School of Mechanical & Manufacturing (SMAE)

Design & Fabrication Project (EC1166)

Automated Guided Vehicle (AGV)

Prepared For:

MR. CHOY FOOK SENG

Prepared By:

KYAW SOE HEIN	**0856566**
KYI HLA WIN	**0856777**
GARRY HAN	**0847007**
AYE MYAT MON	**0856818**

CLASS : DMA/FT/2B/04

1ST FEB, 2010

SINGAPORE POLYTECHNIC

Table of Contents

Pages

Overview

In the second semester of Year 2, all Diploma in Mechatronics (DMA) students have to undergo a Design and Fabrication Project, which requires them to build an Automatic Guided Vehicle (AGV). As part of the criteria, the classes are divided into groups of four students, who collaborate and fuse talents together to create an AGV which has to be outstanding in both design and mechanism.

An AGV is a four-wheeled vehicle with an Infra-Red sensor attached to the underneath of the front part of it that detects a strip of aluminium foil guiding its way. The AGV will read and follow the path as it is programmed.

REPORT ON THE AUTOMATED GUIDED VEHICLE (AGV) PROJECT

1.0 INTRODUCTION

1.1 Objectives

The objective of building this AGV is to equip students with necessary hands-on skills on designing, fabricating and assembling of the mechanical structure and electronic parts of an automated vehicle. The critical part is to understand the concepts of design, drawings, process planning, machining and assembling, which increases in complexity as the mechanical and electrical parts integrate to work together. Students are expected to troubleshoot problems that arise from both mechanical & electronics aspects and be able to overcome these obstacles while completing the project on time.

1.2 Scope of Work

This report will mainly focus on areas involved in creating the AGV, while reflecting the efforts invested by the team. The areas include Design Description, Testing & Troubleshooting, and Discussion & Recommendation.

The section on Design Description will then be sub-divided into two key areas like software description and mechanical assembling description. The section on Testing & Troubleshooting will emphasize on dimensional and tolerance control and machining errors. Moreover, the section on testing PCB (Printed Circuit Board) and troubleshooting of electronics components will be covered in this report. Finally, this report will detail recommendations that will be provided for better and safer applications.

1.3 Design Specifications

A standard AGV measures 25cm in length, 20cm in width and 25cm in height. It has four wheels for its movement, two large motor-driven wheels and two smaller direction-adjustable wheels which perform a task of height balancing. A crucial part is the front fork-lift, which functions very similarly to the counterparts of the industry. A sensor board for track detection is attached to the front lower part. The AGV consists of two main circuit boards, which facilitate for easy investigation and troubleshooting.

1.4 Functions of AGV

A completed AGV performs two fundamental functions. First, it should detect and move along a designated path.

Second is the forklift which is designed to carry small sample weights, the fork-lift should move up and down by at least 10cm, and is controlled by a Microcontroller board, which is tuned with a language called Assembly Language. The Microcontroller board is linked with a geared motor, that turns a screw thread.

1.5 Application of AGV in Industry

An AGV does pre-determined standard procedures at a calculated rate. Its overall advantage is increased efficiency. AGVs are most appropriately used in Industries which require repetitive work. Another advantage is the lack of fatigue frequency of machines. Humans need to rest after carrying heavy objects, and thus wasting time. By using AGVs, the industry could improve quality of its service, mainly transporting components, in the Industry by doing steady, reliable jobs. The most common application of AGV in Industry is in the dockyards (For example, Keppel Shipyard), warehouses and factories. As shown below, Fig 1.1 and 1.2 are two pictures depicting AGVs for Industrial use.

Fig 1.1 - A Fully-Functional AGV for Industrial use

(Source: Google Image Finder - images.google.com)

Fig 1.2 - An AGV carrying bulky items in a factory

(Source: Google Image Finder – images.google.com)

2.0 Design Description

2.1 Mechanical Design

This section will discuss on Mechanical parts of AGV in the aspects of Design & process to obtain Hardware parts.

Parts of the AGV can be broken down into two main groups, namely Chassis (Body of the car) & Fork Lift.

The entire idea of utilizing the AGV in application is to transport objects by means of automated vehicle. In addition, lifting the objects up and down is done by fork lift which is installed onto the main body of the car. Therefore, different parts of each group are machined to fabricate accordingly to be able to fit into one as an AGV.

Group	Name of the parts
Chassis	1. Base Plate 2. Brackets 3. Wheels 4. T-mount(Supporting brackets for wheels) 5. Motor Supporting Brackets 6. PCB support
Fork Lift	1. Fork Column 2. Fork Lift Top 3. Fork Lift Guide Block 4. Fork Base 5. Fork Wedge 6. Guide Rod 7. Lead Screw

Table 1.1 List of the different parts of AGV

The general design of the AGV is fairly similar to locomotive robot which has locomotors (Wheels) and fork Lift at the front end to pick up things and transport them.

2.1.1 Design of AGV

2.1.1.1 Base Plate & Roof

It has a broad base to stabilize itself while moving. In technical terms, the base measures 25 cm in length and 20 cm in width. The height of the fork lift column where the fork base is attached is 25 cm. Thus, it makes the centre of gravity of the AGV (The point at which weight of the AGV, in this case, concentrated) lower, and hence it provides higher stability.

Furthermore, the electronic systems are integrated on the AGV by mounting PCB on the base plate. These electrical boards are covered by a rigid roof which in turn can protect circuit boards from external disturbance. This is particularly the best idea of the design which will provide robustness and safe condition for electronic components.

2.1.1.2 Wheels & Sensor Board

The castor wheels (Rubber wheels) are used to turn left or right depending on the movement of the rear wheels which are driven by stepper motors and they lead the way directed by a sensor board located in the front. Those large diameter wheels are designed for fast movement.

The sensor board is attached to lower surface of the base plate right below the fork lift column.

2.1.1.3 Fork Lift

Particularly, fork lift is another strong point of the design. Normally fork wedges, the two laterally extended appendices, are short and small making it difficult to lift heavy and large size loads. However, in this case, relatively long and strong fork wedges are installed for better efficiency. The wedges are attached to the wide fork lift base which in turn gives stronger and more stable applied force while lifting the loads. The fork lift base, in turn, is attached to the guide block which is placed between the fork lift columns. The guide block is responsible for vertical movement of the fork.

The movement of the fork is being driven by motor which is mounted on the base plate at the front end. At the tip of the power transmitting rod from the motor is a gear with teeth which rotates horizontally. On the other hand, the screw lead which moves the guide block creates the perfect gear system with transmitting rod. The screw lead is in vertical position and transmitting rod from DC motor is in horizontal position hence touching the tips of them at the point making 90 Deg angle and there is teeth fitting in each other.

In other words, when the power transmitting rod is rotating, the screw lead will be rotating as well because they are in same gear system. Therefore the guide block along with the fork will be moving upwards or downwards depending on the direction of rotation of the screw lead. The guide block is supported by two brass rod for its stabilized vertical movement.

Fig 2.1 - A 3D generated image of our AGV, with the most fundamental features visible

2.1.2 Software Application

Before fabricating the actual hardware (mechanical portion) of the AGV, the use of software, AutoCAD software is crucial to be fully accomplished. First of all, the AGV design is developed from the rough sketch. After developing rough sketch, the 3D drawing of AGV design is drawn so that it is possible to visualize and judge dimension of the entire car. As each individual part of the AGV car is to be machined precisely, it is drawn in 2D detail drawing assisting in actual fabricating process as well. These assessments acquire the aid of particular software, AutoCAD, one of the world leading cad software to interpret 2D, 3D, which provides ultimate in flexibility, powerful features, design and speedy documentation software.

2.1.2.1 AutoCAD for AGV

To start off with the assessment of fabricating the AGV car, students are to use AutoCAD software. The main purpose is to teach the students to utilize the knowledge attained from previous module taken, which is, computer-aided drafting module (AutoCAD), in year one stage B as well as to familiarize or practice the use of AutoCAD synchronizing with industrial manufacturing process. Students are provided with relevant templates featuring respective tolerances, dimension style, etc as well as systematic guideline of labeling by whom which part is drawn or checked. Each group performance is guided by supervisor for continuous assessment grading the progress of every week. The procedure for drawing process includes solid modeling 3D, 2D detailed drawings for machining individual parts and assembly drawing according to module map (assessment plan).

2.1.2.2 Solid Modelling 3D Design

Upon the confirmation of design sketching, the engineering drawing for fabricating purpose is completed by using AutoCAD 3D style. Initially, the dimension of standard parts provided by workshops such as miter gears, motors, castor wheels, bearings and screws are closely observed and the general idea of overall dimension from previous cars are measured or noted as reference for upgrading the design. What is more, the additional parts of new innovative design are considerably decided for dimension after the group discussion. While deciding the dimension of the car, the drafter should reckon to make use of the workshop's materials, standard material size available to minimize machining several parts for saving time and cost. As all the dimensions are confirmed, solid model is created using 3D concepts. 3D design can represent the entire volume of the object regardless of the shape of object.

2.1.2.3 UCS System

The entire AGV is drawn by using UCS (user coordinate system), which facilitates the user re-orientating the X-Y plane from 2D profiles to 3D solids so that the different parts of the car are drawn conveniently over different views, different orientation making it easier to draw out the 3D complete design. Each part is drawn from respective view followed by compiling them together.

2.1.2.4 Usage of commands

It is compulsory to know 3D commands to work out the drawing as it is relatively difficult to draw some critical parts such as gears and hinges(attached to the roof of the design).The range of commands need to be revised properly to accomplish the model drawing. The supervisor checked on the proper dimension and appropriate positioning for approval.

2.1.2.5 2D Detailed Drawings

In order to fabricate each component, 2D detailed drawings for individual parts are drawn using orthographic projection principle, projecting a number of separate two dimensional inter-related views which are mutually at right angles to each other.

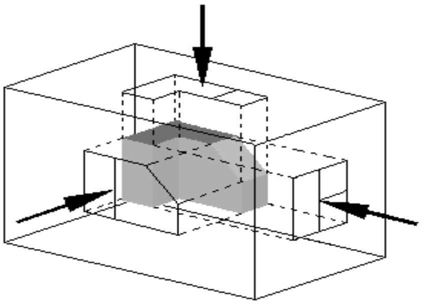


Fig 2.2 Third Angle projection

(Source: http://www.solidworkstutorials.com/en/wp-content/uploads/2009/07/third_angle_projecting.png)

2.1.2.6 First and Third Angle Projection

In third angle projection, object is positioned in third-angle quadrant between two principal planes. The view of object by drawing parallel lines from the object to the vertical plane is front view or elevation. The view similarly projected to the horizontal plane is called a top view or plan. The view projected on to the auxiliary vertical plane is called and end view. For AGV, third angle projection concept is applied to draw out all the 2D drawings. Figure shown below is the planning of third angle projection.

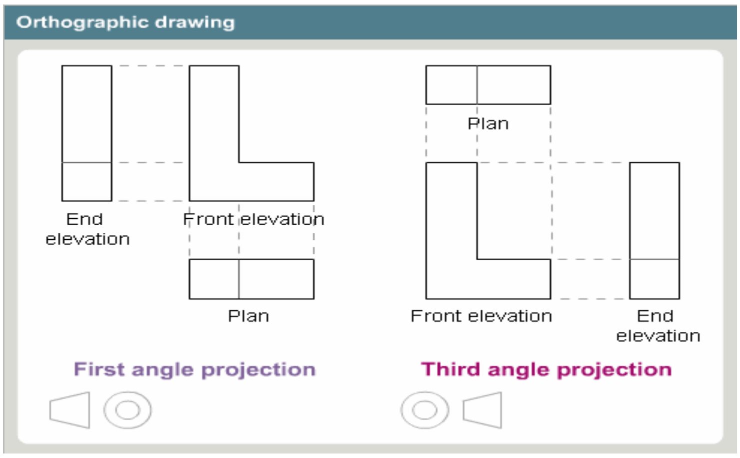


Fig2.3 - Comparison of the positioning plan of first to third angle projection

(Source: http://www.bbc.co.uk/schools/gcsebitesize/design/images/dt_g_dmi_tp9.gif) [accessed on 10/01/10]

2.1.2.7 Assembly Drawing

Finally assembly drawing for the whole car is produced. The guideline to directly transform from 3D isometric drawing to assembly drawing is provided in blackboard. Assembly drawing shows how the parts are fitted to make a complete unit and provide a visual image of how the finished product should look like when the parts are assembled. The individual part is indicated by a balloon with an arrowhead on the drawing of that particular part. The part list with detail description, material used and quantity of all the parts are included in the drawing.(refers to Appendix I).

2.1.3 Fabrication Process

Before commencing machining process, it is crucial to prepare for fabrication procedure worksheets. (Appendix2).

Planning fabrication procedures before machining will help to minimize the material wastage and machining time, as well as improve the quality of the machining work and maximize the utilization of machine tools and other facilities available in the Project Workshop.(This Project Workshop is located at W12 where students have to carry on machining processes)

In addition, a clear understanding of the size and functional requirements of the AGV is acquired as it will enable us to select the right type and size of raw materials for the AGV at the initial design stage. Furthermore, this will again reduce both material wastage and machining time, and most important of all, it will help to meet the design requirements accurately.

2.1.3.1 Machines

AGV fabrication needed drilling holes, making wheels and cutting the aluminium sheets to size. Thus, the following machines are used.

- Lathe Machine
- Milling Machine
- Drilling Machine
- Shearing Machine

2.1.3.2 General Mechanical Tools

The mechanical tools are also essential to carry out the machining process. Tools are used for measuring, marking, drilling, cutting and shaping.

This comprises of the following tools:

- 0-3" outside micrometers
- 6" rigid steel rule (5R graduation)
- Scriber set

- Centre punch set
- 20 oz. dead blow hammer
- Combination square set
- 8" mill/bastard file with handle
- 8" thread file with handle
- Rolling tool box
- Ball peen hammer
- Channel lock and needle nose (small)
- Safety glasses with side shields and so on.

2.1.3.3 Lathe Machine

A lathe is a machine tool which spins a block of material to perform various operations such as cutting, sanding, knurling, drilling, or deformation with tools that are applied to the work piece to create an object which has symmetry about an axis of rotation. Lathes are used in woodturning, metalworking, metal spinning, and glass working.

2.1.3.4 Milling Machine

A milling machine is a machine tool used for the shaping of metal and other solid materials. Milling machines exist in two basic forms: horizontal and vertical, means that cutting spindle can be in horizontal position or vertical position either.

2.1.3.5 Drilling Machine

A drilling machine is a machine which creates holes (usually called boreholes) and/or shafts in the metal work piece.

2.1.3.6 Shearing Machine

A machine for cutting bars, sheets and plates of metal into the dimension required. Especially it is useful for volume production of metal plates.

Figure 2.4 - Drilling machine

Figure 2.5 - Shearing Machine

(Source: Google Image Finder – images.google.com)

The fabrication of work pieces, parts of AGV, is then carried out subsequently. The procedure of machining of each part is described in the Appendix 2 in details.

2.2 Electronics Design

This section of the report covers the details of electronics hardware and software used in the AGV.

2.2.1 Sensor Board

The sensor board is a small circuit board whose function is to detect the reflective strip along which the AGV will move. It connects directly to the microcontroller board, which will translate information for the motor board to read.

2.2.1.1 Structure

The sensor board has a form of a square with a side of 6cm. It has four 3mm (diameter) holes for the purpose of mounting. The board is large enough to house several LEDs (Light Emitting Diodes), a couple of resistors, a transistor and an IC chip.

A copper patch is present on the lower side of the board, and is marked by the admission number of the student. Copper tracks are pre-positioned and all holes for component mounting are drilled by the manufacturer using 1.0mm (diameter) drill bits.

However, before soldering in the components, all copper tracks are tested as a procedure to ensure that cracks and discontinuity are not present, for the benefit of easier troubleshooting subsequently.

2.2.1.2 Components

The table below is a complete list of all components that constitute a sensor board.

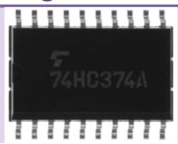
Reference Name	Value	Description	Diagram
IC31	74HC374	Octal Driver SMD	
T31	2N2222	NPN Power Transistor	
L31 – L35	SFH409 RS195-625	Infra-Red Emitter (Diameter = 3mm)	
D31 – D35	SFH309FA RS195-647	Infra-Red Detector (Diameter = 3mm)	
R31	10K	SIL 8 Resistor Array	
R32	2.7K	Resistor (Carbon Film) ¼ Watt	
R33	330	Resistor(Carbon Film)0.5 Watt	

Table 2.2 – All the components required for making a Sensor Board

(Pictures Source: Google Image Finder – images.google.com)

2.2.1.3 Sensor Board Silkscreen & Artwork

The manufacturing process for the Printed Circuit Board (PCB) involves computer drafting. Silkscreen is a layout on the board, which is calculated to the size of an actual board. Components in their respective packaging or footprint are placed on the layout as they should be on the actual board.

The artwork is a semi-final stage. It is a layout with all the copper strips, which would connect components to each other, in position. In other words, the artwork looks exactly the same as its counterpart – the real PCB. The PCB has two sides, each of which employs different copper strips for connecting various fields.

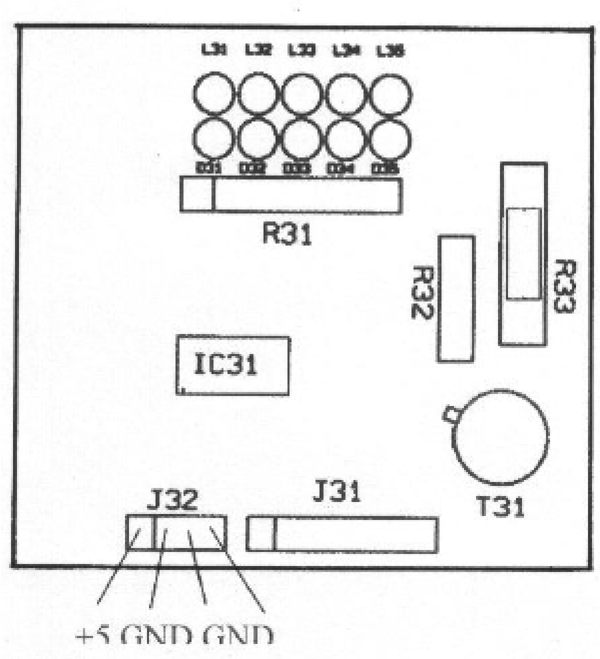


Fig 2.6 - Sensor Board Silkscreen

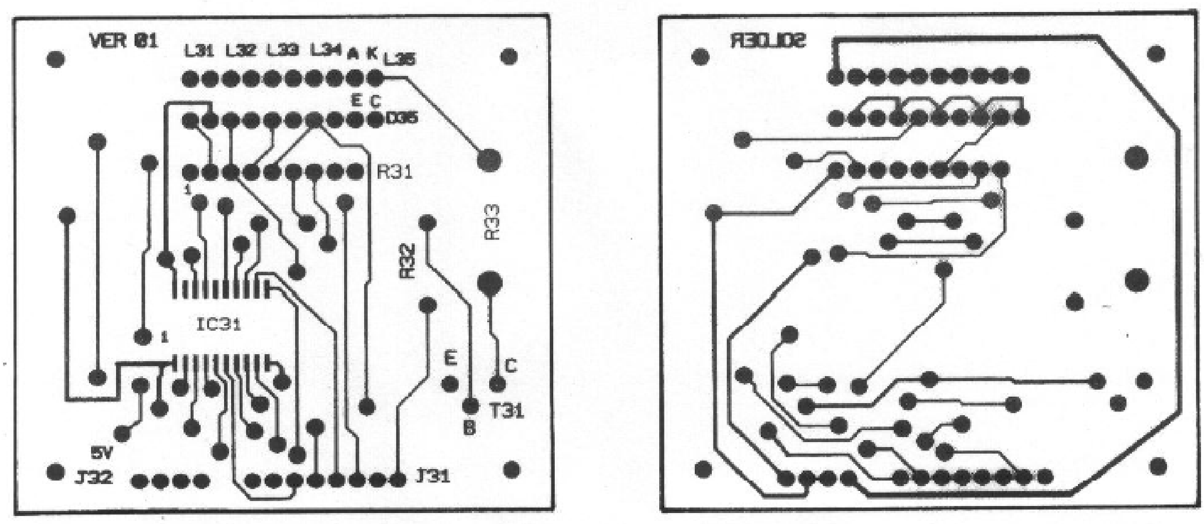


Fig 2.7 - Sensor Board Artwork

(Pictures Source: DFBP Textbook from SP Blackboard)

2.2.1.4 Surface Mounting

Unlike the Motor Board, the Sensor Board is designed to utilize a component that requires *Surface-Mount Technology (SMT)*. The IC chip (IC31) is mounted directly on the surface of the Printed Circuit Board (PCB), which is entirely different from the traditional mounting through holes.

The procedure requires a few steps. The chip is first carefully positioned on the PCB using tweezers. Once the pins are correctly located on the printed copper strips, a few drops of solder are put on two corners of the IC to hold it temporarily. Subsequently, large beads of solder are melted on all pins of the chip using the soldering gun at temperature around 400 degrees Celsius. Finally, excess amount of solder is cleaned off from the legs of IC chip using clay.

It is important not to heat the IC for more than 5 seconds since the high temperature can internally fry the chip.

2.2.1.5 Schematic Diagram

The mechanism running inside the Sensor Board can be explained better with the help of a schematic diagram below.

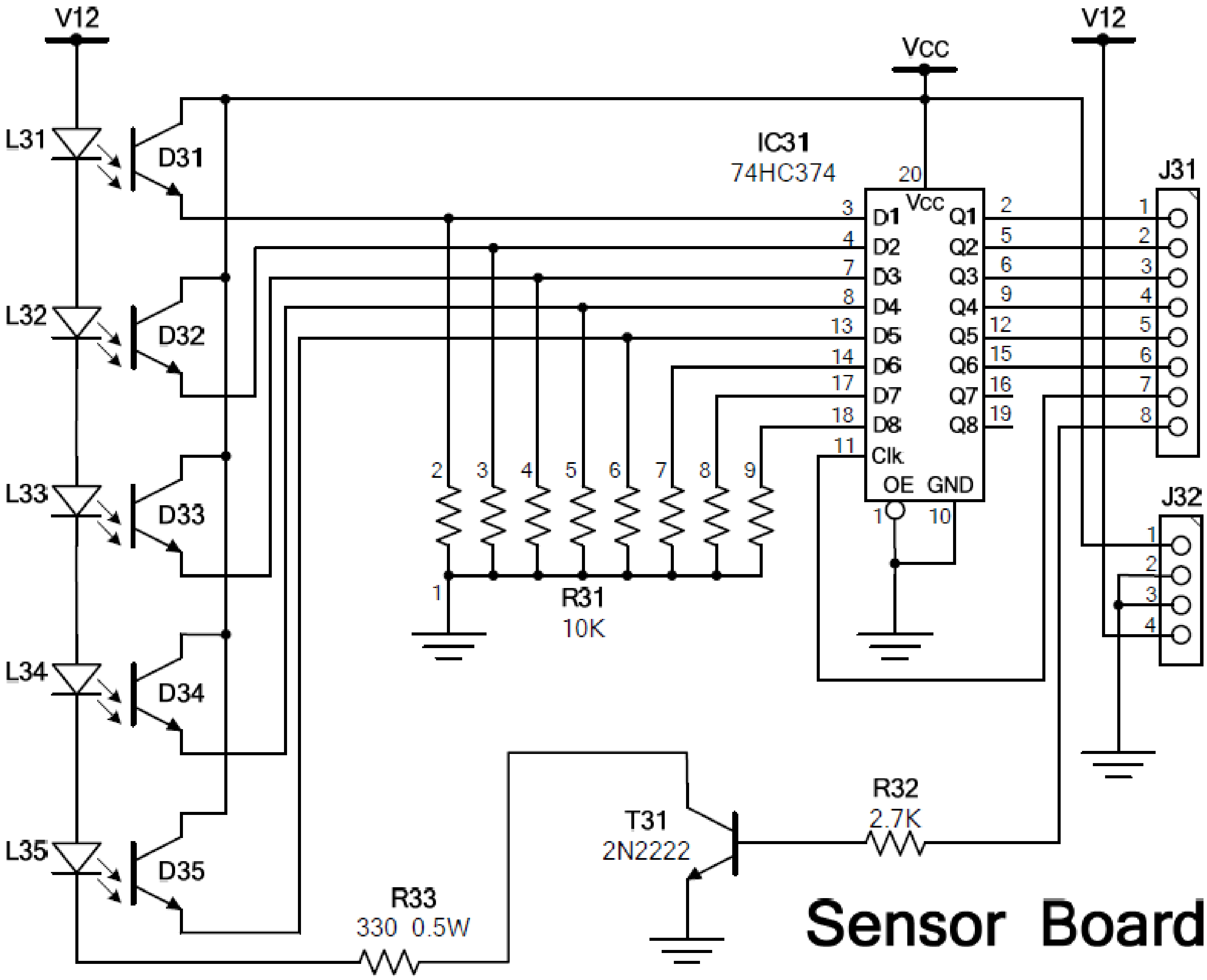


Fig 2.8 - Schematic Diagram of Sensor Board

(Source: DFBP Textbook from SP Blackboard)

2.2.1.6 How It Works

The five Infra-Red (IR) Emitting Diodes (L31 – L35) perform as light sources. They are supplied with 12 Volts on their positive end in *series* and the negative end is connected to the ground through transistor (T31). The transistor is forward –biased during operation so that current can flow through the IR Emitting Diodes.

Infra-Red rays are emitted from the diodes during the process. Those rays are bounced back when a reflective surface is present below. The reflected rays, at the right angle, enter the IR phototransistors (D31 – D35).

The phototransistors act as light sensors. When rays strike the phototransistors, current flows through the respective resistors from R31 to the ground. Pins (3, 4, 7, 8, 13) from IC31 measure voltage from each of the 5 LEDs with respect to the ground. If voltage is higher than 2.4 volts, the respective output from IC31 shows a logic 1, or 5V. Else, it shows logic 0, or 0V.

The signal is then sent to the Microcontroller Board.

When connected to the Microcontroller Board, the pin 1 from J32 is supposed to provide 5 Volts and pin 4 is supposed to supply 12 Volts. The other two pins (2 and 3) are Ground.

Pin 1 (5V) provides potential for the 5 IR Detectors which are connected in *Parallel*, so that each of the five will consume 5 Volts. Therefore, the output of these 5 LEDs can range from 0V to 5V, proportionately to the Infra-Red waves they detect.

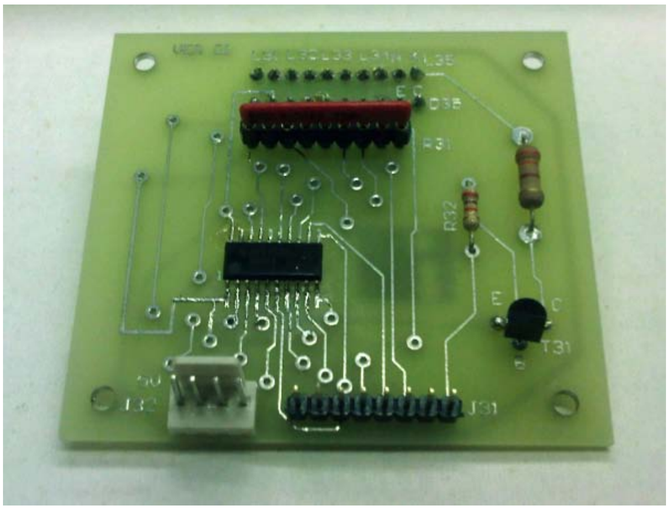
Pin 4(12V) provides potential for the 5 IR Emitters which are connected in *Series.* Each of the emitters will consume around 1.7V to 2.2V according to the conditions.

On the J31 connector, pin 1 through 5 carry signals from the IC according to the 5 LEDs. Pin 7 provides a clock frequency, which is 1.0000 KHz, to the IC chip. Pin 8 is called *Enable,* which means it provides 5V, or logic 1, at all times.

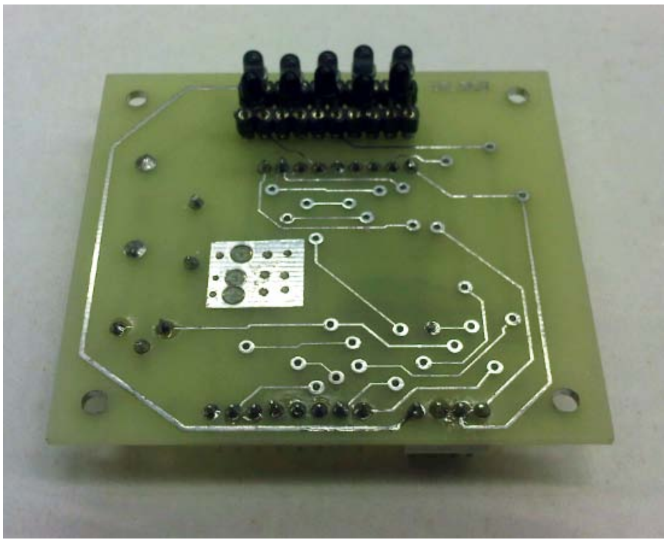
2.2.1.7 Finalized Board

A finished Sensor Board functions as follows. The IR Emitters emit Infra-Red rays when instructed from the Microcontroller Board according to the program. When the AGV is on its path with reflective strips, IR rays will reflect back to IR Detectors which allows current to flow through them to the resistors, which in turn set potential (according to Ohm's Law). The IC chip measures the voltage, and generates logic output, so that the Microcontroller can understand what the detectors are reading.

A Sensor Board functioning as such appears the same way as the one in the following picture.



(A)



(B)

Fig 2.9(a,b) – A finished Sensor Board ready for troubleshooting

(Photo by: Kyi Hla Win)

2.2.2 Motor driver board

The motor driver board plays an essential role in the AGV. It controls the movement of the AGV as well as the up and down movement of forklift.

Fig 1 shows placement and orientation of the different components required for the motor board to function.

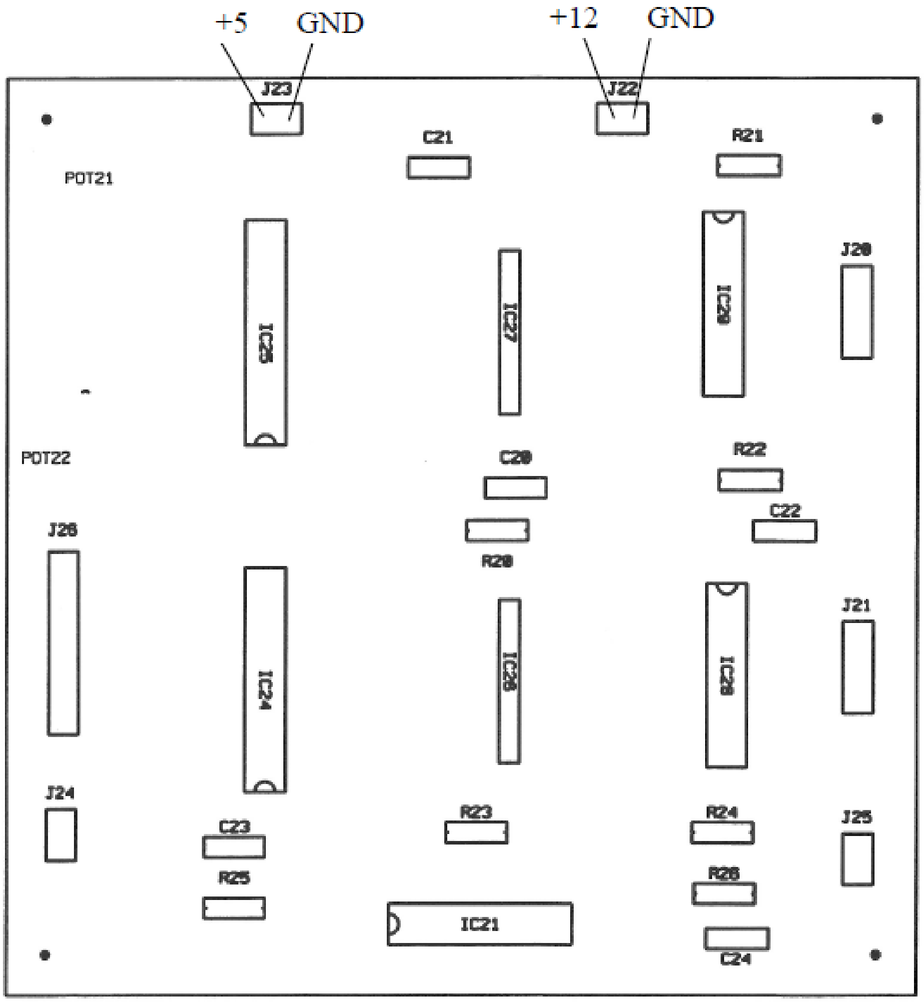


Fig 2.10 - Motor Board Artwork

(Source: DFBP Textbook from SP Blackboard)

To begin with, each component has to be brought individually from shops located in Sim Lim tower.

After the components were bought, our team has to solder each component with precision onto the printed circuit board.
This is a delicate process as poor solder joints may cause short circuits which will in turn damage the components or cause the circuit to malfunction.
When there is a short circuit or damaged components in the circuit, the motor driver may perform undesired operations which may cause further problems.

Fig 2 shows the motor driver board after all the necessary components are soldered onto the printed circuit board.

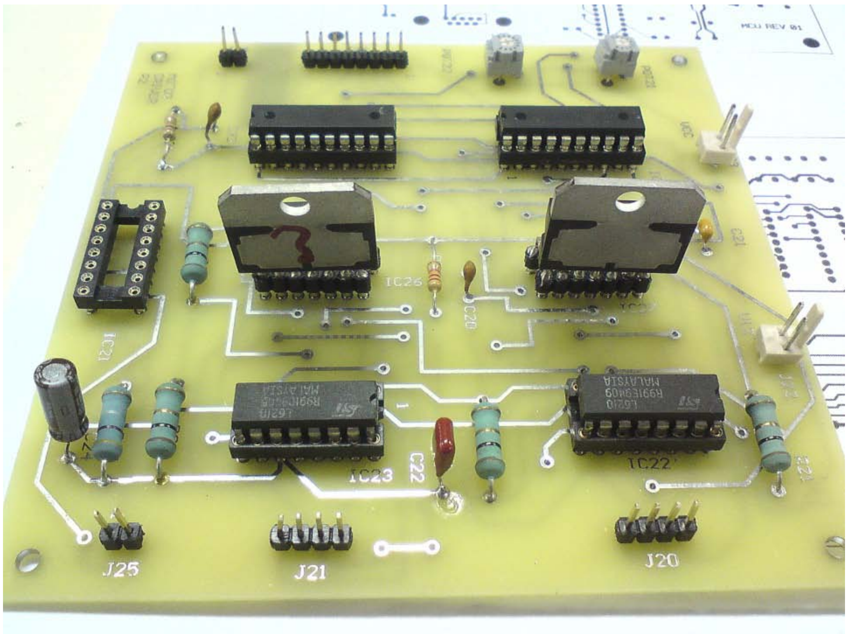


Fig 2.11 - Finalised Motor Board

(Photo by – Garry Han)

2.2.2.1 Components and description

- IC26, IC27 – L298 Dual Full bridge driver
  Used to drive bipolar stepper motor

- IC 24, IC 25- L297 Stepper motor controllers
  Used to control the drive of the stepper motor with half-step or normal step.

- IC21 – A3953 Full bridge PWM motor driver
  Used to control the operation of the DC motor.

- C20, C24 - 47uF/16v

- C21, C22 – 0.1uF/50v

- C23 – 820pF/50v 
- R21-R24 – 1ohm 1 watt 
- R20 – 22Kohm ¼ watt
- R25 – 27Kohm ¼ watt
- R26 – 0.47ohm 1 watt

(Pictures Source: Google Image Finder – images.google.com)

2.2.2.2 How it works

IC 24 and IC 25 are supplied with Vcc 5 volts to pin 12 and ground to pin 2. These 2 IC are used to provide signal to drive the stepper motor. The signals are sent to outputs A, B, C, D, pin 4,6,7,9 respectively.

IC 26,27 are supplied with Vcc 5 volts to pin 9 and 12 volts to pin 4.While ground is connected to pin 8. These 2 IC are used to drive the stepper motors. The signals are sent to outputs 1,2,3,4, pins 2, 13, 3, 14 respectively.

IC22 and IC 23 serve as diodes to protect the circuit from shortage.

Each of these IC works together in order to drive a stepper motor. This is the reason why there are 2 sets for each type of IC used.

Outputs from pins ABCD are sent to IN1, IN2, IN3, IN4 of IC 26, iC27. The signal is then processed and sent from OUT1, OUT2, OUT3, OUT4 to pin 2, 10, 7, 15 respectively.

The signal is then transferred to J20, where the 4 way pins of the stepper motor is connected.
The motor will turn according to the signal received.

IC 24 is used to control the operation of the DC motor. IC 24 is supplied with VCC to pins 1, 2, 6.
The DC motor is controlled by the signal coming from pin7 and pin 8, phase and enable respectively.
These 2 pins use the combination of logic signals to control the output. To control the dc motor to turn in forward motion, enable has to be set to logic low while phase is set to logic

high. For a reverse motion, both enable and phase are to set to logic low.

2.2.2.3 Hardware testing

After soldering is completed, our team checks for poor solder. If there are poor solders, our team will de-solder the particular section and solder it back again.

2.2.2.4 Manual Test

When checking is done, motor board is tested manually to check if it is working.

First, the following equipments are needed:
1: Multimeter
2:Oscilloscope
3:Power Supply
4:Signal Generator

To test if the motor board can control the stepper motor, the following has done.
First, connect J7 of the microcontroller board to J23 and J22 of the motor board using a 4 way wire. Ensure the 5 volt is correctly connected to J23 and 12 volt is correctly connected to J2. If these two volts are switched, the circuit may be damaged by the high voltage.
This action is to pass current to the motor board to activate the board.

Then use the probes to connect reset and enable pins of J26 from the motor board to the 5 volts pin of J6 on the microcontroller board.

Using probes again, connect the 2 clock pins, left and right respectively, of J26 on the motor board to ground on J6 of the microcontroller board.
This is to activate the individual stepper motors.

In order to test if the wheel’s direction of turning can be changed, our team connects right directional pin and left directional pin of J26 of the motor board to ground of J6 on the microcontroller board.

This ends the manual connecting sequence of the two boards.

Then turned on the power supply and adjust the voltage to 12 volts and current to 1 ampere and connected the red and black wire of the microcontroller board to live and ground of the power supply respectively.

After connecting the power supply, we connect a test probe to the oscilloscope. Connect the crocodile clip to common ground and begin probing each pins to check for desired outputs.

Check the output pins of IC24, IC25, IC26, IC27.
Refer to Fig 2.12 for the schematic diagram for respective pins to be probed.

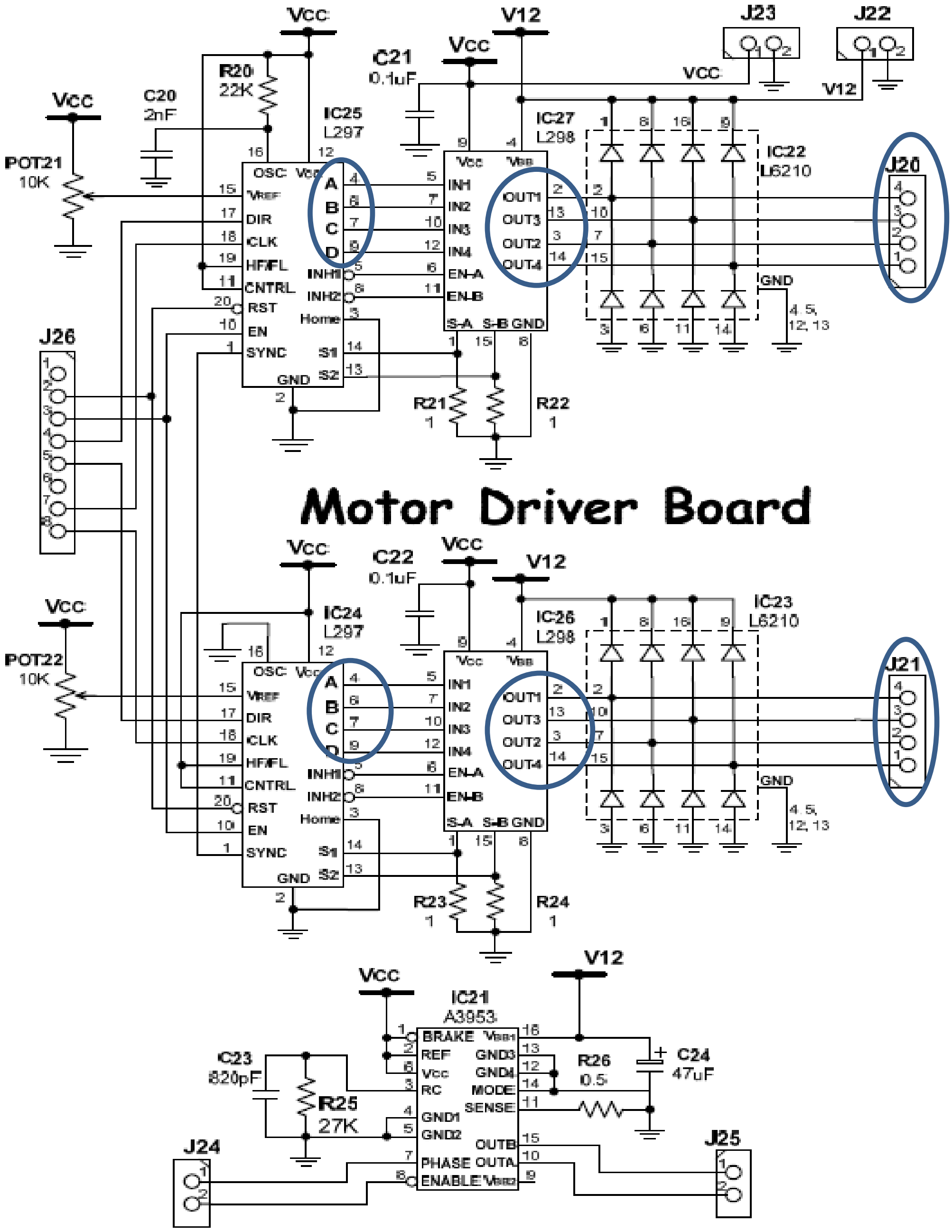


Fig 2.12 Motor Board Schematic Diagram (Source: DFBP Textbook from SP Blackboard)

Pin 4, 6, 7, 9 of IC24 and IC25 has to be logic 1, which is 2.4~5 volt. Pin 2, 13, 3, 14 of IC26 and IC27 has to be 2.4~5volt. While pins 4,3,2 of J20 and J21 has to be 2.4~5volt. Pin 1 of J20 and J21 has to be 0volt.

If the outputs are correct, the motor board is capable of controlling the stepper motor. This ends the manual testing of the stepper motor.

To test if the motor board can control the DC motor, these steps are taken.

Connect pin 1 of J24, enable, to pin 2 of J2. Connect pin 2 of J24,phase, to pin 1 of J2.
To test for forward slow current- decay mode, probe enable to ground, and phase to 5 volt. The output voltage of OutA should be 5 V while OutB should be 0V. This can be checked by using oscilloscope. If the outputs are correct, the motor board is capable of controlling the DC motor.

When the motor board is capable of controlling both the stepper motor and DC motor, the motor is serviceable.
This ends the manual testing of the motor board.

2.2.3 Software Description

As the AGV is a hybrid of mechanical, electrical and electronic concepts, it includes some programming for it to be complete – to be fully automated as it is named. This section will discuss the program codes that tell the AGV when to move, when to turn and when to stop. The programming language used is C-programming.

2.2.3.1 Program

```
#define porta (*(unsigned char volatile*)0x1000)  /* DC Motor */
#define portb (*(unsigned char volatile*)0x1004) /* LED & IR emitters */
#define portc (*(unsigned char volatile*)0x1003) /* Stepper Motor control */
#define porte (*(unsigned char volatile*)0x100A)  /* sensors */
#define ddrc  (*(unsigned char volatile*)0x1007)

#define L 1
#define R 2
#define LR 3
#define CW 1
#define CCW 0

main() {
    int Lmotor;
    int Ldir;
    int Rmotor;
    int Nmotor;
    int Rdir;
    char input;
    char sw;
    int check=0;        /* Boolean */
```

```
    int dblcheck=0;
    int sth=0;          /* Timer */
    int some;

    initport();         /* initialise I/O ports */

while(1)
    {
     sw=porta;
     sw=sw&0x04;
     if(sw==0x00)
     {

       while(1) {

             some=run_dcmotor(dblcheck,check);
             if (some==1)
             {
                 sth=sth+1;
             }
             input = readsensor();
             display(input);

               if (sth==10000)
             {
                  dblcheck=0;
                  check=0;
                  sth=0;
             }
             delay(1);
             if (input==0x00)    /* No Sensor */
             {
                    Lmotor=0;
                    Rmotor=0;
                    Ldir=CW;
                    Rdir=CW;
                    delay(1);
                    input=readsensor();
                    if (input==0x00)
                    {
                         if (dblcheck==1)
                         {
                            check=1;
                         }
                    }
             }
             else if(input==0x1f) /* Full Sensor */
             {
                    Lmotor=LR;
                    Rmotor=0;
```

```
        Ldir=CW;
        Rdir=CW;
        dblcheck=1;
    }
    else if(input == 0x0f)  /* Turning Left */
    {
        Lmotor=L;
        Rmotor=0;
        Ldir=CW;
        Rdir=CW;
    }
    else if(input == 0x07)
    {
        Lmotor=L;
        Rmotor=0;
        Ldir=CW;
        Rdir=CW;
    }
    else if(input == 0x03)
    {
        Lmotor=L;
        Rmotor=0;
        Ldir=CW;
        Rdir=CW;
    }
    else if(input == 0x01)
    {
        Lmotor=L;
        Rmotor=0;
        Ldir=CW;
        Rdir=CW;
    }                           /* End Turning Left */
    else if(input == 0x1e)      /* Start Turning Right */
    {
        Lmotor=0;
        Rmotor=R;
        Ldir=CW;
        Rdir=CW;
    }
    else if(input == 0x1c)
    {
        Lmotor=0;
        Rmotor=R;
        Ldir=CW;
        Rdir=CW;
    }
    else if(input == 0x18)
    {
        Lmotor=0;
        Rmotor=R;
```

```
            Ldir=CW;
            Rdir=CW;
        }
        else if(input == 0x10)
        {
            Lmotor=0;
            Rmotor=R;
            Ldir=CW;
            Rdir=CW;
        }                /* End Turning Right */


        Nmotor=run_motor(Lmotor,Ldir);
        Nmotor=run_motor(Rmotor,Rdir);

    } /* end of while */
   }
 }/*end of while*/
}

/*********************************************************************/
/*                    initport()                                     */
/*                                                                   */
/* This function initialises portc as output port. It also initialises */
/* RESET and EN inputs of L297s to '1'                               */
/*                                                                   */
/*********************************************************************/

initport() {

   ddrc=0xff;   /* set port C as ouput port */
   portc=0x40;  /* en, rst  */
   porta=0x20;  /* disable DC motor */
}

/*********************************************************************/
/*                    readsensor()                                   */
/*                                                                   */
/* This function reads the dipswitch connected to porte.             */
/*                                                                   */
/*********************************************************************/

readsensor() {
   char input;
   portb=0x01;          /* turn on IR emitter */
   portb=0x03;          /* clock in status */
   input=porte;         /* read status of sensor */
   input=input & 0x1f;  /* mask off the unuse bits */
   return input;
}
```

```
/*********************************************************************/
/*                    display(input)                                 */
/*                                                                   */
/* This function will output the data to the LEDBAR connected to portb      */
/*                                                                   */
/*********************************************************************/

display(input) {
   portb=~input;
}

/*********************************************************************/
/*                    delay(count)                                   */
/*                                                                   */
/* This routine will provide a delay of 'count' msec                 */
/*                                                                   */
/*********************************************************************/

delay(count) {
    int w,v;
    v=count;
    while (v>0)
   {
            w=75;
            while (w>0)
           {
              w=w-1;
            }
            v=v-1;
    }
}  /*  end of delay */

/********************************************************************/
/*                run_motor(int motor, int dir)                     */
/*                                                                  */
/* This function will output the direction (CW/CCW) and supply      */
/* a clock pulse to the L297 to drive the respective motor          */
/*                                                                  */
/*      PC7  PC6   PC5  PC4   PC3   PC2  PC1  PC0                  */
/*       0   _RST  EN   DIRL  DIRR   0   CLKL CLKR              */
/********************************************************************/

run_motor(int motor, int dir) {

    char output;
    /* dir=0 : CCW, dir=1 : CW */
    if (dir == 1) output = 0x78; /* set DIR-L & DIR-R to high  */
      else       output = 0x60; /* leave DIR-L & DIR-R as low */
```

```
    /* motor=1 : Lmotor     motor=2 : Rmotor   */
    if (motor == 1) output = (0x02 | output); /* set clk-L high */
    if (motor == 2) output = (0x01 | output); /* set clk-R high */
    if (motor == 3) output = (0x03 | output); /* set clk-L & clk-R high */

    portc=output;         /* output clock           */
    delay(1);           /* delay for 1 msec        */
    output = output & 0xfc;  /* set EN , clk-L and clk-R to low */
    portc=output;
    delay(1);
    return;
}



/*****************************************************************/
/*                      run_dcmotor()                                  */
/*                                                                   */
/* This function will activate the dc motor using switches SW2 & sw3.       */
/*                                                                   */
/*****************************************************************/

run_dcmotor(int dblcheck,int check) {
    int aa=0;
    char sw;
    sw=porta;       /* read switch */
    sw=sw&0x07;      /* mask off 5 msb */
    if (sw==0x06) porta = 0x40;  /* SW2 ON - DC motor CW */
    if (sw==0x05) porta = 0x00;  /* SW3 ON -  Motor CCW */
    if (dblcheck==1 && check==1)
    {
       porta=0x40;
       aa=1;
    }else if (sw==0x07) porta = 0x20;  /* SW2,SW3,SW4 OFF - disable DC Motor */

    return aa;
}

/*************************************************************/
/*                                                           */
/* This part will set the Reset Vector to the start of program          */
/*                                                           */
/*************************************************************/

/* starting address of EEPROM & program */
#define START  (void (*)())0xF800

/* address of reset vector */
#pragma abs_address:0xFFFE
```

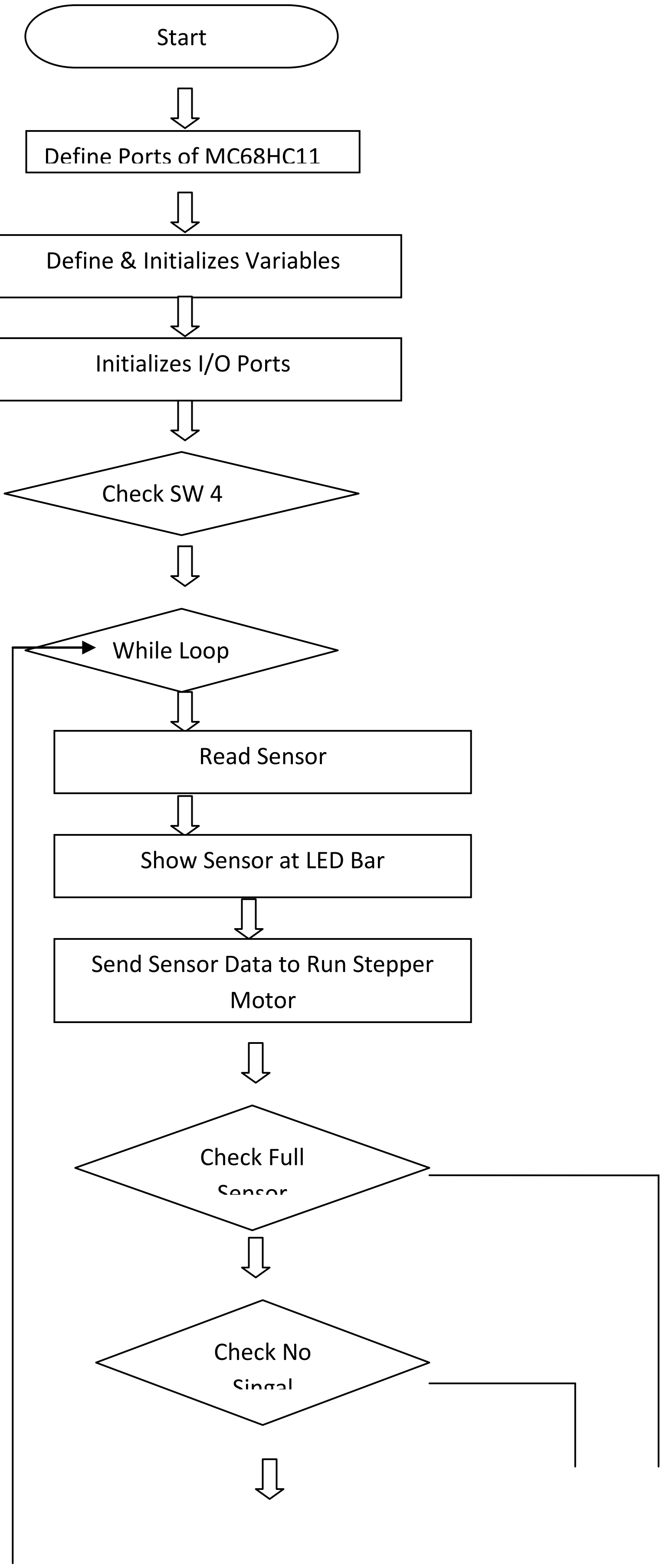
```
void (*interrupt_vectors[])() =
      {
      START,    /* reset vector */
      };

#pragma end_abs_address
```

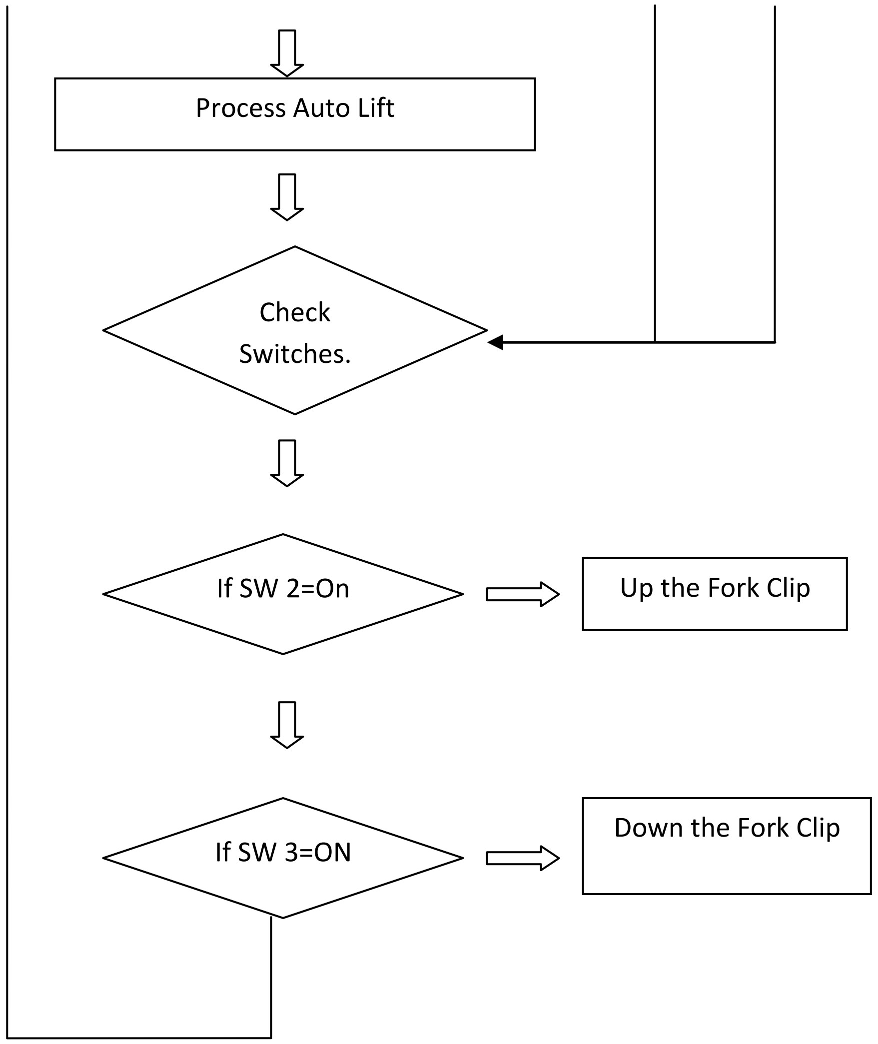
2.2.3.2 Flow Chart

A flow chart is the basis of any project that goes in a sequence. It defines all the intermediate steps and helps to avoid confusion especially when programming. Below is a flow chart that describes all details of the functions of an AGV.

SINGAPORE POLYTECHNIC

Flow Chart




Process Auto Lift
Check
Switches.
If SW 2=On
Up the Fork Clip
If SW 3=ON
Down the Fork Clip

3.0 Testing and Trouble Shooting

3.1 Testing & Troubleshooting of Mechanical Hardware

As individual part of the car is fabricated by each group member, the dimension can be slightly different from it really should be. Therefore, assembling the parts together is needed to check whether alignment of all the parts is correct or there is any part lessen or exceeded the dimension.

3.2 Procedures to check the requirement of AGV

3.2.1 Dimension

The overall dimension of the AGV should not exceed 250Lx200Wx250H. As the car is tested in the limited space of track, the exceeding dimension can disturb the functioning of the car.

3.2.2 Forklift Function

The forklift should be strong enough and installed firmly screwed as it must be able to lift a weigh up to 100 mm from ground level during testing. In addition, all the screws should be screwed properly in the case of breaking down especially for the joints which is exclusively designed unlike typical model of AGV in the past.

3.2.3 The roof accessibility

The particular design of roof should be functioning well with complete installation of hinges which serve the purpose of easily accessible. The height and stability of the roof should be checked if it is disturbing the circuit board cables or forklift.

3.2.4 Checking alignment

In order for AGV to operate appropriately, it requires to have good alignment. For example, motors should be mounted on the respective places on the chassis. The coupling of these motors to their shafts should be detachable as well. The critical point of alignment is forklift base where the joint of power transmitting rod and screw lead. Any mistake in alignment will drive the fork lift out of function. Moreover, the vertical and parallel aligned position of two copper rods is crucial so that the guide block can move vertically to control the movement of the fork.

3.3 Problems Encountered in Assembling

3.3.1 Dimension in Fork Lift Column

The dimension of fork lift column is relatively incorrect in such a way that motor support is slightly larger in size than the gap in between each fork column. Therefore, it is out of vertical upright position. In fact, the two fork lift column should standing upright and parallel to each other. However, as the dimension of forklift column is incorrect, it results in the non-parallel unbalanced rods disturbing the vertical stabilized movement.

3.3.2 Drilling Holes

It sometimes occurs because of the inaccuracy in marking out the holes and machining error or over machining allowance.

3.3.3 Tapping holes

The improper drilling and tapping of holes make it tough to screw them by force. Some cases, the over tapped or drilled holes are not stable enough to hold the screws. Therefore, it is important to take extra care while tapping holes.

3.4 Troubleshooting

The steps to troubleshooting includes

- Use scribers or deburring tools to enlarge the small holes
- Deburr the sharp unequal edges to integrate the entire car parts
- Use tap magic and tap over properly
- Re-measure and dimension the incorrect parts and fabricate again with lowest allowance

.

Fig 3.1 - Scriber

Fig 3.2 - Deburring tool

(Pictures Source: Google Image Finder – images.google.com)

Finally, the overall dimension, each parts dimension are corrected to be able to access and function properly. All the drilling holes are screwed properly and the alignment becomes acceptable by the troubleshooting process in order to continue polishing and coating for colour spray and the hardware mechanical portion of the AGV is successfully achieved.

3.4 Testing & Troubleshooting of Electronics

After finishing the circuit boards, it is necessary to test them and troubleshoot if any failure occurs. As mentioned in previous section, the AGV electrical system is consisted of microcontroller board, motor board and sensor board. In addition, two steppers motors and a DC motor are comprised to form a peripheral electrical subsystem for locomotion.

3.4.1 Microcontroller Board

The microcontroller board is provided by electronic lab to each group and it is merely needed to testify. Nevertheless, the electronic components can be worn out and it is not robust against hostile condition. Therefore, the microcontroller was checked by means of:

1. Visual inspection
2. Multi-meter testing

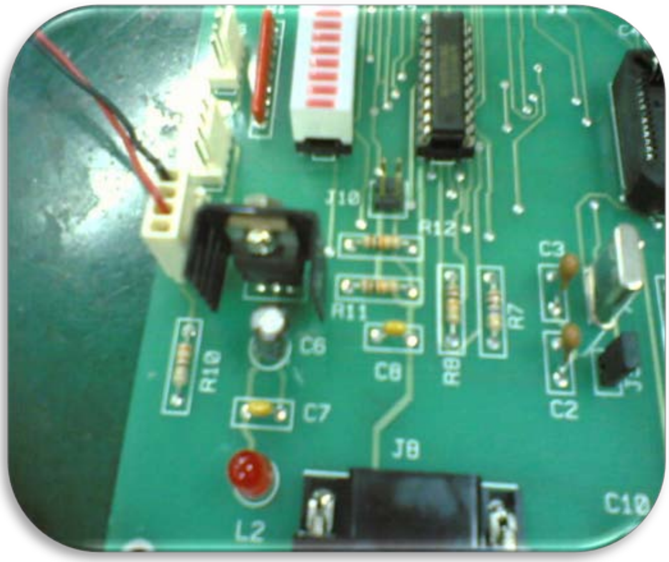


Fig 5.3 – Close-up of Microcontroller Board

(Photo by: Kyaw Soe Hein)

However, this is very important to check the power supply and pushbutton switch as it is vital for all the process. It is clear that power supply plays most important role.

Therefore, it is needed to make sure external power supply is limited to 0.5 A. Another thing is to ensure that the correct logic level will be fed to the µC when the switches (SW2 to SW4) are pushed ON and OFF. If these switches give out wrong logic, the entire system will be giving out wrong sequences and error logic.

The given table is provided for clear understanding purpose of how the circuit is tested & troubleshoot.

Table 3.1

Problem	Solution
Short circuit due to improper solder • **Electricity is connected between the two parts which were not supposed to connected**	De-solder & make the surface clean the solder again
Open circuit due to improper solder • **The resistance value between the two parts is very high while it should be nearly "0" Ohm.**	Solder a light coating of lead on the open part
Malfunction due to poor fabrication	Use shape object to scratch off the copper or remake the PCB

3.4.2 Sensor Board

The sensor board were examined visually and tested with multi-meter to check disconnected parts. Should there be any open circuits; the parts have to be soldered again to make it closed. In order to test the sensor, the voltage is supplied and clock signal is connected to respective pin No: as described in data sheet. Then, connect base of led, pin 1, with red wire from oscilloscope and place the card which has reflective surface and black surface.

When reflective surface pass under the sensor which is connected to oscilloscope, the display will be 5 V. On the other hand, when the black surface is under the sensor, 0 V will be shown on display. The same procedure is carried out repeatedly for all the sensors. If the particular led does not work accordingly, replace the defective sensor with working one.

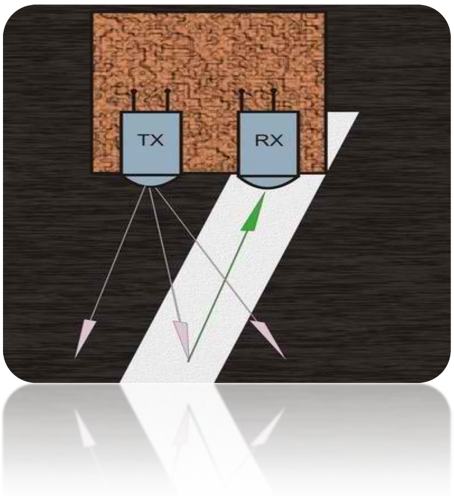


Figure 5.2 Sensor reading the reflective surface
(Source: Google Image Finder – images.google.com)

The problem of improper soldering can happen all the times. However, it can be checked easily using multi-meter and if any found out, De-solder and make it clean and solder again.

3.4.3 Motor Board

When testing the motor board, the power supply should be adjusted to 12V voltage and 1 A current. Then, the board is connected to the power supply, oscilloscope and signal generator. And, the ground wire of each component is connected together and carried with single main ground wire to earth plug.

After that, the VCC and 12V pin of the motor board and the microcontroller board are connected respectively. The average working voltage of IC chips are set to 2.5 V by tuning potentiometer. The pin No: 20(Reset) and 10(Enable) of IC 25 connected to the 5 V and Pin No: 7 and 8 are connected to the signal generator which provides signal.

The Pin 18 of IC 24 and 25 is supplied clock frequency both. Then output Pins 4,6,7 and 9 are examined if they give out clear clock signal. On the display, the base of the discrete steps can have some thick and distorted line. It indicates that soldering is not well. The perfect soldering should yield in single clear base line while displaying on oscilloscope.

The stated out pins should show clear signal. If they do not, the each IC and copper tract should be checked to find out the discontinuity. Then, the parts which lost contact will be soldered again. If problem still persists, replacing the defective ones with working ones will be required.

3.4.4 Testing & Troubleshooting of stepper motor

The stepper motors are used for supplying rotational energy to wheels. The 4-ways wire from each motor is connected to J20 and J21 respectively. A test is carried out to check whether the axis is rotating. If it does not, visual inspection and multi-meter testing should be done. Generally, the fault is due to the lost contact of wire coming out from motor. It is rectified by solder the wire to its base.

3.4.5 Testing & Troubleshooting of DC motor

In order to run the DC motor, J22 and J23 were supplied with 12V and 5V respectively. The pin 2 of J24 is connected to 0V for logic 0 to enable the DC motor. The DC motor should be working at some speed.

If it doesn’t work, visual inspection will be carried to check any lost contact between wire and base of the DC motor or using multi-meter for open and short circuit. If any lost contact found out, lost contacts will be soldered to join.

4.0 Conclusion

Finally, the last of all parts was fabricated. The last component was soldered. The last screw was tightened on the AGV and the last sentences of the report are being typed in. The moment for determining the AGV as a success or a failure has arrived, finally.

Although it is not yet the end of the semester and many of the AGVs still not being tested yet, it is not difficult to foresee the future of a project that is at the stage of showing characteristics such as failing or achieving. Our group confidently says a ‘success’. And we are not gurgling words in nonchalance. We have many reasons to say so.

It has been fortunate enough for us to meet wonderful lecturers, knowledgeable workshop & lab assistants and helpful classmates. The lack of one of these people would have meant a great loss. But we lacked none, and suffered least in most things.

Our group members have been enthusiastic and efficient. We have people who are skilled in certain areas. Working alone could have never brought a better AGV.

With evidence of perfect design, durability and feasibility, our AGV serves its purpose well. This project has completed its goals and taught us the talent of fusing mechanical, electrical and electronics – in other words, the art of ‘Mechatronics’.

5.0 Discussions & Recommendations

Evidently, the team have encountered many difficulties through the stages of constructing a functional AGV. This section is intended to discuss and find solutions to problems that project group felt significant.

Tools Missing in Workshop

Many tools were missing from the tool cabinet beside each machine in the workshop.

Tools had to be borrowed from other machine's tool cabinet. Most tools are in use and not always available. Such case can cause delay, inconvenience and frustration. We recommend the technical service officer in charge to replace all respective missing tools.

Materials Availability and Financial Support

Some other barriers are lack of modern machines, limited size of materials and a missing financial support, such as one provided for Final Year Project. Due to these barriers, a perfect vehicle is almost impossible to build. If we can overcome these barriers, the AGV could become a state-of-the art robot and its appearance and functionality could compete with those produced by industry. Thus, we strongly recommend that SP should offer much more support and install modern automated machines for fabrication. That would bring excellent changes in the result so that students can have a more valuable and memorable experience.

Time Constraint

Several times, our progress was hindered when staff in charge were not present or when resources were at minimum. For example, the workshop is notorious for having frequent breaks. It is understandable for elder lecturers to grow tired easily when instructing with machines. Our workshop assistants and lecturers are not to be blamed, but the school should rather hire more staff so that they can take turns having breaks and keep students working on machines at all times.

Electrical Components

It was somehow interesting why the School of Electrical and Electronics Engineering could not provide basic electronic components. Such components cost low in average, and it should not be a problem. However, due to the awkward unavailability, students have to travel to Sim Lim Square and purchase components for their school projects, at the risk of buying wrong components at the same time. It gets frustrating when some parts have gone wrong. While the team could easily have grabbed the necessary components if it is provided by school, the project group have to go through all the trouble of riding MRT to Sim Lim Square again just to delve into a myriad of poorly-staffed stores.