DEVELOPMENT OF CONTROL SYSTEM OF AUTOMATED GUIDED VEHICLE

ABU HARFIZ BIN HASSAN

Thesis submitted in fulfilment of the requirements
for the award of the degree of
Bachelor of Mechanical Engineering with Automotive Engineering

Faculty of Mechanical Engineering
UNIVERSITI MALAYSIA PAHANG

NOVEMBER 2009

SUPERVISOR’S DECLARATION

We hereby declare that we have checked this project report and in our opinion this project is satisfactory in terms of scope and quality for the award of the degree of Bachelor of Mechanical Engineering with Automotive.

Signature:
Name of Supervisor:
Position:
Date:

STUDENT'S DECLARATION

I hereby declare that the work in this report is my own except for quotations and summaries which have been duly acknowledged. The report has not been accepted for any degree and is not concurrently submitted for award of other degree.

Signature:
Name:
ID Number:
Date:

DEDICATED TO MY BELOVED PARENTS, FAMILY, AND FRIENDS..

THANK YOU FOR ALL YOUR SUPPORT, IDEAS, AND COOPERATION..

ALL OF YOU ALWAYS IN MY HEART FOREVER..

ACKNOWLEDGEMENTS

First of all, I would like to express my grateful to ALLAH s.w.t. for the blessing given to me to complete my project.

In order to complete my work, I have engaged with many people in helping me especially my supervisor Mr. Mohd Fadzil Faisae. I wish to express my sincere appreciation to him for his encouragement, guidance, advices and motivation. Without his support and interest, this thesis would not have been the same as presented here. I also want to thank to all my friends that involve direct or indirect to complete my task.

Not forgotten, I would like to express my gratitude to all university staff from Mechanical Engineering Faculty for being so cooperative and not forgetting all my friends for the support, advice and information sharing.

Finally, I acknowledge my sincere indebtedness and gratitude to my parents for their love, dream and sacrifice throughout my life. I cannot find the appropriate words that could properly describe my appreciation for their devotion, support and faith in my ability to attain my goals. Without endless love and relentless support from my family, I would not have been here.

ABSTRACT

The thesis focuses on the development of control system for automated guide vehicle. The development is referring either develop something new or based on existed on earth. To achieve that, we go to basic concepts of how controlling the microcontroller. It is important to know the features of microcontroller, and what suitable port to used. This development was carried out by doing some fabrication of AGV and programs it using C language. Another additional innovative for improving the control system of AGV is sensor. Research concentrates on type of sensor. It is determining their function, whether line follower AGV or light follower. The AGV can be commercialized and replace human work.

ABSTRAK

Tesis ini menumpukan tentang pembangunan system kawalan kepada AGV. Pembangunan ini boleh merujuk kepada membangunkan sesuatu yang baru atau berdasarkan yang telah wujud atas muka bumi. Untuk mencapai itu, kita merujuk kepada konsep asas dalam mengawal mikrokontroller. Penting untuk mengetahui ciri-ciri mikrokontroller, dan pin kaki yang sesuai. Pembangunan ini akan dijalankan dengan menjalankan fabrikasi kepada AGV dan memprogramkannya menggunakan bahasa C. Lain-lain inovatif tambahan untuk membaiki system kawalan AGV ialah alat deria. Penyelidikan menekankan jenis alat deria. Ia menentukan fungsinya, samada AGV pengikut garisan atau AGV pengikut cahaya. AGV dapat dikomersilkan dan menggantikan kerja manusia.

TABLE OF CONTENTS

Page

LIST OF TABLE

LIST OF FIGURES

LIST OF ABBREVIATIONS

AGV	Automated Guided Vehicle
IR	Infrared Sensor

CHAPTER 1

INTRODUCTION

1.1 Background of the study

The study of this project is more likely focusing on the control system of the AGV, since AGV has other system called mechanical system. It was undertaken by 3 main objectives which will be explained later. Control system of AGV is included taking data from input, process it before send the data into output.

From here, the basic knowledge of programming of microcontroller is important; how to relate every components become cooperate to each other. In addition, some information regarding circuit connection must have. This may be useable when the circuit facing problem. Then, troubleshooting is needed.

1.2 Problem Statement

In industries, some of factories or warehouses are using hands material handling to doing some process. For example Archive Systems, Cross Docking Distribution, High Density Storage, High Speed Sortation, Material Flow and Transport, Production and Manufacturing Delivery Systems, Production and Manufacturing Support Systems, Warehouse Management and Control, Work-In-Process Buffers.[1]

In fact, the hands material is not very efficient in order to increase the quality of their services. There were so many issues such as fatality and injuries related to this hands material handling. Therefore, technology of AGV was introduced to overcome this situation which is AGV technology. [2]

AGV has their own control system which enables it to do their tasks. Control system of AGV is quiet simple idea but to complete it very challenging job. So for here, some modification in control system of AGV to make it capable to overcome the problems related. Perhaps also, the existed control system of AGVs is not very suitable; higher cost, not ergonomic and friendly to human being.

So, for the project of line following of AGV here can overcome the problem. Here, the cost of the project is lowered, and maybe the style to use it very friendly.

Overall, the objective of using an AGV is to eliminate the cost and variance of the human operator in a process that adds no value to the end product – movement between processes.

1.3 Project Objectives

- To develop a control system for automated guided vehicle (AGV).
- To integrate developed control system with automated guided vehicle mechanical system (AGV body).
- To develop guided path of AGV.

1.4 Project Scopes

In this project there are several limitations. The first one is AGV navigation. AGV is moving on line base on sensor. This line is built with white tape. The second one is fabricate option. AGV is just a prototype.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

AGV is one of the most commonly used transportation vehicles in automatic systems. The vehicle, together with AS/RS or production line, is used to carry material, parts, subassemblies, and products from one workstation to another one. [1]

The automated or self-guided vehicle (AGV) is essentially a mobile robot used in the industrial arena to move materials from point-to-point. AGVs can tow objects behind them in small trailers which they can autonomously hook up to. These trailers can be used to move raw materials into line to get them ready to be manufactured.

The AGV can also store objects on a bed. The objects can be placed on a set of motorized treads and then pushed off by reversing them. Some AGVs use forklifts to lift objects for storage. Transporting materials such as medicine in a hospital situation is also done.

2.2 History

The first AGV was brought to market in the 1950s, by Barrett Electronics of Northbrook, Illinois, (now Savant Automation of Walker, Michigan) and at the time it was simply a tow truck that followed a wire in the floor instead of a rail. Over the years the technology has become more sophisticated and today automated vehicles are mainly Laser navigated e.g. LGV (Laser Guided Vehicle). In an automated process, [2] LGVs are programmed to communicate with other robots to ensure product is moved smoothly through the warehouse, whether it is being stored for future use or sent directly to shipping areas. Today, the LGV plays an important role in the design of new factories and warehouses, safely moving goods to their rightful destinations.

2.3 Instructional Robotics

2.3.1 TeRK (Telepresence Robot Kit)

TeRK project was launched on April 26 2007 [6]. It was aimed at combating the dropping numbers in American student's interest in the science and engineering field.

TeRK takes advantage of the fact that there has been a major influx of multimodal technology (USB cameras, cell phones, etc) thus reducing the price. Their aim was to make use of motion, sight, sound and connectivity into the robot and to cost the same as the LEGO Mindstorm (around $250).[12]

These Robots use a Qwerk Microprocessor, which provides high functionality at a low cost (around $150) see figure 3. The Qwerk Processor includes analog ports, digital ports, sensor interfacing, and even two USB sockets which can be used for webcams or even a wireless network adapter.

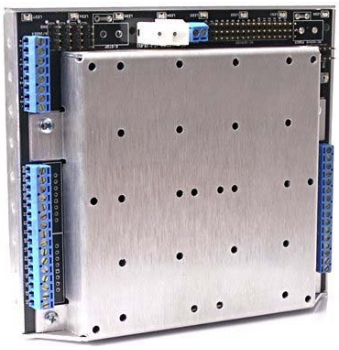


Figure 2.1: Qwerk Microprocessor

Source: Lloyd [12]

From this basis the TeRK Robot has many different configurations, they call recipes. These can be downloaded via the wireless network adapter. None of the recipes require anything more than hand tools and hardware readily available at hobby shops.

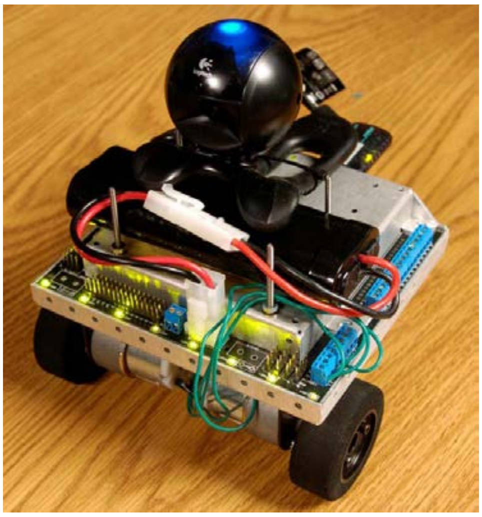


Figure 2.2: TeRK Robot

Source: Lloyd [12]

The Microprocessor does all the work in the robot, there is no possibility of a hardwire solution. The basis of the robot consists of the motors, the chassis, and the microcontroller, so if you wanted to run quite a few robots at once, the basis turns out to be costly.

The robot is not as modular as it sounds, you don’t just put on another component/circuit and change its functionality, and the TeRK system requires mounting, connection and reprogramming.

2.3.2 LEGO Mindstorm

LEGO Mindstorm is a synthesis between toys and engineering. Their product range includes, Mindstorm NXT, which includes building blocks, legs, claws to create a chassis, then made moveable using servo motors, controlled by the brain which is connected to a series of input sensors see figure 5a . The robot is then programmed by a LEGO created software based on a drag and drop system designed for school level education.[13]

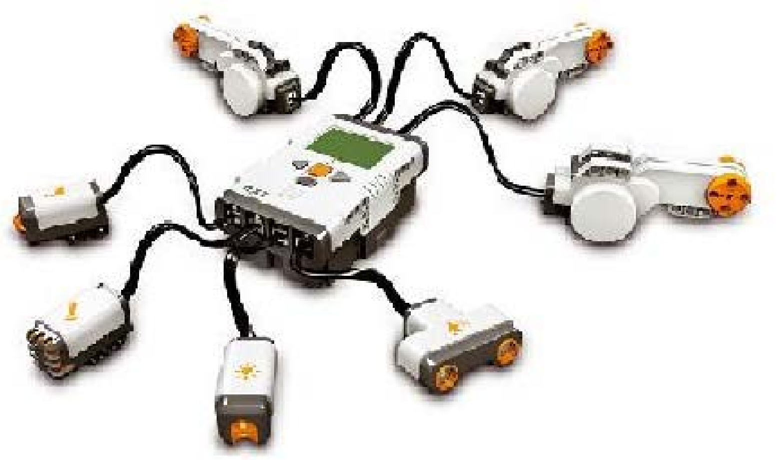


Figure 2.3: LEGO Mindstorm NXT Robotics

Source: Lloyd [12]

The Mindstorms system is very modular, but still only on a toy level. The chassis and robot configuration could to be built to anything (eg humanoid, cranes, etc), but still only runs of the same electro mechanical components.

2.4 Hardware and Software

AGV differs from common vehicles in the guidance system as it is autonomously guided though the sensoring and control system. An AGV consists of hardware and software. The former includes the mechanical structure of the vehicle and the power transmission system as well as electronic hardware for sensoring and control. These two parts are integrated by the control system driven by control schemes, coded in software.

2.4.1 Microcontroller

A microcontroller is a kind of miniature computer that you can find in all kinds of gizmos. Some examples of common, every-day products that have microcontrollers built-in. Popular microcontrollers that have being in great demand for realizing mechatronics system are Motorola's 68HC11, Intel's 8096 and PIC. But here is more focusing on PIC microcontroller.[2]

PIC is a family of Harvard architecture microcontrollers made by Microchip Technology, derived from the PIC1640[1] originally developed by General Instrument's Microelectronics Division. The name PIC initially referred to "Peripheral Interface Controller".[2][4]

PICs are popular with developers and hobbyists alike due to their low cost, wide availability, large user base, extensive collection of application notes, availability of low cost or free development tools, and serial programming (and re-programming with flash memory) capability.

Microchip announced on February 2008 the shipment of its six billionth PIC processor. The PIC architecture is distinctively minimalist. It is characterized by the following features:[4]

- Separate code and data spaces (Harvard architecture)
- A small number of fixed length instructions
- Most instructions are single cycle execution (4 clock cycles), with single delay cycles upon branches and skips
- A single accumulator (W), the use of which (as source operand) is implied (i.e. is not encoded in the opcode)
- All RAM locations function as registers as both source and/or destination of math and other functions.
- A hardware stack for storing return addresses
- A fairly small amount of addressable data space (typically 256 bytes), extended through banking

- Data space mapped CPU, port, and peripheral registers
- The program counter is also mapped into the data space and writable (this is used to implement indirect jumps).

Unlike most other CPUs, there is no distinction between memory space and register space because the RAM serves the job of both memory and registers, and the RAM is usually just referred to as the register file or simply as the registers.

PICs have a set of registers that function as general purpose RAM. Special purpose control registers for on-chip hardware resources are also mapped into the data space. The addressability of memory varies depending on device series, and all PIC devices have some banking mechanism to extend the addressing to additional memory. Later series of devices feature move instructions which can cover the whole addressable space, independent of the selected bank. In earlier devices (i.e., the baseline and mid-range cores), any register move had to be achieved via the accumulator.[4]

To implement indirect addressing, a "file select register" (FSR) and "indirect register" (INDF) are used: A register number is written to the FSR, after which reads from or writes to INDF will actually be to or from the register pointed to by FSR. Later devices extended this concept with post- and pre- increment/decrement for greater efficiency in accessing sequentially stored data. This also allows FSR to be treated almost like a stack pointer.

External data memory is not directly addressable except in some high pin count PIC18 devices. The PIC architectures have several limitations:

- Only a single accumulator
- A small instruction set
- Operations and registers are not orthogonal; some instructions can address RAM and/or immediate constants, while others can only use the accumulator
- Memory must be directly referenced in arithmetic and logic operations, although indirect addressing is available via 2 additional registers

- Register-bank switching is required to access the entire RAM of many devices, making position independent code complex and inefficient
- Conditional skip instructions are used instead of conditional branch instructions

2.4.2 Stepper Motor

A stepper motor (or step motor) is a brushless, synchronous electric motor that can divide a full rotation into a large number of steps. The motor's position can be controlled precisely, without any feedback mechanism (see open loop control). Stepper motors are similar to switched reluctance motors (which are very large stepping motors with a reduced pole count, and generally are closed-loop commutated.)[5]

Stepper motors operate differently from normal DC motors, which rotate when voltage is applied to their terminals. Stepper motors, on the other hand, effectively have multiple "toothed"[6] electromagnets arranged around a central gear-shaped piece of iron.

The electromagnets are energized by an external control circuit, such as a microcontroller. To make the motor shaft turn, first one electromagnet is given power, which makes the gear's teeth magnetically attracted to the electromagnet's teeth. When the gear's teeth are thus aligned to the first electromagnet, they are slightly offset from the next electromagnet. When the next electromagnet is turned on and the first is turned off, the gear rotates slightly to align with the next one, and from there the process is repeated. Each of those slight rotations is called a "step," with an integral number of steps making a full rotation. In that way, the motor can be turned by a precise angle.

Stepper motors are constant-power devices (power = angular velocity x torque). As motor speed increases, torque decreases. The torque curve may be extended by using current limiting drivers and increasing the driving voltage.

Steppers exhibit more vibration than other motor types, as the discrete step tends to snap the rotor from one position to another. This vibration can become very bad at some speeds and can cause the motor to lose torque. The effect can be mitigated by accelerating quickly through the problem speed range, physically damping the system, or using a micro-stepping driver. Motors with a greater number of phases also exhibit smoother operation than those with fewer phases. There are three main types of stepper motors:[5]

- Permanent Magnet Stepper
- Hybrid Synchronous Stepper
- Variable Reluctance Stepper

2.4.3 Sensor

A sensor is a device that measures a physical quantity and converts it into a signal which can be read by an observer or by an instrument.[7] For example, a mercury thermometer converts the measured temperature into expansion and contraction of a liquid which can be read on a calibrated glass tube. A thermocouple converts temperature to an output voltage which can be read by a voltmeter. For accuracy, all sensors need to be calibrated against known standards.[6]

Sensors are used in everyday objects such as touch-sensitive elevator buttons and lamps which dim or brighten by touching the base. There are also innumerable applications for sensors of which most people are never aware. Applications include cars, machines, aerospace, medicine, manufacturing and robotics.

A sensor's sensitivity indicates how much the sensor's output changes when the measured quantity changes. For instance, if the mercury in a thermometer moves 1 cm when the temperature changes by 1 °C, the sensitivity is 1 cm/°C. Sensors that measure very small changes must have very high sensitivities. Sensors also have an impact on what they measure; for instance, a room temperature thermometer inserted into a hot cup of liquid cools the liquid while the liquid heats the thermometer.

Sensors need to be designed to have a small effect on what is measured, making the sensor smaller often improves this and may introduce other advantages. Technological progress allows more and more sensors to be manufactured on a microscopic scale as microsensors using MEMS technology. In most cases, a microsensor reaches a significantly higher speed and sensitivity compared with macroscopic approaches. [7]

- Light time-of-flight. Used in modern surveying equipment, a short pulse of light is emitted and returned by a retroreflector. The return time of the pulse is proportional to the distance and is related to atmospheric density in a predictable way.[7]
- Light sensors, or photodetectors, including semiconductor devices such as photocells, photodiodes, phototransistors, CCDs, and Image sensors; vacuum tube devices like photo-electric tubes, photomultiplier tubes; and mechanical instruments such as the Nichols radiometer.
- Infra-red sensor, especially used as occupancy sensor for lighting and environmental controls.
- Proximity sensor- A type of distance sensor but less sophisticated. Only detects a specific proximity. May be optical - combination of a photocell and LED or laser. Applications in cell phones, paper detector in photocopiers, auto power standby/shutdown mode in notebooks and other devices. May employ a magnet and a Hall effect device.
- Scanning laser- A narrow beam of laser light is scanned over the scene by a mirror. A photocell sensor located at an offset responds when the beam is reflected from an object to the sensor, whence the distance is calculated by triangulation.
- Focus. A large aperture lens may be focused by a servo system. The distance to an in-focus scene element may be determined by the lens setting.
- Binocular. Two images gathered on a known baseline are brought into coincidence by a system of mirrors and prisms. The adjustment is used to determine distance. Used in some cameras (called range-finder cameras) and on a larger scale in early battleship range-finders[8]
- Interferometry. Interference fringes between transmitted and reflected lightwaves produced by a coherent source such as a laser are counted and the distance is calculated. Capable of extremely high precision.
- Scintillometers measure atmospheric optical disturbances.
- Fiber optic sensors.

- Short path optical interception - detection device consists of a light-emitting diode illuminating a phototransistor, with the end position of a mechanical device detected by a moving flag intercepting the optical path, useful for determining an initial position for mechanisms driven by stepper motors.

2.4.4 Programming

C is a general-purpose computer programming language developed in 1972 by Dennis Ritchie at the Bell Telephone Laboratories to implement the Unit operating system.[9] Although C was designed for writing architecturally independent system software, it is also widely used for developing application software.

Worldwide, C is the first or second most popular language in terms of number of developer positions or publicly available code. It is widely used on many different software platforms, and there are few computer architectures for which a C compiler does not exist. C has greatly influenced many other popular programming languages, most notably C++, which originally began as an extension to C, and Java and C# which borrow C lexical conventions and operators.[9]

C is an imperative (procedural) systems implementation language. It was designed to be compiled using a relatively straightforward compiler, to provide low-level access to memory, to provide language constructs that map efficiently to machine instructions, and to require minimal run-time support. C was therefore useful for many applications that had formerly been coded in assembly language.

Despite its low-level capabilities, the language was designed to encourage machine-independent programming. A standards-compliant and portably written C program can be compiled for a very wide variety of computer platforms and operating systems with little or no change to its source code, while approaching highest performance. The language has become available on a very wide range of platforms, from embedded microcontrollers to supercomputers.

C is designed to provide high-level abstracts for all the native features of a general-purpose CPU, while at the same time allowing modularization, structure, and code re-use. Features specific to a particular program's function (features that are not general to all platforms) are not included in the language or library definitions.

However any such specific functions are implementable and accessible as external reusable libraries, in order to encourage module dissemination and re-use. C is somewhat strongly typed (emitting warnings or errors) but allows programmers to override types in the interests of flexibility, simplicity or performance; while being natural and well-defined in its interpretation of type overrides.

C's design is tied to its intended use as a portable systems implementation language. Consequently, it does not require run-time checks for conditions that would never occur in correct programs, it provides simple, direct access to any addressable object (for example, memory-mapped device control registers), and its source-code expressions can be translated in a straightforward manner to primitive machine operations in the executable code. Some early C compilers were comfortably implemented (as a few distinct passes communicating via intermediate files) on PDP-11[10] processors having only 16 address bits; however, C99 assumes a 512 KB minimum compilation platform.

Like most imperative languages in the ALGOL tradition, C has facilities for structured programming and allows lexical variable scope and recursion, while a static type system prevents many unintended operations. In C, all executable code is contained within functions. Function parameters are always passed by value. Pass-by-reference is achieved in C by explicitly passing pointer values. Heterogeneous aggregate data types (struct) allow related data elements to be combined and manipulated as a unit. C program source text is free-format, using the semicolon as a statement terminator (not a delimiter).

- C also exhibits the following more specific characteristics:[10]
- non-nestable function definitions
- variables may be hidden in nested blocks

- partially weak typing; for instance, characters can be used as integers
- low-level access to computer memory by converting machine addresses to typed pointers
- function and data pointers supporting ad hoc run-time polymorphism
- array indexing as a secondary notion, defined in terms of pointer arithmetic
- a preprocessor for macro definition, source code file inclusion, and conditional compilation
- complex functionality such as I/O, string manipulation, and mathematical functions consistently delegated to library routines [11]
- A relatively small set of reserved keywords (originally 42, now 47 in C99)
- A lexical structure that resembles B more than ALGOL, for example
  - { ... } rather than ALGOL's begin ... end
  - the equal-sign is for assignment (copying), much like Fortran
  - two consecutive equal-signs are to test for equality (compare to .EQ. in Fortran or the equal-sign in BASIC)
  - && and || in place of ALGOL's and and or (these are semantically distinct from the bit-wise operators & and | because they will never evaluate the right operand if the result can be determined from the left alone (short-circuit evaluation)).
  - a large number of compound operators, such as +=, ++, ...

2.5 Project information

Project is more emphasizing the control system of AGV. Types of decision (move, halt and turn 90^{o}) made by AGV when confront several situations is decided. This project is required the AGV to move on line with coordinates system.

Then the AGV to move on lines (tape) to the desired coordinates. To realize all above requirements, some hardware and even software is needed. Hardware are included microcontroller (PIC), five of detectors, dc supplier 9 volts and other electronics part. The purpose of microcontroller can be said as central processing unit

(CPU). It processes the entire code scheme (.hex file) those already downloaded. Then, why must have five detectors? First of three detector act as main detector which placed at the centre of bottom of AGV while the rest as subdetector. They help to count row and column.

2.6 Summary of literature

The summarization for overall control system of AGV is here. The control system of AGV is line following type. It is consists of 3 main elements which are input data, processing centre and output data.

For input data, specific sensor will be used such as infrared sensor. This sensor will detect the line as input data and send it through comparator to processing centre. In processing centre, the input data will be analyzed. So, PIC microcontroller will act as processing centre and send output to servo motor. Servo motor purposed to accept the output data from microcontroller and give result; whether the AGV is able to follow the line or not.

There are several reasons for choosing this type of control system. Among of them are very suitable to be implemented in industry. It can be produced for huge quantity since the lower cost and next replace the human energy.

CHAPTER 3

METHODOLOGY

3.1 INTRODUCTION

This chapter is about all methodology that has been used. Methodology is already planned from the beginning of this project. There are several things that must be known before planned the methodology. The area of study for this project must be known, and then the objectives are decided. Beside that also have criteria of choosing the components. Through this chapter hopefully the readers can give an impression about the AGV of this project.

3.2 Flow chart of methodology

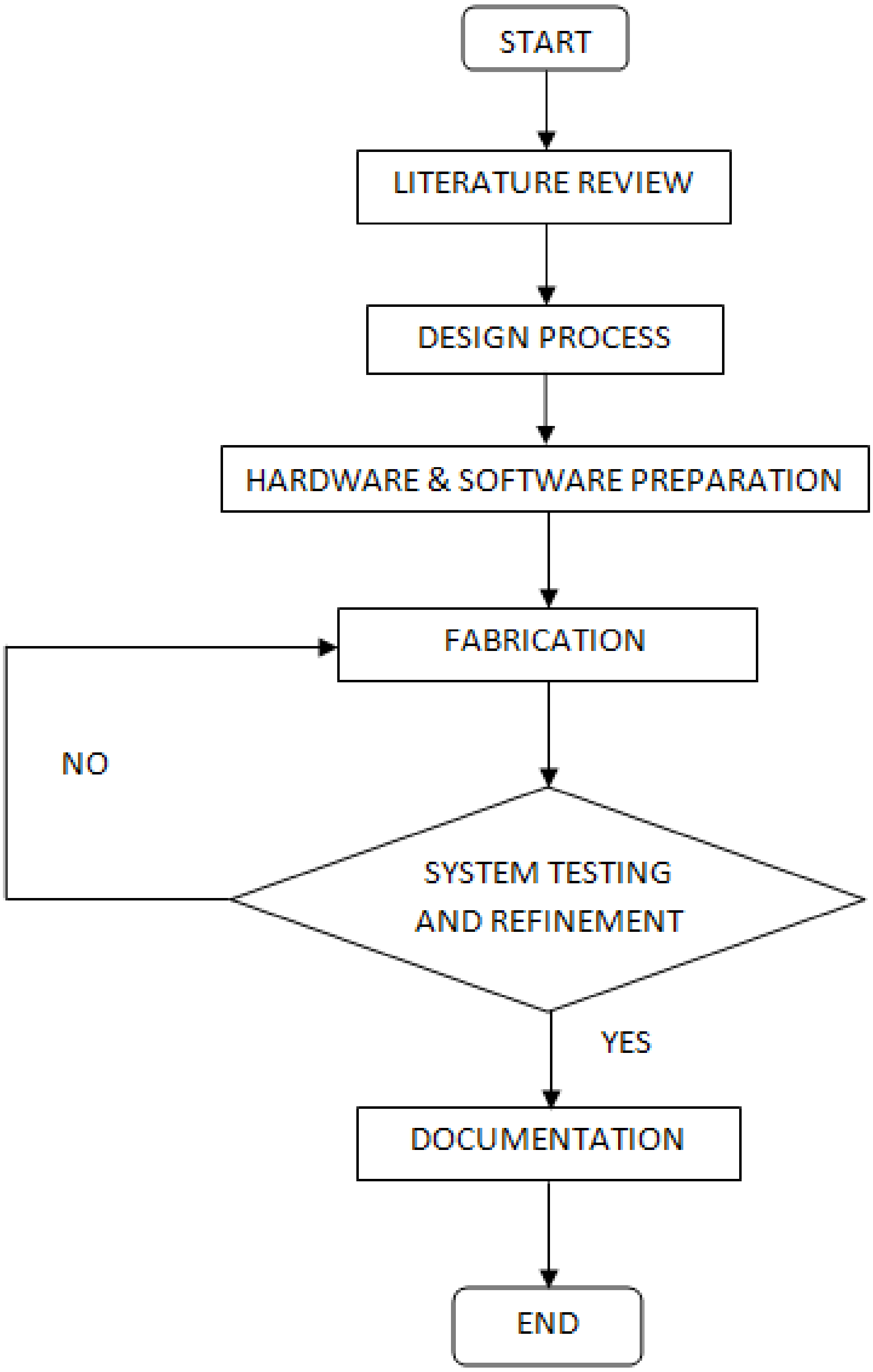


Figure 3.1: Methodology expressed in flow chart

3.3 Literature review

Literature review is to gather journals, information, articles and other sources about AGV as much as can. Then, gain knowledge from it. The theoretical framework of AGV or subject area is established. Some key terms, definitions and terminology are defined. Besides that, studies, models, case studies of project are identified. The three key points of a literature review of AGV:

- what the research says (theory).
- how the research was carried out (methodology).
- what is missing, example the gap that research intends to fill.

3.4 Design process

This process purposes to give rough sketched view of AGV which means that what tasks of AGV and how it complete the tasks. For that, must refer back to the objectives and scopes of the project. The objective is all about the development of control system of AGV. Besides that, this process can be done by referring to some key characteristics below. Key characteristics:

a. Structure and power system

   Structure of an AGV often based on its loading, motion characteristics. The structure can be very different for and AGV used in FMS (flexible manufacturing system) and in an office as the former must be stiff with high power in order to carry parts used in a production line while the latter is often light with silent power system as it is often used to transfer tools among tables.

b. Path planning and guidance system

   This characteristic depends on the function of the AGV. AGV employed in a factory needs simple path and robust guidance. Several sensing and guidance methods were surveyed in order to understand the state of the art of sensoring and positioning technologies.

c. Sensing and control

This item is close related to the guidance system as the accuracy of movement and positioning are determined mainly by the resolution and sensitivity or sensors and controller. Several different sensoring technologies and corresponding control schemes are reviewed for designing the AGV.

3.5 Hardware and software preparation

After design process, gather all the relevant hardware and software is one of the most important steps before proceed to the fabrication of AGV. Some hardware are hand-self made. Some software are available free and ready to be downloaded.

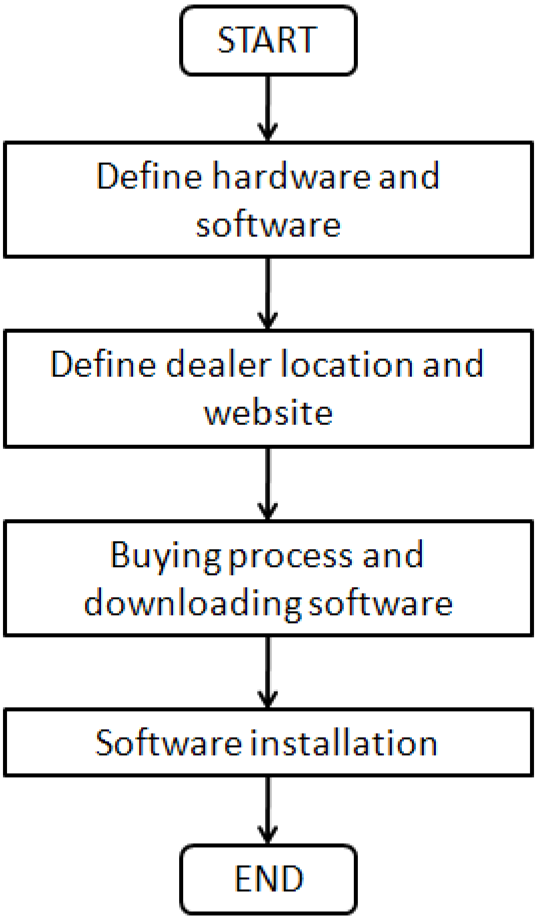

START
Define hardware and software
Define dealer location and website
Buying process and downloading software
Software installation
END

Figure 3.2: Flow chart of hardware and software preparation

From the flow chart in figure 3.2, the hardware and software process begins from define the component and software being used. The entire components are being listed according units for each. Software are also defined.

After define the hardware and software, location of dealer is being located. This is important for buying process. Besides that, the website which provides free trial of software that desired is already targeted. Then, software can be downloaded.

After prepare all the software and hardware, software installation and setting the pc is being operated. Setting the pc is regarding the RS323 cable which is important for downloading the hex file into microcontroller.

3.6 Fabrication

After gather all the component, it is time to fabricate the AGV. This process requires wise plans and patient especially during the soldering. It is very tedious and easy to make silly mistakes. Fabrication process is including soldering, assemble, programming the microcontroller and teaching the AGV. The program will try to follow the flow chart in figure 3.3.

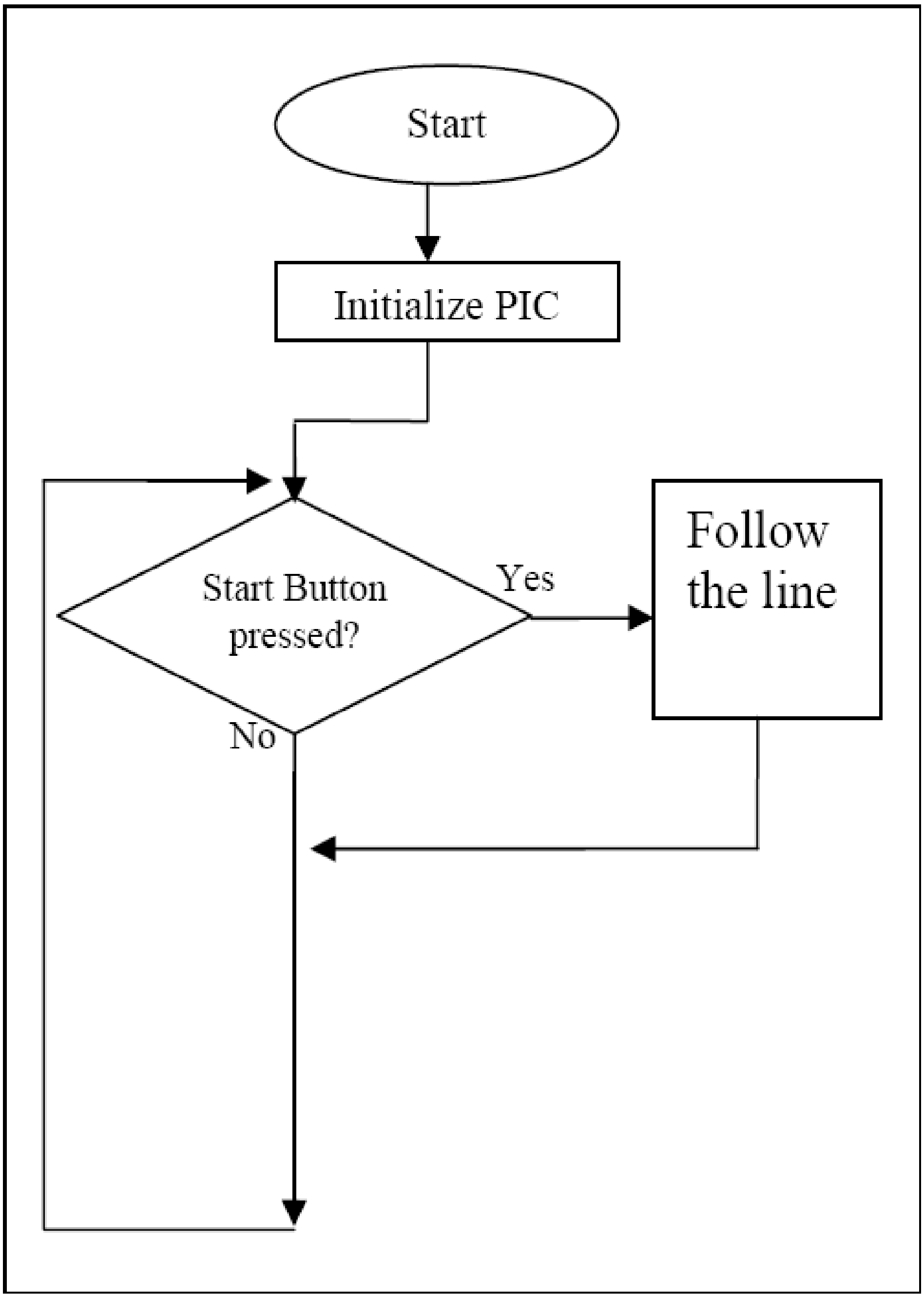


Figure 3.3: Flow chart of programming

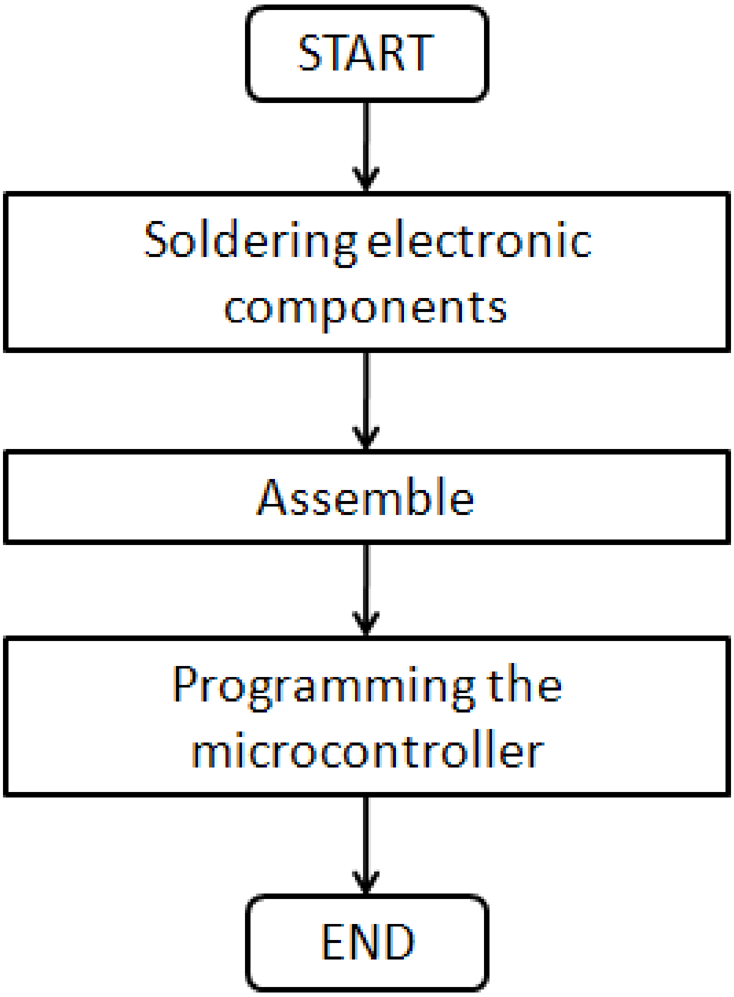

START
Soldering electronic components
Assemble
Programming the microcontroller
END

Figure 3.4: Flow chart of fabrication

Flow chart of fabrication is about what process fabrication was taken. The first one is soldering the electronics circuit. This is required additional tool like soldering tool. Then, assembling the mechanical system and control system is following up. In this process, some modifications are added into mechanical system to make it suitable. After that, program the microcontroller. This is required the software which is already downloaded. Compile the file into hex file and download it into microcontroller using RS323 cable.

3.7 System testing and refining

This process purpose to check all the system includes mechanical system and control system. Some tools such as digital multimeter are being used to check for short circuit. Besides that, it can be used for voltage checking which is important in tuning process. Tuning process is finalizing all the process before the AGV is being used or work. Then, all the results are being recorded and the following discussions emerge to give explanations.

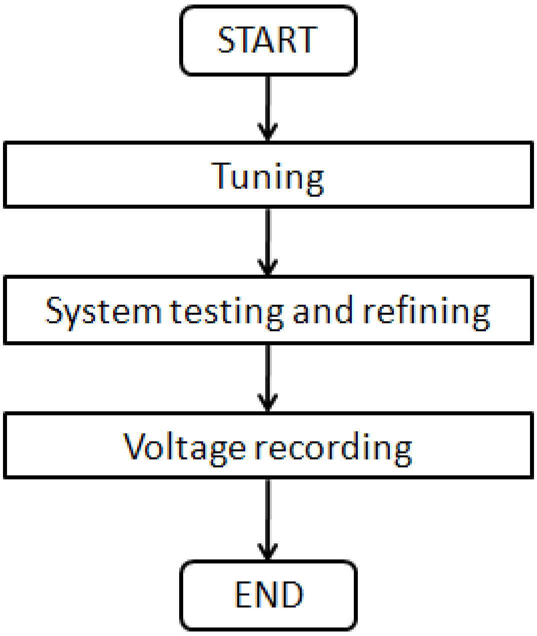


Figure 3.5: Flow chart system testing and refining

CHAPTER 4

RESULT AND DISCUSSION

4.1 INTRODCUTION

This chapter is containing the results and discussions of previous chapter. Results are needed since it determines the successful of the project. Then, purposes of discussions are to give the explanations of results.

4.2 Design the AGV

4.2.1 Control System of AGV

AGV is line following type. It will follow the white line made from tape. Then, the AGV is equipped with several of sensors as line detector. The sensors will detect the line send as data input to microcontroller. Then, microcontroller will processes all the data base on programmed code which inside it. After that, the microcontroller will send out data output through motors. The impacts can be seen either motors rotating or any of it. The explanation can be simplified through the diagram below.

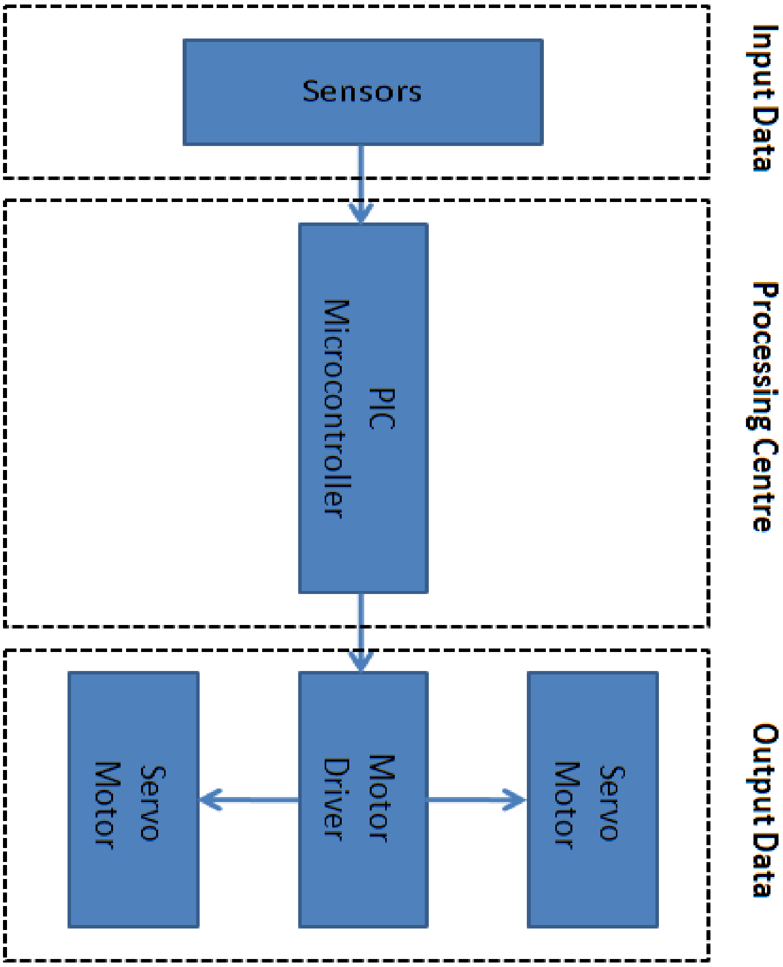


Figure 4.1: Control System of AGV

4.2.2 Sketching and Viewing of Future AGV

The purpose of sketching and viewing here is to give an impression of AGV to readers. Here are included some pictures, basically about the whole body and where the main component is located.

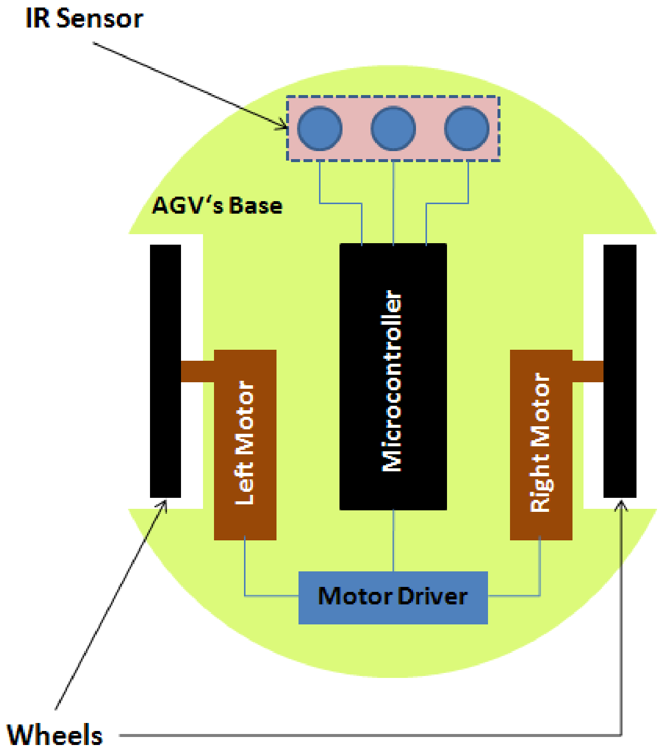


Figure 4.2: Top view of AGV

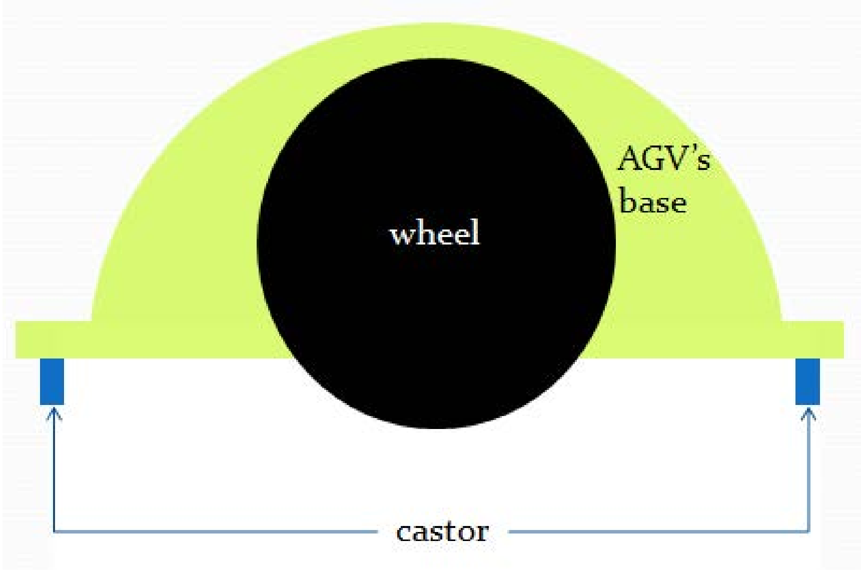


Figure 4.3: Side view of AGV

4.2.3 Features of AGV

AGV is moving base on white line. It equipped with three IR sensors which is arranged in array. Each sensor is build from transmitter and receiver. The transmitter emits the IR ray onto the floor. Then, ir ray being reflected by white line and reflected IR ray detects by receiver.

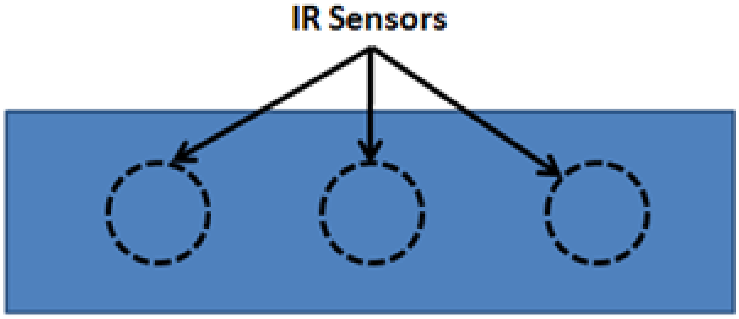


Figure 4.4: Array of IR sensors

When left sensor detects the line, it sends data input to microcontroller. Then, microcontroller processes the data according to the program. It sends data output to the right motor. The result is right motor rotating but left motor halting. Otherwise, if only right sensor detects line the process almost the same. Left motor is rotating but now right motor is halting. But, in case neither left sensor nor right sensor detect line, both motors are rotating.

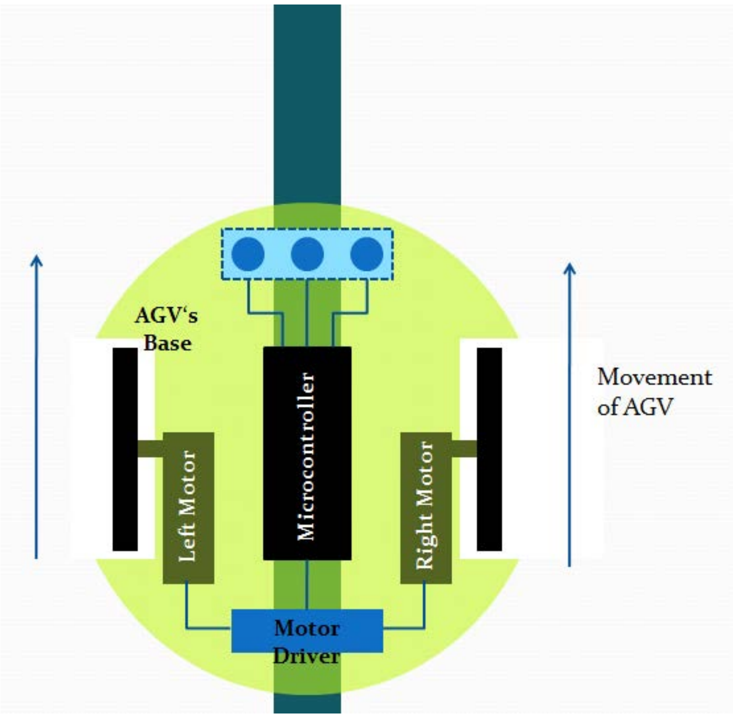


Figure 4.5: Both motors are rotating when left and right motor detect no line at the same time.

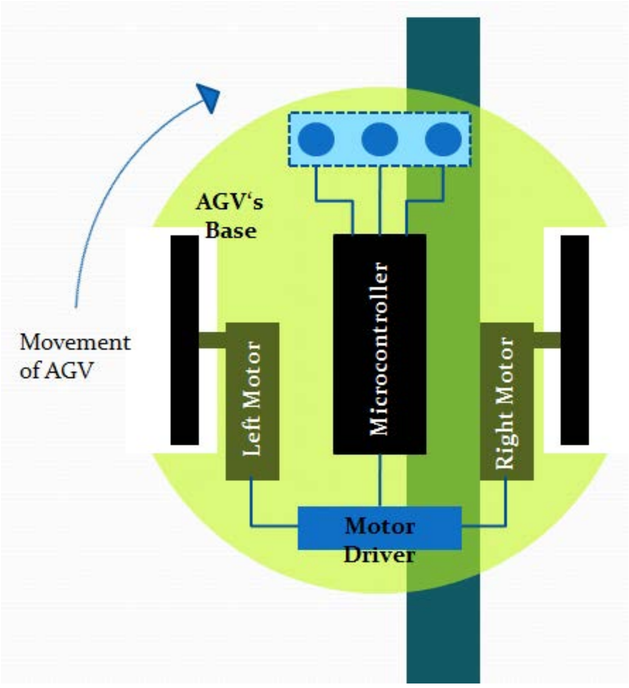


Figure 4.6: Movement of AGV in case right sensor detects the line.

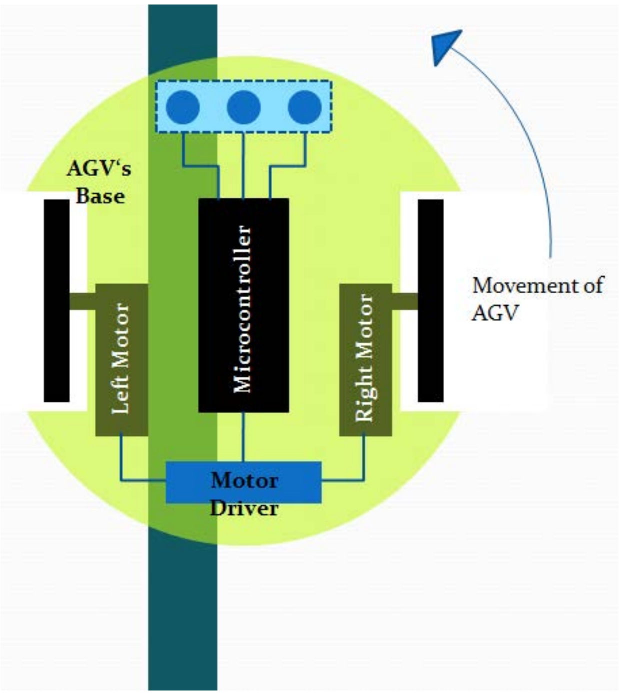


Figure 4.7: Movement of AGV in case left sensor detects the line.

4.3 Hardware and Software Assembling.

4.3.1 EBox-877 Kit and Microcontroller

This powerful (200 nanosecond instruction execution) yet easy-to-program (only 45 single word instructions) CMOS FLASH-based 8-bit microcontroller packs Microchip's powerful PIC® architecture into an 40- or 44-pin package and is upwards compatible with the PIC16C5X, PIC12CXXX and PIC16C7X devices. Features of the device:

- 256 bytes of EEPROM data memory
- Self programming
- Comparators
- 8 channels of 10-bit Analog-to-Digital (A/D) converter
- capture/compare/PWM functions

- synchronous serial port can be configured as either 4 wire Serial Peripheral Interface (SPI™) or the 2-wire Inter-Integrated Circuit (I²C™) bus
- Universal Asynchronous Receiver Transmitter (UART).

All of these features make it ideal for more advanced level A/D applications in automotive, industrial, appliances and consumer applications.

Figure 4.8: EBox-877 Kit and Microcontroller

4.3.2 Servo Motor

This project will be focusing on servo motor. Typical servo motors only allow 180° rotate which is not suitable for a wheeled AGV to move from a place to another place. Thus, some modifications had been made to the servo motor so that it can rotate 460° continuously. To move the motor, just connect the power supply to the terminal on the motor while to move it in opposite direction, change the polarity of the connection between the power supply and the terminal. This AGV needs two servo motors.

Figure 4.9: Servo motor

Source: cytron.com

To be more detailed, type of servo motor is C46S. The specifications are listed in table below:

Table 4.1: Specification of servo motor

Speed (sec/60deg)	0.16/4.8V, 0.14/6.0V
Torque (Kg-cm)	4.5/4.8V, 4.5/6.0V (Maximum 6.0V)
Size (mm)	40.8x20.18x46.5
Pulse width range	0.546ms to 2.4ms (estimation)
Weight (g)	46
Other	Designed for "closed feedback". Able to control the position of the motor

4.3.3 Motor driver L294D

Out of which is the microcontroller. Therefore, an additional component is required to enable the microcontroller to control the motors. For this project, L294D will be used for this purpose. A servo motor usually needs 4.8V or 6.0V to operate. Higher voltage will generate more power to the motor, thus the 6 pins, 4 is for the left motor and the other 4 is for the right motor.

Concentrate on only one side of L294D, 4 pins for the microcontroller and 2 pins for the motor. From the 4 pins, 2 pins are for the direction of the motor and 1 pin (connected to Pin C1 or C2) is for PWM which is to control the speed of the motor. The direction of the motors depends on the connection of the terminal but can also be determined through the program.

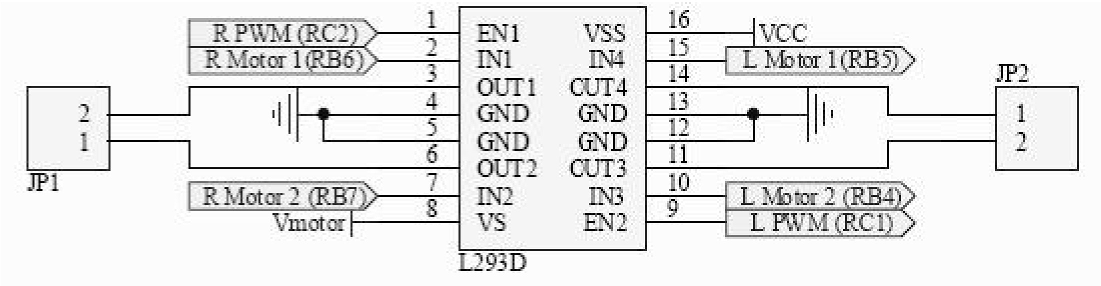


Figure 4.10: L294D

4.3.4 Array of infrared sensors

For this project, three pairs of IR sensors which will be attached to the bottom of the AGV. These 4 sensors will be classified as left sensor, middle sensor and right sensor. The distance between 2 sensors is 2.5cm.

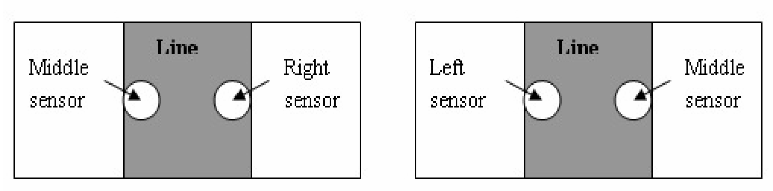


Figure 4.11: The sensor should be placed in such a way that maximum distance of two sensors is equal to the width of the line.

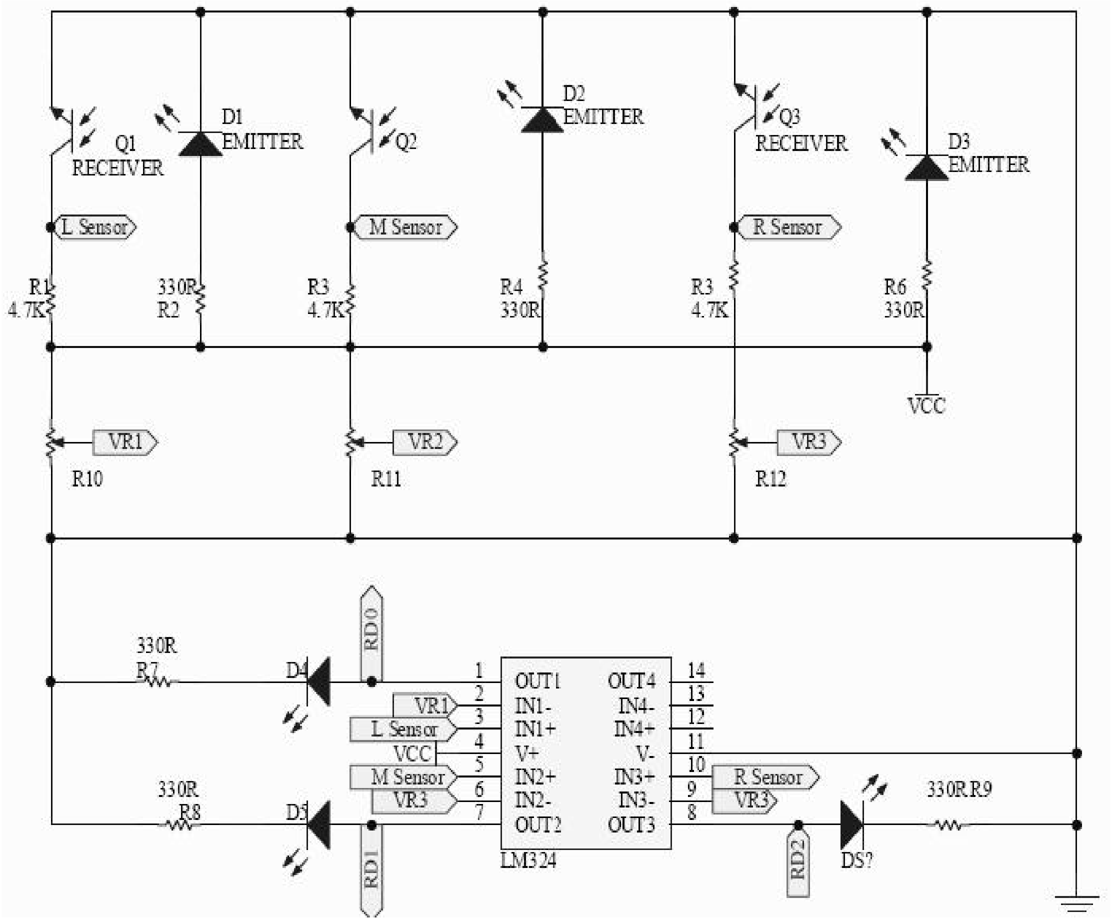


Figure 4.12: Infrared sensor circuit

The value from the IR sensor is compared with the value from the preset which can be adjusted by user. The output is logic 1 if IR sensor value is larger than preset and vice versa. The LED is used to indicate logic 1 or logic 0. The LED will turn ON when logic 1 and turn OFF when logic 0.

4.3.5 Chassis of AGV

These goods are available at Cytron. It is a startup platform for small size mobile AGV. Complete set including base, servo motor C46S (460 degree rotate) x 2, servo wheel x 2, castor x 2.

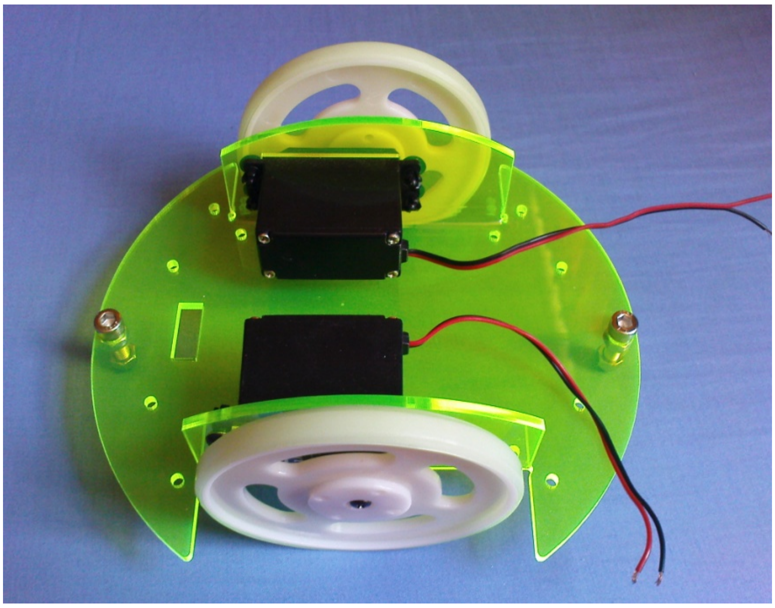


Figure 4.13: Chassis of the AGV

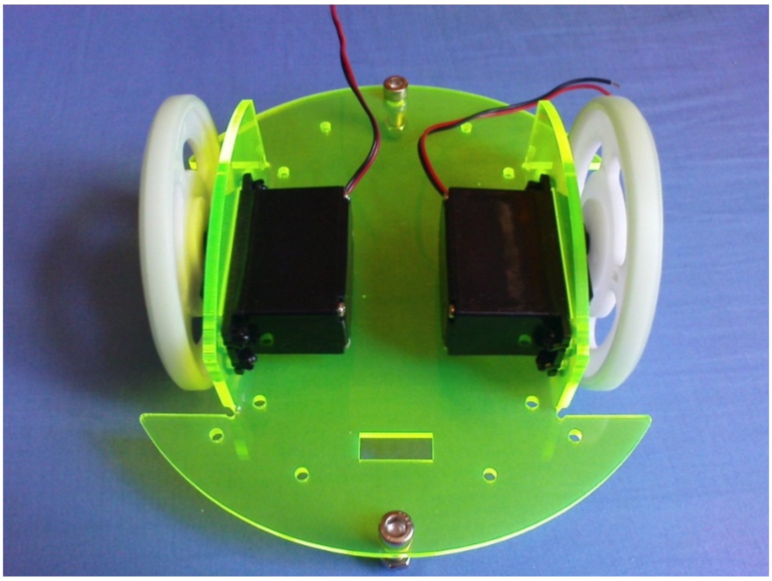
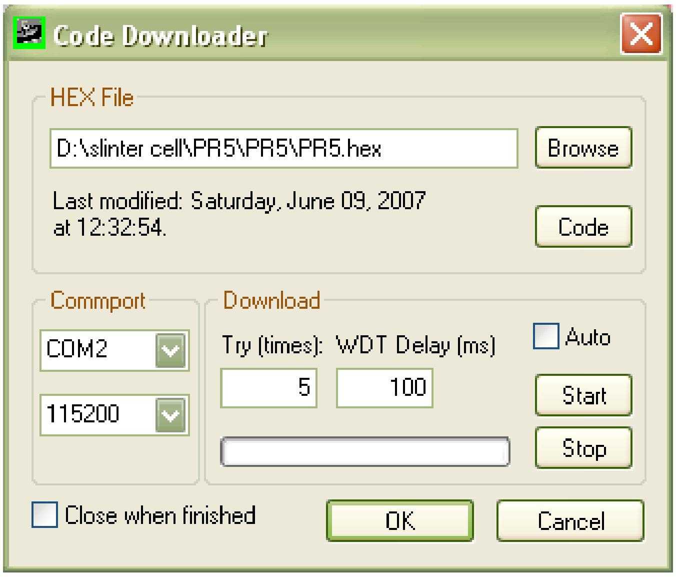


Figure 4.14: Chassis of AGV

4.3.6 Code Downloader

This software is free and available at www.microchipc.com. Besides that, it requires no installation. Just simply run the CD2M and code downloader window will appears.



Code Downloader

HEX File

D:\slinter cell\PR5\PR5\PR5.hex

Last modified: Saturday, June 09, 2007 at 12:32:54.

Commport

COM2

115200

Download

Try (times): 5

WDT Delay (ms): 100

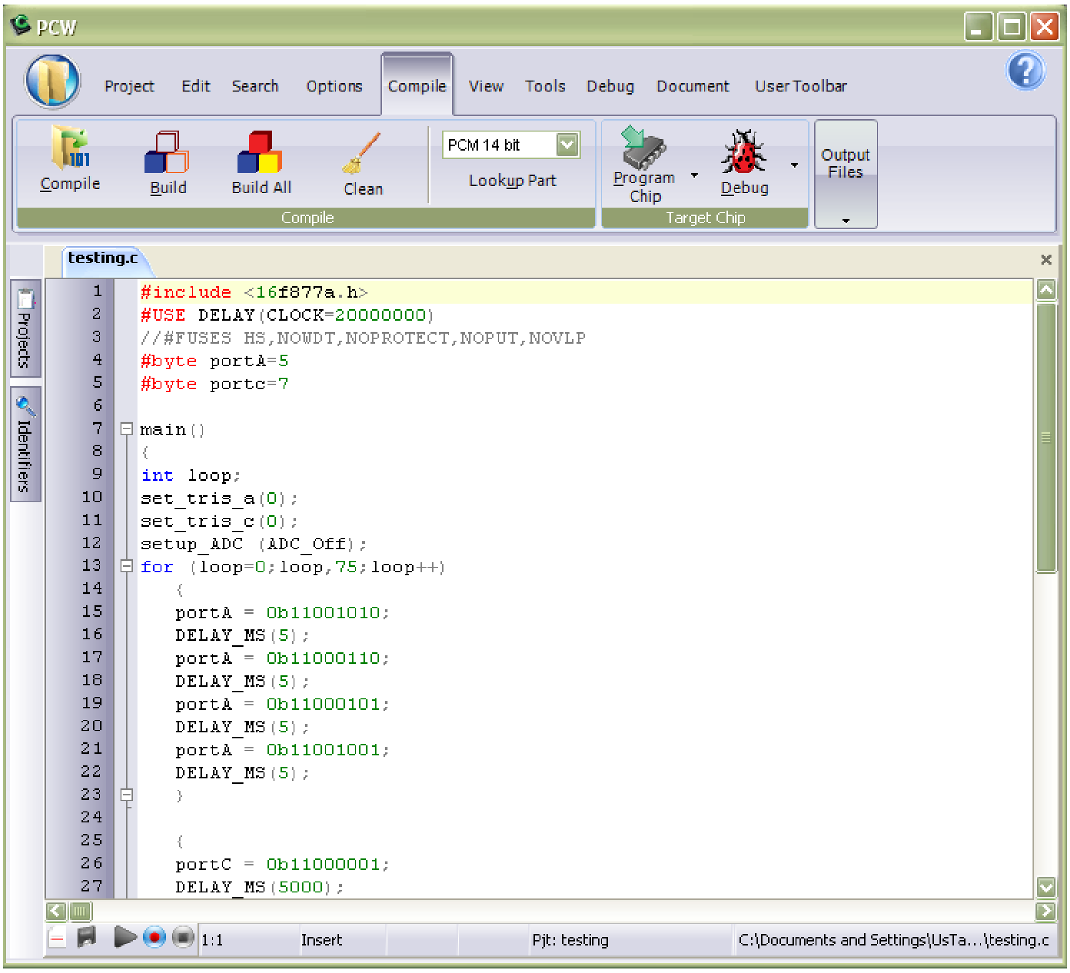
□ Auto

□ Close when finished

Figure 4.15: Code downloader window

4.3.7 PIC C Compiler

This software is to compile the program code in c language into hex file. C language is languages that can be understand by human being but hex file is only microcontroller understand. The PIC C Compiler window is shown below.



```
#include <16f877a.h>
#USE DELAY(CLOCK=20000000)
//#FUSES HS,NOWDT,NOPROTECT,NOPUT,NOVLP
#byte portA=5
#byte portc=7

main()
{
int loop;
set_tris_a(0);
set_tris_c(0);
setup_ADC (ADC_Off);
for (loop=0;loop,75;loop++)
   {
   portA = 0b11001010;
   DELAY_MS(5);
   portA = 0b11000110;
   DELAY_MS(5);
   portA = 0b11000101;
   DELAY_MS(5);
   portA = 0b11001001;
   DELAY_MS(5);
   }

   {
   portC = 0b11000001;
   DELAY_MS(5000);
```

Figure 4.16: PIC C Compiler window

4.3.8 Relevant electronic components

Here is the list of all electronic components that may be required in this project. Mostly all the components are available in the market.

Table 4.2: Electronic components

No	Components	Unit
1	eBox-877 Kit	1
2	Comparator LM424	1
4	Motor Driver L294D	1
4	LED	4
5	Resistor 4.7K	4
6	Resistor 440R	6
7	Trimmer 10K	4
8	PIC16F877A	1
9	Independent Board	1
10	Rainbow Cable	1
11	RC Servo Motor C46S (modified)	2
12	9V battery	2
14	Jumper wire	1
14	2 way header connector	4

4.4 Fabrication

4.4.1 Soldering electronic components

Prepare the independent board and cuts into two portions as shown in figure 4.17. The idea is smaller portion of independent board is for A and greater one is for residue.

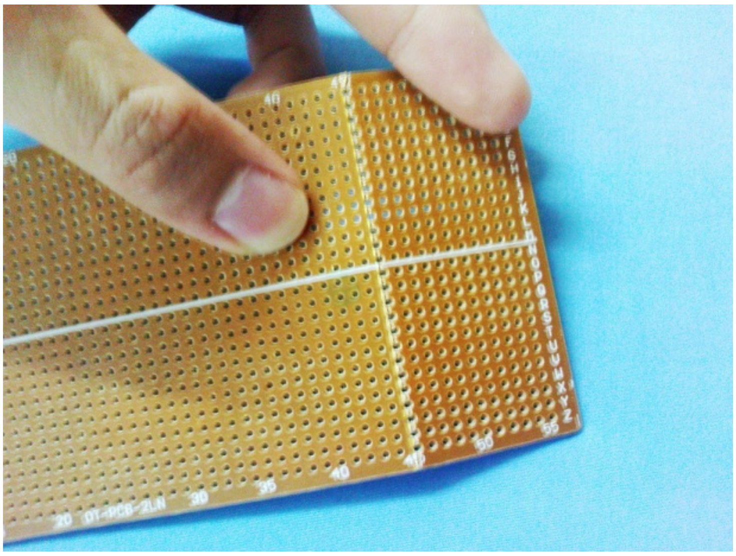


Figure 4.17: Cut the board

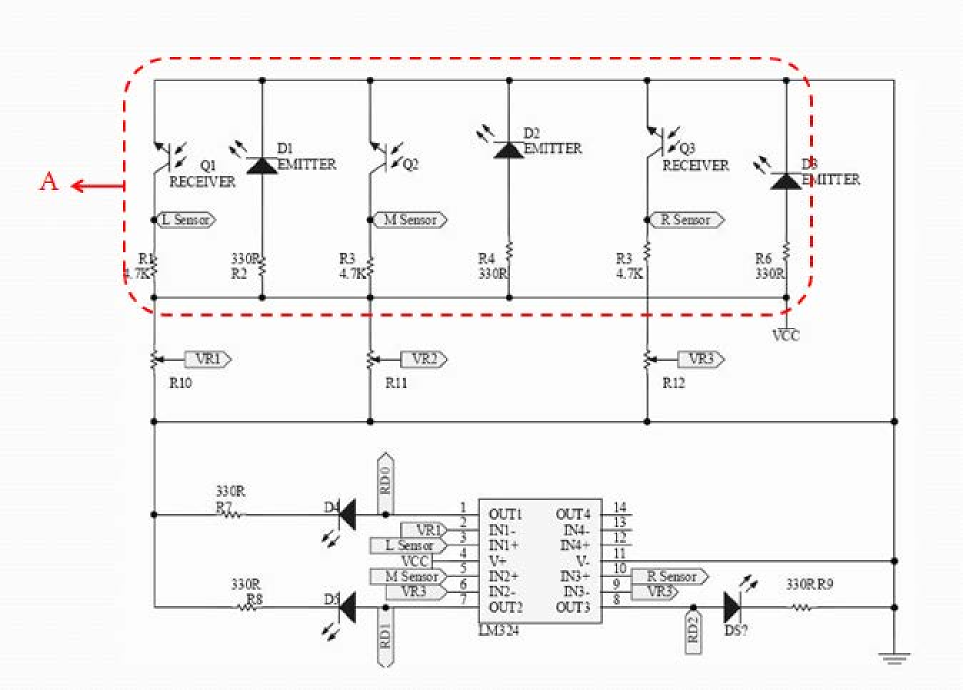


Figure 4.18: Area of A for array of IR sensors

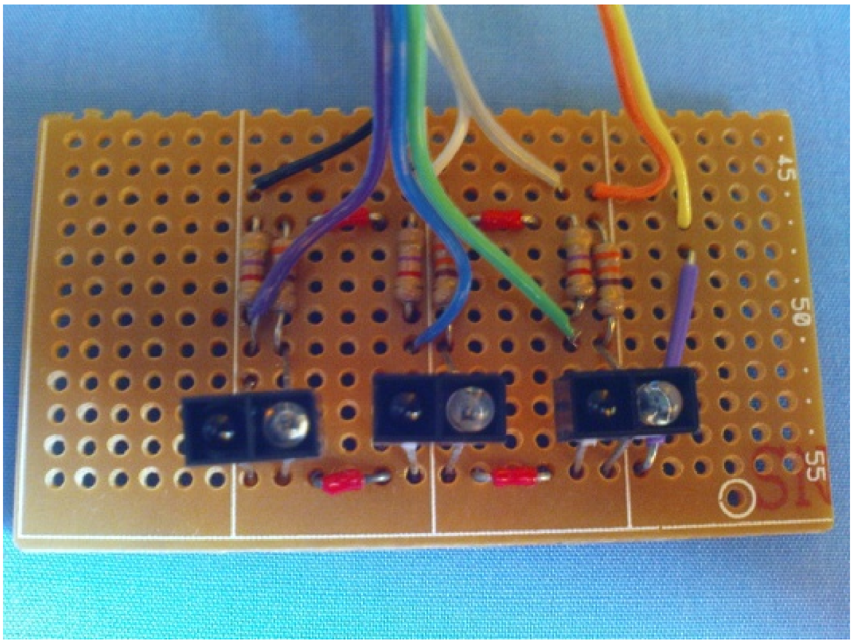


Figure 4.19: Array of sensors

Area of A is for array of sensors which will be attached to the bottom front of AGV. This is shown in the figure 4.18.

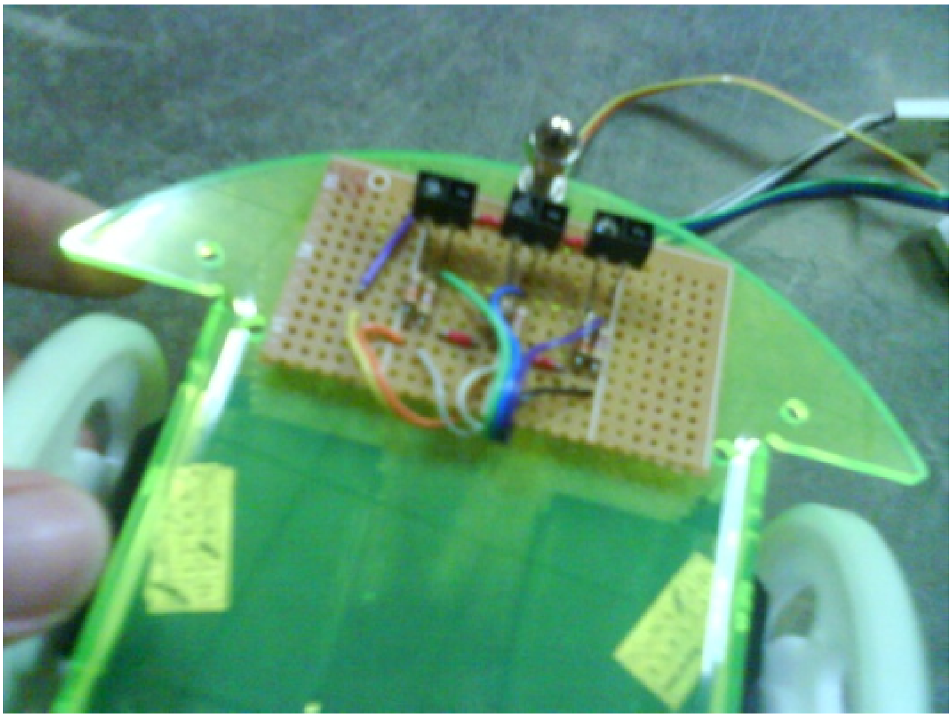


Figure 4.20: Bottom front of AGV

If unconfident, test it out on breadboard. It is advisable to start with the sensor part and test it first before soldering the motor part. This is because the sensors can be tested without need to program the microcontroller. Test the sensors by placing it on a white surface and after that on a black surface.

If the LEDs lights when it is placed on the white surface and turn off when placed on the black surface, then your sensors are connected correctly. Then, proceed with motor part. Make sure that all the polarity of the components is according to the schematic.

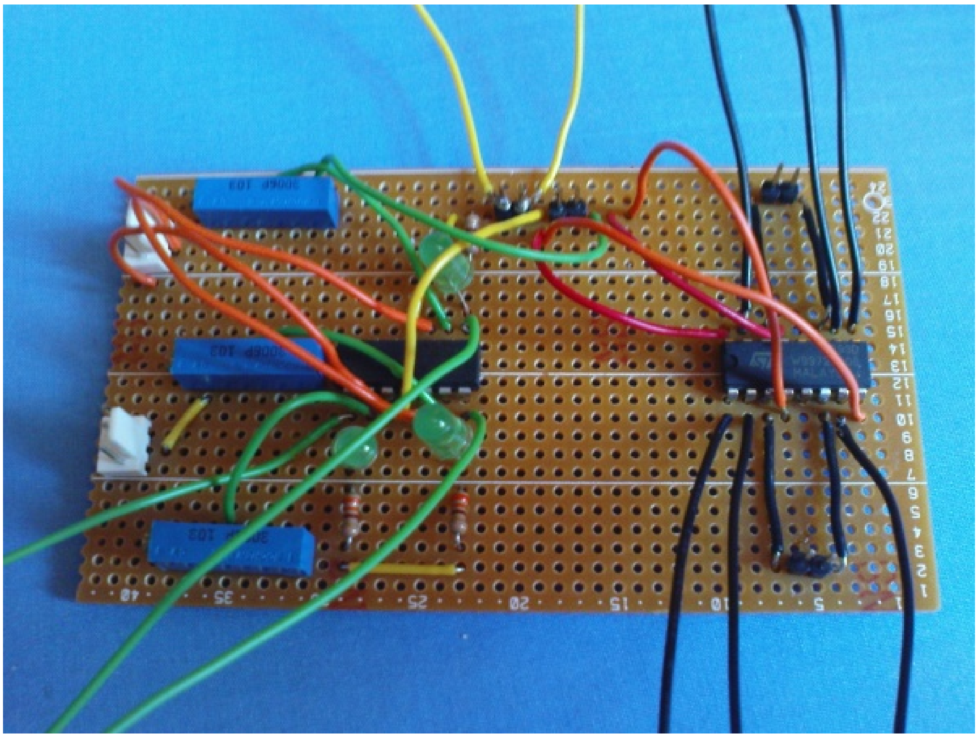


Figure 4.21: Motor and sensor part

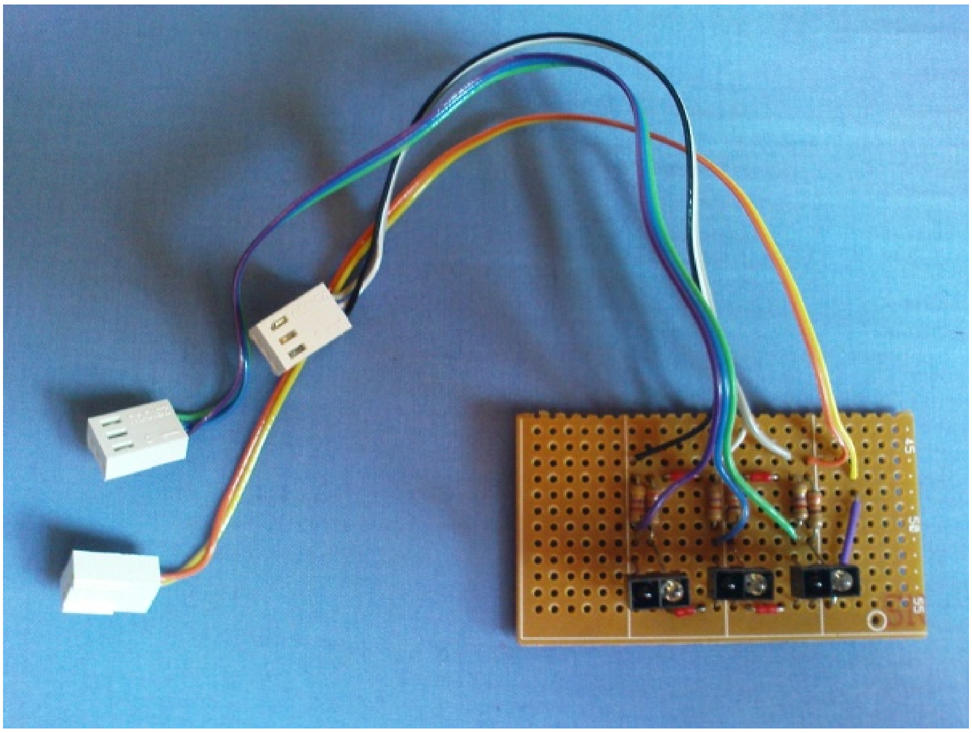


Figure 4.22: The finished circuit

4.4.2 Assemble

This process is integration of control system with mechanical system. Mechanical system included chassis of AGV, wheels, and castors. Castors are important to balance the AGV. Control system is what explained in the earlier.

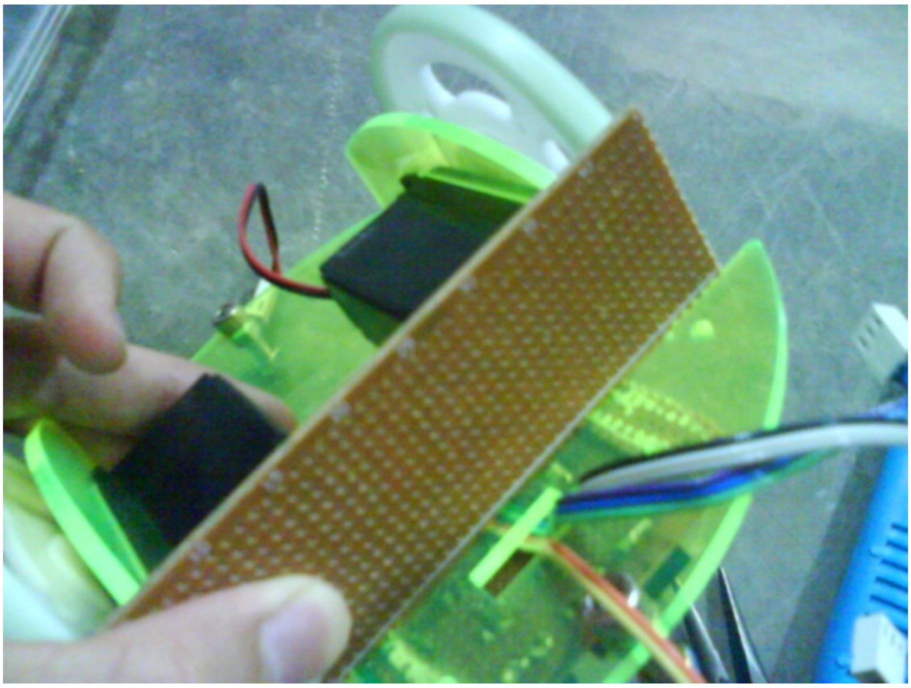


Figure 4.23: Modifying the chassis of AGV

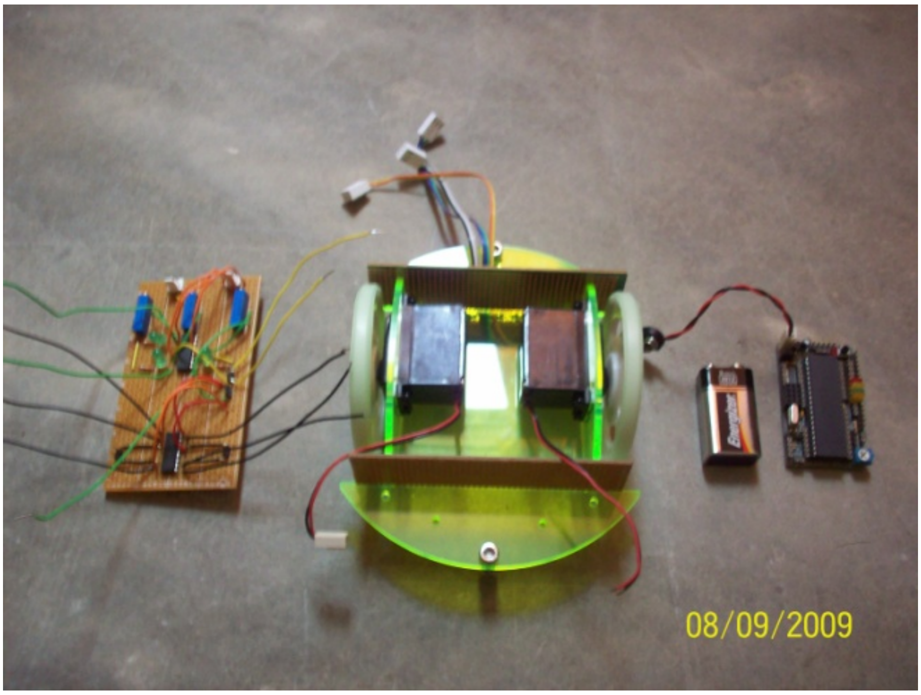


Figure 4.24: Mechanical and control system

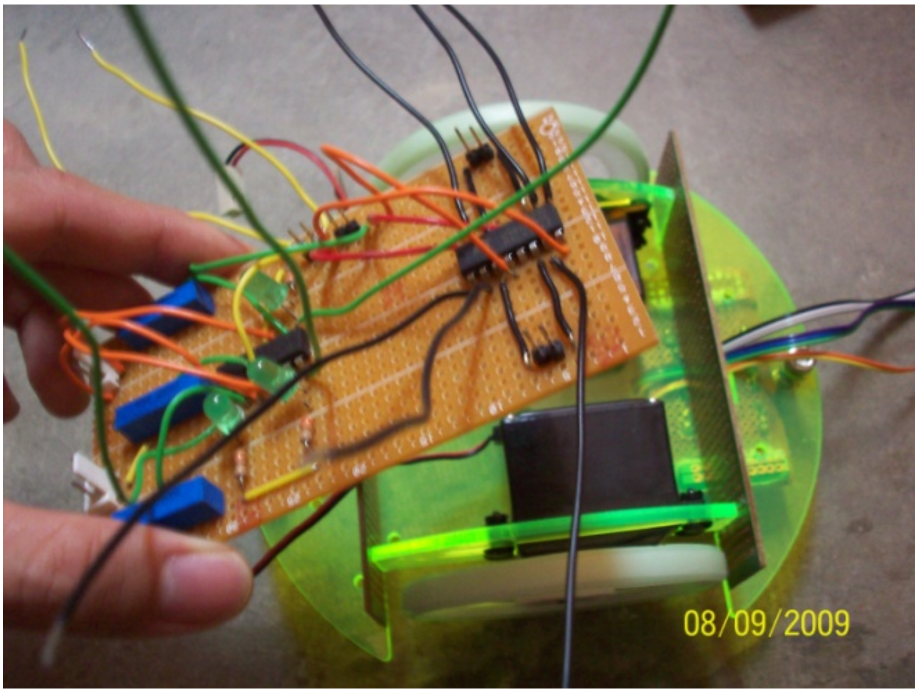


Figure 4.25: Place the circuit on chassis

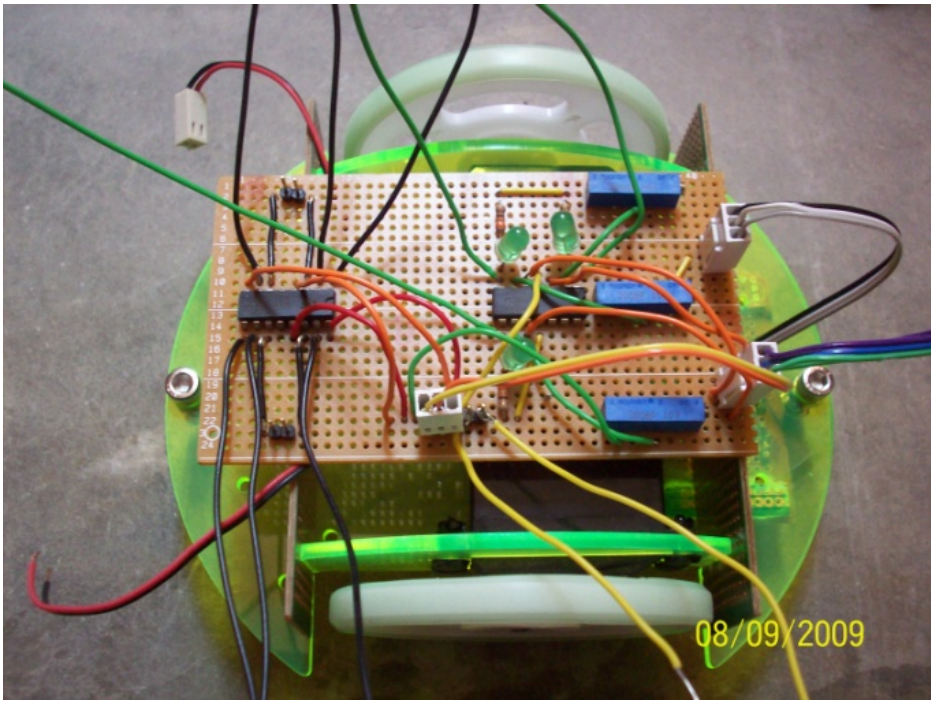
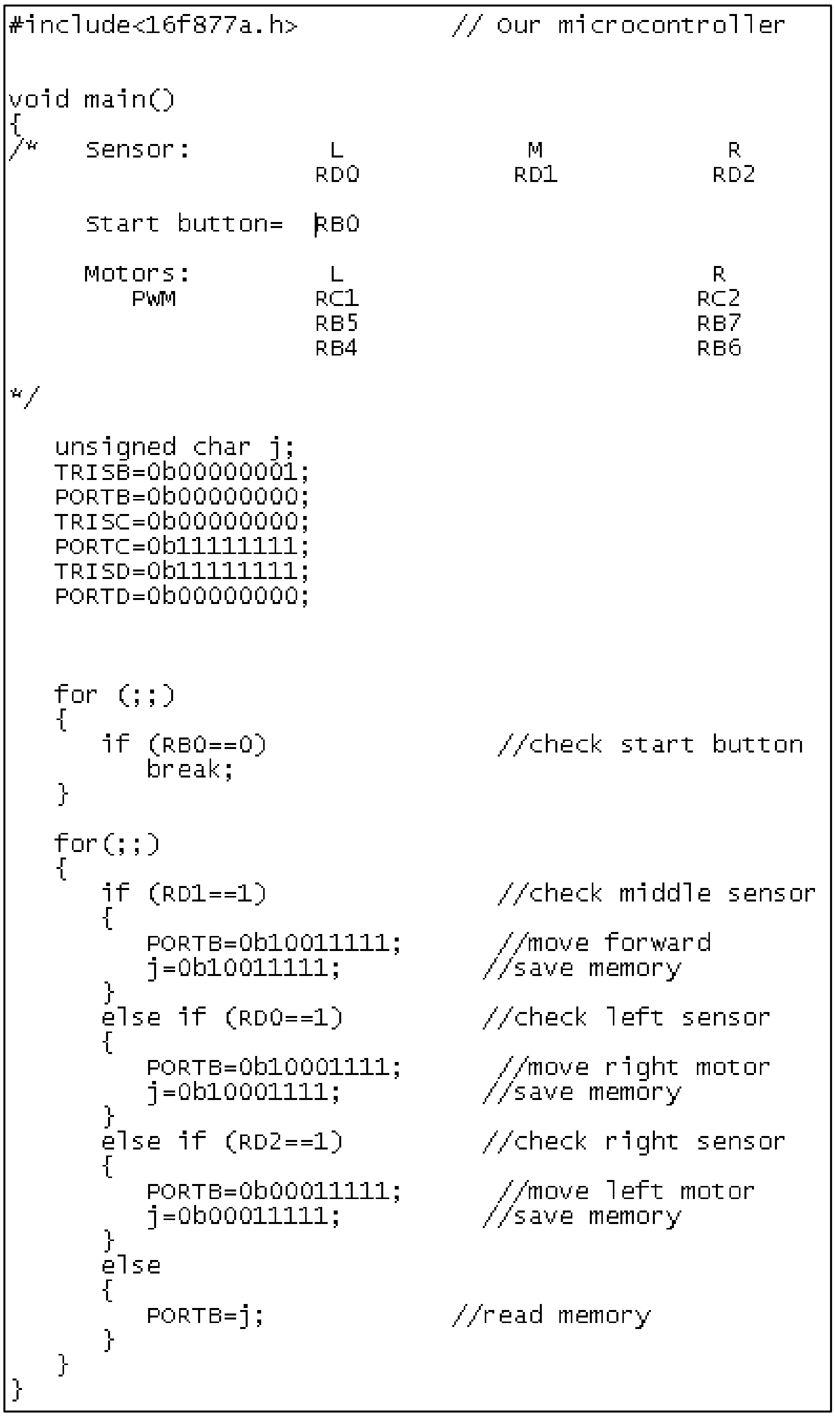


Figure 4.26: After assemble

4.4.3 Programming the microcontroller

After assemble everything, it is time to program the microcontroller. PIC microcontroller has several ports which can be output or input. So, for this project the port D will become input port while port B will become output port. Program code is in C language. Then, the finished program will be compiled into hex file.



```
#include<16f877a.h>          // our microcontroller


void main()
{
/*   Sensor:         L            M            R
                    RD0          RD1          RD2

     Start button=  RB0

     Motors:         L                         R
        PWM         RC1                       RC2
                    RB5                       RB7
                    RB4                       RB6

*/

   unsigned char j;
   TRISB=0b00000001;
   PORTB=0b00000000;
   TRISC=0b00000000;
   PORTC=0b11111111;
   TRISD=0b11111111;
   PORTD=0b00000000;



   for (;;)
   {
      if (RB0==0)                //check start button
         break;
   }

   for(;;)
   {
      if (RD1==1)                //check middle sensor
      {
         PORTB=0b10011111;       //move forward
         j=0b10011111;          //save memory
      }
      else if (RD0==1)          //check left sensor
      {
         PORTB=0b10001111;       //move right motor
         j=0b10001111;          //save memory
      }
      else if (RD2==1)          //check right sensor
      {
         PORTB=0b00011111;       //move left motor
         j=0b00011111;          //save memory
      }
      else
      {
         PORTB=j;              //read memory
      }
   }
}
```

Figure 4.27: Program code in c language

Here are some explanations of program code. In line 1 which is (#include <16f877a.h>), referring to PIC 16f877a microcontroller. Then, start from line 5 until line 15 which is within (/*…..*/) referring to comment. This comment tells that port D (RD0, RD1 and RD2) is for IR sensors and port C and B are for servo motor.

In line 17, data type of ‘j’ is character which range value from 0 to 255. In line 18 until line 23is to declare the port pin output or input. If 0 is output but 1 is otherwise. In line 26 until line 30 which are under ‘for’ loop. It is checking whether the switch is ON or OFF. If ON, the program will move to next ‘for’ loop which in line 32 until line 53. The ‘for’ loop from line 32 until line 53 related to moving style of AGV. If left sensor detects the line which is (RD==1), the right motor will rotates (PORTB=0b1000111) and vice versa. If middle sensor detects line (RD==1), both motor will rotate which is (PORTB=0b10011111). Both ‘for’ loop is rotate until infinity since no condition was put in there (for(; ;)).

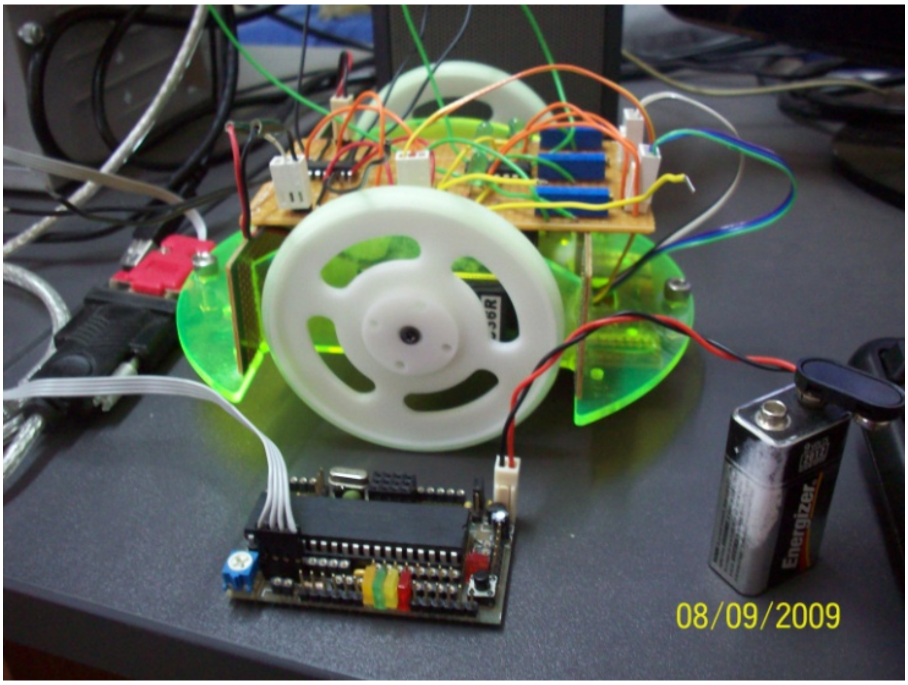
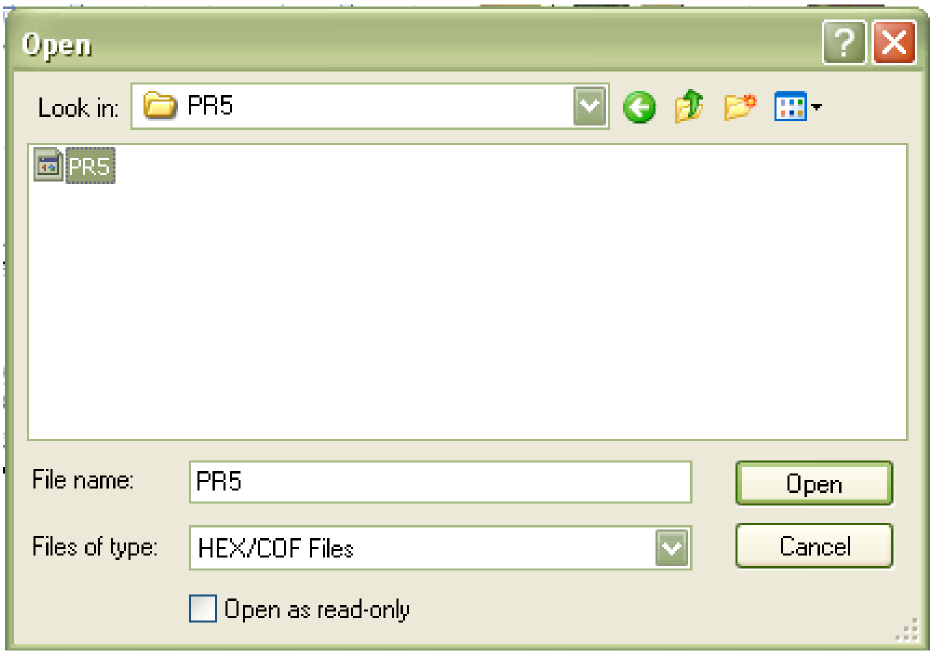


Figure 4.28: Connect the cable



Open

Look in: PR5

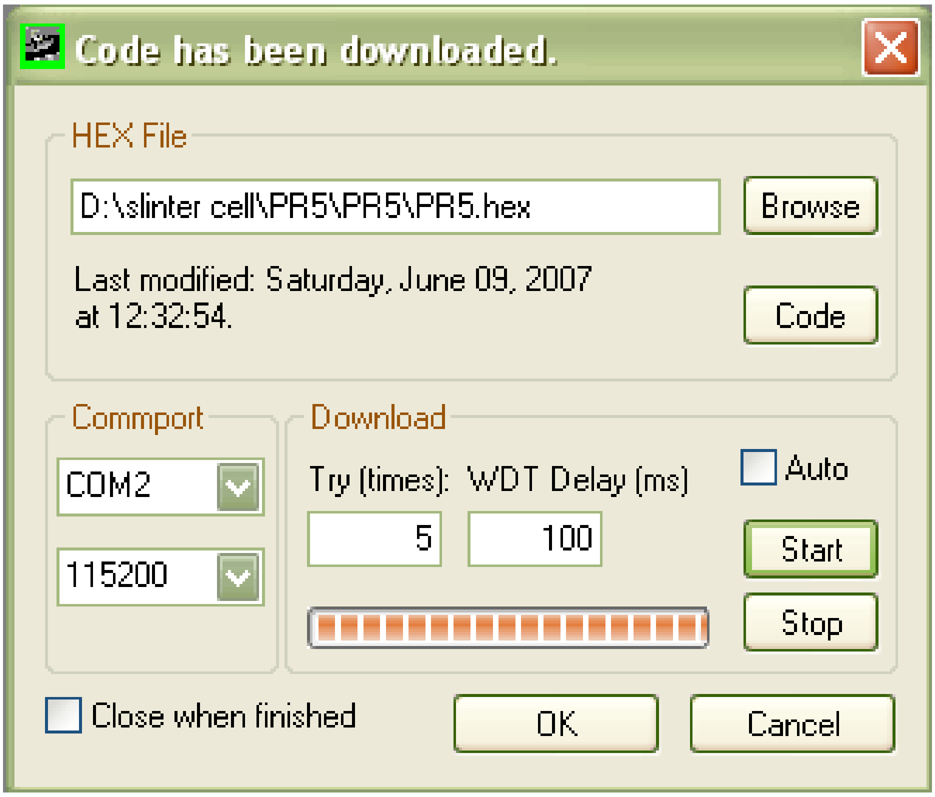
PR5

File name: PR5

Files of type: HEX/COF Files

□ Open as read-only

Figure 4.29: Open the hex file



Code has been downloaded.

HEX File

D:\slinter cell\PR5\PR5\PR5.hex

Last modified: Saturday, June 09, 2007
at 12:32:54.

Commport

COM2

115200

Download

Try (times): 5

WDT Delay (ms): 100

□ Auto

□ Close when finished

Figure 4.30: Download successful

4.5 Tuning

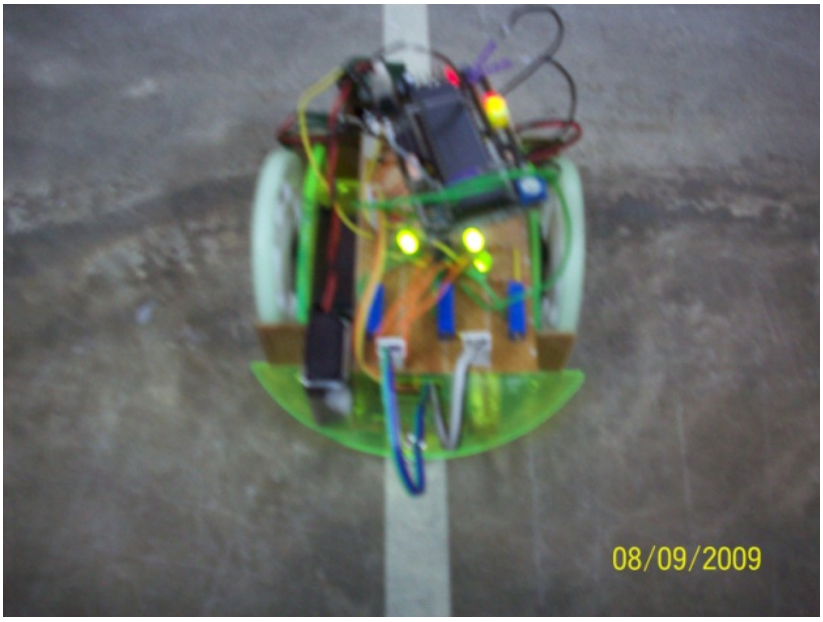


Figure 4.31: AGV on white line

Place the AGV on a white floor with white tape (line) as shown in figure 4.30. To teach the AGV for line detection, you may follow these steps:

1. Adjust the AGV so that the center infrared sensor is on top of white floor. Make sure the wheels and castor of the AGV touches the floor properly.
2. Use the screw driver to adjust the preset of center sensor until indicator LED (center) light ON.
3. Now adjust the AGV to move the center sensor towards the black line where the reflection of infrared is poor.
4. At this point, make sure the indicator LED is OFF. If the LED is still on, it means you have over tune the preset. Tune it back so that the indicator is OFF.
5. Repeat step 1. - 4. for a few times and make sure the indicator LED ON and OFF correctly at the right spot.
6. Repeat step 1. - 5. to "teach" left sensor and right sensor.

Now your AGV has been "taught".

4.6 Relevant voltage

The AGV is following the line according to the plan. The voltage of IR sensors and voltage of preset when detect line or not are taken using digital multimeter:

Table 4.3: Voltage when detect line

	Voltage of Presets (V)	**Voltage of IR sensors (V)**
Right	2.462	4.167
Middle	2.461	4.111
Left	2.462	4.137

Table 4.4: Voltage when detect no line

	Voltage of Presets (V)	**Voltage of IR sensors (V)**
Right	2.462	0.016
Middle	2.461	0.013
Left	2.462	0.012

The voltages of presets are no changes since it is fixed. Voltages of IR sensors are increasing when detect the line and decreasing when detect no line. When voltage of IR sensor greater than preset, the logic is 1 and vice versa. Maximum range detection of IR sensor is about 2cm.But here, the distance of sensor to floor is 1cm.

4.7 Problem and Solution

The problems when execute this project are difficult to find the dealers and circuit is not functioning. For the first problem, there are lots of helps from friends' especially FKEE students. They have suggested what appropriate dealer such as cytron, keyence and farnell.

The second problem is being resolved with the help from Mr. Mohd Fadzil Faisae. Some changing in voltage must be known to solve the problems. Then, reset the resistance of preset to make it running.

CHAPTER 5

CONCLUSION

5.1 Conclusion

For overall conclusion, this project is successful in developing the control system of AGV. The readers may understand what type of control system of the AGV. Control system of the AGV is line following type.

This project is successful to integrate the control system with mechanical system. Control system is more likely to electronic circuit and how to program it while mechanical system is including the chassis of the AGV

Finally, the project is successful to build a guided path of AGV. AGV is following the line which is guide path for the AGV.

5.2 Recommendations

The functions of AGV can be varied. This will lead the AGV to do some more tasks. Put other sensor onto the AGV such as optic sensor. Optic sensor can make AGV to detect the light source. Besides that, to avoid obstacle other sensor can be added. Therefore, it could make the AGV wiser.

The mechanical system of the AGV also can be focused; what the maximum load can be put on AGV and what ideal speed of the AGV. Some of manufacturers or individual have tried to build the faster AGV. Some of them might bother the speed but emphasize the maximum load can be carried.

Finally, AGV should be introduced to the factory or warehouses since it can reduce the human works. It also might reduce the risk of fatality and disable.

5.3 Summary

Automated guided Vehicle (AGV) is a very useful tool when it comes to load distribution applications. After this project, it was clearly that the basics of controlling the PIC microcontroller are important. The AGV to be built was the line follower. It used simple program and circuit, understandable by the students. The AGV could be used to do simple tasks and may be applied to real world applications.

Fabrication of AGV is really challenging since it included circuit. There are so many troubleshooting to make it run. Dysfunction of each component can affects the AGV. Besides that, the polarities must be correctly aligned to make it successfully run.

Other than that, the timeline of the project is quiet ideal. During Final Year Project 1, the knowledge about what going to do must be covered all. Afterward, in Final Year Project 2 follow up the planned plan. Six months are a short periods.

References

[1] Siemens. 2006. The Basics of Automated Guided Vehicles. pp.9-75.

[2] University of Kentucky. 2006. Flexible Manufacturing Systems.

[3] Mobile Platforms. 2006. Nav 200 Absolute Navigation System.

[4] Jervis B. 2007. Guidance options for AGVs Webb Company.

[5] University of Birmingham. 2006. Sonar sensor and mounting.

[6] Kececioglu. 2006. Battery Charging Systems for Automated Guided Vehicles. AGV Battery Charging Systems. Egemin Automation Inc.

[7] Stone, R. 2008. Internal Logistics Systems: from operating requirement to opportunity for succes. OCME Srl.

[8] HK Systems, Inc. 2007. The Appropriate Application of Automated Guided Vehicles.

[9] Swisslog. 2006. Automatic Guided Vehicle-Description.

[10] Swisslog. 2006. SALMOIRAGHI Automatic Handling. The Salmoiraghi Approach

[11] Elettric. 2008. A Guide to Robotic Logistics – LGVs (AGVs). 80 http://en.wikipedia.org/wiki/Automated guided vehicle.html (14 March 2009)

[12] Lloyd,G.P. 2007. Simple Robotic Vehicles for Instructional Purposes

[13] Mindstorms. 2007. Telepresence Robot Kit. http://mindstorms.lego.com (17 June 2009)

[14] Jervis B. 2008. Webb Automatic Guided Carts. Webb Company

[15] Siemens. 2006. *The Basics of Automated Guided Vehicles.* Flexible Manufacturing Systems. AGV Systems.

[16] Jervis B. 2007. Guidance options for AGVs. Webb Company.

[17] Nav. 2006. Absolute Navigation System. Mobile Platforms. pp 5

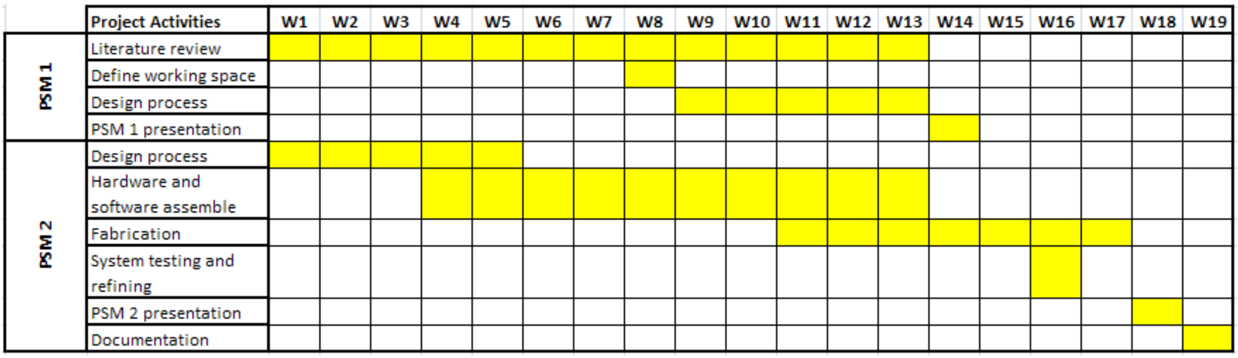
[18] University of Birmingham. 2006. Sonar sensor and mounting.

[19] JBT Corporation. Common AGV Applications: Raw Material Handling. 18 March 2009

[20] Basu, Dipak. 2001. Dictionary of Pure and Apllied Physics, CRC Press. pp. 75

APPENDIX A

GANTT’S CHART



	Project Activities	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17	W18	W19
PSM 1	Literature review	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨						
	Define working space								🟨											
	Design process									🟨	🟨	🟨	🟨	🟨						
	PSM 1 presentation														🟨					
PSM 2	Design process	🟨	🟨	🟨	🟨	🟨														
	Hardware and software assemble				🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨						
	Fabrication											🟨	🟨	🟨	🟨	🟨	🟨	🟨		
	System testing and refining																🟨			
	PSM 2 presentation																		🟨	
	Documentation																			🟨