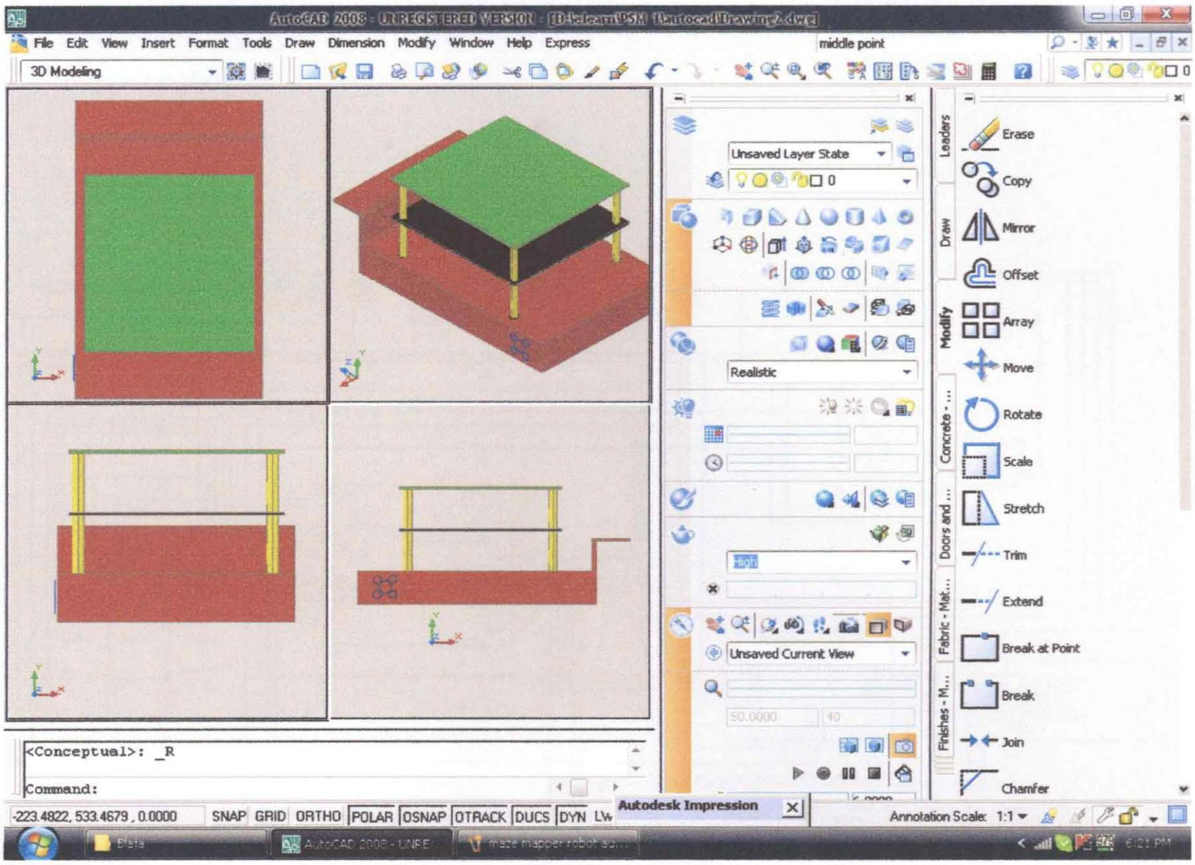


Figure 3.1.2 (b): Realistic of Maze Mapper Robot

3.1.3 Electronic Circuit Design

For the electronic circuit design, the schematic must be design depends on the creativity, and well function based on he application that needed . The schematics must be have the PIC microcontroller, power supply, the voltage regulator, infrared sensor, motor driver and the dc motor. For this schematic diagram design, an OrCAD 2008 software is used to design the schematic diagram of the robot. This figure is shown of design the electronic circuit maze mapper robot.

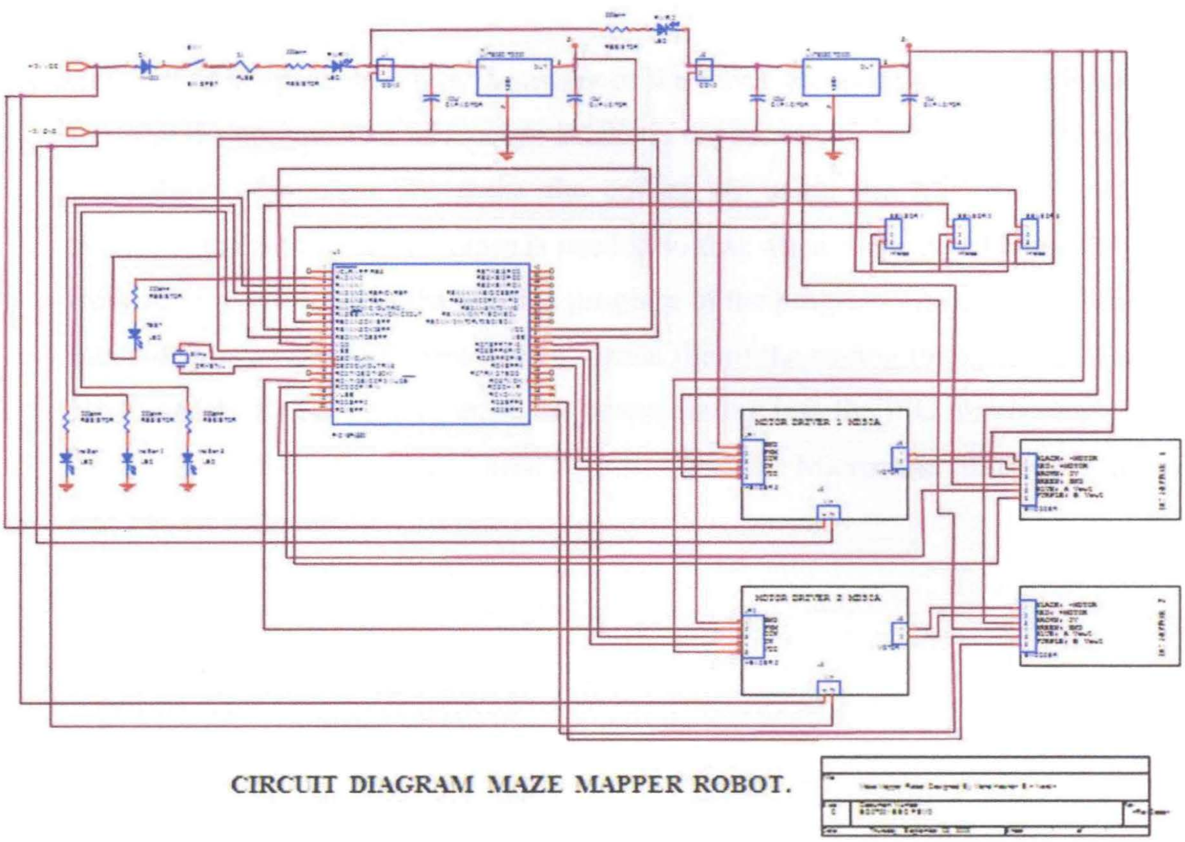


Figure 3.1.3: Schematic Diagram For Maze Mapper Robot

3.1.4 Firmware Design

For the programming / firmware of the robot, the PBasic language is used to make the coding. The Microcode Studio is the assembler software of PIC code for PBasic language. To make the coding by using the Microcode studio software, the setting of PIC types is needed so that when the code had assembled there are no error during the burning progress of the program into the PIC. When the coding is assembled, there have 4 format file of the coding that is LST, ASM, HEX, and PBP. To write or erase the programming into the PIC microcontroller 18F4550, the PICkit is used. These figures shows the Microcode studio software and PICkit software.

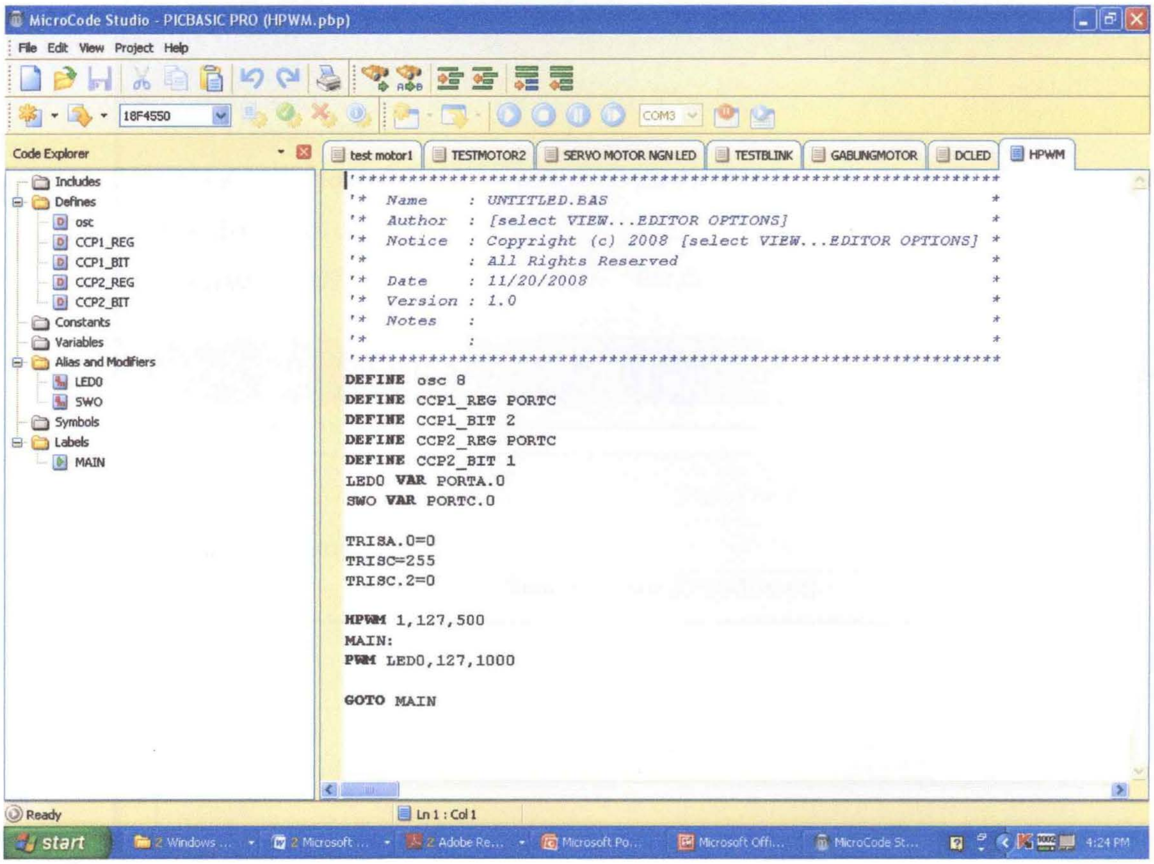


Figure 3.1.4 (a): Microcode Studio Software

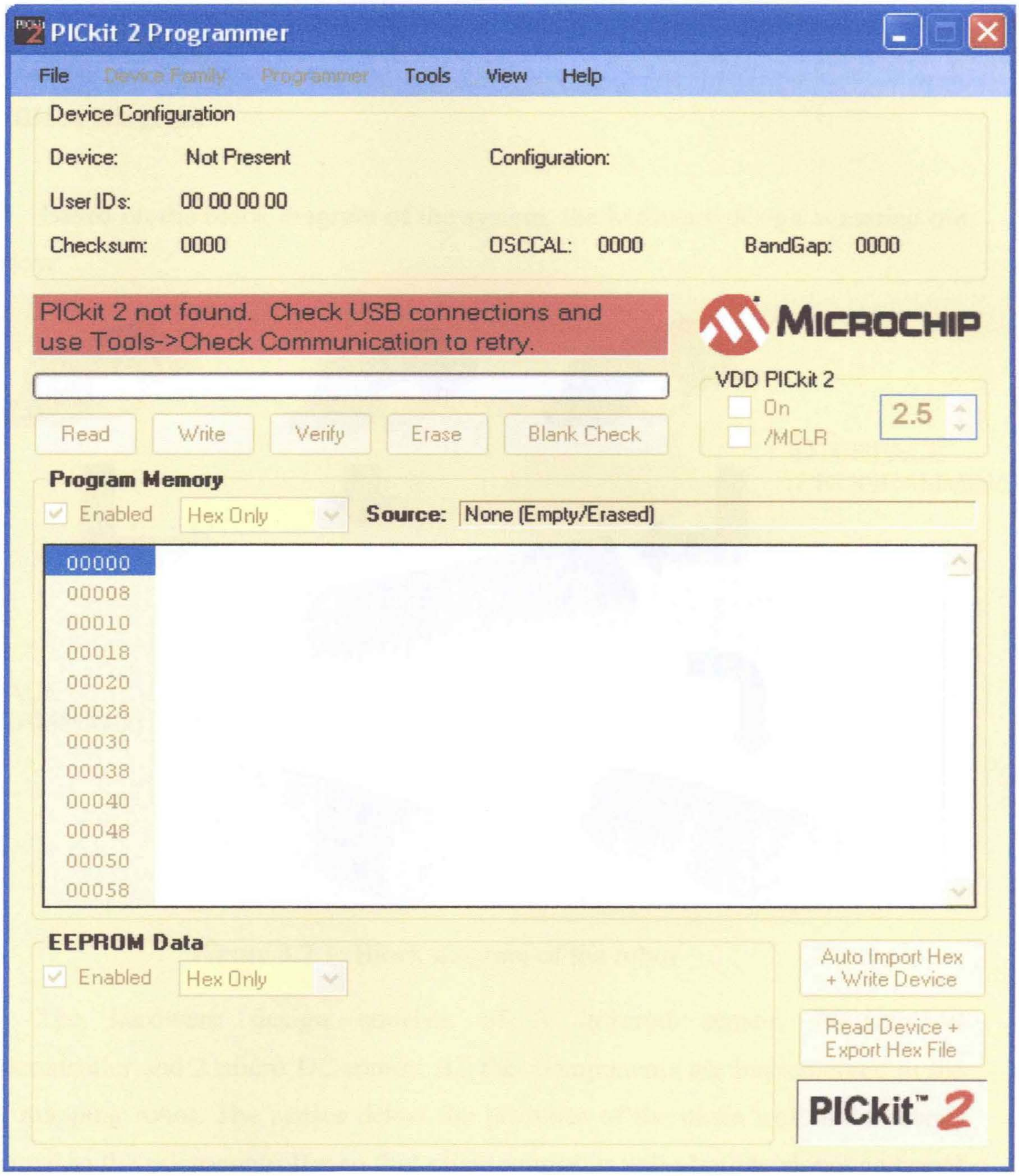


Figure 3.1.4 (b): PICkit Programmer Software

3.2 Block Diagram

Based on the block diagram of the system, the hardware design is carried out as below

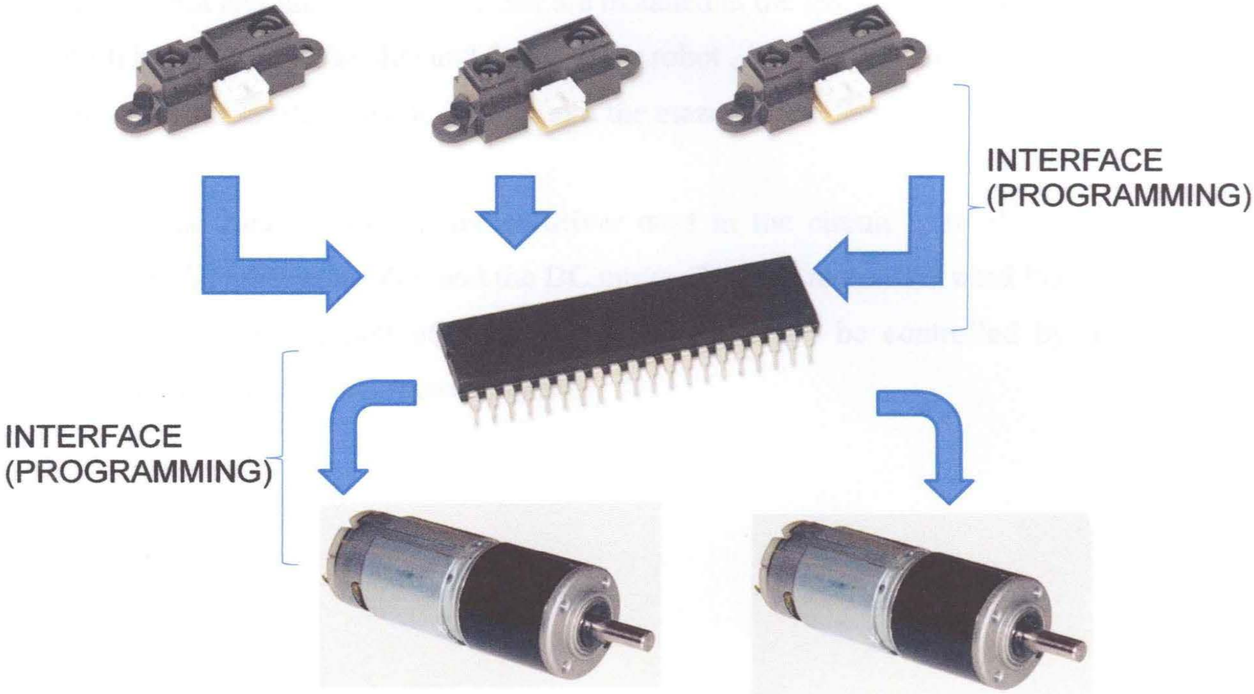


Figure 3.2.1: Block diagram of the robot

The hardware design consists of 3 infrared sensor, PIC18F4550 microcontroller and 2 micro DC motor. All the components are implemented in the maze mapping robot. The sensor detect the presence of the maze wall and transmit the signal to the microcontroller so that microcontroller will alert the signal and send the information to the dc motor to change the direction and avoid the robot to crash the maze wall. The PIC microcontroller also can memorize the pathway which the sensor send the data of the map path. This microcontroller have advantage than other microcontroller such as HC11. PIC18F4550 not just only made the instruction for the robot to make the movement but it can memorize which path that the robot had went so that when the robot get into the maze map for the second time, it can move in the map perfect without figured which path it must go. The firmware that use to program the robot is the PBasic language.

The infrared sensor that used is GP2D120XJ00F manufactured by SHARP. It can detect the maze wall with range 4cm until 30cm. For this project, 3 infrared sensor are used that implemented in the robot. One of the sensor installed at the front of the robot and the other two sensor are installed at the left and right of the robot. With installation at the side and front of the robot , it will avoid the maze wall and minimize the crashing while it try to exit the maze map.

The function of DC motor driver used in the circuit is to allow interface between the microcontroller and the DC motor. The DC motor that used PD3046 and it use as the movement of the robot. This speed can be controlled by make a programming in the PIC microcontroller.

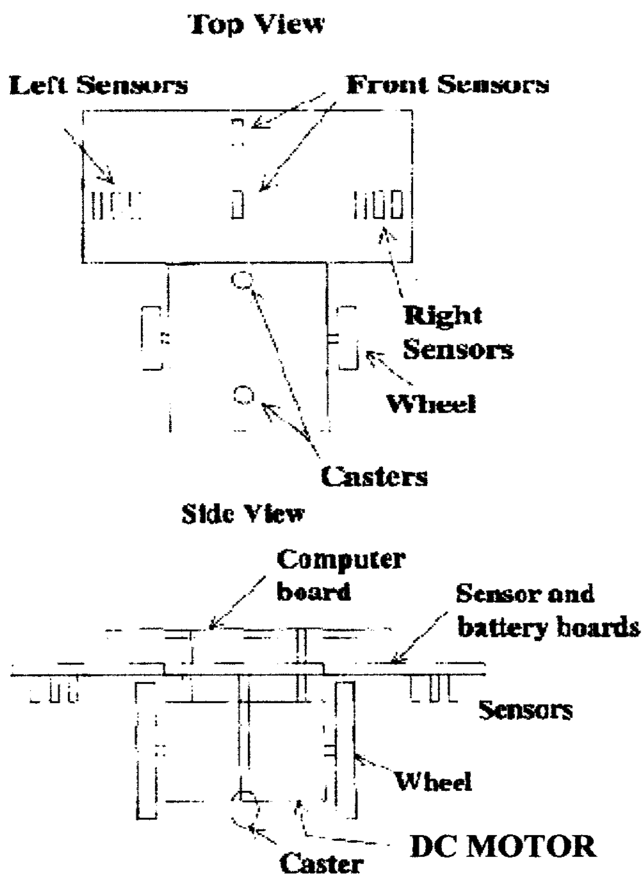


Figure 3.2.2: Example of a maze mapper robot design

3.3 Flow Chart

The flow of the project progress is shown below during the project.

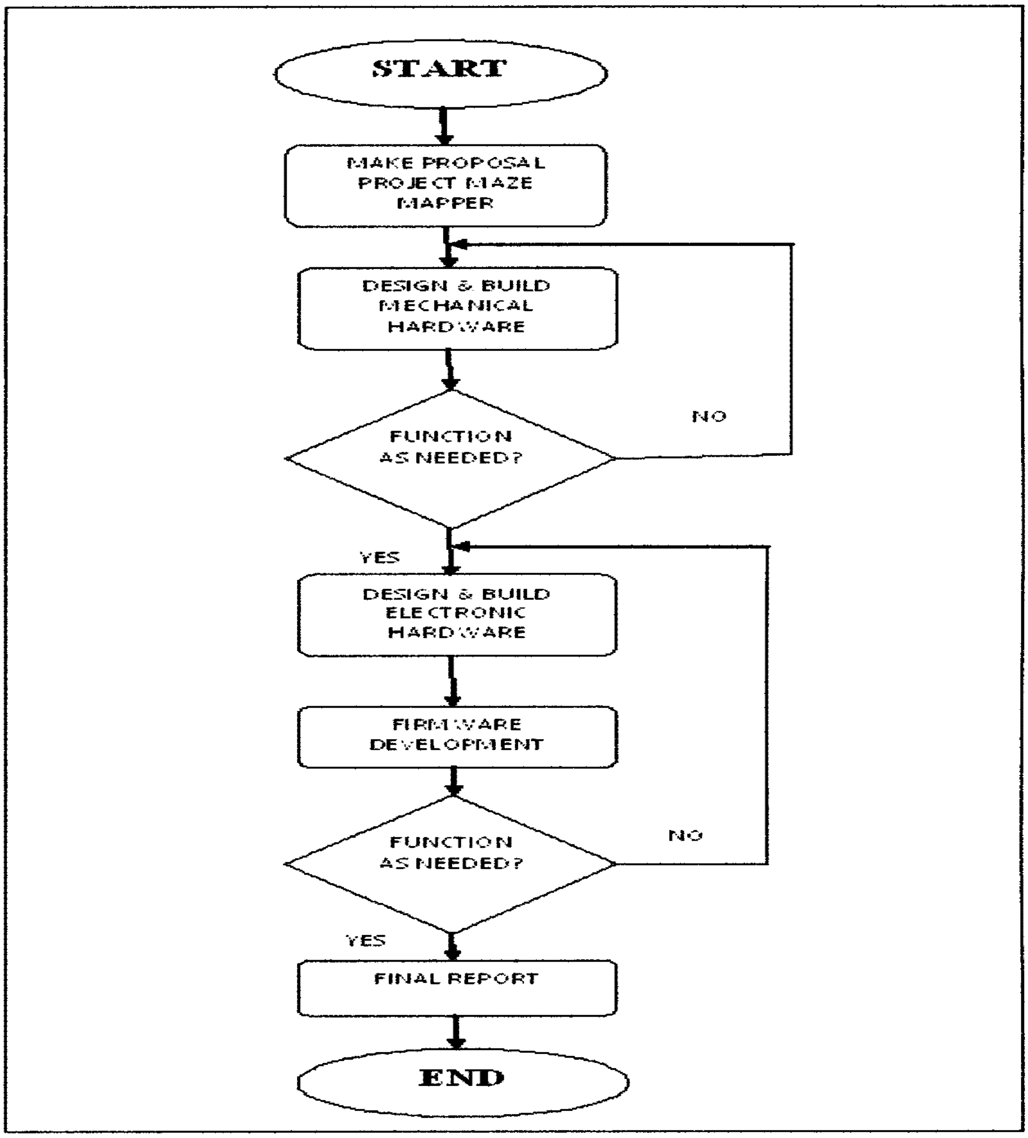
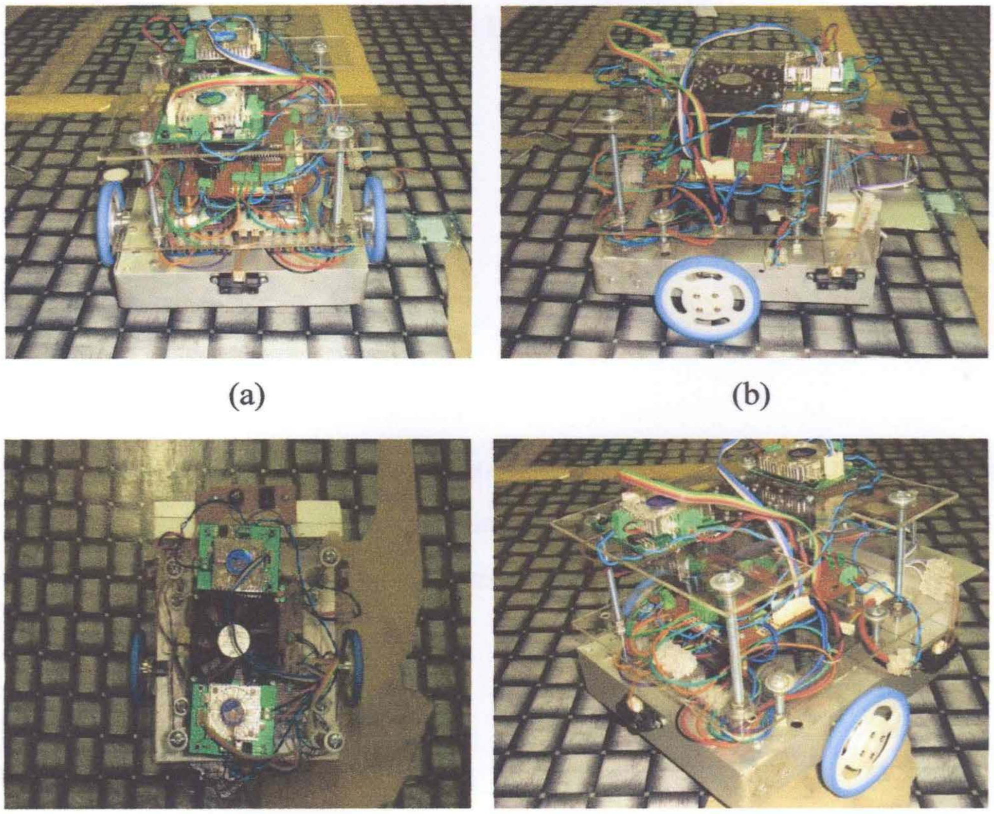


Figure 3.3: Flow chart of the project

3.4 Maze Mapper Robot Design

The maze mapper robot are designed based on the graphic in the OrCAD.



(a) (b) (c) (d)

Figure 3.4: The maze mapper robot (a) front; (b) side; (c) above; (d) Isometric

CHAPTER 4

RESULT & DISCUSSION

4.1. Introduction

In this project, the result from the execution of maze mapper robot is to find it's own way out when the robot in the maze area zone. To get the result, the maze algorithm of the robot movement is considered. This chapter will explain the result and the discussion of the project.

4.2 Maze Map Design

In unknown maze map design, it designed with the specification:

i. Size of map 5 block x 5 block
ii. Each block have size 40cm x 45c0

The unknown maze is designed with variable designed so that the maze mapper robot can execute in any environment in the maze map. One of the unknowns maze map is designed below:

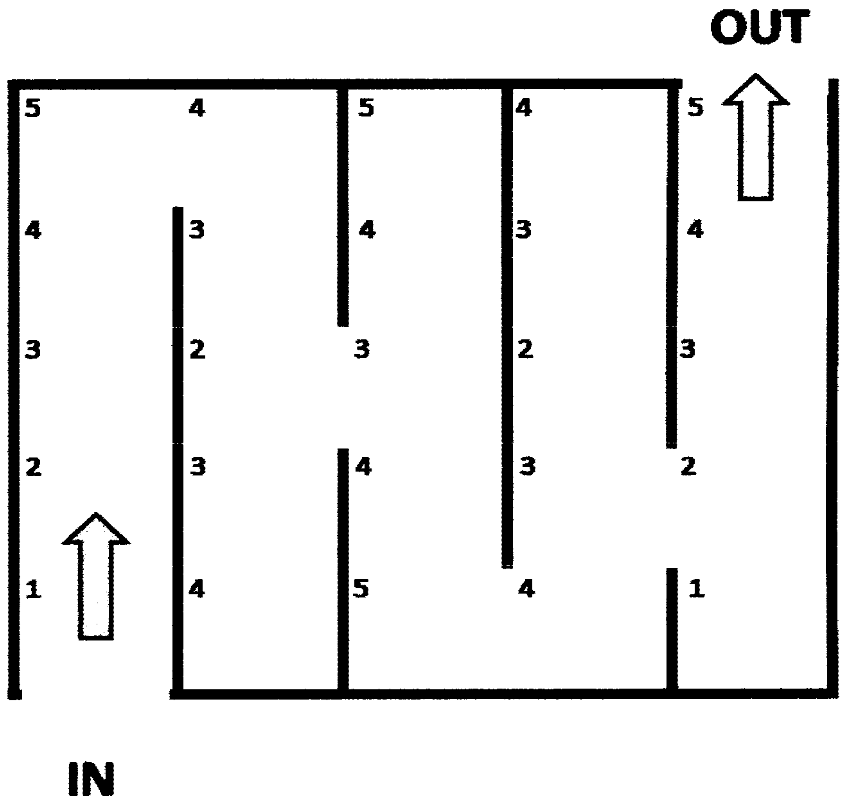


Figure 4.1: Unknown maze map

4.3 PIC18F4550 Microcontroller Programming Analysis

For the PIC18F4550 programming, the PicBasic Pro Compiler (or PBP) it used. It makes the programming even quicker and easier to program Microchip Technology's powerful PICmicro microcontrollers (MCUs). The English-like BASIC language is much easier to read and write than the quirky Microchip assembly language. PBP has over 80 commands. Some commands are similar to the PicBasic commands with minor changes. Decisions were made that we hope improve the language overall.

4.3.1 PicBasic Pro Commands

SELECT CASE command, "SELECT CASE *Var*". This command is used to compare a variable with different values. Case statements are sometimes easier to use than multiple IF…Then. These statements are used to compare a variable with different values or ranges of values, and take action based on the value. The variable to be used in all of the comparisons is specified in the SELECT CASE statement. Each case is followed by the *Statements* to be executed if the case is true. Is may be used to specify a comparison other than equal to. If none of the cases are true, the *Statements* under the optional CASE ELSE statement are executed. An END SELECT closes the SELECT CASE. In this command there are 8 cases are considered which is:

Case	Sensor detection	Command
0	none	Robot forward & stop
1	Sensor1	Robot turn right
2	Sensor 2	Robot go backward
3	Sensor 1 & Sensor 2	Robot turn right & forward
4	Sensor 3	Robot turn left
5	Sensor 1 & Sensor 3	Robot go forward
6	Sensor 2 & Sensor 3	Robot turn left & forward
7	Sensor 1, Sensor 2 & Sensor 3	Robot go backward, make uturn & forward

Table 4.1: Case of the maze mapper robot

The command of the select case of the robot is shown below:

```
SELECT CASE Var
CASE Expr1 {, Expr...}
Statement...
CASE Expr2 {, Expr...}
Statement...
{CASE ELSE
Statement...}
END SELECT
```

4.3.2 Memorize Maze Map

To memorize the map, the robot execute the countnstop instruction. This instruction will execute the robot to make movement at the starting of each block inside the map and stop at the end of the each block. The length distance of the each block is 205 distance per cm. While it make the countnstop , the robot will scan & calculate each block that it had move. This calculation i=0 in this instruction is from starting point of first block. When the robot move forward & reached at the end of the block, it will calculate by using formula i=i+1. When the robot move backward & reached at the end of the block, the robot make the formula i=i-1. This formula is the algorithm of maze & memorize the unknown map.

4.3.3 Calculation of Countnstop

To calculate the distance of each block, the calculation is shown below:

$$size\ distance\ in\ 1\ pulse = \frac{size\ of\ \text{circumference of a wheel}}{1\ cycle\ of\ wheel}$$

$$size\ distance\ in\ 1\ pulse = \frac{25cm}{114}$$

$$size\ distance\ in\ 1\ pulse = 0.22\text{cm}$$

$$\frac{cm}{distance} = \frac{0.22\text{cm}}{1}$$

$$\frac{distance}{cm} = \frac{1}{0.22cm}\ x\ length\ of\ 1\ block$$

$$\frac{distance}{cm} = \frac{1}{0.22cm}\ x\ 45cm$$

$$\frac{distance}{cm} = 205\ \text{distance per cm}$$

4.3.4 Initializing Sensor

There are 3 sensors that installed on the robot which 2 sensors are implemented at the side by side of the robot and the other one sensor is implemented at the front of the robot.

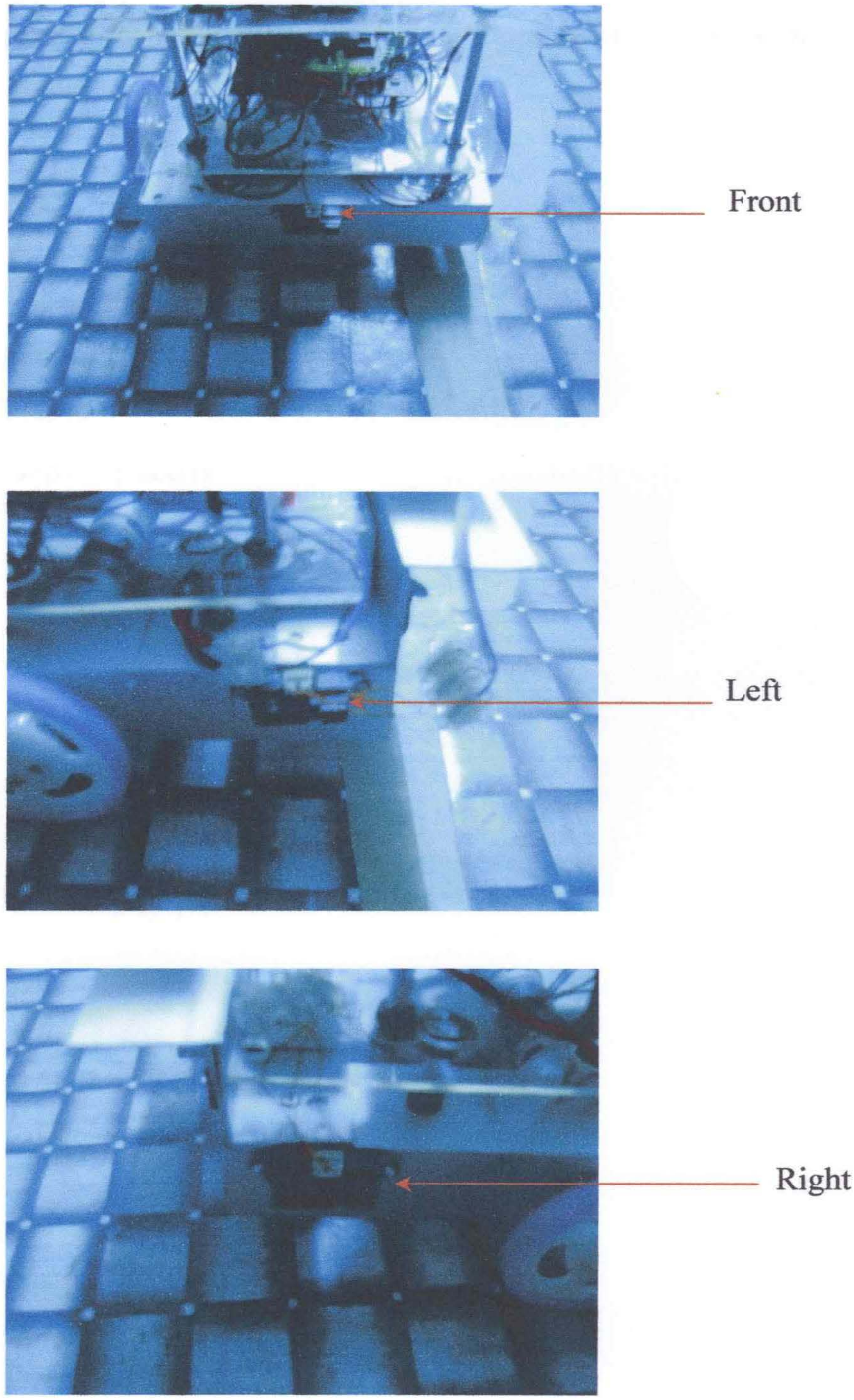


Figure 4.3.4: Sensors position

These sensors are analog device. So to initialize the sensor from analog to digital, the instruction of analog to digital conversion is added. Initializing of the sensors by the program that shown below:

```
ADCON1=%00000000

IR1:
  ADCIN 0, SENSOR1
  IF SENSOR1>26 THEN
  status_sensor.0=1 'sensor 1 active
  ENDIF

RETURN

IR2:
  ADCIN 1,SENSOR2
  IF SENSOR2>60 THEN
  status_sensor.1=1 'sensor 2 active
  ENDIF

RETURN

IR3:
  ADCIN 2,SENSOR3
  IF SENSOR3>26 THEN
  status_sensor.2=1 'sensor 3 active
  ENDIF

RETURN
```

4.4 Circuitry Development Analysis

During the project, the most problematic part is when to make the interface between the sensors, motor driver, dc motor to the PIC circuit. It's because of the current for the circuit is quite high cause of the load of the circuit and load from dc motor. To solve the problem, fuse 5amp is used to protect the motor driver rpm and the component at the circuit board. Then the sensors detection problem because one of the sensor does not response when an object or obstacle gets near to the sensor.

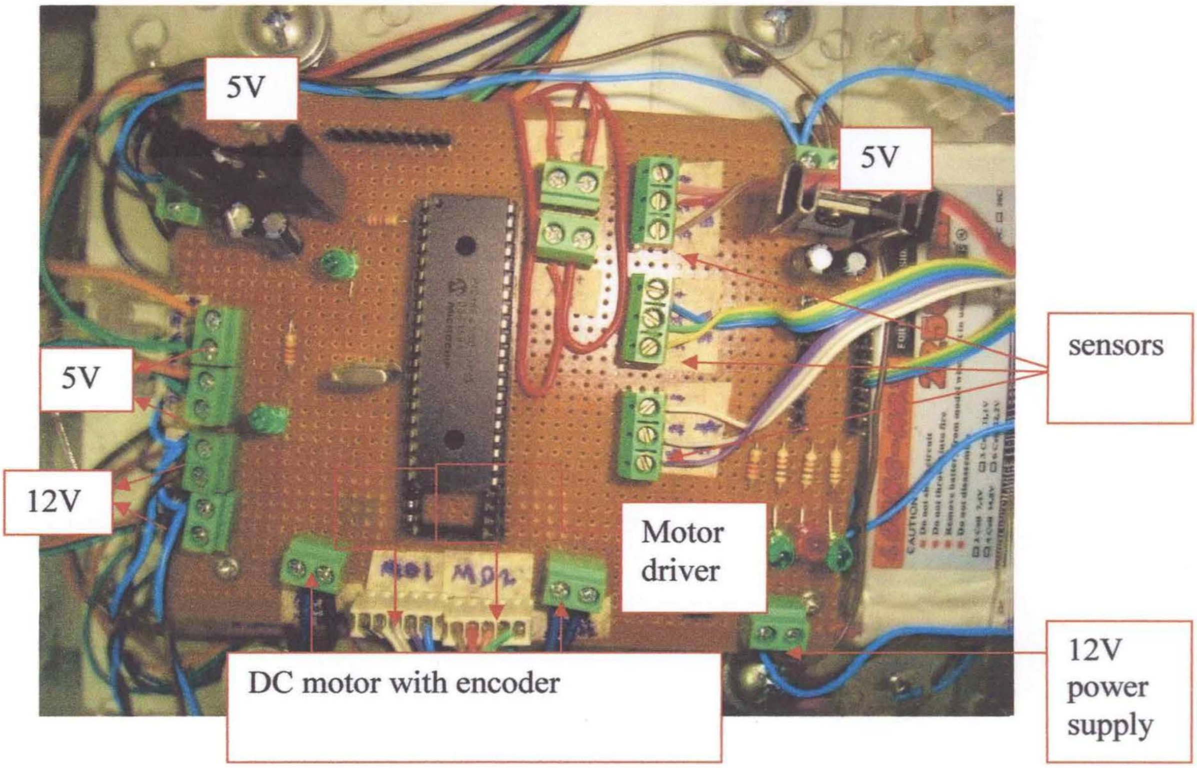


Figure 4.4 (a): Circuit of Maze Mapper Robot

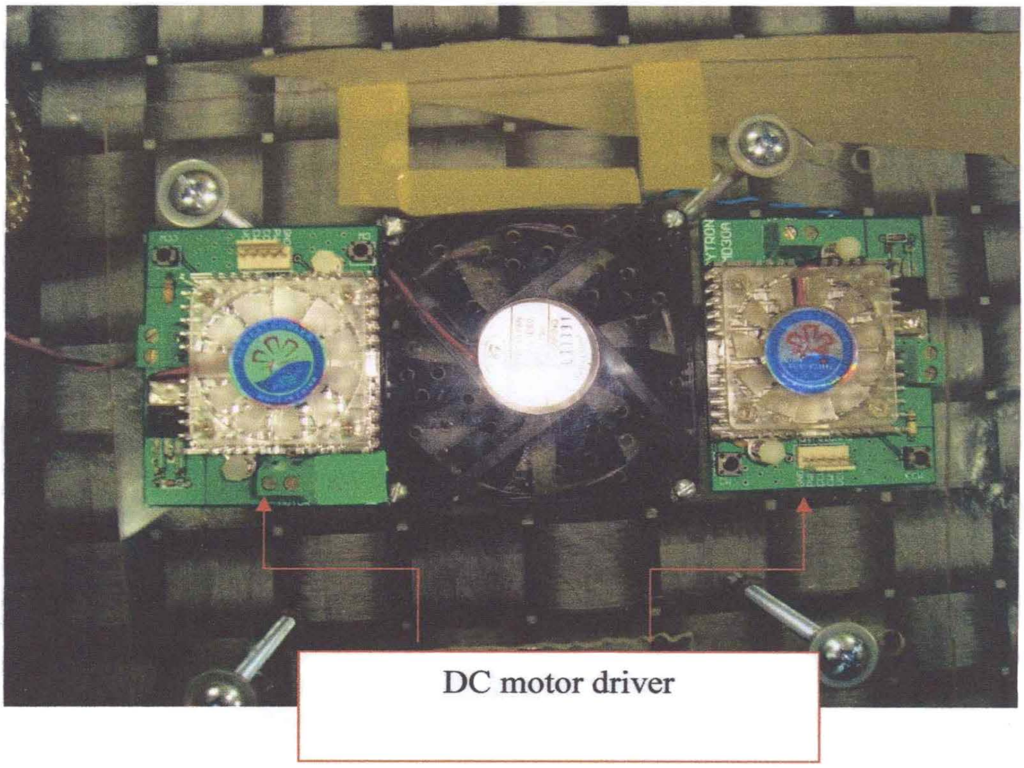


Figure 4.4 (b): Circuit of DC Motor Driver

4.4.1 IR sensor

The sensors range between 0.25V to 2.55V. For this project the minimum detection with the wall is the 1.18V and the max range detection with the wall is 0.51V.

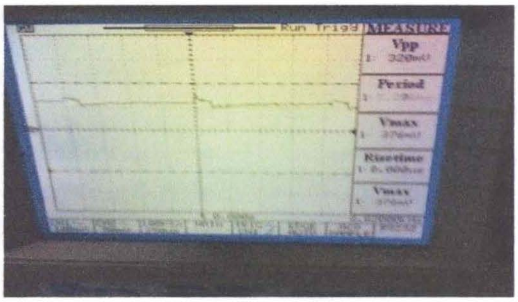


Figure 4.4.1(a): Wave of Sensor Using Oscilloscope

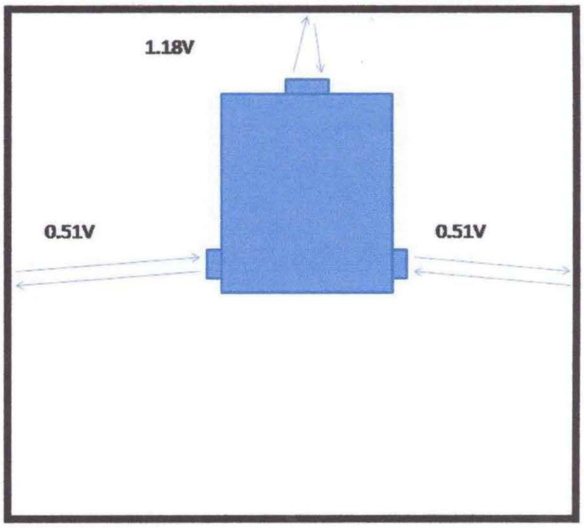
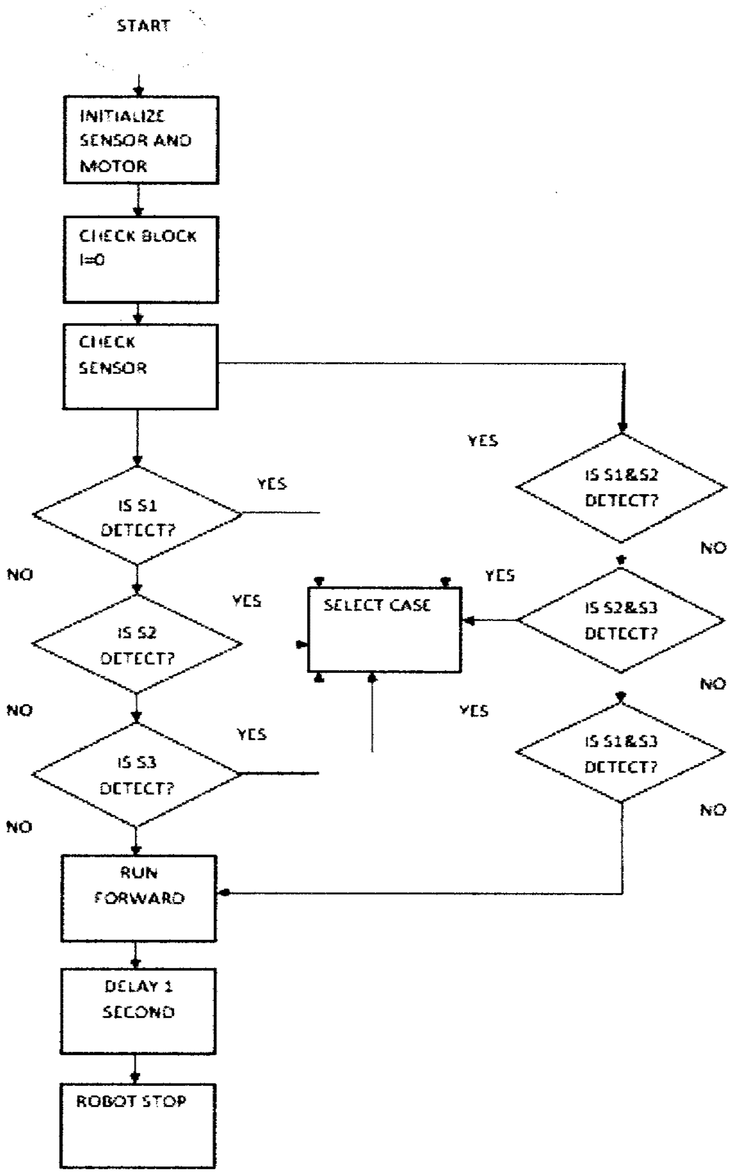


Figure 4.4.1(b): Range Detection of 3 Sensors

4.5 Flowchart of Maze Map Algorithm

The flow of the maze map algorithm is considered to ensure the robot is exactly execute same as the instruction. This flowchart is designed first before make the coding for the algorithm. Figure 4.5 shows the alignment corrective flow chart which is used to ensure that the robot is moving parallel to the maze to minimize wall contact and according to the flow path algorithm in Chapter 2


START
INITIALIZE SENSOR AND MOTOR
CHECK BLOCK I=0
CHECK SENSOR
IS S1 DETECT?
IS S2 DETECT?
IS S3 DETECT?
YES
NO
SELECT CASE
IS S1&S2 DETECT?
IS S2&S3 DETECT?
IS S1&S3 DETECT?
RUN FORWARD
DELAY 1 SECOND
ROBOT STOP

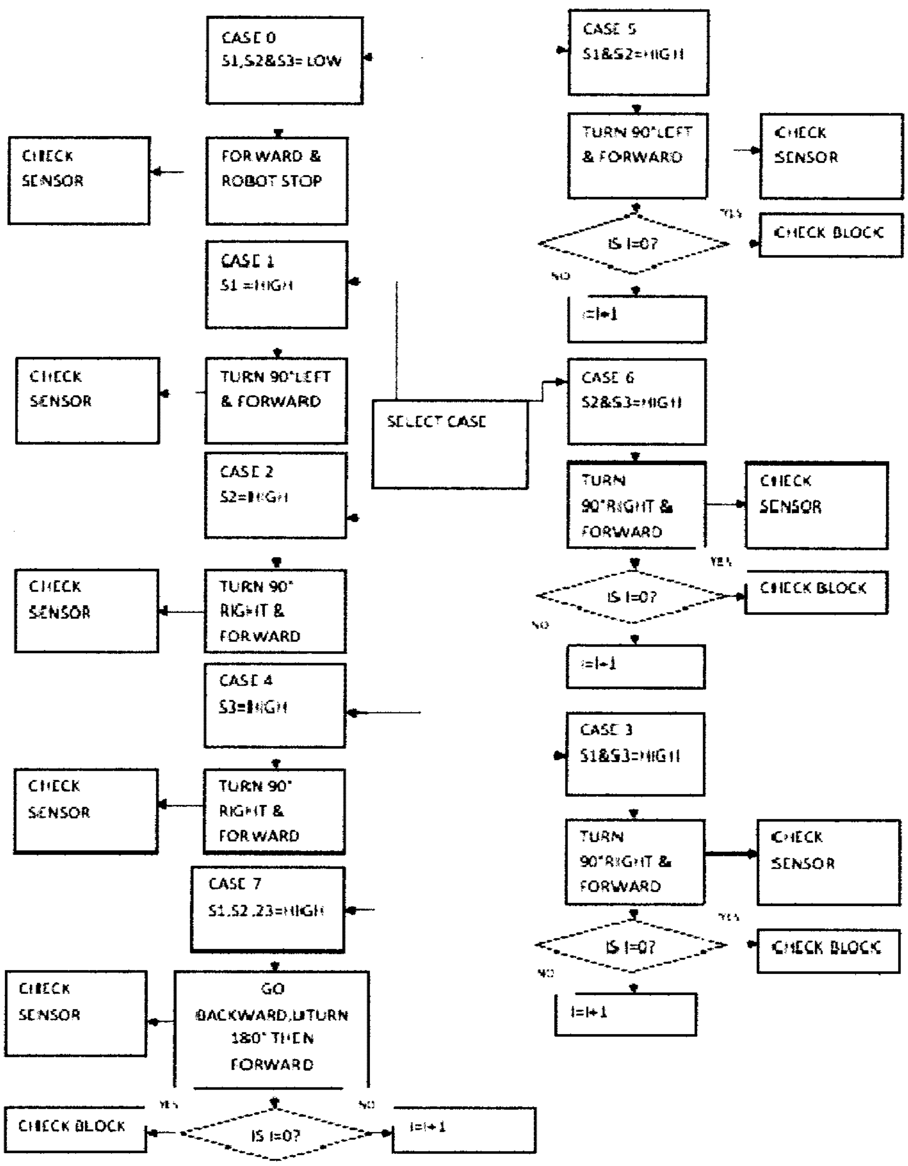
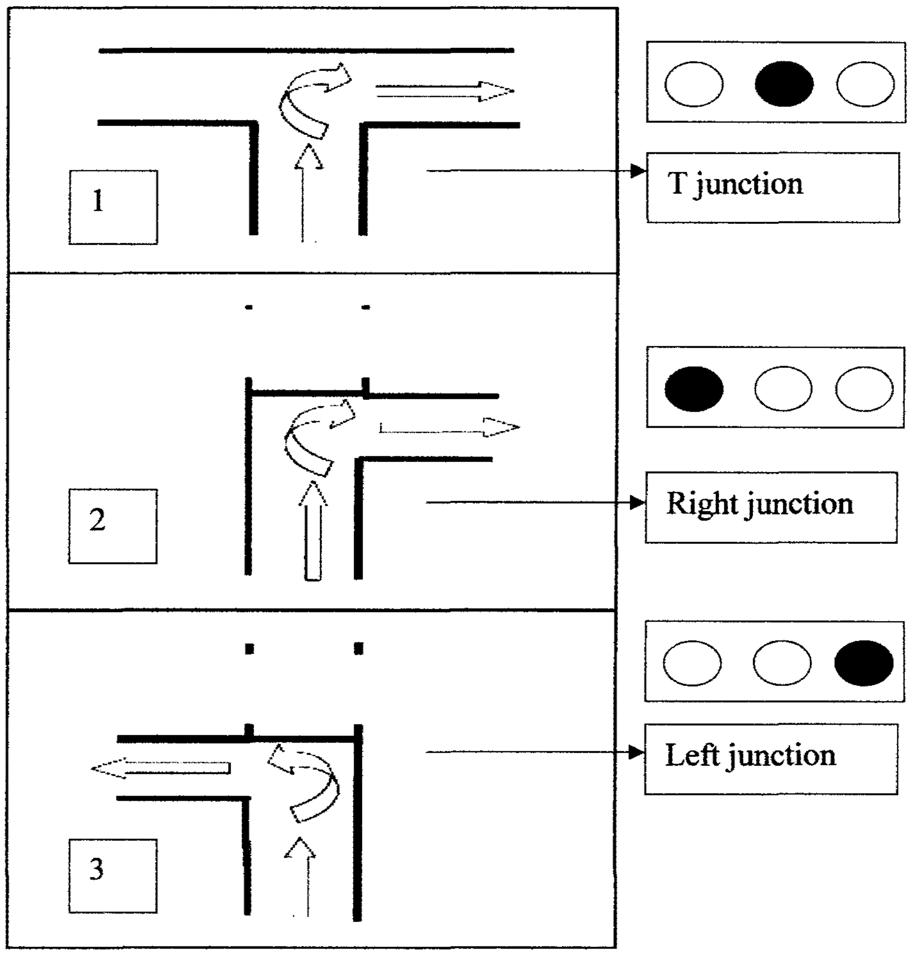


Figure 4.5: Flowchart of The Maze Map Algorithm

4.6 Path of Maze Map Algorithm

The path of the maze map algorithm is designed to make the robot is exactly execute same as the instruction. This path has 6 pattern for the maze mapper robot which what it should do when it reach at the 6 pattern of path. Maze mapper robot will proceed according to the path algorithm. In the maze, there are seven possible states the robot might encounter. For instances in Figure at a T-junction where there are 3 paths to be chosen, the robot will prioritize to turn left rather than going for the other 3 choices. This figures below shows the 6 path of the map.



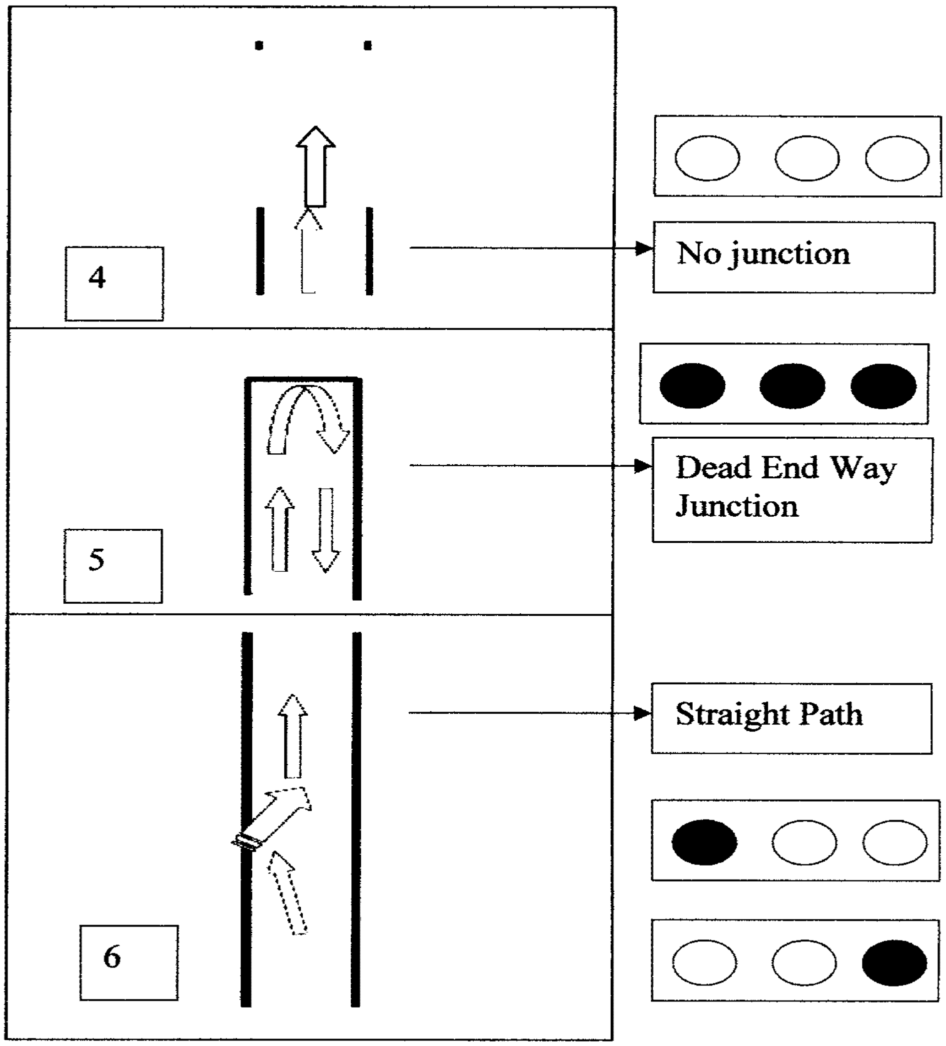


Figure 4.6: Types of Path Maze Map

From the figures above, Maze Mapper robot reacts based on the outputs of the top down wall sensing. The black dot indicates that there is a successful detection of the reflected infrared light from the wall. In short, it means wall is detected. The white dot on the other hand tells the PIC microcontroller that there is an opening in the maze where the reflected light was way beyond the detectors' sensitivity. In Figure 4.6 (6) the sensor at the middle detect the wall and the other 2 sensor at side by side of the robot have no detection so that the robot have reach the T-junction. In this case, it can be said that the robot is turn right and with the parallel.

For figure 4.6 (2) the 2 sensors of the robot which is sensor left and middle are detect the wall. The right sensor is open means no detection. This detection will send the signal to the PIC microcontroller to read the signal then make the instruction for the motor to make another movement. So the robot will turn 90 ° right and moving forward parallel with the maze.

For figure 4.6(3), this situation is same but only sensor at the middle and the right are detect the wall, so the robot will make turn 90 ° left and then make forward movement. When the robot reach the no junction at figure 4.6(4), all of 3 sensors are not detect the maze wall. So the signal send to the microcontroller and microcontroller will read no signal come from the sensor. The robot will stop based on the case which the robot have reach the objectives and finally get out of maze.

For figure 4.6(5), is the case when the robot had reach dead end path. All 3 sensors are detect and send to the microcontroller as the signal. The processing unit will readthe signal and make the instruction backward for the motor to move backward until 2 seconds then the robot will make the uturn. The first motor will make spinning forward and the second motor will make spinning backward.these motor will spin about 180 °. After the robot turn 180 °, it continue move forward.

In Figure 4.6(6) the 2 sensors are available at the straight path because the wall are at the side by side of the robot. In this case, it can be said that the robot is moving parallel and aligned with the maze. This is the ideal case.

In this situation, there is a slight change on the sensor's detection on the left side causes by the movement of the robot which is not parallel to the wall anymore. The robot has skidded to the left and this will prompt the alignment corrective subroutine to adjust the inconsistency. PIC18F4550 will then react by disabling the right motor and enabling the left motor until the ideal state is achieved.

In this situation, there is a minor discrepancy on the sensors detection on the right side causes by the movement of the robot which is not parallel to the wall anymore. The robot has skidded to the right and this will prompt the alignment corrective subroutine to adjust the inconsistency. PIC18F4550 will then respond by making the right motor spin and stopping the left motor until ideal state is achieved.

CHAPTER 5

CONCLUSION

5.1 Main Conclusion

The aim of this project was to design a prototype Maze Mapper rovot. The robot itself is finally finished and functional after a very strong ground developed to reach this objective. Some recommendations on ways to improve and upgrade the performance of this robot are discussed. The Maze Mapper robot project was a very useful experience for building a practical device. During the different stages, a lot of problems occurred, which had to be solved. The construction, testing and improvement of the robot were very tiring but worthwhile. The microcontroller part was very well executed. Here is the list of main conclusion.

i. The microcontroller works well with the sensor and the motor.
ii. The are two power supply, each for PIC18F4550 and motor.
iii. The motor controller works well with the motor driver MD30A.
iv. The IR sensor works the best during the night as it is prone to sunlight.
v. The communication link is set up and downloading of program is successful.

vi. Microcode studio, MPlab and PIC kit v2.55 are used for software development.

vii. Infrared sensor must be chosen carefully as not all detectors will go with the sensor circuit.

viii. Keeping the mass low, smaller circuit diagram and a square base an improve acceleration and stabilization.

5.2 Problems and Solutions

There were a few problems encounter during the course of this work. The problems encountered will be discussed by dividing it into two phases, which is during the progress of PSM 1 and during PSM 2.

5.2.1 Problems Encountered During PSM 1

The first major problem encountered was designing circuit diagram for the PIC, motor and sensor board using OrCAD. As such, sample of time was spent in learning the software. Then the designed of the mechanical hardware design was designed by using AutoCAD. A lot of pressure to make the designed because it make time to make drawing of mechanical designed.

The second problem was built the mechanical hardware. With no knowledge in the mechanical design make the robot looks like unsmart.

5.2.2 Problems Encountered During PSM 2

This table shows the problem encountered during PSM 2 and the solution:

Problem	Solution
One of the sensor blow. Can't detect the range.	change new sensor
Voltage regulator LM7805 exceed more than 5v when measured by using multimeter.	change the voltage regulator
Coding for test sensor and dc motor programming not function because some instruction were wrong	make the correction for the programming
One of the shaft for wheel broken and it make the robot run unbalanced	make the new shaft for the wheel
The room space for making maze map limit	search new room bigger
Coding for maze map algorithm were wrong and make the robot execute wrong instruction	make the correction for the programming

Table 5.2.2: Problems Encountered During PSM2

5.3 Recommendation

As previously stated, there are several improvement that can be made to the system. Some of the improvement is additional while others is crucial in making the system stay reliable in the future.

- Make the PID programming to make the robot move straight.
- Put the IR range sensor with width angle range.
- Installation the camera at the robot for another objective of navigation so that robot can find the image while find its way out.

5.3.1 PID Programming

First recommendation is to make the PID programming to make the robot move straight and smoothly. This concept will make the reducing error detection in the PIC during movement.

5.3.2 IR Sensor With Angle Range

Second recommendation is the robot can avoid the wall contact accurate and precisely by using the IR sensor with angle range detection. These sensor can detect the angle of wall between 125° to 180° although during the turning position.

5.3.1 Camera installation

The objectives of the robot can be improved by find the object in order to find its way out of the maze map. By using the camera which installed at the robot, it can search the object that it captured before and also it can try to find its way out simultaneously. For the future, this robot not only for the toy competition but it can be a rescue robot when there have some accident in the building like searching people which trapped in the building during there have some fire or earthquake.

5.4 Costing and Commercialization

5.4.1 Costing

The costs of this project which contains three different parts that is sensory, processing and output. The table 5.1 shows all components used in this project.

PIC BOARD COMPONENTS			
Component	Quantity	Unit price (RM)	Amount (RM)
PIC 18F4550	2	23.00	46.00
Crystal 8MHz	2	2.25	4.50
Voltage Regulator LM 7805 C	2	1.80	3.60
Capacitor 10µF	2	1.00	2.00
Capacitor 100µF	2	1.20	2.40
Stripboard	2	5.00	5.00
IC Base 40-pin	1	0.70	0.70
Zif Socket	1	20.00	20.00
Fius Soket	1	2.00	2.00
Suiz	1	3.00	3.00
LED	5	0.20	1.00
Resistor 220 ohm	5	0.30	1.50
		Total :	91.70

BODY COMPONENTS			
Component	Quantity	Unit price (RM)	Amount (RM)
Driver Motor MD30A	2	135.00	270.00
Fulhaber Dc Miromotor with encoder	2	300	600
IR sensor	3	65.00	195.00
Dc Power Supply	1	90.00	90.00
Wheel	2	40.00	80.00
		Total:	1235.00
		Total cost:	1326.70

Table 5.4.1: Cost of components

5.4.1 Commercialization

This system has high commercial value by revolutionize the way mobile robot move and can be applied in new technology for optimum performance.

REFERENCES

REFERENCE

[1] M. Tan Loong Peng, (March 2003), *Micromouse Robot Thesis* URLhttp://mail.fke.utm.my/~michael/download/Michael%20Tan%20Loong%20Peng%20-%20Micromouse.pdf

[2] Verena V. Hafner; Adaptive Behavior, (Jun 2005), vol. 13: pp. 87 – 96 *Cognitive Maps in Rats and Robots* URL http://portal.acm.org/citation.cfm?id=1064672

[3] Stephen Se, David Lowe, and Jim Little, The International Journal of Robotics Research, (Aug 2002); vol. 21: pp. 735 – 758, *Mobile Robot Localization and Mapping with Uncertainty using Scale-Invariant Visual Landmarks* URLhttp://www.cs.ubc.ca/~lowe/papers/ijrr02.pdf

[4] I. Matijevics, (August, 2007), vol 5: pp. 179-181, *Infrared Sensors Microcontroller Interface System for Mobile Robots* URLhttp://www.waset.org/pwaset/v32/v32-113.pd

APPENDIX A

DATASHEET

MICROCHIP

PIC18F2455/2550/4455/4550 Data Sheet

28/40/44-Pin High-Performance, Enhanced Flash USB Microcontrollers with nanoWatt Technology

MICROCHIP **PIC18F2455/2550/4455/4550**

28/40/44-Pin High-Performance, Enhanced Flash USB Microcontrollers with nanoWatt Technology

Universal Serial Bus Features:

- USB V2.0 Compliant
- Low Speed (1.5 Mb/s) and Full Speed (12 Mb/s)
- Supports Control, Interrupt, Isochronous and Bulk Transfers
- Supports up to 32 endpoints (16 bidirectional)
- 1-Kbyte dual access RAM for USB
- On-chip USB transceiver with on-chip voltage regulator
- Interface for off-chip USB transceiver
- Streaming Parallel Port (SPP) for USB streaming transfers (40/44-pin devices only)

Power-Managed Modes:

- Run: CPU on, peripherals on
- Idle: CPU off, peripherals on
- Sleep: CPU off, peripherals off
- Idle mode currents down to 5.8 µA typical
- Sleep mode currents down to 0.1 µA typical
- Timer1 oscillator: 1.1 µA typical, 32 kHz, 2V
- Watchdog Timer: 2.1 µA typical
- Two-Speed Oscillator Start-up

Flexible Oscillator Structure:

- Four Crystal modes including High Precision PLL for USB
- Two External Clock modes, up to 48 MHz
- Internal oscillator block:
  - 8 user-selectable frequencies, from 31 kHz to 8 MHz
  - User-tunable to compensate for frequency drift
- Secondary oscillator using Timer1 @ 32 kHz
- Dual oscillator options allow microcontroller and USB module to run at different clock speeds
- Fail-Safe Clock Monitor
  - Allows for safe shutdown if any clock stops

Peripheral Highlights:

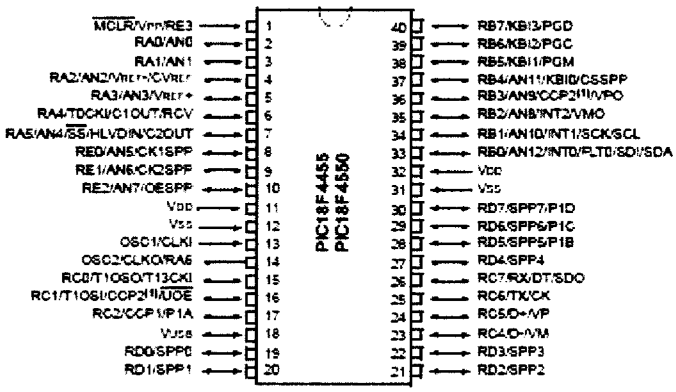
- High-current sink/source 25 mA/25 mA
- Three external interrupts
- Four Timer modules (Timer0 to Timer3)
- Up to 2 Capture/Compare/PWM (CCP) modules:
  - Capture is 16-bit, max. resolution 6.25 ns (TCY/16)
  - Compare is 16-bit, max. resolution 100 ns (TCY)
  - PWM output: PWM resolution is 1 to 10-bit
- Enhanced Capture/Compare/PWM (ECCP) module:
  - Multiple output modes
  - Selectable polarity
  - Programmable dead time
  - Auto-Shutdown and Auto-Restart
- Enhanced USART module:
  - LIN bus support
- Master Synchronous Serial Port (MSSP) module supporting 3-wire SPI™ (all 4 modes) and I²C™ Master and Slave modes
- 10-bit, up to 13-channels Analog-to-Digital Converter module (A/D) with programmable acquisition time
- Dual analog comparators with input multiplexing

Special Microcontroller Features:

- C compiler optimized architecture with optional extended instruction set
- 100,000 erase/write cycle Enhanced Flash program memory typical
- 1,000,000 erase/write cycle Data EEPROM memory typical
- Flash/Data EEPROM Retention: > 40 years
- Self-programmable under software control
- Priority levels for interrupts
- 8 x 8 Single-Cycle Hardware Multiplier
- Extended Watchdog Timer (WDT):
  - Programmable period from 41 ms to 131s
- Programmable Code Protection
- Single-Supply 5V In-Circuit Serial Programming™ (ICSP™) via two pins
- In-Circuit Debug (ICD) via two pins
- Optional dedicated ICD/ICSP port (44-pin devices only)
- Wide operating voltage range (2.0V to 5.5V)

Device	Program Memory		Data Memory		I/O	10-bit A/D (ch)	CCP/ECCP (PWM)	SPP	MSSP		EAUSART	Comparators	Timers 8/16-bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)					SPI™	Master I²C™			
PIC18F2455	24K	12288	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F2550	32K	16384	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F4455	24K	12288	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3
PIC18F4550	32K	16384	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3

40-Pin PDIP

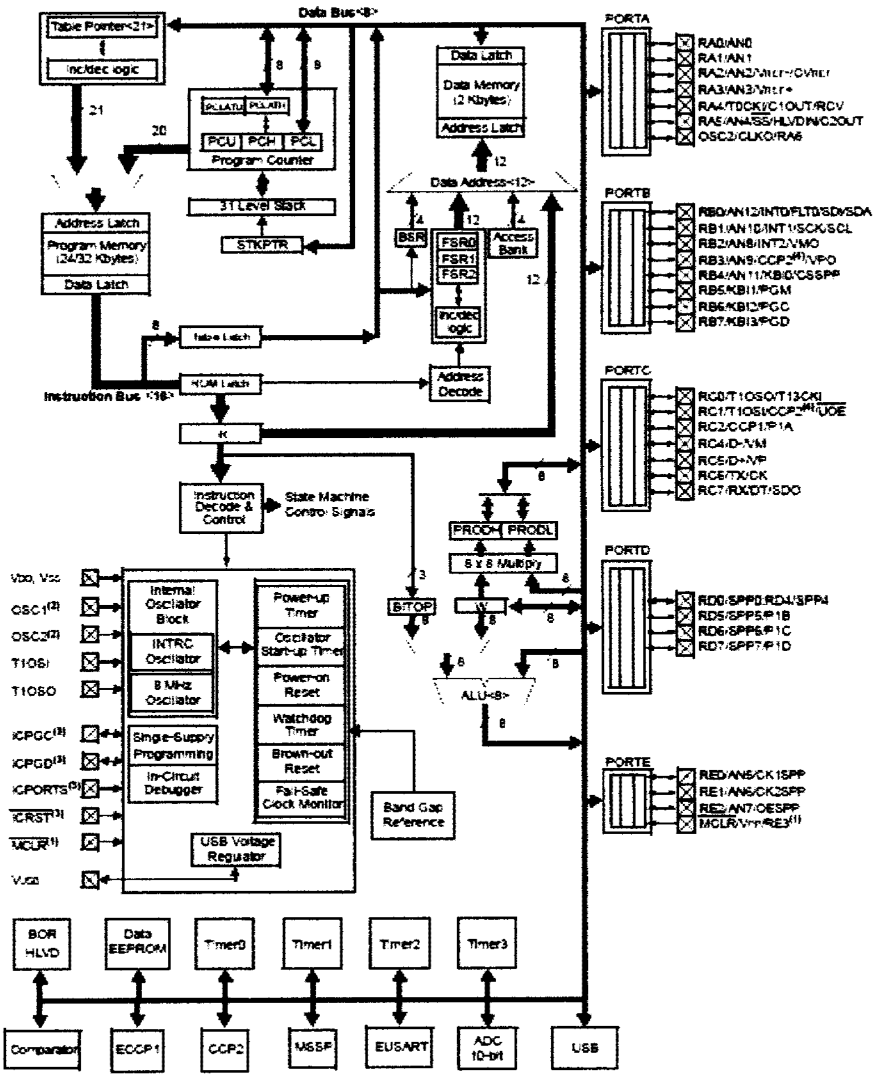


Note 1: RB3 is the alternate pin for CCP2 multiplexing.

TABLE 1-1: DEVICE FEATURES

Features	PIC18F2455	PIC18F2550	PIC18F4455	PIC18F4550
Operating Frequency	DC – 48 MHz	DC – 48 MHz	DC – 48 MHz	DC – 48 MHz
Program Memory (Bytes)	24576	32768	24576	32768
Program Memory (Instructions)	12288	16384	12288	16384
Data Memory (Bytes)	2048	2048	2048	2048
Data EEPROM Memory (Bytes)	256	256	256	256
Interrupt Sources	19	19	20	20
I/O Ports	Ports A, B, C, (E)	Ports A, B, C, (E)	Ports A, B, C, D, E	Ports A, B, C, D, E
Timers	4	4	4	4
Capture/Compare/PWM Modules	2	2	1	1
Enhanced Capture/ Compare/PWM Modules	0	0	1	1
Serial Communications	MSSP, Enhanced USART	MSSP, Enhanced USART	MSSP, Enhanced USART	MSSP, Enhanced USART
Universal Serial Bus (USB) Module	1	1	1	1
Streaming Parallel Port (SPP)	No	No	Yes	Yes
10-bit Analog-to-Digital Module	10 Input Channels	10 Input Channels	13 Input Channels	13 Input Channels
Comparators	2	2	2	2
Resets (and Delays)	POR, BOR, $\overline{\text{RESET}}$ Instruction, Stack Full, Stack Underflow (PWRT, OST), $\overline{\text{MCLR}}$ (optional), WDT	POR, BOR, $\overline{\text{RESET}}$ Instruction, Stack Full, Stack Underflow (PWRT, OST), $\overline{\text{MCLR}}$ (optional), WDT	POR, BOR, $\overline{\text{RESET}}$ Instruction, Stack Full, Stack Underflow (PWRT, OST), $\overline{\text{MCLR}}$ (optional), WDT	POR, BOR, $\overline{\text{RESET}}$ Instruction, Stack Full, Stack Underflow (PWRT, OST), $\overline{\text{MCLR}}$ (optional), WDT
Programmable Low-Voltage Detect	Yes	Yes	Yes	Yes
Programmable Brown-out Reset	Yes	Yes	Yes	Yes
Instruction Set	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled
Packages	28-pin PDIP 28-pin SOIC	28-pin PDIP 28-pin SOIC	40-pin PDIP 44-pin QFN 44-pin TQFP	40-pin PDIP 44-pin QFN 44-pin TQFP

FIGURE 1-2: PIC18F4455/4550 (40/44-PIN) BLOCK DIAGRAM



Note 1: RE3 is multiplexed with $\overline{MCLR}$ and is only available when the $\overline{MCLR}$ Resets are disabled.

2: OSC1/CLKI and OSC2/CLKO are only available in select oscillator modes and when these pins are not being used as digital I/O. Refer to **Section 2.0 "Oscillator Configurations"** for additional information.

3: These pins are only available on 44-pin TQFP under certain conditions. Refer to **Section 26.8 "Special ICPORT Features (Designated Packages Only)"** for additional information.

4: RB3 is the alternate pin for CCP2 multiplexing.

PIC18F2455/2550/4455/4550

TABLE 1-3: PIC18F4455/4550 PINOUT I/O DESCRIPTIONS

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	QFN	TQFP			
MCLR/VPP/RE3	1	18	18			Master Clear (input) or programming voltage (input).
MCLR				I	ST	Master Clear (Reset) input. This pin is an active-low Reset to the device.
VPP				P		Programming voltage input.
RE3				I	ST	Digital input.
OSC1/CLKI	13	32	30			Oscillator crystal or external clock input.
OSC1				I	Analog	Oscillator crystal input or external clock source input.
CLKI				I	Analog	External clock source input. Always associated with pin function OSC1. (See OSC2/CLKO pins.)
OSC2/CLKO/RA6	14	33	31			Oscillator crystal or clock output.
OSC2				O	—	Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode.
CLKO				O	—	In RC mode, OSC2 pin outputs CLKO which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
RA6				I/O	TTL	General purpose I/O pin.

Legend: TTL = TTL compatible input; CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; I = Input; O = Output; P = Power

Note 1: Alternate assignment for CCP2 when CCP2MX configuration bit is cleared.

2: Default assignment for CCP2 when CCP2MX configuration bit is set.

3: These pins are No Connect unless the ICPRT configuration bit is set. For NC/ICPORTS, the pin is No Connect unless ICPRT is set and the DEBUG configuration bit is cleared.

TABLE 1-3: PIC18F4455/4550 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	QFN	TQFP			
						PORTA is a bidirectional I/O port.
RA0/AN0	2	19	19			
RA0				I/O	TTL	Digital I/O.
AN0				I	Analog	Analog input 0.
RA1/AN1	3	20	20			
RA1				I/O	TTL	Digital I/O.
AN1				I	Analog	Analog input 1.
RA2/AN2/VREF-/CVREF	4	21	21			
RA2				I/O	TTL	Digital I/O.
AN2				I	Analog	Analog input 2.
VREF-				I	Analog	A/D reference voltage (low) input.
CVREF				O	Analog	Analog comparator reference output.
RA3/AN3/VREF+	5	22	22			
RA3				I/O	TTL	Digital I/O.
AN3				I	Analog	Analog input 3.
VREF+				I	Analog	A/D reference voltage (high) input.
RA4/T0CKI/C1OUT/RCV	6	23	23			
RA4				I/O	ST	Digital I/O.
T0CKI				I	ST	Timer0 external clock input.
C1OUT				O	—	Comparator 1 output.
RCV				I	TTL	External USB transceiver RCV input.
RA5/AN4/$\overline{SS}$/HLVDIN/C2OUT	7	24	24			
RA5				I/O	TTL	Digital I/O.
AN4				I	Analog	Analog input 4.
$\overline{SS}$				I	TTL	SPI™ slave select input.
HLVDIN				I	Analog	High/Low-Voltage Detect input.
C2OUT				O	—	Comparator 2 output.
RA6	—	—	—	—	—	See the OSC2/CLKO/RA6 pin.

Legend: TTL = TTL compatible input; CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; I = Input; O = Output; P = Power

Note 1: Alternate assignment for CCP2 when CCP2MX configuration bit is cleared.

2: Default assignment for CCP2 when CCP2MX configuration bit is set.

3: These pins are No Connect unless the $\overline{ICPRT}$ configuration bit is set. For NC/ICPORTS, the pin is No Connect unless ICPRT is set and the $\overline{DEBUG}$ configuration bit is cleared.

TABLE 1-3: PIC18F4455/4550 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number PDIP	QFN	TQFP	Pin Type	Buffer Type	Description
						PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs.
RB0/AN12/INT0/FLT0/SDI/SDA	33	9	8			
RB0				I/O	TTL	Digital I/O.
AN12				I	Analog	Analog input 12.
INT0				I	ST	External interrupt 0.
FLT0				I	ST	Enhanced PWM Fault input (ECCP1 module).
SDI				I	ST	SPI™ data in.
SDA				I/O	ST	I²C™ data I/O.
RB1/AN10/INT1/SCK/SCL	34	10	9			
RB1				I/O	TTL	Digital I/O.
AN10				I	Analog	Analog input 10.
INT1				I	ST	External interrupt 1.
SCK				I/O	ST	Synchronous serial clock input/output for SPI mode.
SCL				I/O	ST	Synchronous serial clock input/output for I²C mode.
RB2/AN8/INT2/VMO	35	11	10			
RB2				I/O	TTL	Digital I/O.
AN8				I	Analog	Analog input 8.
INT2				I	ST	External interrupt 2.
VMO				O	—	External USB transceiver VMO output.
RB3/AN9/CCP2/VPO	36	12	11			
RB3				I/O	TTL	Digital I/O.
AN9				I	Analog	Analog input 9.
CCP2(1)				I/O	ST	Capture 2 input/Compare 2 output/PWM 2 output.
VPO				O	—	External USB transceiver VPO output.
RB4/AN11/KBI0/CSSPP	37	14	14			
RB4				I/O	TTL	Digital I/O.
AN11				I	Analog	Analog input 11.
KBI0				I	TTL	Interrupt-on-change pin.
CSSPP				O	—	SPP chip select control output.
RB5/KBI1/PGM	38	15	15			
RB5				I/O	TTL	Digital I/O.
KBI1				I	TTL	Interrupt-on-change pin.
PGM				I/O	ST	Low-Voltage ICSP™ Programming enable pin.
RB6/KBI2/PGC	39	16	16			
RB6				I/O	TTL	Digital I/O.
KBI2				I	TTL	Interrupt-on-change pin.
PGC				I/O	ST	In-Circuit Debugger and ICSP programming clock pin.
RB7/KBI3/PGD	40	17	17			
RB7				I/O	TTL	Digital I/O.
KBI3				I	TTL	Interrupt-on-change pin.
PGD				I/O	ST	In-Circuit Debugger and ICSP programming data pin.

Legend: TTL = TTL compatible input; CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; I = Input; O = Output; P = Power

Note 1: Alternate assignment for CCP2 when CCP2MX configuration bit is cleared.

2: Default assignment for CCP2 when CCP2MX configuration bit is set.

3: These pins are No Connect unless the ICPRT configuration bit is set. For NC/ICPORTS, the pin is No Connect unless ICPRT is set and the DEBUG configuration bit is cleared.

TABLE 1-3: PIC18F4455/4550 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	QFN	TQFP			
						PORTC is a bidirectional I/O port.
RC0/T1OSO/T13CKI	15	34	32			
RC0				I/O	ST	Digital I/O.
T1OSO				O	—	Timer1 oscillator output.
T13CKI				I	ST	Timer1/Timer3 external clock input.
RC1/T1OSI/CCP2/ $\overline{\text{UOE}}$	16	35	35			
RC1				I/O	ST	Digital I/O.
T1OSI				I	CMOS	Timer1 oscillator input.
CCP2[2]				I/O	ST	Capture 2 input/Compare 2 output/PWM 2 output.
$\overline{\text{UOE}}$				O	—	External USB transceiver $\overline{\text{OE}}$ output.
RC2/CCP1/P1A	17	36	36			
RC2				I/O	ST	Digital I/O.
CCP1				I/O	ST	Capture 1 input/Compare 1 output/PWM 1 output.
P1A				O	TTL	Enhanced CCP1 PWM output, channel A.
RC4/D-/VM	23	42	42			
RC4				I	TTL	Digital input.
D-				I/O	—	USB differential minus line (input/output).
VM				I	TTL	External USB transceiver VM input.
RC5/D+/VP	24	43	43			
RC5				I	TTL	Digital input.
D+				I/O	—	USB differential plus line (input/output).
VP				I	TTL	External USB transceiver VP input.
RC6/TX/CK	25	44	44			
RC6				I/O	ST	Digital I/O.
TX				O	—	EUSART asynchronous transmit.
CK				I/O	ST	EUSART synchronous clock (see RX/DT).
RC7/RX/DT/SDO	26	1	1			
RC7				I/O	ST	Digital I/O.
RX				I	ST	EUSART asynchronous receive.
DT				I/O	ST	EUSART synchronous data (see TX/CK).
SDO				O	—	SPI™ data out.

Legend: TTL = TTL compatible input; CMOS = CMOS compatible input or output
ST = Schmitt Trigger input with CMOS levels; I = Input
O = Output; P = Power

Note 1: Alternate assignment for CCP2 when CCP2MX configuration bit is cleared.
2: Default assignment for CCP2 when CCP2MX configuration bit is set.
3: These pins are No Connect unless the ICPRT configuration bit is set. For NC/ICPORTS, the pin is No Connect unless ICPRT is set and the $\overline{\text{DEBUG}}$ configuration bit is cleared.

PIC18F2455/2550/4455/4550

TABLE 1-3: PIC18F4455/4550 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number PDIP	QFN	TQFP	Pin Type	Buffer Type	Description
						PORTD is a bidirectional I/O port or a Streaming Parallel Port (SPP). These pins have TTL input buffers when the SPP module is enabled.
RD0/SPP0	19	38	38			
RD0				I/O	ST	Digital I/O.
SPP0				I/O	TTL	Streaming Parallel Port data.
RD1/SPP1	20	39	39			
RD1				I/O	ST	Digital I/O.
SPP1				I/O	TTL	Streaming Parallel Port data.
RD2/SPP2	21	40	40			
RD2				I/O	ST	Digital I/O.
SPP2				I/O	TTL	Streaming Parallel Port data.
RD3/SPP3	22	41	41			
RD3				I/O	ST	Digital I/O.
SPP3				I/O	TTL	Streaming Parallel Port data.
RD4/SPP4	27	2	2			
RD4				I/O	ST	Digital I/O.
SPP4				I/O	TTL	Streaming Parallel Port data.
RD5/SPP5/P1B	28	3	3			
RD5				I/O	ST	Digital I/O.
SPP5				I/O	TTL	Streaming Parallel Port data.
P1B				O	—	Enhanced CCP1 PWM output, channel B.
RD6/SPP6/P1C	29	4	4			
RD6				I/O	ST	Digital I/O.
SPP6				I/O	TTL	Streaming Parallel Port data.
P1C				O	—	Enhanced CCP1 PWM output, channel C.
RD7/SPP7/P1D	30	5	5			
RD7				I/O	ST	Digital I/O.
SPP7				I/O	TTL	Streaming Parallel Port data.
P1D				O	—	Enhanced CCP1 PWM output, channel D.

Legend: TTL = TTL compatible input; CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; I = Input; O = Output; P = Power

Note 1: Alternate assignment for CCP2 when CCP2MX configuration bit is cleared.

2: Default assignment for CCP2 when CCP2MX configuration bit is set.

3: These pins are No Connect unless the ICPRT configuration bit is set. For NC/ICPORTS, the pin is No Connect unless ICPRT is set and the $\overline{DEBUG}$ configuration bit is cleared.

TABLE 1-3: PIC18F4455/4550 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	QFN	TQFP			
						PORTE is a bidirectional I/O port.
RE0/AN5/CK1SPP	8	25	25			
RE0				I/O	ST	Digital I/O.
AN5				I	Analog	Analog input 5.
CK1SPP				O	—	SPP clock 1 output.
RE1/AN6/CK2SPP	9	26	26			
RE1				I/O	ST	Digital I/O.
AN6				I	Analog	Analog input 6.
CK2SPP				O	—	SPP clock 2 output.
RE2/AN7/OESPP	10	27	27			
RE2				I/O	ST	Digital I/O.
AN7				I	Analog	Analog input 7.
OESPP				O	—	SPP output enable output.
RE3	—	—	—	—	—	See $\overline{MCLR}$/VPP/RE3 pin.
VSS	12, 31	6, 30, 31	6, 29	P	—	Ground reference for logic and I/O pins.
VDD	11, 32	7, 8, 28, 29	7, 28	P	—	Positive supply for logic and I/O pins.
VUSB	18	37	37	O	—	Internal USB 3.3V voltage regulator output.
NC/ICCK/ICPGC	—	—	12			No Connect or dedicated ICD/ICSP™ port clock.(3)
ICCK				I/O	ST	In-Circuit Debugger clock.
ICPGC				I/O	ST	ICSP programming clock.
NC/ICDT/ICPGD	—	—	13			No Connect or dedicated ICD/ICSP port clock.(3)
ICDT				I/O	ST	In-Circuit Debugger data.
ICPGD				I/O	ST	ICSP programming data.
NC/$\overline{ICRST}$/ICVPP	—	—	33			No Connect or dedicated ICD/ICSP port Reset.(3)
$\overline{ICRST}$				I	—	Master Clear (Reset) input.
ICVPP				P	—	Programming voltage input.
NC/ICPORTS	—	—	34	P	—	No Connect or 28-pin device emulation.(3)
ICPORTS						Enable 28-pin device emulation when connected to VSS.
NC	—	13	—	—	—	No Connect.

Legend: TTL = TTL compatible input; CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; I = Input; O = Output; P = Power

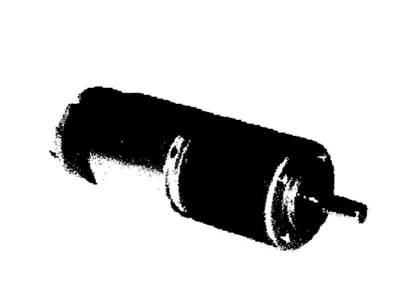
Note 1: Alternate assignment for CCP2 when CCP2MX configuration bit is cleared.

2: Default assignment for CCP2 when CCP2MX configuration bit is set.

3: These pins are No Connect unless the ICPRT configuration bit is set. For NC/ICPORTS, the pin is No Connect unless ICPRT is set and the $\overline{DEBUG}$ configuration bit is cleared.

PD3046 GEARED MOTOR SERIES 12 AND 24V TYPE

Brush commutator planetary dc gear motors with strong metal gears of straight type manufactured by milling machine process. Only the motor shaft gear pinion is made of bakelite for noise reduction purpose. The output shaft is made in hardened stainless steel and has a ball bearing output as standard. This is marked as a B in the part description. For electric noise suppression the dc motors have as standard and if available integral filters as either with varistor and / or capacitor. This is marked as a F in the part description. There are options for encoder assembly and special adaptations. All motors are produced under strict high quality production standards and are tested and inspected prior to delivery.



TRANSMOTEC SWEDEN AB

MODEL NO. DESIGNATIONS

MODEL - VOLTAGE - RATIO - B - F - E - C

B = Ball bearing
F = Filter
C = Cover
E = Encoder

Example: PD3046-12-14-BFEC

ENCODER VERSION

SEE SEPARATE DATA SHEET

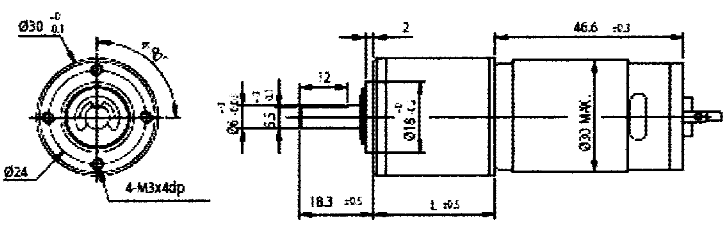
GEAR HEAD DATA

- Ball bearing
- Planetary gear
- Backslash at no load ≤ 2.5 °
- Radial (10mm from flange) load ≤ 3.5kg)
- Axial load ≤ 2.5kg
- Press fit max. ≤ 10kg
- Radial play ≤ 0.04mm
- Thrust play ≤ 0.3mm

GEARED MOTOR DATA - 24V

Reduction	5:1	14:1	19:1	27:1	39:1	51:1	71:1	100:1	139:1	189:1	264:1	516:1	721:1	939:1
Reduction absolute	5 2/11	13 212/289	19 38/187	[illegible] 103/121	34 43/44	50 [illegible]	71 [illegible]	99 [illegible]	139 184/1331	188 [illegible]	263 [illegible]	515 [illegible]	720 [illegible]	939 [illegible]
Number of stages	1	2	2	2	2	3	3	3	3	4	4	4	4	4
Rated torque (kg-cm)	0.54	1.2	1.7	2.4	3.1	4.0	5.5	7.8	8	10	10	10	10	10
Rated speed (rpm)	1170	445	320	229	176	121	87	62	46	34.5	26	13.5	9.8	7.3
Length L1 (mm)	23.6	30	30	30	30	36.4	36.4	36.4	36.4	42.8	42.8	42.8	42.8	42.8
Weight (g)	188	201	201	201	201	216	216	216	216	231	231	231	231	231

APPEARANCE SIZE

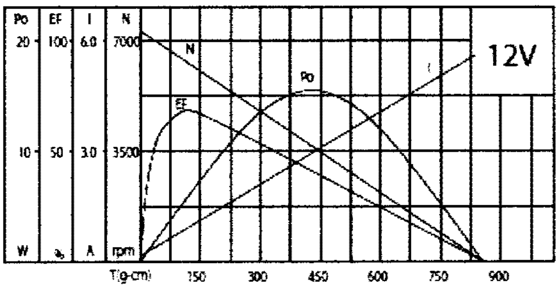


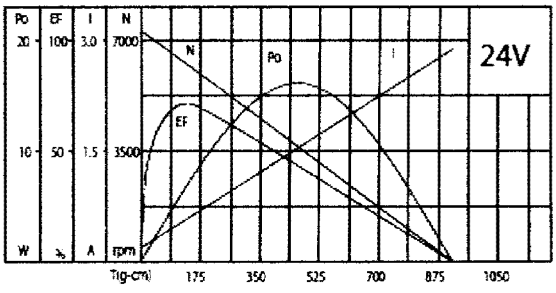
MOTOR FEATURES

- Precious metal brush commutator dc motors
- Gears in steel manufactured by milling process
- Ball bearing supported gear shaft if (B)
- Integrated electric noise suppression filter if (F)
- Custom adaptations are possible
- Encoder version is available
- Operating temp. range -10-..+80 °C

MOTOR DATA

Rated volt (V)	Rated torque (g-cm)	Rated speed (rpm)	Rated Current (mA)	No load speed (rpm)	No load current (mA)	Rated output (W)
12	110	5950	≤900	7300	≤150	7.0
24	130	6160	≤500	7300	≤80	8.5





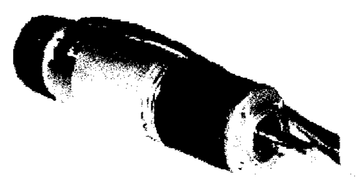
Go to Table of Contents

Transmotec

ENCODER VERSIONS FOR PD, SD and SDC MOTOR SERIES

Encoders are available for all our planetary and spur dc gear motors in this product range. This document describes the data for only the encoders. For motor data please refer to the separate motor data sheets. The encoder design is based on a rotating magnetic disk and two hall sensors. It is a simple, long life, robust and low cost technology. In standard our encoders generate the maximum number of pulses that are possible in relation to the diameter of the magnet and encoder.

MAGNETIC TWO CHANNEL HALL EFFECT ENCODERS



PD3047-12-5-VCEC

SDC3729-12-6-VCEC

SD3729-12-10-VCEC

TRANSMOTEC SWEDEN AB

MODEL NO. DESIGNATIONS

MODEL - VOLTAGE - RATIO - B - F - E - C

B = Ball bearing
F = Filter
C = Cap
E = Encoder

Example: PD3346-12-14-BFEC

ENCODER FEATURES

- Magnetic hall sensor encoders
- Two sensors channels is standard
- Available for all our gear motors
- Simple design
- Robust and long lifetime
- Low cost
- High number of pulses

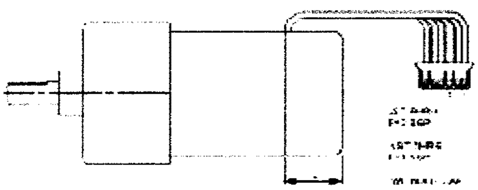
ENCODER SHAPE AND OUTPUT

MOTOR	CAP ø	LENGTH	MAGNETIC POLES N + S	PPR	CPR
PD1427	NO CAP		3	6	12
PD2132	NO CAP		3	6	12
PD3047	27.2		13	26	52
PD3347	36.2		13	26	52
PD3257	27.2		13	26	52
PD4366	42.2		19	38	76
SD3729	32.2		13	26	52
SDC3729	32.2		13	26	52
SD3839	27.2		13	26	52
SD3157	36.2		13	26	52
SD4860	36.2		13	26	52

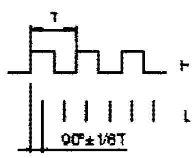
GEARED MOTOR DATA

CHARACTERISTICS	SYMBOL	TEST CONDITIONS	MIN	REF	MAX	UNITS
Supply voltage	Vcc	—	3.5	-	20	V
Output saturation voltage	Vce sat	Vcc = 14V ; Ic = 20mA	-	300	700	mV
Output leakage current	Icex	Vce = 14V ; Vcc = 14V	-	<0.1	10	uV
Supply Current	Icc	Vcc = 20V Output open	-	5	10	mA
Output rise time	tr	Vcc = 14V; RL = 820 Ohm; CL = 20pF		0.3	1.5	uS
Output fall time	tf	Vcc = 14V; RL = 820 Ohm; CL = 20pF	-	0.3	1.5	uS
Hall sensor	LTC Unisonic KS1816					

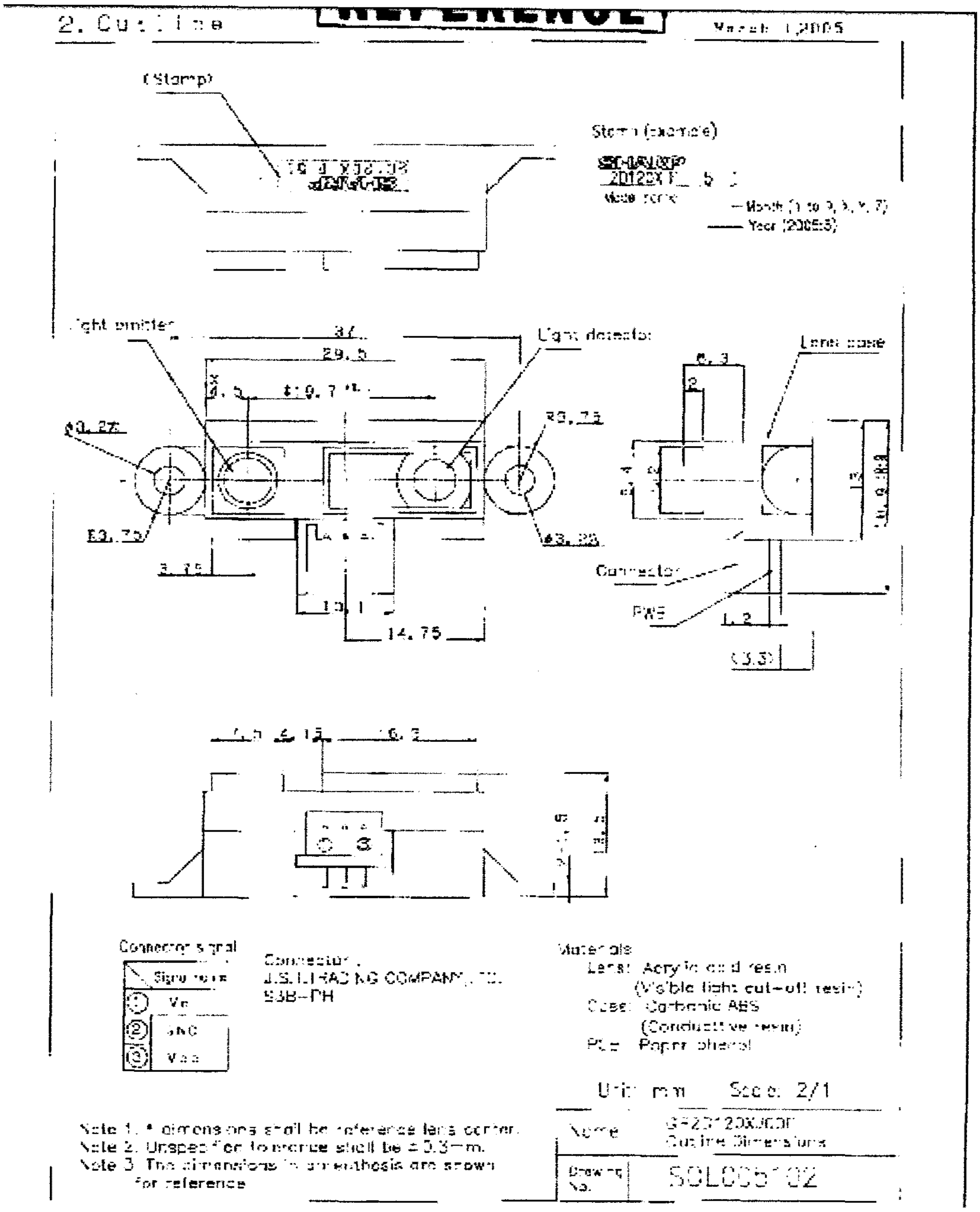
APPEARANCE SIZE



OUTPUT WAVE



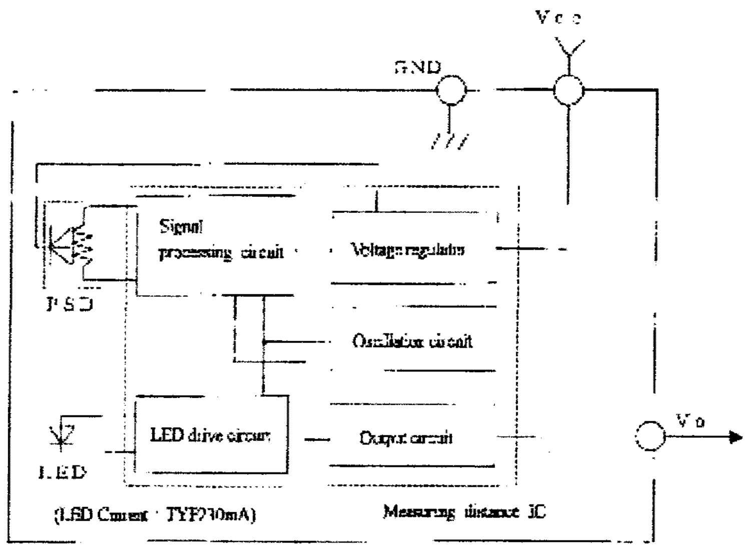
TWO CHANNEL ENCODER	
CONNECTIONS:	
1. BLACK	- MOTOR
2. RED	+ MOTOR
3. BROWN	: HALL SENSOR Vcc
4. GREEN	: HALL SENSOR GND
5. BLUE	: HALL SENSOR A Vout
6. PURPLE	: HALL SENSOR B Vout



2. Outline
(Stamp)
Light emitter
Light detector
Lens case
Connector
PWB
37
29.5
14.75
(3.3)
Connector signal
Signal name
Vo
GND
Vcc
Connector :
J.S.T. TRADING COMPANY, LTD.
S3B-PH
Materials
Lens: Acrylic acid resin
(Visible light cut-off resin)
Case: Carbonic ABS
(Conductive resin)
PCB: Paper phenol
Unit: mm
Scale: 2/1
Note 1. * dimensions shall be reference lens center.
Note 2. Unspecified tolerance shall be ±0.3mm.
Note 3. The dimensions in parenthesis are shown for reference
Name
Outline Dimensions
Drawing No.
SOL005102

REFERENCE

3-1 Schematic



3-2 Absolute maximum ratings (Ta=25°C, Vcc=5V)

Parameter	Symbol	Ratings	Unit	Remark
Supply voltage	V_{cc}	-0.3 to +7	V	-
Output terminal voltage	V_o	-0.3 to Vcc+0.3	V	-
Operating temperature	Topr	-10 to +60	°C	-
Storage temperature	Tstg	-40 to +70	°C	-

Operating supply voltage

Symbol	Rating	Unit	Remark
V_{cc}	4.5 to 5.5	V	-

REFERENCE

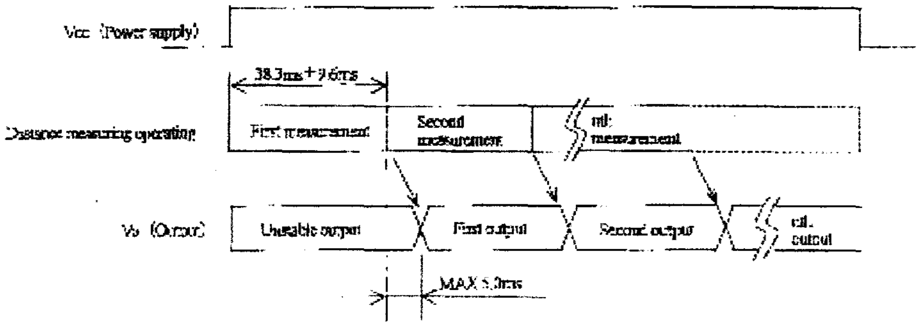
3-3 Electro-optical Characteristics

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Measuring distance range	ΔL	(Note 1)	4	-	30	cm
Output terminal voltage	V_O	L=30cm (Note 1)	0.25	0.4	0.55	V
Output voltage difference	ΔV_O	Output change at L change (30cm → 4cm) (Note 1)	1.95	2.25	2.55	V
Average supply current	I_{CC}	L=30cm (Note 1)	-	33	50	mA

※ L : Distance to reflective object

(Note 1) Using reflective object : White paper (Made by Kodak Co., Ltd. gray cards R-27 · white face, reflective ratio ; 90%)

3-4 Timing chart



REFERENCE

4. Reliability

The reliability of products shall be satisfied with items listed below.

Confidence level : 90%

LTPD : 20 or 30

No.	Test Items	Test Conditions	Failure Judgement Criteria	Samples (n) Defective (c)
1	Temperature cycling	1 cycle -40°C ⇄ +70°C (30min.) (30min.) 25 cycle test	Initial × 0.8 > Vo Vo > Initial × 1.2 (Note 1)	n=11, c=0
2	High temp. and high humidity storage	+40°C, 90%RH, 500h		n=11, c=0
3	High temp. storage	+70°C, 500h		n=11, c=0
4	Low temp. storage	-40°C, 500h		n=11, c=0
5	Operation life (High temp.)	+60°C, Vcc=5V, 500h		n=11, c=0
6	Mechanical shock	1000m/s^2, 6.0ms 3times/±X, ±Y, ±Z direction		n=8, c=0
7	Variable frequency vibration	10 to 55 to 10Hz/1min. 2h/X, Y, Z direction overall amplitude : 1.5mm		n=8, c=0

(Note 1) Test conditions are according to 3-3 Electro-optical characteristics.

(Note 2) After test, measurement shall be measured after leaving under the normal temperature and the normal humidity for two hours. But no dew point.

5. Outgoing inspection

(1) Inspection lot

Inspection shall be carried out per each delivery lot.

(2) Inspection method

A single sampling plan, normal inspection level II based on ISO 2859 is applied.

The AQL according to the inspection items are shown below.

Defect	Inspection item	AQL (%)
Major defect	Electro-optical characteristics defect (in para. 3-3)	0.4
Minor defect	Defect on appearance and dimension ※ Crack, chip, scratch, stain	1

※ Crack, chip, scratch, stain

One which affects the characteristics of para. 3-3 shall be defect.

7 Notes [redacted]

[Advice for the optics]

7-1 Lens of this device shall be kept cleanly. There are cases that dust, water or oil and so on deteriorate the characteristics of this device. Please consider in actual application.

7-2 In case that protection is set in front of the emitter and detector portion, the protection cover which has the most efficient transmittance at the emitting wavelength range of LED for this product (λ=850nm±70nm), shall be recommended to use. The face and back of protection cover should be mirror polishing. Also, as there are cases that the characteristics may not be satisfied with according to the distance between the protection cover and this product or the thickness of the protection cover, please use this product after confirming the operation sufficiently in actual application.

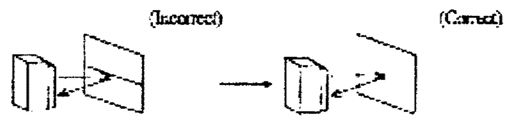
[Advice for the characteristics]

7-3 In case that there is an object near to light axis of the sensor between the sensor and the detected object, please use this device after confirming sufficiently what the characteristics of this sensor do not change by the object.

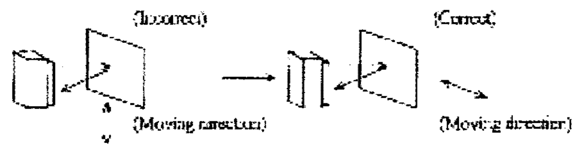
7-4 When the detector surface receive direct light from the sun, tungsten lamp and so on, there are cases that it can not measure the distance exactly. Please consider the design that the detector does not receive direct light from such light source.

7-5 Distance between sensor and mirror reflector can not sometimes measure exactly. In case of changing the mounting angle of this product, it may measure the distance exactly.

7.6 In case that reflective object has boundary line clearly, there is cases that distance can not measure exactly. At that time, if direction of boundary line and the line between emitter center and detector center parallels, it is possible to decrease deviation of measuring distance.



7-7 In order to decrease measuring error by moving direction of object, we recommend to mount the sensor like below drawing.



7-8 In order to stabilize power supply line, we recommend to connect a by-pass capacitor of 10μF or more between Vcc and GND near this product.

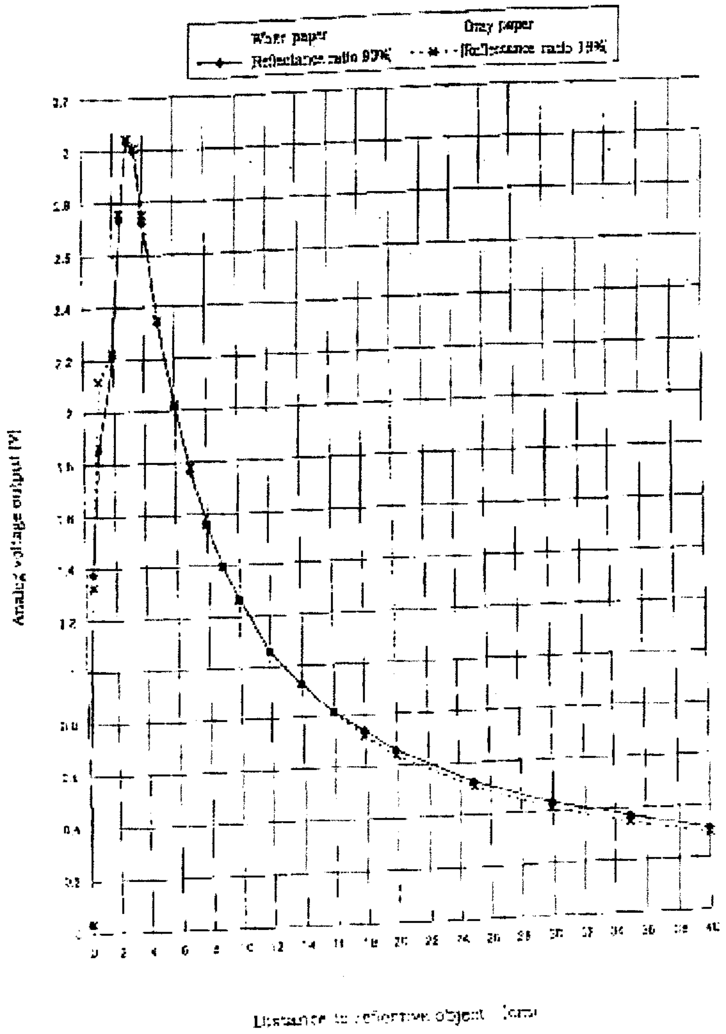
[Notes on handling]

7-9 Please don't do washing. Washing may deteriorate the characteristics of optical system and so on. Please confirm resistance to chemicals under the actual usage since this product has not been designed with the consideration

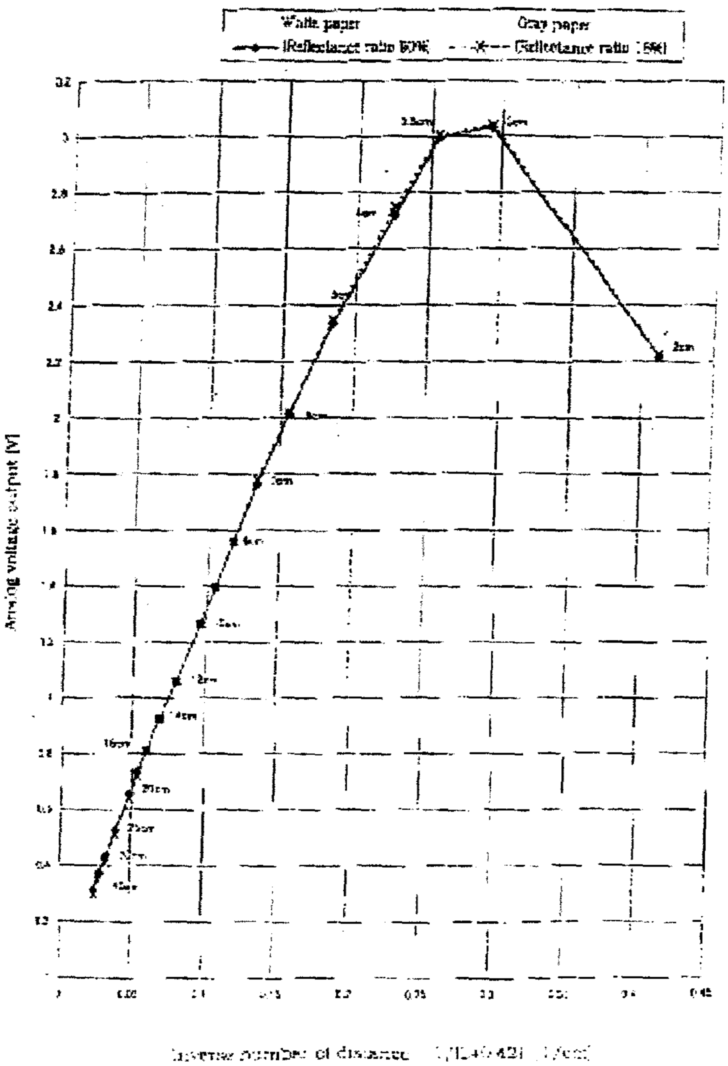
7-10 There are some possibilities that the sensor inside the case package with lens may be exposed to the excessive mechanical stress. Please be careful not to cause any excessive pressure on the case package with lens and also on the PCB at the assembly and inserting of the set.

REFERENCE

6-1 GP2D120XJ00F Example of output distance characteristics



6-2 GP2D120XJ00F Example of output distance characteristics with inverse number of distance



National Semiconductor

February 1995

LM78XX Series Voltage Regulators

General Description

The LM78XX series of three terminal regulators is available with several fixed output voltages making them useful in a wide range of applications. One of these is local on card regulation, eliminating the distribution problems associated with single point regulation. The voltages available allow these regulators to be used in logic systems, instrumentation, HiFi, and other solid state electronic equipment. Although designed primarily as fixed voltage regulators these devices can be used with external components to obtain adjustable voltages and currents.

The LM78XX series is available in an aluminum TO-3 package which will allow over 1.0A load current if adequate heat sinking is provided. Current limiting is included to limit the peak output current to a safe value. Safe area protection for the output transistor is provided to limit internal power dissipation. If internal power dissipation becomes too high for the heat sinking provided, the thermal shutdown circuit takes over preventing the IC from overheating.

Considerable effort was expended to make the LM78XX series of regulators easy to use and minimize the number of external components. It is not necessary to bypass the output, although this does improve transient response. Input bypassing is needed only if the regulator is located far from the filter capacitor of the power supply.

For output voltage other than 5V, 12V and 15V the LM117 series provides an output voltage range from 1.2V to 57V.

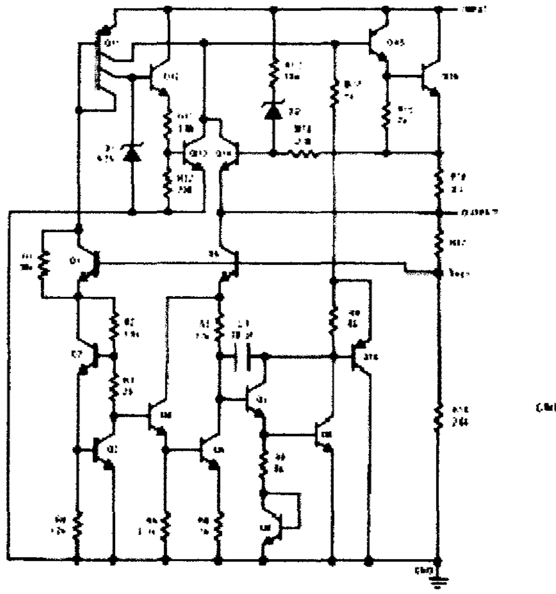
Features

- Output current in excess of 1A
- Internal thermal overload protection
- No external components required
- Output transistor safe area protection
- Internal short circuit current limit
- Available in the aluminum TO-3 package

Voltage Range

LM7805C	5V
LM7812C	12V
LM7815C	15V

Schematic and Connection Diagrams



TL/H/7746-1

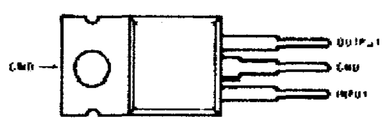
Metal Can Package
TO-3 (K)
Aluminum

TL/H/7746-2

Bottom View

Order Number LM7805CK, LM7812CK or LM7815CK
See NS Package Number KC02A

Plastic Package
TO-220 (T)



TL/H/7746-3

Top View

Order Number LM7805CT, LM7812CT or LM7815CT
See NS Package Number T03B

Absolute Maximum Ratings

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Input Voltage (V_O = 5V, 12V and 15V)	35V
Internal Power Dissipation (Note 1)	Internally Limited
Operating Temperature Range (T_A)	0°C to +70°C
Maximum Junction Temperature	
(K Package)	150°C
(T Package)	150°C
Storage Temperature Range	−65°C to +150°C
Lead Temperature (Soldering, 10 sec.)	
TO-3 Package K	300°C
TO-220 Package T	230°C

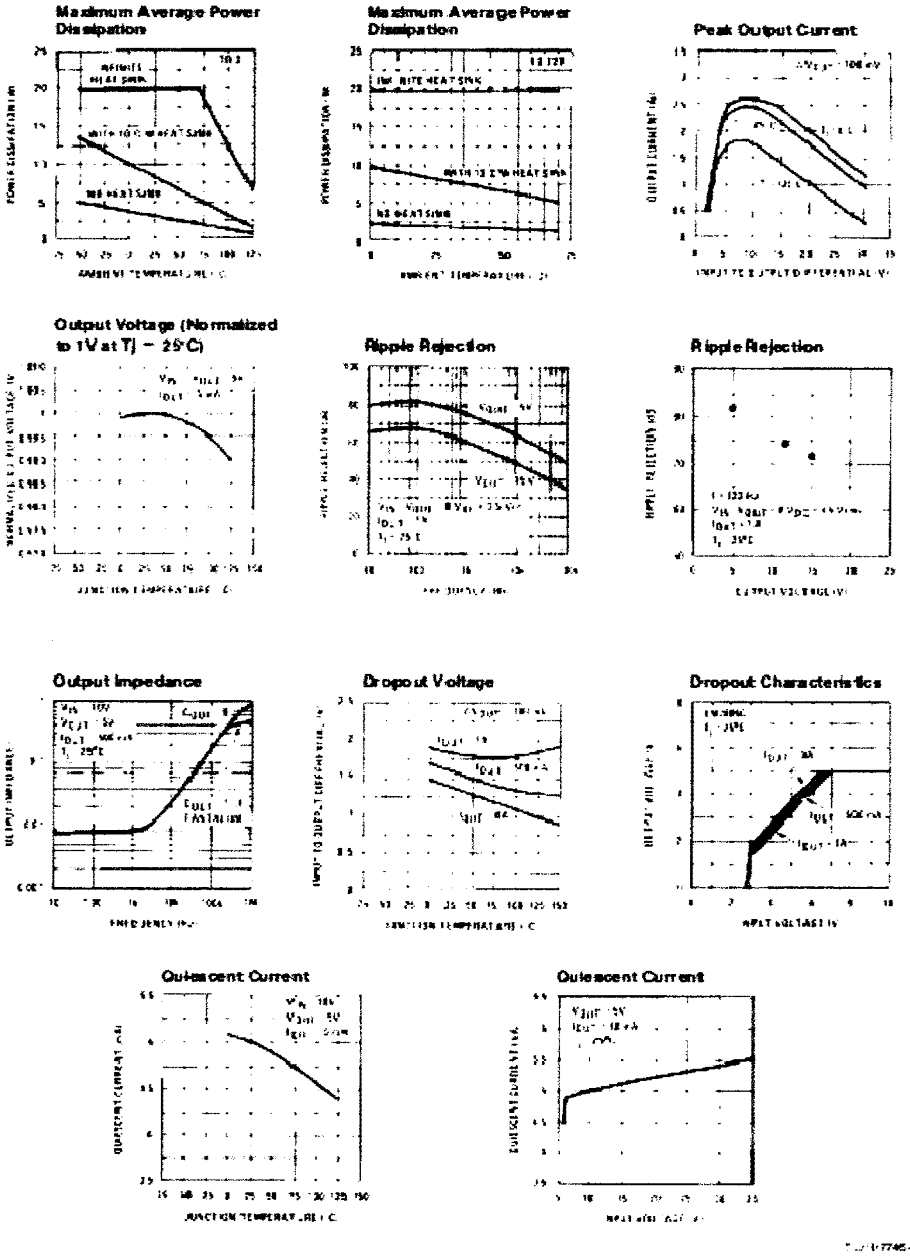
Electrical Characteristics LM78XXC (Note 2) 0°C ≤ T_J ≤ 125°C unless otherwise noted.

Output Voltage				5V			12V			15V			Units
Input Voltage (unless otherwise noted)				10V			19V			23V			
Symbol	Parameter	Conditions		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
V_O	Output Voltage	T_J = 25°C, 5 mA ≤ I_O ≤ 1A		4.8	5	5.2	11.5	12	12.5	14.4	15	15.6	V
		P_D ≤ 15W, 5 mA ≤ I_O ≤ 1A		4.75		5.25	11.4		12.6	14.25		15.75	V
		V_{MIN} ≤ V_{IN} ≤ V_{MAX}		(7.5 ≤ V_{IN} ≤ 20)			(14.5 ≤ V_{IN} ≤ 27)			(17.5 ≤ V_{IN} ≤ 30)			V
ΔV_O	Line Regulation	I_O = 500 mA	T_J = 25°C		3	50		4	120		4	150	mV
			ΔV_{IN}	(7 ≤ V_{IN} ≤ 25)			14.5 ≤ V_{IN} ≤ 30)			(17.5 ≤ V_{IN} ≤ 30)			V
			0°C ≤ T_J ≤ +125°C			50			120			150	mV
			ΔV_{IN}	(8 ≤ V_{IN} ≤ 20)			(15 ≤ V_{IN} ≤ 27)			(18.5 ≤ V_{IN} ≤ 30)			V
		I_O ≤ 1A	T_J = 25°C			50			120			150	mV
			ΔV_{IN}	(7.5 ≤ V_{IN} ≤ 20)			(14.6 ≤ V_{IN} ≤ 27)			(17.7 ≤ V_{IN} ≤ 30)			V
			0°C ≤ T_J ≤ +125°C			25			60			75	mV
			ΔV_{IN}	(8 ≤ V_{IN} ≤ 12)			(16 ≤ V_{IN} ≤ 22)			(20 ≤ V_{IN} ≤ 26)			V
ΔV_O	Load Regulation	T_J = 25°C	5 mA ≤ I_O ≤ 1.5A		10	50		12	120		12	150	mV
			250 mA ≤ I_O ≤ 750 mA			25			60			75	mV
		5 mA ≤ I_O ≤ 1A, 0°C ≤ T_J ≤ +125°C				50			120			150	mV
I_Q	Quiescent Current	I_O ≤ 1A	T_J = 25°C			8			8			8	mA
			0°C ≤ T_J ≤ +125°C			8.5			8.5			8.5	mA
ΔI_Q	Quiescent Current Change	5 mA ≤ I_O ≤ 1A				0.5			0.5			0.5	mA
		T_J = 25°C, I_O ≤ 1A				1.0			1.0			1.0	mA
		V_{MIN} ≤ V_{IN} ≤ V_{MAX}		(7.5 ≤ V_{IN} ≤ 20)			(14.8 ≤ V_{IN} ≤ 27)			(17.9 ≤ V_{IN} ≤ 30)			V
		I_O ≤ 500 mA, 0°C ≤ T_J ≤ +125°C				1.0			1.0			1.0	mA
		V_{MIN} ≤ V_{IN} ≤ V_{MAX}		(7 ≤ V_{IN} ≤ 25)			(14.5 ≤ V_{IN} ≤ 30)			(17.5 ≤ V_{IN} ≤ 30)			V
V_N	Output Noise Voltage	T_A = 25°C, 10 Hz ≤ f ≤ 100 kHz			40			75			90		μV
$\frac{\Delta V_{IN}}{\Delta V_{OUT}}$	Ripple Rejection	f = 120 Hz	I_O ≤ 1A, T_J = 25°C or	62	80		55	72		54	70		dB
			I_O ≤ 500 mA, 0°C ≤ T_J ≤ +125°C	62			55			54			dB
		V_{MIN} ≤ V_{IN} ≤ V_{MAX}		(8 ≤ V_{IN} ≤ 18)			(15 ≤ V_{IN} ≤ 25)			(18.5 ≤ V_{IN} ≤ 28.5)			V
R_O	Dropout Voltage	T_J = 25°C, I_{OUT} = 1A			2.0			2.0			2.0		V
	Output Resistance	f = 1 kHz			8			18			19		mΩ
	Short-Circuit Current	T_J = 25°C			2.1			1.5			1.2		A
	Peak Output Current	T_J = 25°C			2.4			2.4			2.4		A
	Average TC of V_{OUT}	0°C ≤ T_J ≤ +125°C, I_O = 5 mA			0.6			1.5			1.8		mV/°C
V_{IN}	Input Voltage Required to Maintain Line Regulation	T_J = 25°C, I_O ≤ 1A			7.5		14.6			17.7			V

Note 1: Thermal resistance of the TO-3 package (K, KC) is typically 4°C/W junction to case and 35°C/W case to ambient. Thermal resistance of the TO-220 package (T) is typically 4°C/W junction to case and 50°C/W case to ambient.

Note 2: All characteristics are measured with capacitor across the input of 0.22 μF, and a capacitor across the output of 0.1 μF. All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques (t_w ≤ 10 ms, duty cycle ≤ 5%). Output voltage changes due to changes in internal temperature must be taken into account separately.

Typical Performance Characteristics



APPENDIX B

FIRMWARE SOURCE CODE

PBP FORMAT

```
'****************************************************************
'*  Name     : MAZE MAPPER ROBOT.BAS                          *
'*  Author   : [MOHD HASNAN BIN NORDIN (ED07001) (4BEC)]       *
'*  Notice   : Copyright (c) 2009 [select VIEW...EDITOR OPTIONS] *
'*           : All Rights Reserved                            *
'*  Date     : 11/12/2009                                     *
'*  Version  : 1.0                                            *
'*  Notes    :                                                *
'*           :                                                *
'****************************************************************
define osc 8
'**************************DEFINITION *************************
'defenition
DEFINE CCP1_REG PORTC   ' Hpwm 1 pin port
DEFINE CCP1_BIT 2       ' Hpwm 1 pin bit
DEFINE CCP2_REG PORTC   ' Hpwm 2 pin port
DEFINE CCP2_BIT 1       ' Hpwm 2 pin bit
DEFINE ADC_BITS 8       ' Set number of bits in result (8, 10 or 12)
DEFINE ADC_CLOCK 3      ' Set clock source (rc = 3)
DEFINE ADC_SAMPLEUS 50  ' Set sampling time in microseconds
MWPWM1 VAR PORTC.2      'PWM MOTOR 1
MWPWM2 VAR PORTC.1      'PWM MOTOR 2
MRCW1 VAR PORTD.4
MRCCW1 VAR PORTD.5      'MOTOR RIGHT DRIVE
MRCW2 VAR PORTD.6
MRCCW2 VAR PORTD.7      'MOTOR LEFT DRIVE
TRISD.4=0
TRISD.5=0
TRISD.6=0
TRISD.7=0
TRISA.2=1
TRISA.1=1
TRISA.0=1
ADCON1=%00000000
SENSOR1 VAR BYTE     'right sensor
SENSOR2 VAR BYTE     'middle sensor
SENSOR3 VAR BYTE     'left sensor
i var byte 'step in one box
junction var byte 'at the jubction
junction_case var byte  'last case at junction
status_sensor var byte '0 - 1, 1 - 2, 2 -3
porte.7=1      'activate internal resistor

'kire distance
distance var word   'distance in pulse
```

```
counter var word
stat var byte
portin var portd.0
input portin

HPWM 1,50,500
HPWM 2,50,500
i=0
'**************************MAIN PROGRAMMING**********************
'MAIN PROGRAM
MAIN:
    status_sensor=0

    GOSUB IR1    'JUMP TO IR1 SUBOUTINE
    GOSUB IR2    'JUMP TO IR2 SUBROUTINE
    GOSUB IR3    'JUMP TO IR3 SUBROUTINE

    'maze algorithm'

    select case status_sensor

        case 0            'no detection
        gosub fwd          'go straight
         pause 2000        'delay for 1 second
        gosub stopmotor    'robot stop

'**************************start of junction saving************************

        case 4          's3 detect         right  left & forward
          junction=i
          junction_case=4
          gosub bwd1   'go straight
          gosub left    'turn left
          pause 2000    'delay for 1 second
          gosub fwd  'go straight
           gosub bwd1   'go straight
          i=i+1
          pause 2000

        case 2  's2 detect                 forward  left & right
          junction=i
          junction_Case=2
          gosub bwd1   'go straight
          gosub right   'turn right
          pause 2000    'delay for 1 second
          gosub fwd 'go straight
           gosub bwd1   'go straight
```

```
            I=I+1
'           pause 2000

          case 1 's1 detect                  left    FORWARD & RIGHT
            junction=i
            junction_case=1
            gosub bwd1   'go straight
            gosub right  'turn right
'           pause 2000   'delay for 1 second
            gosub fwd 'go straight
'            gosub bwd1   'go straight
'           pause 2000

'****************** end of junction savings******************************

          case 6  's2&s3 detect      forward and right.
            gosub bwd1   'go straight
            gosub left  'turn left
'           pause 1000  'delay for 1 second
            gosub fwd  'go straight
'           pause 2000
            i=i+1      'block+1
'            gosub bwd1   'go straight


          case 5  's1&s3 detect   right and left
            gosub fwd 'go straight
'            pause 2000
            i=i+1      'block+1
'            gosub S1
'            gosub S2
            'gosub left1  'go left a little
            'gosub fwd1 'go straight
            'gosub right1 'go right a little
'            gosub bwd1   'go bwd a little

          case 3  's1&s2 detect      forward and left
            gosub bwd1   'go straight
            gosub right  'turn right
'           pause 2000   'delay for 1 second
            gosub fwd  'go straight
'           pause 2000
            i=i+1      'block+1
'            gosub bwd1   'go straight

'************************* back track case ****************************
```

```
            case 7  's1,s2,s3 detect
               ' go back to the last junction
               while i<>junction
                  gosub bwd  'reverse
                  i=i-1      'block-0
'                 pause 2000  'delay for 1 second
               wend
               if junction_case=4 then
                  gosub right
               else
                  if junction_case=2 then
                     gosub left  '180deg turn
                     gosub left
                  else
                     if junction_case=1 then
                        gosub left
                     endif
                  endif
               endif
               'gosub uturn  'make uturn
'              pause 2000    'delay for 1 second
               gosub fwd  'go straight
               i=i+1
               gosub bwd1   'go straight

'************************* end of back track case ********************
            END SELECT

'gosub fwd
'pause 1000
'gosub right
'pause 1000
'gosub left
'pause 1000
'gosub bwd
'pause 1000
'gosub uturn
'pause 1000

GOTO MAIN    'LOOP TO MAIN FOREVER

'************************* subroutine sensors***************************
'SUBROUTINE SENSOR PROGRAM
IR1:
   adcin 0, SENSOR1
  IF SENSOR1>26 THEN
     status_sensor.0=1
```

```
  ENDIF

RETURN

IR2:
  ADCIN 1,SENSOR2
   IF SENSOR2>48 THEN
      status_sensor.1=1
    ENDIF

RETURN

IR3:
  ADCIN 2,SENSOR3
IF SENSOR3>26 THEN
      status_sensor.2=1
    ENDIF
RETURN

'S1:
'if sensor1>sensor3 then
'gosub right1
'else
'gosub fwd1
'endif
'return

'S2:
'if sensor1<sensor3 then
'gosub left1
'else
'gosub fwd1
'endif
'return

'if sensor1>26 then
'   gosub right1
'endif

'return

'S3:
'if sensor3>26 then
'   gosub left1
'endif

'return
```

```
'************************* subroutine DC motor**************************
'SUBROUTINE DC MOTOR PROGRAM
left:
HPWM 1,50,500
MRCCW1=0
MRCW1=1
HPWM 2,50,505
MRCCW2=0
MRCW2=1
gosub countnstop4
gosub stopmotor
pause 1000
return

'left1:
'HPWM 1,50,500
'MRCCW1=0
'MRCW1=1
'HPWM 2,50,500
'MRCCW2=0
'MRCW2=1
'gosub countnstop6
'gosub stopmotor
'pause 1000
'return

'uturn:
'HPWM 1,50,500
'MRCCW1=1
'MRCW1=0
'HPWM 2,50,500
'MRCCW2=1
'MRCW2=0
''distance=179
'gosub countnstop3
'gosub stopmotor
'pause 1000
'return

fwd:
  HPWM 1,50,530
  MRCCW1=0
  MRCW1=1
  HPWM 2,50,550
  MRCCW2=1
  MRCW2=0
```

```
   gosub countnstop
   gosub stopmotor
   pause 1000

RETURN

'fwd1:
'  HPWM 1,55,500
'  MRCCW1=0
'  MRCW1=1
'  HPWM 2,53,520
'  MRCCW2=1
'  MRCW2=0
'  gosub countnstop7
'  gosub stopmotor
'  pause 1000

stopmotor:
   MRCCW1=0
   MRCW1=0
   MRCCW2=0
   MRCW2=0
RETURN

bwd:
   HPWM 1,50,500
   MRCCW1=1
   MRCW1=0
   HPWM 2,50,530
   MRCCW2=0
   MRCW2=1
   gosub countnstop2
   gosub stopmotor
   pause 1000
return

bwd1:
   HPWM 1,50,500
   MRCCW1=1
   MRCW1=0
   HPWM 2,50,500
   MRCCW2=0
   MRCW2=1
   gosub countnstop5
   gosub stopmotor
   pause 1000
return
```

```
right:
HPWM 1,50,500
MRCCW1=1
MRCW1=0
HPWM 2,50,540
MRCCW2=1
MRCW2=0
gosub countnstop1
gosub stopmotor
pause 1000
return

'right1:
'HPWM 1,50,500
'MRCCW1=1
'MRCW1=0
'HPWM 2,50,500
'MRCCW2=1
'MRCW2=0
'gosub countnstop6
'gosub stopmotor
'pause 1000
'return

'************************** subroutine countnstop ************************'
'SUBROUTINE COUNTNSTOP PROGRAM
countnstop:   'step by step forward
  distance=121
  stat=0
  counter=0
  while distance>counter
    if portin=1 then
      stat=1
    endif
    if stat=1 and portin=0 then
      counter=counter+1
      stat=0
    endif
  wend
  gosub stopmotor
return

countnstop1:     'step by step right and left
  distance=95
  stat=0
  counter=0
```

```
    while distance>counter
      if portin=1 then
        stat=1
      endif
      if stat=1 and portin=0 then
        counter=counter+1
        stat=0
      endif
    wend
    gosub stopmotor
return

countnstop2:  'step by step backward
    distance=121
    stat=0
    counter=0
    while distance>counter
      if portin=1 then
        stat=1
      endif
      if stat=1 and portin=0 then
        counter=counter+1
        stat=0
      endif
    wend
    gosub stopmotor
return

'countnstop3:    'step by step uturn
'   distance=195
'   stat=0
'   counter=0
'   while distance>counter
'     if portin=1 then
'       stat=1
'     endif
'     if stat=1 and portin=0 then
'       counter=counter+1
'       stat=0
'     endif
'   wend
'   gosub stopmotor
'return

countnstop4:      'step by step right and left
    distance=80
    stat=0
```

```
   counter=0
   while distance>counter
      if portin=1 then
         stat=1
      endif
      if stat=1 and portin=0 then
         counter=counter+1
         stat=0
      endif
   wend
   gosub stopmotor
return

countnstop5:     'step by step backward a little
   distance=46
   stat=0
   counter=0
   while distance>counter
      if portin=1 then
         stat=1
      endif
      if stat=1 and portin=0 then
         counter=counter+1
         stat=0
      endif
   wend
   gosub stopmotor
return

'countnstop6:     'step by step left & right a little
'  distance=10
'  stat=0
'  counter=0
'  while distance>counter
'     if portin=1 then
'        stat=1
'     endif
'     if stat=1 and portin=0 then
'        counter=counter+1
'        stat=0
'     endif
'  wend
'  gosub stopmotor
'return

'countnstop7:     'step by step left & right a little
'  distance=10
```

```
'   stat=0
'   counter=0
'   while distance>counter
'      if portin=1 then
'         stat=1
'      endif
'      if stat=1 and portin=0 then
'         counter=counter+1
'         stat=0
'      endif
'   wend
'   gosub stopmotor
'return
```