Smart Shopping Cart

by

KOK LIANG YI
15291

FINAL YEAR PROJECT REPORT

Dissertation submitted to the
Electrical & Electronic Engineering Programme
in partial fulfillment of the requirements
for the Degree
Bachelor of Engineering (Hons)
(Electrical & Electronic Engineering)

SEPTEMBER 2014

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Smart Shopping Cart

by

Kok Liang Yi
15291

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Approved by,

____

Dr. Syed Saad Azhar Ali

Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

September 2014

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

____

KOK LIANG YI

ABSTRACT

This project aims to design and develop a prototype of an automated motorized shopping cart, capable of trailing shoppers along with the extensive ability to avoid obstacles with its built-in intelligence.

This feature has enabled the study of embedding the extensive ability of trailing shoppers in a shopping complex. Given a situation with obstacles in a complex mall, the shopping cart would have the ability to evade them. This can be done because of the system is governed by a simple micro-controller (Arduino Mega 2560), with ultrasonic positioning system (Transmitter and Receiver based approach) for identifying and locating the target person, and infrared sensors system to measure the direction of obstacles. In hope of adding intelligence values the shopping cart would be design in such a way that, the purpose of extending convenience service and safety to the customer can be delivered.

Keywords: *Smart shopping cart, shopping trolley, automation motorized shopping cart*

ACKNOWLEDGEMENTS

First, I would like to express my utmost gratitude to Universiti Teknologi PETRONAS (UTP) for providing sufficient project and/or research materials to ease the research of my project. The excellent environment contributes to the life of me as an undergraduate student.

Secondly, I would like to acknowledge the constant guidance and instruction of my advisor who is also my project supervisor as well, Dr. Syed Saad Azhar Ali for his patience and sincere guidance towards me. Dr Syed Saad has shared with me a lot of his experience throughout his research years that helps in inspiring me in my project.

Besides, I thank all my friends, lecturers, technicians, and PETROBOT club members that contribute to the success of my project. Their participation in the experimental part of the project is highly appreciated.

Last but not least, I would like to thank my family the most. I am very thankful for having their moral support by my side always throughout my undergraduate student's life.

List of Figures

List of Tables

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1. Background of Study

In this rapidly changing era, the innovation of technologies evolves at an exponential rate. Many companies are investing into developments which ensure satisfaction for all stages of consumers. One of the most common development is technologies in customer service which promises new and robust systems for shoppers.

Shopping carts, which are also commonly known as shopping trolleys are a method of transportation of goods temporarily before cashing out. Shopping carts have limited changes made since its invention. Most of the expansions have been done is to modify its capacity and weight. However, due to the improvement of technology, some of the company's research have been developed a convenient shopping system to customer. For example there is a touch panel and bar code scanner attached on the shopping trolley, the customer can be informed about the promotion and location of goods from the touch panel. Besides that, the bar code scanner will scan the purchasing item when the customer place it into the trolley, and the amount of the bill will be displayed on the touch panel display. The customer will just have to perform the payment based on the amount on the display to the cashier without the long queue of payment procedure.

The improvement of shopping cart as mentioned can save shopping time of customer, but it requires manually navigation by the customer. Therefore, the purpose of this project is to allow hands free shopping experience to the customer, so that the need of self-driven and artificial intelligence shopping trolley is inevitable.

1.2. Problem Statement

This project's challenges are inclusive of elimination of accidents caused by shopping trolleys operated by humans. Human error in this matter, as we all know, causes minor collisions frequently bumping to toes and feet as well as racks and shelves in the mall. Meanwhile, the carelessness of human may contribute to more serious accidents involving trolleys with loaded goods, children and other shoppers. We are hopeful to eradicate these accidents using our proposed solution.

1.3. The Relevancy of the Project

Relevancy of the project includes structured programming. The language which governs the system will be in C language. Aside from structured programming, electrical machines will be incorporated because there will be need of calculating torques for motor usage for the trolley drive wheels.

1.4. Objective and Scope of Study

The objective of this project is to design and develop a prototype of an automated motorized shopping cart, capable of trailing shoppers along with the extensive ability to avoid obstacles with its built-in intelligence.

The scope of study for this project is to study and apply the suitable method for human tracking, obstacles avoid and interface the whole system with microcontroller.

CHAPTER 2

LITERATURE REVIEW

Automation is basically the delegation of human control function to technical equipment. It is the use of control systems such as computers, Programmable Logic Controller (PLC) and microcontrollers are the common devices that use to control machinery and processes to reduce the need for human sensory and mental requirements as well.

There is the comparison between Programmable Logic Controller (PLC) and microcontroller (μC) as shown in the table below.

Table 1: Comparison between Programmable Logic Controller (PLC) and microcontroller (μC)

	Programmable Logic Controller (PLC)	**Microcontroller (μC)**
Power supply	Most work with 24VDC or 230VAC	Most work with 5VDC
Programming language	Ladder diagrams, Statement list	Assembly, C, C++, Basic
Application	Designed for industrial purposes	Designed for used in consumer electronic system
Current driving capacity	Large	Small
Size	Large	Small
Cost	High	Low

In order to design and develop automation system, we need input for the system. For example, switches, timer, thermocouple, sensor and etc. Sensor is a complicated device, it converts the physical parameter such as pressure, temperature, proximity, etc. into a measured electrically signal. It is often used to detect and respond to electric or optical signal due to its characteristic.

The idea is to detect the objects that are nearly placed without any point of contact is using proximity sensors. Proximity Sensors are using high frequency oscillation to detect metal and nonmetal objects that nearly to the sensors.

The main principle of proximity sensor is it emits signal such as infrared light, sound wave, radio wave, etc. and receives and evaluates the returned signal or echo. For example, when the time interval between the signal emitted and received is long, this determines the object is far away from the sensors.

There are different types of proximity sensor in the market, such as ultrasonic proximity sensor, capacitive proximity sensor, inductive proximity sensor, photoelectric sensor, and so on.

Ultrasonic sensors utilize the reflection of high frequency sound waves to detect parts or distances to the parts; capacitive proximity sensors use the variation of capacitance between the sensor and the object being detected. Inductive proximity sensors use currents induced by magnetic fields to detect nearby metal objects; photoelectric sensor use light sensitive elements to detect objects.

In order to track target users, the very first step for the system is to determine the position of target. There are several methods that have been used to detect the position of target. Based on the research of Nagumo and Ohya[1], the target person carries 2 Light Emitting Device (LED) which fixed perpendicular on a stick, the camera on the robot will capture the image of it. After that, they were using the direction of LED on the captured image to compare with the central axes and determined the distance and direction of target.

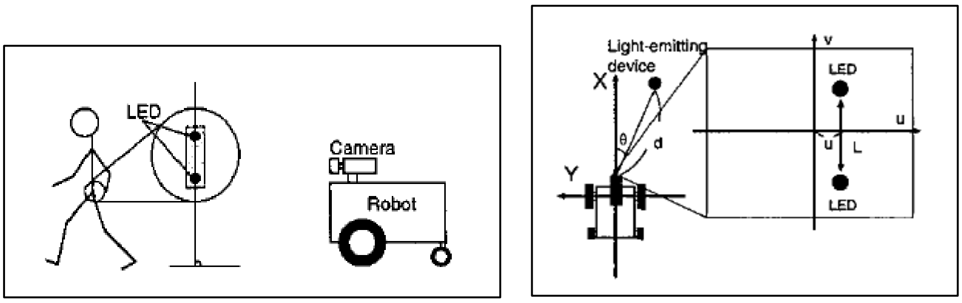


Figure 1: Human Following Using Light Emitting Device [1]

Besides that, vision based target detection were used in the study of Yoshimi et. al [2]. In the beginning, the robot identifies an individual with its image processing system by detecting a person's region and identifying the registered color and texture of his/her outfits. Although vision based system enables obtain of various kinds of information, but computing the image processing is complex.

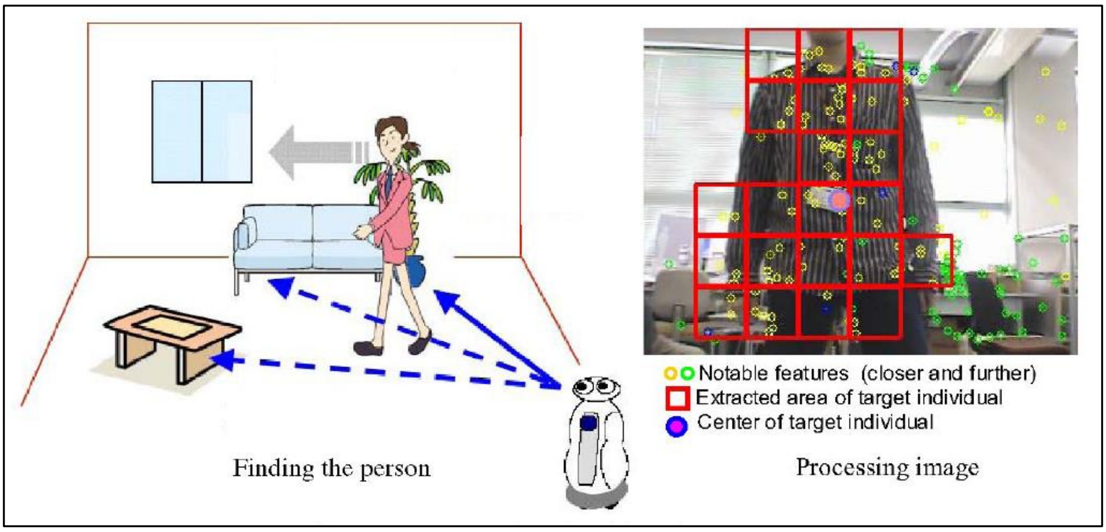


Figure 2: Vision Based Target Detection [2]

Other than that, the method that used by Kohtsuka et.al [3] to recognize the target's position is by calculating the central angle and the boundaries of target which the data obtained from the Laser Range Sensor (LRS).

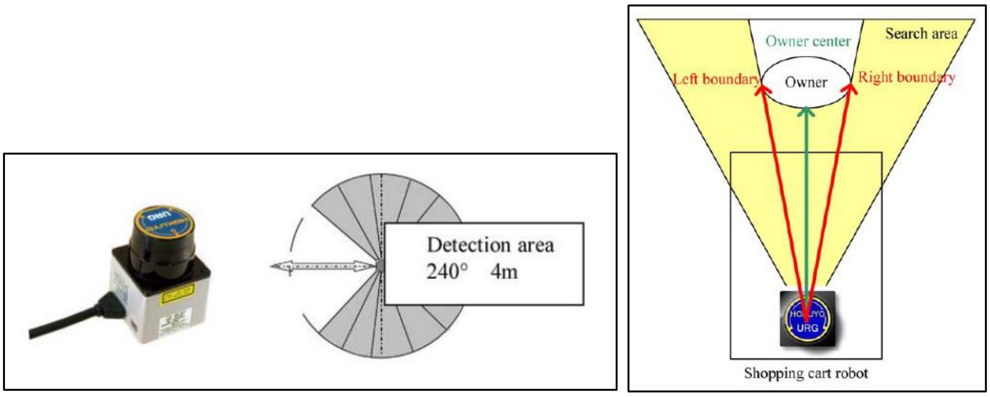


Figure 3: Target Recognize Using Laser Range Sensor [3]

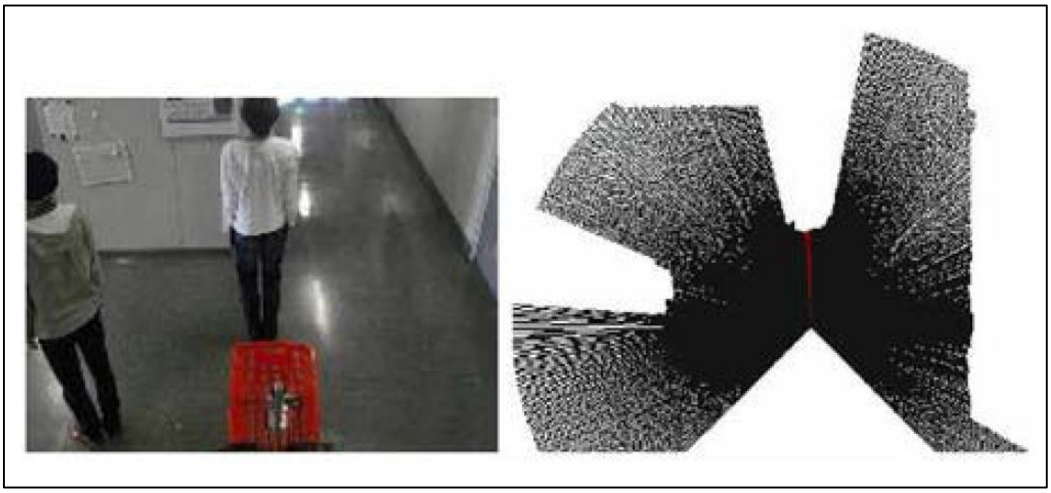


Figure 4: Example of Data Obtained from the Laser Range Sensor [3]

According to the research of Schulz et al. [4] we can get high range finding accuracy and high resolution of target by using Laser Range Finder (LRF), but it has a major disadvantage where it could not recognize the specific target whenever there are more than 1 target in the detection range. Thus, we have to combine other functions with the method in order to detect target by using distance sensors.

There is an example research [5] that uses 2 different types of sensors: large range sensor and Kinect sensor to detect target. The advantage of Kinect sensor is that it can perform tracking of target and large range sensor can monitor in wide range, thus this combined application of sensors results a wide range of tracking performance.

Another method that used by Bigos et al. [6] is transmitter-receiver based approach. The target user wears the Infrared (IR) emitter bracelet, and mounted 2 IR detectors on the robot, so that the robot can detect the target user.

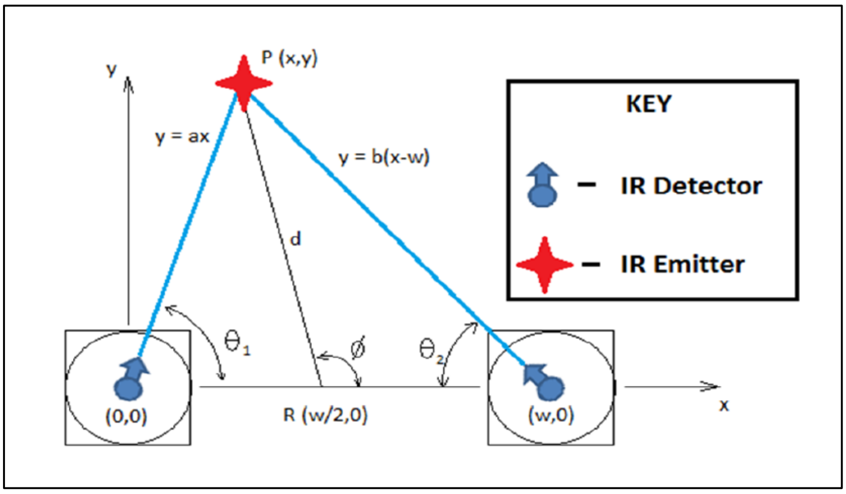


Figure 5: IR Triangulation [6]

CHAPTER 3

METHODOLOGY/ PROJECT WORK

3.1. Research Methodology

In methodology, the first step was to do the feasibility study which is intending to be a preliminary review of the facts to see if it is worthy of proceeding to the analysis phase. It is because the feasibility analysis is the primary tool for recommending whether to proceed to the next phase or to discontinue the project.

The aims of the smart shopping cart are:

i. be safe,
ii. be user friendly,
iii. able to tracking and follow target user, and
iv. able to avoid obstacle.

Automation is the use of control systems to control processes, reducing the need for human intervention. Due to human error and/or carelessness of the customer while control their trolley causes incident happen. Thus, the use of robotics and automated equipment is an effective way to prevent injuries.

Next, design the project. The method of 4W1H (What, Who, Where, Why and How) which is a question and answer based method has been use to starting the project design.

Table 2: The Question and Answer Based Method of Project

No.	Question and Answer
1.	Q: What does the modern shopping experience need? A: An intelligent automatic operation of shopping cart.
2.	Q: Who will use this smart shopping cart? A: Shopping customers and consumers.
3.	Q: Where is this project applied at? A: In uncrowded shopping environment.
4.	Q: Why doing this project? A: To allow hands free shopping experience to the customer.
5.	Q: How is the project work? A: The author gets and studies the knowledge and information from research, evaluate from experience supervisor, and then comes out a prototype design of intelligent automatic operation of shopping cart.

The following components are selected, after studied and analyzed the literatures and references as mention in previous chapter (Chapter 2).

i. Microcontroller (Arduino Mega 2560) as the control system device, because of its user friendly capability, low power consumption and low in cost, compare with computer and PLC.
ii. Battery as the power supply due to the application of shopping cart is dynamic, not stationary in a place/point.
iii. DC motors to drive the wheels of shopping cart.
iv. Ultrasonic sensor as tracking function element, because we can specific the frequency of ultrasonic for each of the individual user, thus the system can keep tracking the specific user easily.
v. Infrared sensor as obstacle detection element, because it can detects most of the object in the shopping mall, user friendly and low in cost.

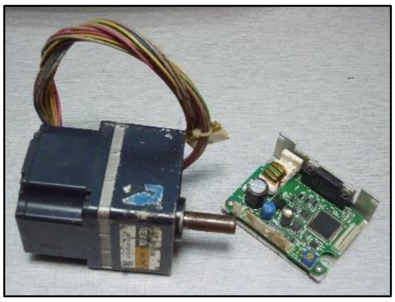


Figure 6: Shopping Cart

(100 Liter Capacity)

Figure 7: 24V DC Motor

and Driver

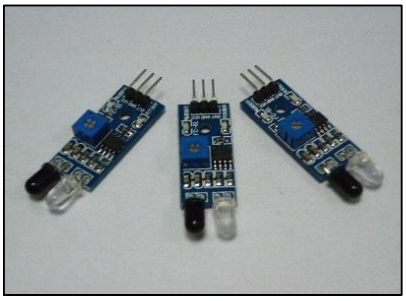


Figure 8: 12V Battery

Figure 9: Infrared Sensors

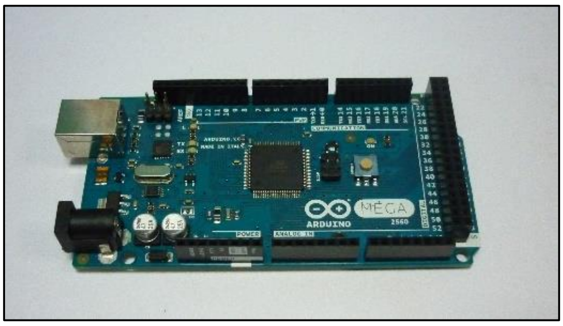
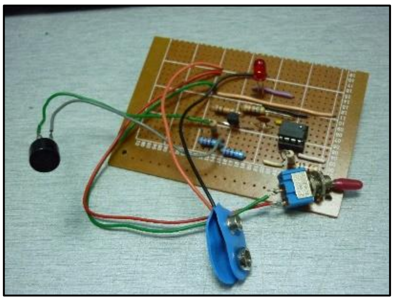


Figure 10: Arduino Mega 2560



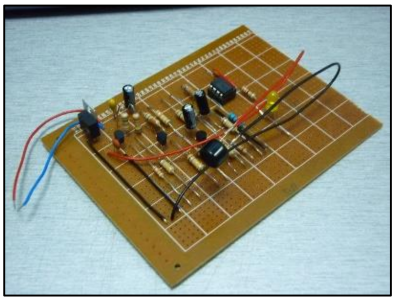


Figure 11: Ultrasonic Transmitter

Circuit

Figure 12: Ultrasonic Receiver

Circuit

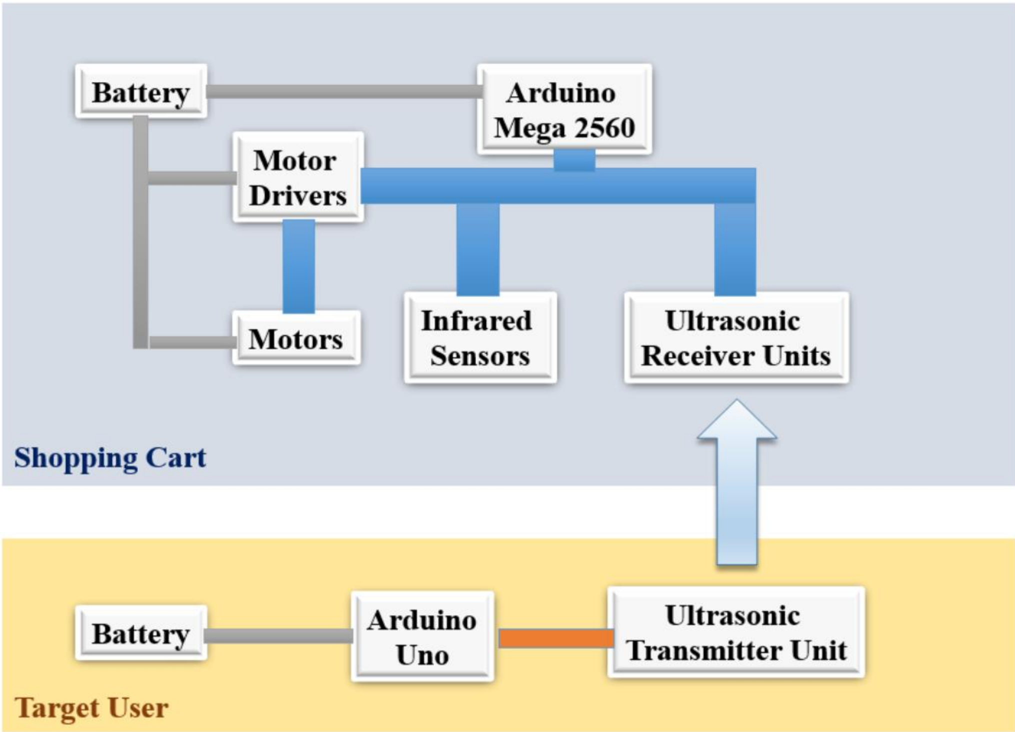


Figure 13: System Block Diagram

Figure 13 shows the block diagram of the system. The ultrasonic transmitter circuit in the target user part contains battery, Arduino Uno, and ultrasonic transmitter unit. The battery will power up and activate the controller (Arduino Uno), and the controller is connected to the ultrasonic transmitter unit. Therefore, the ultrasonic transmitter unit will transmit the signal according to the frequency that set in the controller while the power is ON.

Power unit (battery), Arduino Mega 2560, motor drivers, motors, infrared sensors, and ultrasonic receiver circuits are the essential components. The battery will power up the motors, motor drivers and the controller (Arduino Mega 2560). The ultrasonic receiver circuit is to receiver the signal that transmitted from the target user. The infrared sensor is to detect the position of the obstacle. The controller is to analyze the data/signal from the infrared sensors and ultrasonic receiver circuits, and feedback the shopping cart movement by sending signal to the motor drivers to control the motors.

3.2. Electrical/Electronics

3.2.1. Ultrasonic Sensor Circuits

The ultrasonic sensor circuits design (both transmitter and receiver circuits) are referred the “Ultrasonic Transmitter Receiver Circuit Ideas” [9] which prepared by the HobbyTronixStore EBAY STORE. Aside from this prototype ultrasonic sensory circuit, there are also commercialized sensory circuits manufactured with higher accuracy readings and suitable in term of functionality which is suggested to be used to enhance the performance of the smart shopping cart’s sensing ability.

3.2.1.1. Transmitter

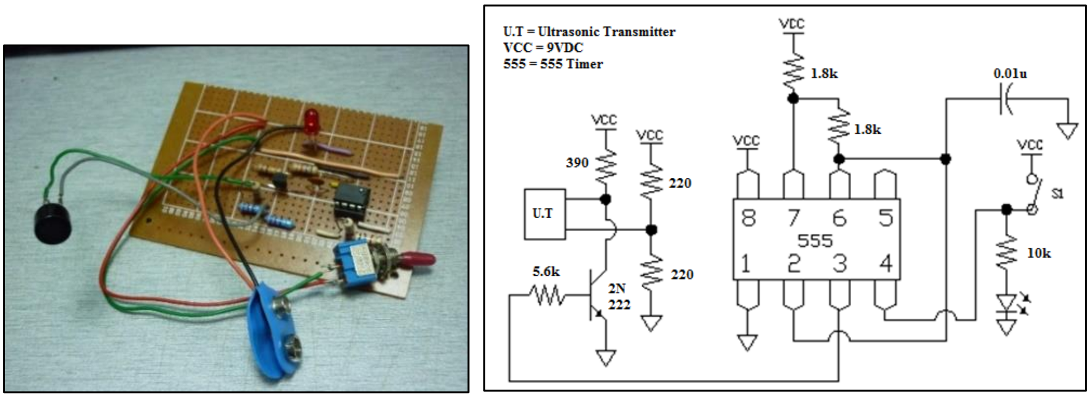


Figure 14: Ultrasonic Transmitter Circuit Diagram

In this transmitter circuit, the single pull single throw switch (S1) is to allow or cut off the power to the circuit; the 555 timer (555) is using to create a transmission frequency; the ultrasonic transmitter (U.T) is to transmit signal; and the red LED indicator is to indicate the power supply of the circuit.

The components that determine the frequency and duty cycle of the 555 timer in astable mode are: the resistor (R1) between VCC and pin No. 7, the resistor (R2) between pin No. 6 and 7, and the capacitor (C).

Calculation of 555 timer in astable mode:

$$\text{Frequency} = \frac{1}{0.693 \times C \times (R1+2R2)}$$

$$= \frac{1}{0.693 \times (0.01\mu) \times [1.8k+2(1.8k)]}$$

$$= \frac{1}{37.422\mu}$$

= 26.72 kHz

≈ 27 kHz

Time ON = 0.693 (R1 + R2) C

= 0.693 (3.6 k)(0.01 µ)

= 24.948 µs

Time OFF = 0.693 (R2) C

= 0.693 (1.8 k)(0.01 µ)

= 12.474 µs

$$\text{Duty Cycle} = \frac{Time\ ON}{Time\ ON + Time\ OFF} \times 100\% = \frac{24.948\ \mu s}{37.422\ \mu s} \times 100\% = 66.67\%$$

3.2.1.2. Receiver

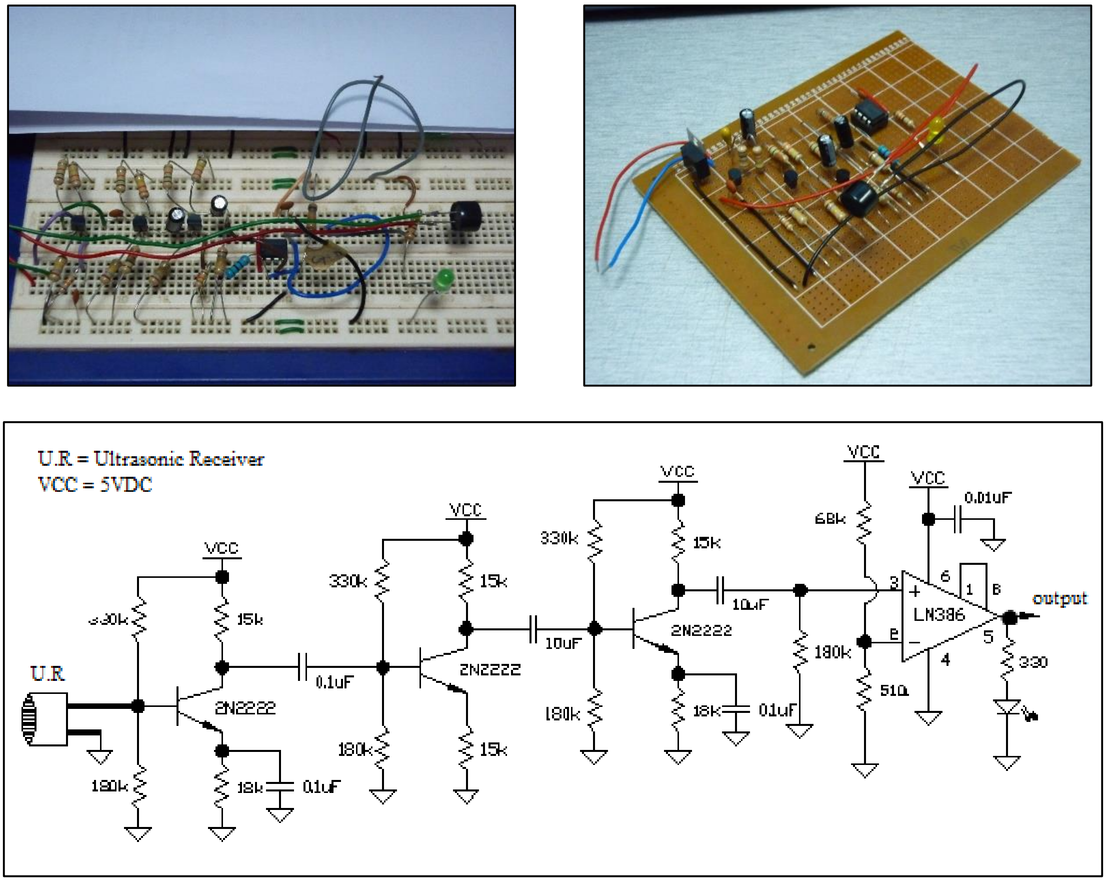


Figure 15: Ultrasonic Receiver Circuit Diagram

In this receiver circuit, there are 4 stages of operation, which are 3 amplifier stages and 1 comparator. Each stage is coupled by a capacitor to get rid of the DC component or remove the bias. The first 3 stages with NPN transistor (2N222) are the amplifier stage which is to amplify the received signal, and the last stage with amplifier chip (LM386) is the comparator. The function of the comparator is to set a threshold value, whenever the received signal is strong enough to exceed the threshold value, a square wave will get at the output. The LED indicator which placed at the output stage is to indicate whether the received signal exceed the threshold value. (LED will illuminate if the received signal exceeds the threshold value, otherwise the LED will not be illuminate).

3.3. Programming

3.3.1. Motors Movement

There is several movements of the shopping cart:

i. When the shopping cart is in idle mode, the both motors will stop and are freewheel;

ii. In order to move the shopping cart to forward, the left motor has to rotate in clockwise (CW) direction, at the same time the right motor has to rotate in counter-clockwise (CCW) direction;

iii. In order to make a left turn, the left motor has to brake, at the same time the right motor has to rotate in CCW direction;

iv. In order to make a right turn, the left motor has to rotate in CW direction, at the same time the right motor has to brake.

```
void freewheel()
{
  digitalWrite(motorL1,HIGH);     //
  digitalWrite(motorL2,LOW);      //  Stop
  digitalWrite(motorL3,LOW);      //
  analogWrite(motorL5,200);       //

  digitalWrite(motorR1,HIGH);     //
  digitalWrite(motorR2,LOW);      //  Stop
  digitalWrite(motorR3,LOW);      //
  analogWrite(motorR5,200);       //
}
```

```
void forward()
{
  digitalWrite(motorL1,LOW);      //
  digitalWrite(motorL2,LOW);      //  Move CW
  digitalWrite(motorL3,LOW);      //
  analogWrite(motorL5,200);       //

  digitalWrite(motorR1,LOW);      //
  digitalWrite(motorR2,LOW);      //  Move CCW
  digitalWrite(motorR3,HIGH);     //
  analogWrite(motorR5,200);       //
}
```

```
void left()
{
  digitalWrite(motorL1,LOW);      //
  digitalWrite(motorL2,HIGH);     //  Brake
  digitalWrite(motorL3,LOW);      //
  analogWrite(motorL5,200);       //

  digitalWrite(motorR1,LOW);      //
  digitalWrite(motorR2,LOW);      //  Move CCW
  digitalWrite(motorR3,HIGH);     //
  analogWrite(motorR5,200);       //
}
```

```
void right()
{
  digitalWrite(motorL1,LOW);      //
  digitalWrite(motorL2,LOW);      //  Move CW
  digitalWrite(motorL3,LOW);      //
  analogWrite(motorL5,200);       //

  digitalWrite(motorR1,LOW);      //
  digitalWrite(motorR2,HIGH);     //  Brake
  digitalWrite(motorR3,LOW);      //
  analogWrite(motorR5,200);       //
}
```

Figure 16: Programming Code of Motors Movement

3.3.2. Human Tracking

The idea for the design of human tracking is analyze the output of the ultrasonic receivers, (which the receivers are placed at the front left and front right of the shopping cart) and responds to the motor movement as followed:

i. If the target user is in front of the shopping cart, both of the ultrasonic receivers will get HIGH output, then the forward function will be executed and the shopping cart will move forward towards the target user.

ii. If the target user is at the front right of the shopping cart, the left ultrasonic receiver will get HIGH output, and the right ultrasonic receiver will get LOW output, then the right function will be executed and move the shopping cart to the right towards the target user.

iii. If the target user is at the front left of the shopping cart, the left ultrasonic receiver will get LOW output, and the right ultrasonic receiver will get HIGH output, then the left function will be executed and move the shopping cart to the left towards the target user.

iv. If the target user is not in the detectable range of the shopping cart, both of the ultrasonic receivers will get LOW output, then freewheel function will be executed and the shopping cart will stop and stay in the place.

```
if ( (sensorValueUL == HIGH) && (sensorValueUR == HIGH) )
{
  forward();
}
else if ( (sensorValueUL == LOW) && (sensorValueUR == HIGH) )
{
  right();
}
else if ( (sensorValueUL == HIGH) && (sensorValueUR == LOW) )
{
  left();
}
else
{
  freewheel();
}
```

Figure 17: Programming Code of Human Tracking

3.3.3. Obstacle Avoidance

The idea for the design of obstacle avoidance is to analyze the output of the Infrared sensors, (which placed at left and right side of the shopping cart) and responds to the motor movement. As following:

i. If the obstacle appears in front of the shopping cart, then freewheel function will be executed, and the shopping cart will stop and stay in the place.
ii. If the obstacle places at the front left of shopping cart, then the right function will be executed, and the shopping cart will make a right turn if there is no obstacle places at the turning direction as well.
iii. If the obstacle places at the front right of shopping cart, then the left function will be executed, and the shopping cart will make a left turn if there is no obstacle places at the turning direction as well.

```
if ( ( sensorValueL1 == LOW) && ( sensorValueR1 == LOW) )  // path block
{
    freewheel();
} */
if ( ( sensorValueL1 == LOW) && ( sensorValueR1 == HIGH) )  // left hit
{
    right();
}
else if ( ( sensorValueL1 == HIGH) && ( sensorValueR1 == LOW) )  // right hit
{
    left();
}
```

Figure 18: Programming Code of Obstacle Avoidance

3.4. Mechanical

3.4.1. Calculation of Drive Wheel Motor Torque Required

Design criteria	:	
• Total shopping cart weight	:	30 kg (66.14 lb)
• Number of drive wheel	:	2
• Weight on each drive wheel	:	8.6 kg (18.96 lb)
• Radius of wheel	:	62.5 cm (2.46 in)
• Desired top speed	:	1.5 ft/sec
• Desired acceleration time	:	1 sec
• Maximum incline angle	:	2 degree
• Worst working surface	:	concrete (good)

This calculation is referred from the manual of drive wheel motor torque calculation prepared by MAE Design and Manufacturing Laboratory [7]. In order to select the motors capable of producing enough torque to move the shopping cart, the motors torque must be greater than or equal to the calculated wheel motor torque. Hence the selected motor for this project is VEXTA AXH Series Brushless DC Motor systems (AXH230KC-30) which can produces the torque up to 3.1 Nm (27 lb-in). (For more information about the selected motor please refer to Appendix 3).

The calculation steps are shown as the following:

Step 1: Determine Rolling Resistance (RR)

Rolling resistance = Total shopping cart weight $\times$ Surface friction

= 66.14 lb $\times$ 0.01

= 0.6614 lb

Step 2: Determine Grade Resistance (GR)

Grade resistance = Total shopping cart weight $\times$ sin (Maximum incline angle)

= 66.14 lb $\times$ sin $(2°)$

= 2.31 lb

Step 3: Determine Acceleration Force (FA)

$$\text{Acceleration force} = \frac{\text{Total shopping cart weight} \times \text{Top speed}}{32.2\ (\frac{ft}{s^2}) \times \text{Acceleration time}}$$

$$= \frac{66.14\ \text{lb} \times 1.5\ \frac{ft}{s}}{32.2\ \left(\frac{ft}{s^2}\right) \times 1\text{s}}$$

= 3.08 lb

Step 4: Determine Total Tractive Effort (TTE)

Total tractive effort = Rolling resistance + Grade resistance + Acceleration force

= RR + GR + FA

= 0.6614 lb + 2.31lb + 3.08 lb

= 6.0514 lb

≈ 6.05 lb

Step 5: Determine Wheel Motor Torque

Wheel motor torque = Total tractive effort × Radius of wheel × Resistance factor

= 6.05 lb × 2.46 in × 1.1

= 16.37 lb-in

= 1.85 Nm

Step 6: Reality Check

Maximum tractive torque = Weight on wheel × Friction coefficient between wheel and ground × Radius of wheel

= 18.96 lb × 0.4 × 2.46 in

= 18.66 lb-in

= 2.11Nm

Noted:
∗ The wheel motor torque must be less than the maximum tractive torque for all drive wheels to avoid slipping occur.
** An appropriate acceleration time must be chosen such that the required
Wheel motor torque < (Maximum tractive torque × number of drive wheels)

3.4.2. Assembly Motor With Wheel to Shopping Cart

Figure 19: Modifying Shopping Cart's Wheels

The top left of Figure 19 shows the original design of the shopping cart, and the modified shopping cart is showing in the top right of Figure 19. The wheels of the trolley were not originally designed for electricity driven, therefore, modification is performed to the wheels in order to allow automatic function to be applied. The shaft of the motors were modified additionally with coupling method to mechanically suite the wheels. The coupling method is used to attach the motor and the wheel. A steel base is also mounted underneath the trolley which act as the platform to secure the control system of the automatic shopping cart.

3.5. Flow Chart

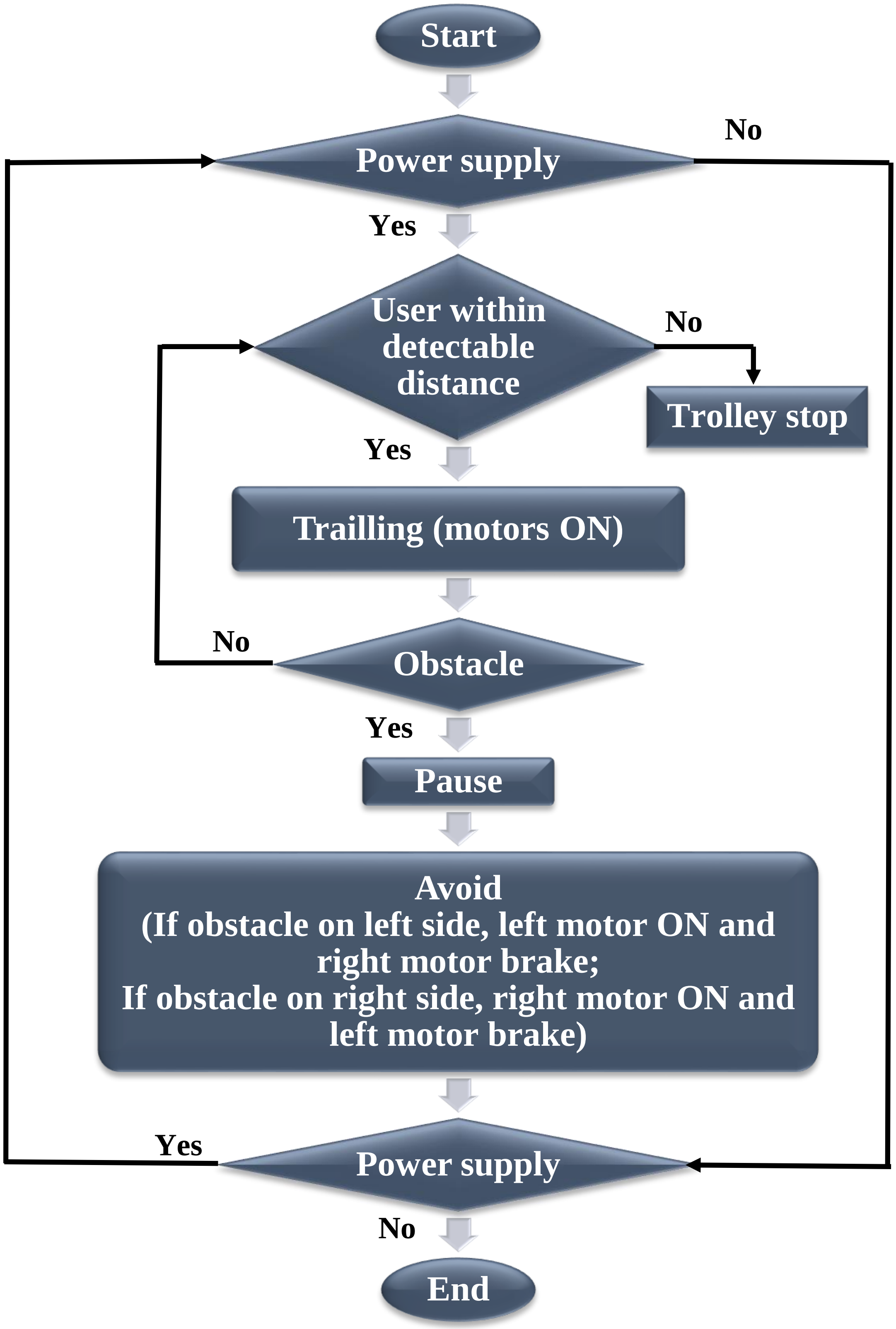


Figure 20: Flow Chart of Smart Shopping Cart

At initial stage, when there is power supplied, the ultrasonic transmitter starts to transmit signal, and the ultrasonic sensor (receiver) from shopping cart will then start to detect the signal from the user's tag (ultrasonic transmitter). When the target user is within the detectable range, the sensor will send the signal to the microcontroller, and the microcontrollers will activate the motors to move the trolley towards the user; otherwise, the microcontrollers will remain idle and the shopping trolley will stay in place.

The Infrared (IR) sensor will identify any objects within the distance range of the shopping cart, if there is an object close to the shopping cart, the sensor will communicates with microcontroller, and the microcontroller will collect the other IR sensor signal to respond and evade the object.

For example, if:

i. There is an obstacle close to the left side of the shopping cart, the shopping cart will make a right turn to avoid/ keep a distance from the obstacle (the microcontroller will activate the left side motor and deactivate the right side motor, in order to make a right turn).
ii. There is an obstacle close to the right side of the shopping cart, the shopping cart will make a left turn to keep a distance from the obstacle and evade it (the microcontroller will activate the right side motor and deactivate the left side motor, in order to make a left turn).

After passing the obstacle, the shopping cart will carry on its operation until the power supply is OFF.

3.6. Project Key Milestones

The milestones of project as shown in the following:

Table 3: Project Milestones

Progress
Selection of project topic
Preliminary research work
Analysis about the project
Preparation and submission of extended proposal
Preparation and presentation of proposal defense
Preparation of project materials list
Preparation of project materials (purchase, borrow from EE store and labs)
Circuit design and assembly
Modify the base of shopping cart and made coupling to attach motor with wheel
Programming design and writing
Performance testing and troubleshooting
Preparation and submission of reports and technical paper

3.7. Project Timeline (Gantt Chart)

Table 4: Project Gantt chart

		June				July				Aug				Sept				Oct				Nov				Dec			
Task	**Duration**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**
Preliminary Research Work	**5 Weeks**	■	■	■	■	■																							
Analysis Research	**3 Week**			■	■	■																							
Preparation of Proposal	**3 Weeks**					■	■	■																					
Preparation of Material list	**4 Weeks**							■	■	■	■																		
Preparetion of Project material	**6 Weeks**											■	■	■	■	■	■												
Circuit Design	**2 Weeks**															■	■												
Hardware Assembly and Construction	**4 Weeks**															■	■	■	■										
Programming Design and Writing	**5 Weeks**																	■	■	■	■	■							
Performance Testing and Troubleshooting	**4 Weeks**																					■	■	■	■				
Project Completed	**1 Week**																									■			
ELECTREX	**1 Week**																									■			
Technical Paper and Final Report	**1 Week**																											■	
Viva	**1 Week**																												■

3.8. Summary of Project Progress & Activities

The progress of project is divided into 4 levels as shown in the following:

i. 1st level (FYP 1)

- Research and development for the project.
- Gathering of informative source about the topic from multiple sources, such as books, journals, internet and also magazines.
- Analysis about the project and at the same time search for solutions for this project.
- Held meetings with supervisor to get the guidelines so the project could be implemented according to the planning.

ii. 2nd level (FYP 1)

- Finalization of the proposal.
- Held discussions with supervisor about the materials preparations
- Preparation of material list to obtain the needed material.
- Understanding about the hardware development.

iii. 3rd level (FYP 2)

- Collecting project materials.
- Hardware designing.
- Hardware assembly and construction.
- Source code design and writing.
- Integration of the developing hardware and software.

iv. Final level (FYP 2)

- Performance testing of developed prototype.
- Locate, analyse and troubleshooting if there is any problem occurred during the performance testing.
- Finally, the smart shopping cart project is complete and functioning, and then final report preparation.

CHAPTER 4

RESULT AND DISCUSSION

4.1. Result/ Experiment

4.1.1. Ultrasonic Transmitter Circuit

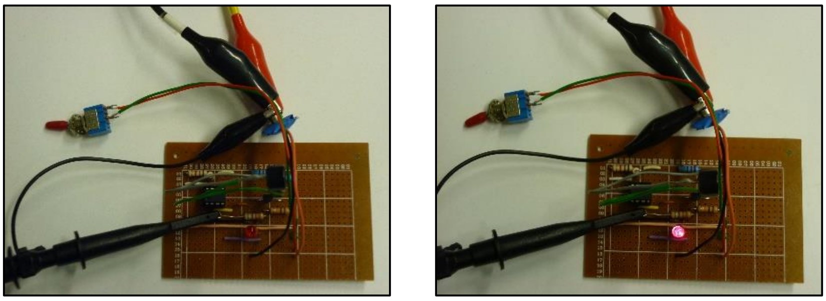


Figure 21: Ultrasonic Transmitter Circuit OFF (Left); Ultrasonic Transmitter Circuit ON (Right)

This experiment is to verify the functioning of the transmitter circuit. The figure 21 shows an LED indicator that indicates the power supply of the circuit. When the circuit is ON, the 555 timer will operate and starts transmitting signals. Figure 22 shows the output waveform of the circuit under different conditions. The leftmost figure is the output waveform of the 555 timer in OFF mode; the second left figure is the waveform of output of 555 timer in ON mode; the second right figure is the waveform of the positive (+ve) terminal of transmitter unit; the rightmost figure is the waveform of the negative (-ve) terminal of transmitter unit.

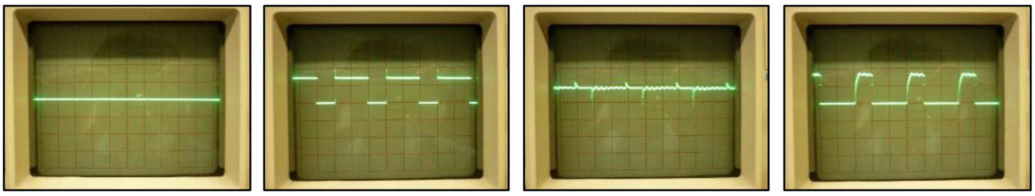


Figure 22: Output Waveform of the Transmitter Circuit

4.1.2. Ultrasonic Receiver Circuit

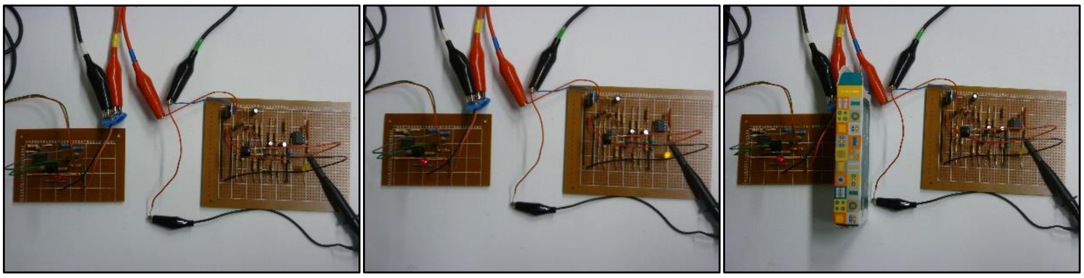


Figure 23: Experiment of Ultrasonic Receiver Circuit

Several experiments has been performed to verify the capability of the receiver circuit. The figures above shows the procedure of this experiment. Figure 23 (left most) shows that, the LED in the receiver circuit is OFF, as the transmitter circuit is in OFF mode, therefore no signal is transmitted, and the receiver circuit will not receives any signal; The middle figure shows the transmitter circuit is in ON mode, there is signal transmitting, and the receiver circuit is receiving the signal, so the LED in the receiver circuit is ON; The right figure shows the transmitter circuit is in ON mode and transmitting signal, but there is an obstacle blocking the transmitted signal and causes the receiver circuit could not receives any signal, therefore the LED in the receiver circuit is OFF. Figure 24 (left) shows the output waveform of the circuit when receiving signal; right figure shows the output waveform of the circuit when there is no signal is received (either there is no signal transmit from transmitter circuit or the transmitted signal has been blocked by the obstacle).

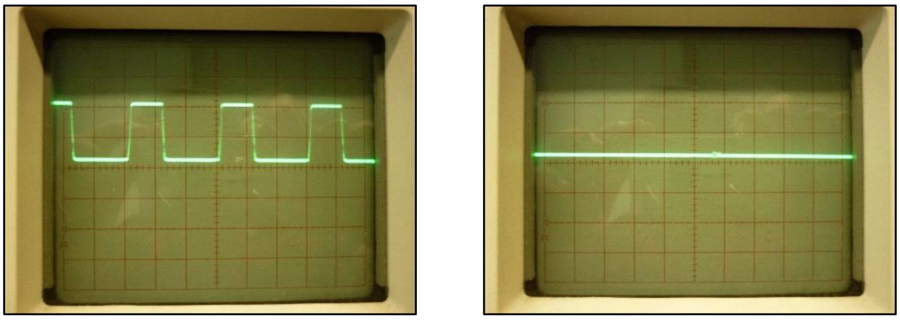


Figure 24: Output Waveform of the Receiver Circuit

4.1.3. Human Tracking Experiment without Obstacle Between the Shopping Cart and the Target User

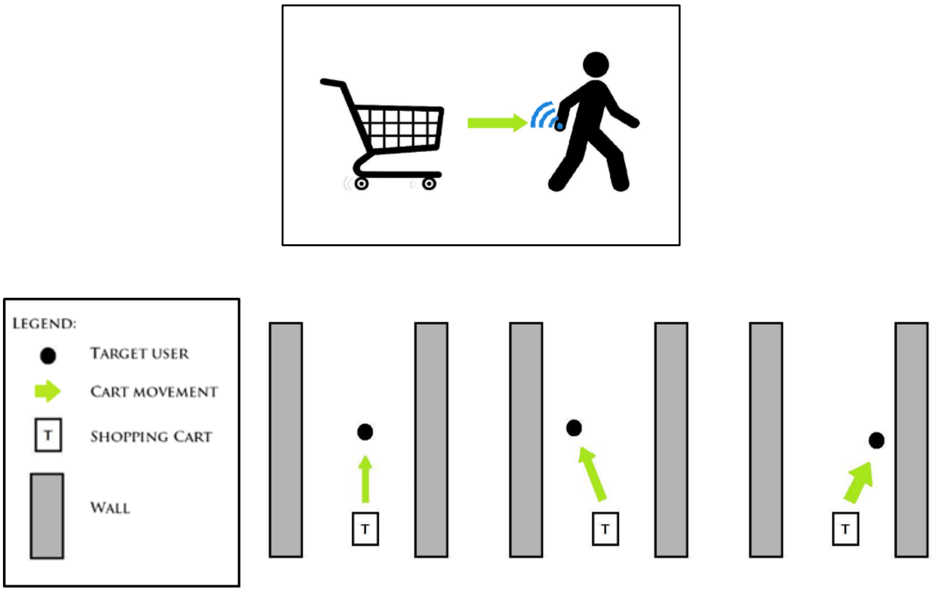


Figure 25: Experiments of Smart Shopping Cart in Obstacle Free Situation

In this section, there are some experiments implemented to verify the ability of the human tracking system in an obstacle free situation. The figures above show the procedure of this experiment. The shopping cart attempts to follow when the target user is moving, and finally keeps itself within a certain range when the target user stops. The shopping cart will stop and stay in the place, if the target user is out of the detectable range. According to the result of this fundamental test, the completeness of the normal human tracking ability performed by the shopping cart is ensured.

4.1.4. Obstacle Recognize/Detection Experiment

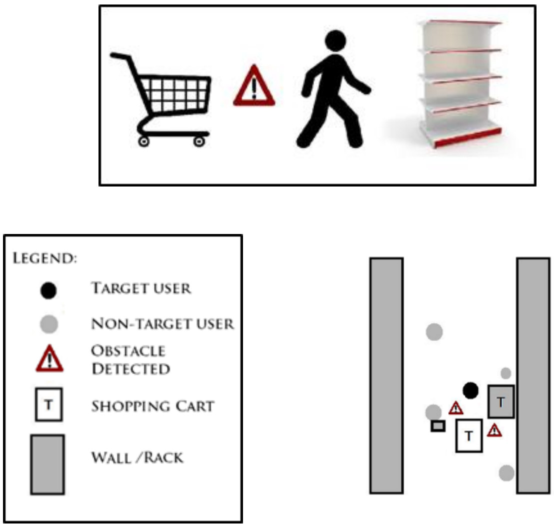


Figure 26: Obstacle Detection Experiment of Smart Shopping Cart

In this section, the experiments were carried out to verify the ability of the shopping cart to implement obstacle recognise/detection. The figure above shows the procedure of this experiment. The experiment shown that the shopping cart will recognised all of the objects where nearby it as obstacle. According to the result of this fundamental test, the completeness of the obstacle detection ability performed by the shopping cart is ensured.

4.1.5. Human Tracking Experiment with Obstacle Between the Shopping Cart and the Target User

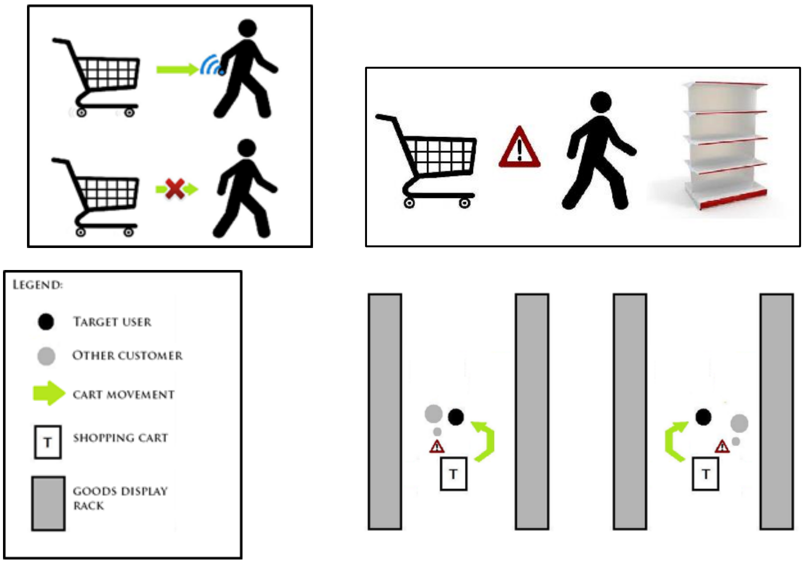


Figure 27: Experiment with Obstacle at the Side of the Front

In this section, the experiments were carried out to verify the ability of the shopping cart to implement obstacle avoidance during human tracking. The first experiment corresponding to Figure 27 is to test the behaviour of shopping cart when the shopping cart encounters an obstacle at the side of the front (left front or right front) during human tracking. When the shopping cart detects the obstacle, it will makes an opposite turn where the obstacle is, until the obstacle is out of the detectable range, and then turn towards to the target user, if the target user is in the detectable range. Otherwise, the shopping cart will stop and stay in the place if the target user is not in the detectable range.

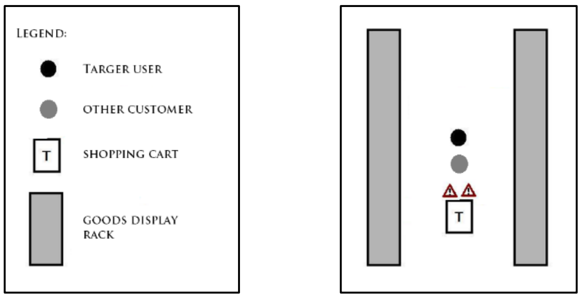


Figure 28: Experiments with Obstacle at the Both Sides of the Front

The next experiment corresponding to Figure 28 is the obstacle is at the both sides of the front in the human tracking task. The experiment states that the shopping cart will stop and stay in the place unless the obstacle has been remove.

4.1.6. Smart Shopping Cart (Integrated System)

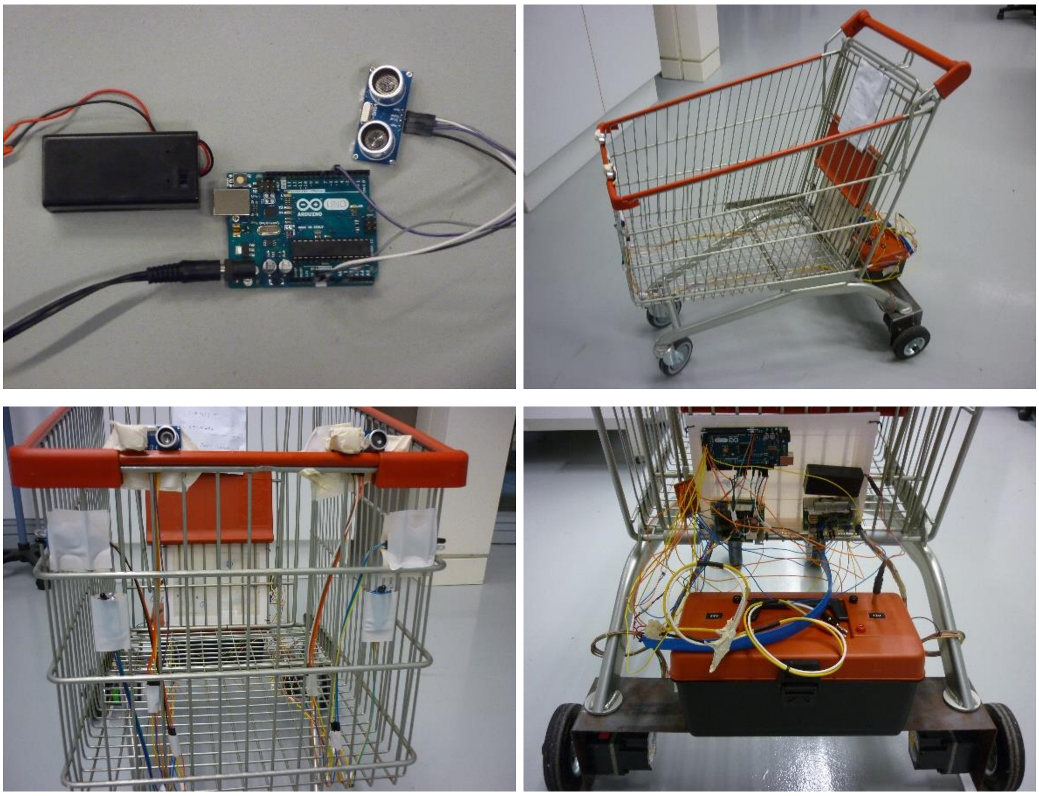


Figure 29: Actual Design of Smart Shopping Cart

Figure 29 shows the actual design of the smart shopping cart. The ultrasonic transmitter tag (first row left of Figure 29); the overview of the smart shopping cart (first row right of Figure 29); the ultrasonic receivers and 4 infrared sensors are mounted on the front of shopping cart (second row left of Figure 29); and the batteries, motor drivers and controller (Arduino Mega 2560) are mounted at the back of shopping cart (second row right of Figure 29).

At initial stage, when there is no power supplied, the shopping cart is staying in place, and both motors are in freewheel mode.

When there is power supplied, the ultrasonic transmitter starts to transmit signal, and the ultrasonic sensor (receiver) from shopping cart will then start to detect the signal from the user’s tag (ultrasonic transmitter). When the target user is within the detectable range (approx. 60cm), the sensor will send the signal to the microcontroller, and the microcontrollers will activate the motors to move the trolley towards the user; otherwise, the microcontrollers will remain idle and the shopping trolley will stay in place. The result is shown in the following table:

Table 5: Output Response of Shopping Cart

Transmitter's Position	Receiver's Output (Left)	Receiver's Output (Right)	Motor's Condition (Left)	Motor's Condition (Right)	Cart's Movement
Left	High (1)	Low (0)	Brake	Rotate CW	Turn Left
Right	Low (0)	High (1)	Rotate CCW	Brake	Turn Right
Centre	High (1)	High (1)	Rotate CCW	Rotate CW	Move Forward
Missing	Low (0)	Low (0)	Freewheel	Freewheel	Stop

*CW = Clockwise; CCW = Counter-Clockwise

The Infrared (IR) sensor will identify any objects within the distance range of the shopping cart, if there is an object close to the shopping cart, the sensor will communicates with microcontroller, and the microcontroller will collect the other IR sensor signal to respond and evade the object.

Results:

i. Whenever there is an obstacle close to the left side of the shopping cart, the shopping cart will make a right turn to avoid/ keep a distance from the obstacle (the microcontroller will activate the left side motor and deactivate the right side motor, in order to make a right turn).

ii. Whenever there is an obstacle close to the right side of the shopping cart, the shopping cart will make a left turn to keep a distance from the obstacle and evade it (the microcontroller will activate the right side motor and deactivate the left side motor, in order to make a left turn).

The shopping cart will carry on its operation until the power supply is OFF.

4.2. Discussion

The smart shopping cart has been tested under two conditions:

i. When the target user takes a small turn - Passed

   The shopping cart was able to follow the target user as the "User Detector" (ultrasonic sensors) was able to still locate the user.

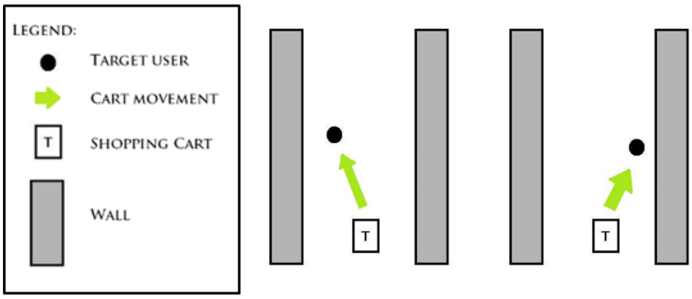


Figure 30: Small Turn

ii. When the target user takes a turn on a corner – Pending

   When the target user takes a turn on an obstacle/wall, the shopping cart needs to be implemented with "Automated User Finding" program/function to automatically find back the target user. In order to implement such program, the "User Detector" which depends on the ultrasonic device was not versatile enough to detect the user for it has exited it line of detection which results in the system being unable to further locate the user. With this, the "Automated User Finding" program could not be implemented thus this condition is yet to be achieved.

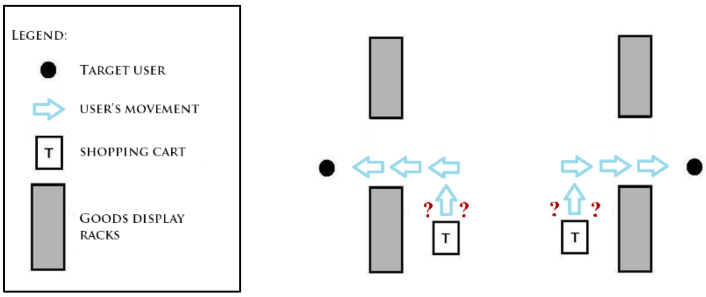


Figure 31: Corner Turn

Possible solution: A better type of ultrasonic sensors should be replaced with the current sensors to increase the range of detection. This will allow the "Automated User Finding" program to execute and continue finding the user.

However, as the results/experiments shown, the smart shopping cart has been motorized and also automatize. This results the need for human intervention to operate with the shopping cart is reducing, it provides the customer hand-free shopping experience during shopping which is the experience that the normal shopping cart cannot be provide to the customer.

Besides that, the safety of smart shopping cart has been ensured. This shopping cart will provide safer operating compare to the normal shopping cart, as it is able to avoid the obstacle when the obstacle is at the side of the front of it, and also stop when there is obstacle at the both sides of the front of it.

There is a big challenge of creating a human tracking system in the project. There are two main issues associated with this challenge.

The first issue is to equip the system with proper sensory devices so that it is able to identify and locate the target person in real time. Various approaches have been investigated and also studied by many researchers in literature, including infrared sensors, ultrasonic sensors, vision, and other approaches. The second issue is to control and navigate the shopping cart so that it follows the target user within a certain distance. This seems simple, but in reality it is a fairly difficult task.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

5.1. Conclusion

The main objective of this project to design and develop a prototype of an automated motorized shopping cart, capable of trailing shoppers along with the extensive ability to avoid obstacles with its built-in intelligence.

In order to accomplish this objective, the following goals have been achieved in this project:

1. Created the interface between the ultrasonic positioning system and the control system in the operating program.
2. Developed the system that able to simultaneously avoid obstacles and track the targeted user.
3. Completed human tracking experiment when there is no obstacle between the shopping cart and the target user.
4. Completed human tracking experiment when there is obstacle between the shopping cart and the target user.

According to the results of several experiments in previous chapter (Chapter 4), the goals have been reached. Since all the goals have been achieved, the feasibility of the main idea in this project was verified, which is the smart shopping cart is able to perform human tracking and obstacle avoidance tasks simultaneously.

5.2. Recommendation/ Future Work

Since the feasibility of developing a human tracking system using an ultrasonic positioning system has been ensured, but there are several limitations with this shopping cart.

The circuitry involving transmitter and receiver as smart shopping carts sensory system is a straight forward method for human tracking is bound to unforeseen limitations. It might not be the most suitable circuitries when there exists large obstacles between the target user and the shopping cart. This might be due to the fact that the transmitted signal will be fully blocked by the obstacle.

Additionally, the effective target bearing for the system to detect the target signal has limited range. Thus, the system may fail to track the target if the target person intentionally moves out of this range. In other words, whenever the target user moves away from the line of sight of the shopping cart, the shopping cart unable receive the signal. Thus, the next step of improvement in terms of efficiency of this project is the evaluation of replacing better ultrasonic sensors or other types of sensors.

In order to improve the efficiency of the human tracking system, there are many types of approach, such as Radio Frequency (RF), Bluetooth, vision based approach, and others. In recommendation works, perhaps the combination of these human tracking system may generate more efficient outcomes.

Other than that, the smart shopping cart needs to be implemented with "Automated User Finding" program/function to automatically find back the target user, when the target user is not in the detectable range after takes a corner turn or blocking by a big obstacle.

It is possible to add other features/functions to the shopping cart to make it smarter. For example, additional of function of "items finder" to the system shall be introduced, so that the shopping cart could guide the customer to find the desired items in the shopping mall.

Additionally, the queue-less payment can be installed onto the shopping cart where it can calculate the bill of shopping goods after the customer place the goods into the shopping cart. This can enhance the time of shopping and the features of the smart shopping cart. By adding these features, the shopping cart will be “smarter” and more convenient for the customer to use.

REFERENCES

[1] Y. Nagumo and A. Ohya, "Human following behavior of an autonomous mobile robot using light-emitting device," in *Robot and Human Interactive Communication, 2001. Proceedings. 10th IEEE International Workshop on*, 2001, pp. 225-230.

[2] T. Yoshimi, M. Nishiyama, T. Sonoura, H. Nakamoto, S. Tokura, H. Sato*, et al.*, "Development of a Person Following Robot with Vision Based Target Detection," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, 2006, pp. 5286-5291.

[3] T. Kohtsuka, T. Onozato, H. Tamura, S. Katayama, and Y. Kambayashi, "Design of a Control System for Robot Shopping Carts," in *Knowledge-Based and Intelligent Information and Engineering Systems*. vol. 6881, A. König, A. Dengel, K. Hinkelmann, K. Kise, R. Howlett, and L. Jain, Eds., ed: Springer Berlin Heidelberg, 2011, pp. 280-288.

[4] D. Schulz, W. Burgard, D. Fox, and A. B. Cremers, "Tracking multiple moving objects with a mobile robot," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, 2001, pp. I-371-I-377 vol.1.

[5] F. Hoshino and K. Morioka, "Human following robot based on control of particle distribution with integrated range sensors," in *System Integration (SII), 2011 IEEE/SICE International Symposium on*, 2011, pp. 212-217.

[6] M. Bigos, J. Doiron-LaRue, R. Mui, E. Wu. (2011). *Final Project Report: Cart-A-Long.* [Online]. Available: http://www.ecs.umass.edu/ece/sdp/sdp11/pishro-nik/FPR.pdf

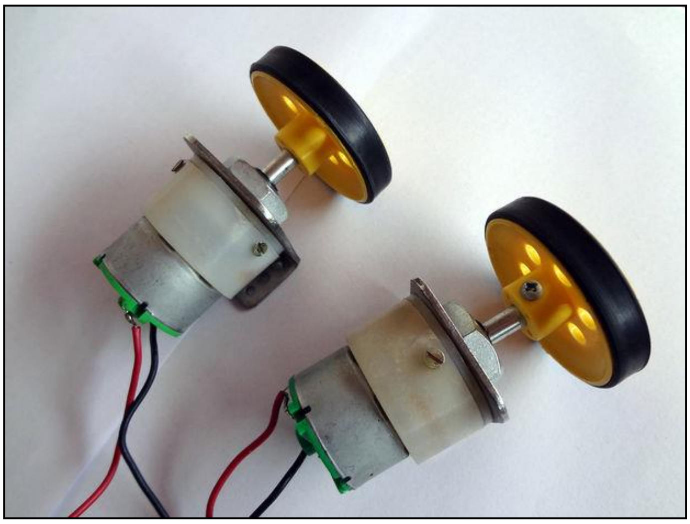
[7] MAE Design and Manufacturing Laboratory. (n.d.). *Drive Wheel Motor Torque Calculation.* [Online]. Available: http://www2.mae.ufl.edu/designlab/motors/EML2322L%20Drive%20Wheel%20Motor%20Torque%20Calculations.pdf

[8] *Oriental Motor General Catalog 2003/2004. Brushless DC Motor Systems,* Oriental Motor Corp., USA. 2003. pp B60-68.

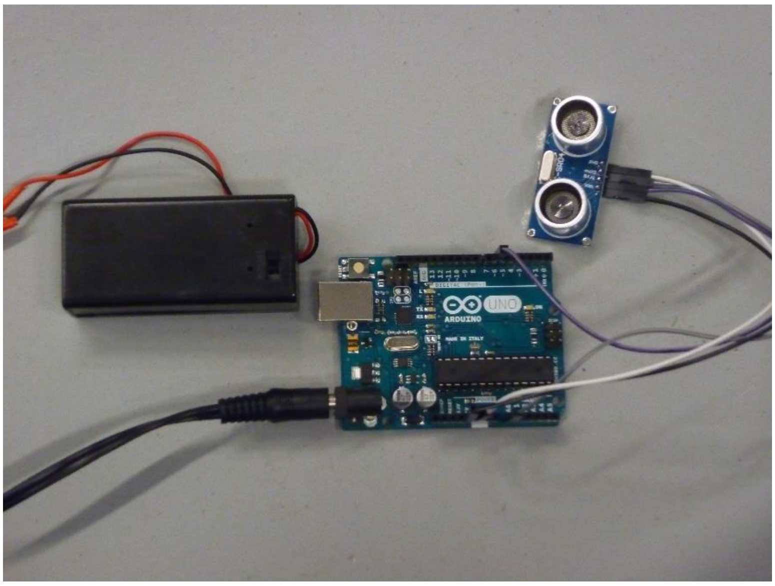
[9] P. Mitchell. (n.d.). *Ultrasonic Transmitter Receiver Circuit Ideas.* [Online]. Available: http://www.electroniclessons.com/Rangefinderkit.doc

APPENDICES

Appendix 1: Design Concept



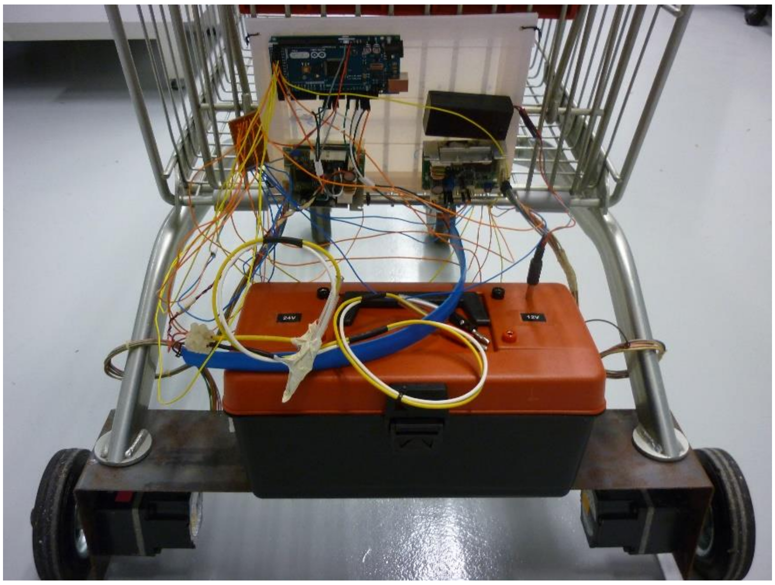
Appendix 2: Actual Project Design



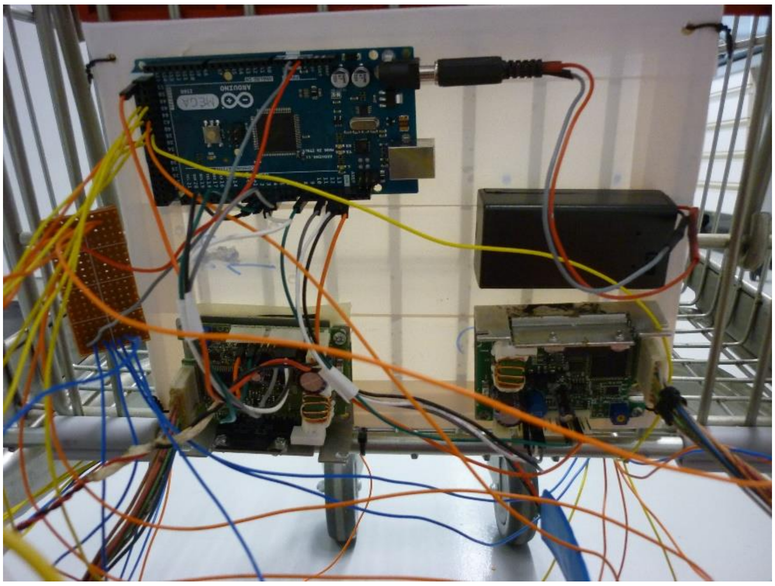
Transmitter Tag

Overview of Smart Shopping Cart

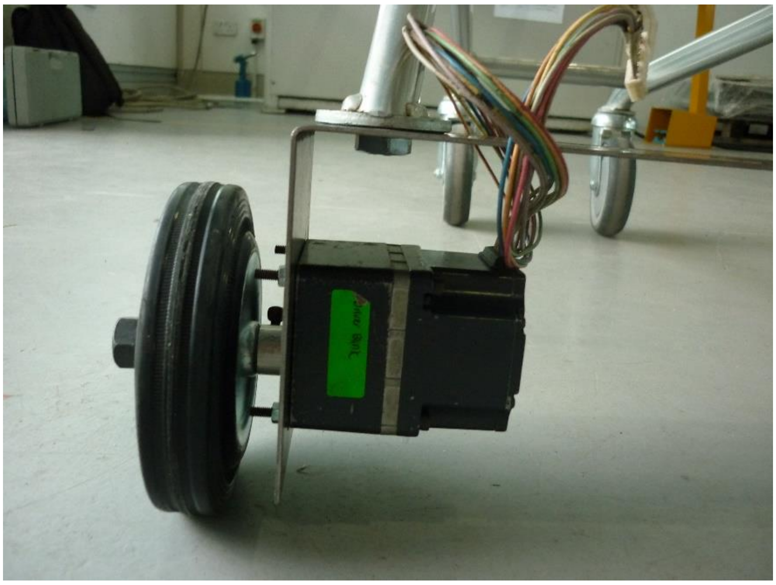
Front View of Smart Shopping Cart



Back View of Smart Shopping Cart

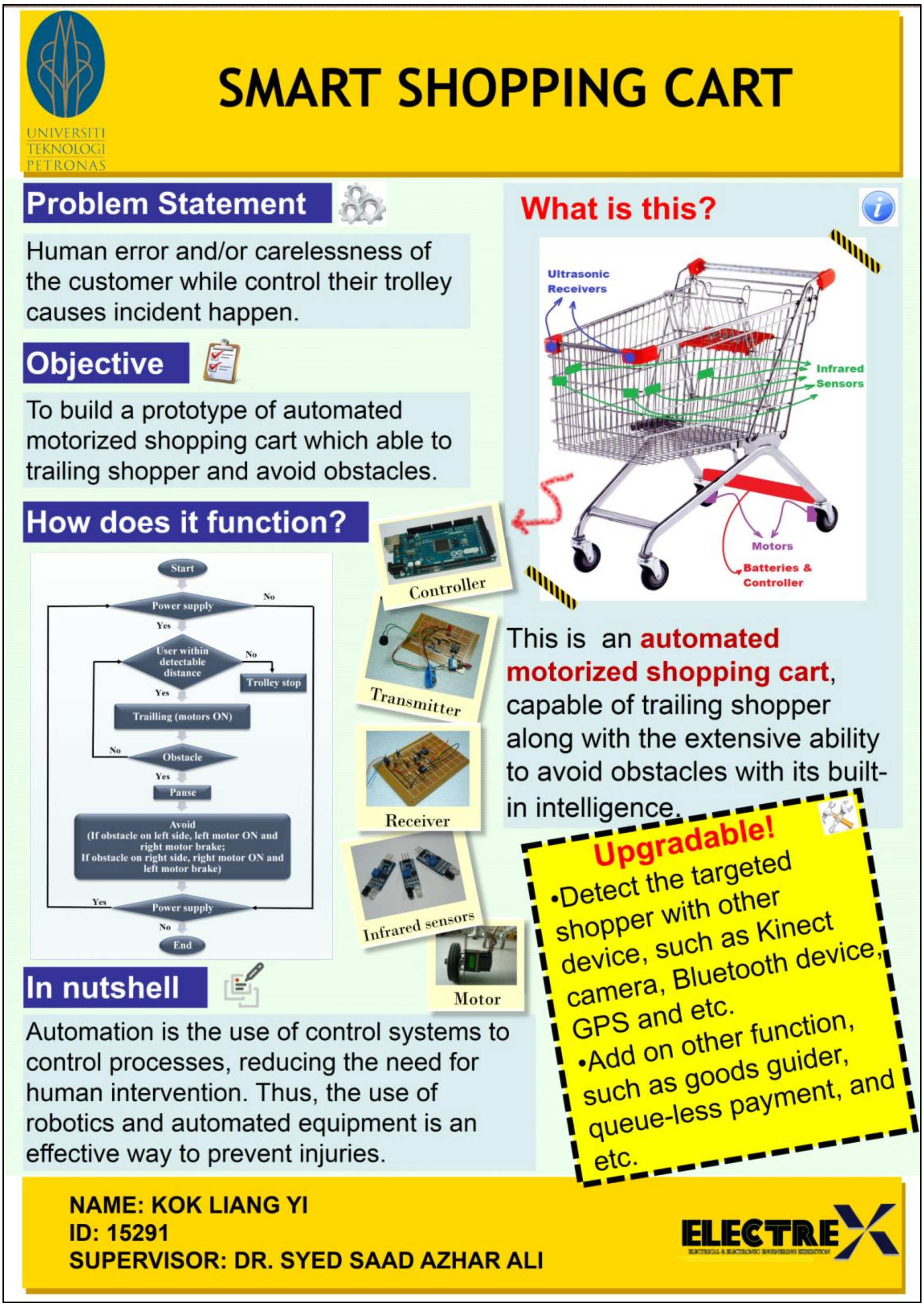


Close View of the Control System Components/Modules



Close View of Mounted Motor

Appendix 3: ELECTREX Poster



Appendix 4: Specification of VEXTA AXH Series Brushless DC Motor Systems [8]

Specifications

Model		Geared Type/Combination Type	AXH015K-□	AXH230KC-□	AXH450KC-□	AXH5100KC-□
		Round Shaft Type	AXH015K-A	AXH230KC-A	AXH450KC-A	AXH5100KC-A
Rated Output Power		HP (W)	1/50 (15)	1/25 (30)	1/15 (50)	1/8 (100)
Power Source	Voltage		24 VDC ±10%			
	Rated Input Current	A	1.0	2.1	3.1	6.0
	Maximum Input Current	A	2.0	3.5	5.0	9.0
Rated Torque		oz-in (N·m)	7.1 (0.05)	17 (0.12)	28 (0.20)	56 (0.40)
Starting Torque		oz-in (N·m)	10.6 (0.075)	21 (0.15)	34 (0.24)	71 (0.50)
Permissible Load Inertia J *		$oz\text{-}in^2$ ($\times 10^{-4}$ $kg \cdot m^2$)	2.7 (0.5)	9.8 (1.8)	18.1 (3.3)	31 (5.6)
Maximum Speed		r/min	3000			
Rated Speed		r/min	3000	2500		
Variable Speed Range		r/min	100~3000 (30:1)			
Speed Regulation	Load		±1% Max. (0~rated torque, at rated speed)			
	Voltage		±1% Max. (Power supply voltage ±10%, at rated speed with no load)			
	Temperature		±1% Max. (32°F~122°F [0°C~+50°C] at rated speed with no load)			

* The permissible load inertia specified above is only applicable for round shaft type. Permissible Load Inertia for Geared Type and Combination Type → Page B-61
- Enter the gear ratio in the box (□) with the model name.
- The values for each item is for the motor only.

Common Specifications

Item	Specifications		
Speed Control Method	Any one of the following methods. 1. By built-in potentiometer 2. By external potentiometer 3. By DC voltage (0~5 VDC)		
Input Signals	C-MOS negative logic	L: (ON) : 0~0.5 VDC	H: (OFF) : 4~5 VDC
	START/STOP input	L: START	H: STOP
	Brake input	L: RUN	H: Instantaneous stop
	Direction of rotation input	L: CW	H: CCW
	Speed setting method	L: Internal	H: External
	Alarm reset	L: Reset	H: Normal
Output Signals	Open collector output External use conditions 26.4 VDC, 10 mA Max. Speed Signal Output (SPEED OUT) 30 P/R, Alarm Signal Output (ALARM OUT)		
Protection Functions *1	When the following are activated, the alarm signal will be output and the motor will come to a natural stop. ● Overload Protection: Activated when a load exceeding the rated torque is applied to the motor for approximately 5 seconds or more. ● Out-of-Phase Protection: Activated when the sensor wire inside the motor cable is disconnected. ● Overvoltage Protection: Activated when the voltage applied to the driver exceeds 24 VDC by approximately 15% or more. ● Undervoltage Protection: Activated when the voltage applied to the driver falls at least 25% below 24 VDC. ● Over Speed Protection: Activated when the motor rotates at an abnormal speed above 3500 r/min.		
Motor Insulation Class *2	Class E [248°F (120°C)]		
Rating	Continuous		

*1 With the **AXH** Series the motor speed cannot be controlled in applications where the motor shaft is turned by the load, as in lowering operations.
Also, the motor will stop naturally if the load exceeds the permissible load inertia or the overvoltage protection function is activated during load lowering operations.
*2 Motor insulation is recognized as class A [221°F(105°C)] by UL and CSA standards.

General Specifications

Item	Motor	Driver
Insulation Resistance	100 MΩ or more when 500 VDC megger is applied between the windings and the frame after continuous operation under normal ambient temperature and humidity.	100 MΩ or more when 500 VDC megger is applied between the power supply input and the frame after continuous operation under normal ambient temperature and humidity.
Dielectric Strength	Sufficient to withstand 0.5 kVAC at 50 Hz applied between the windings and the frame for 1 minute after continuous operation under normal ambient temperature and humidity.	Sufficient to withstand 0.5 kVAC at 50 Hz applied between the power supply input and the frame for 1 minute after continuous operation under normal ambient temperature and humidity.
Temperature Rise	90°F (50°C) or less measured by the thermocoupler method after the temperature of the coil has stabilized under normal operation at the rated voltage and frequency under normal ambient temperature and humidity, with a connected gearhead or equivalent heat radiation plate.*	—
Ambient Temperature	32°F~122°F (0°C~+50°C) (nonfreezing)	
Ambient Humidity	85% maximum (noncondensing)	
Atmosphere	No corrosive gases or dust	
Degree of Protection	15W Type: IP 40 30W~100W Type: IP65 (except for the mounting surface)	IP 00

* Size of heat radiation plate (Material: Aluminum)
AXH230KC-A: 4.53 in. × 4.53 in. (115 mm × 115 mm), 0.20 in. (5 mm) thick
AXH450KC-A: 5.31 in. × 5.31 in. (135 mm × 135 mm), 0.20 in. (5 mm) thick
AXH5100KC-A: 7.87 in. × 7.87 in.(200 mm × 200 mm), 0.20 in. (5 mm) thick

Features B-58 | System Configuration B-59 | Specifications B-60 | Characteristics B-62

Gearmotor — Torque Table (Geared Type/Combination Type)

Unit = Upper values: lb-in/Lower values: N·m

Model	Speed Range * r/min	20~500 (20~600)	10~250 (10~300)	6.7~167 (6.7~200)	5~125 (5~150)	3.3~83 (3.3~100)	2~50 (2~60)	1~25 (1~30)	0.5~12.5
	Gear Ratio	**5**	**10**	**15**	**20**	**30**	**50**	**100**	**200**
AXH015K-□		2.0 0.23	3.9 0.45	6.0 0.68	7.6 0.86	11.5 1.3	17.7 2.0	17.7 2.0	—
AXH230KC-□		4.7 0.54	9.7 1.1	14.1 1.6	19.4 2.2	27 3.1	46 5.2	53 6.0	53 6.0
AXH450KC-□		7.9 0.9	15.9 1.8	23 2.7	31 3.6	46 5.2	76 8.6	141 16	141 16
AXH5100KC-□		15.9 1.8	31 3.6	47 5.4	63 7.2	91 10.3	152 17.2	260 30	260 30

- Enter the gear ratio in the box (□) within the model name.
- A colored background indicates gear shaft rotation in the same direction as the motor shaft; a white background indicates rotation in the opposite direction.

* Values inside parentheses ( ) are for the **AXH015K-□** model.

Permissible Overhung Load and Permissible Thrust Load

Geared Type/Combination Type

Model	Gear Ratio	Permissible Overhung Load				Permissible Thrust Load	
		0.39 in. (10 mm) from shaft end		0.79 in. (20 mm) from shaft end			
		lb.	N	lb.	N	lb.	N
AXH015K-□	**5~100**	11.2	50	—	—	6.7	30
AXH230KC-□	**5**	22	100	33	150	9	40
	10~20	33	150	45	200		
	30~200	45	200	67	300		
AXH450KC-□	**5**	45	200	56	250	22	100
	10~20	67	300	78	350		
	30~200	101	450	123	550		
AXH5100KC-□	**5**	67	300	90	400	33	150
	10~20	90	400	112	500		
	30~200	112	500	146	650		

- Enter the gear ratio in the box (□) within the model name.

Round Shaft Type

Model	Permissible Overhung Load			
	0.39 in. (10mm) from shaft end		0.79 in. (20 mm) from shaft end	
	lb.	N	lb.	N
AXH015K-A	11.2	50	—	—
AXH230KC-A	15.7	70	22	100
AXH450KC-A	27	120	31	140
AXH5100KC-A	36	160	38	170

- Permissible Thrust Load: Avoid thrust loads as much as possible. If thrust load is unavoidable, keep it to no more than half the motor weight.

Permissible Load Inertia J for Geared Type/Combination Type

Unit = Upper values: oz-in²/Lower values: ×10⁻⁴ kg·m²

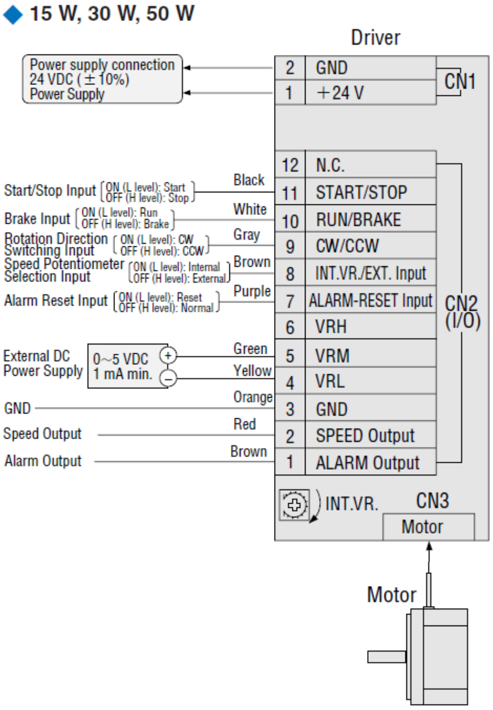
Model \ Gear Ratio	**5**	**10**	**15**	**20**	**30**	**50**	**100**	**200**
AXH015K-□	2.2 0.4	9.3 1.7	21 3.9	38 7.0	86 15.7	240 43.7	240 43.7	—
AXH230KC-□	8.5 1.55	34 6.2	77 14.0	136 24.8	310 55.8	850 155	850 155	850 155
AXH450KC-□	30 5.5	120 22	270 49.5	480 88	1080 198	3000 550	3000 550	3000 550
AXH5100KC-□	137 25	547 100	1230 225	2188 400	4923 900	13675 2500	13675 2500	13675 2500

- Enter the gear ratio in the box (□) within the model name.

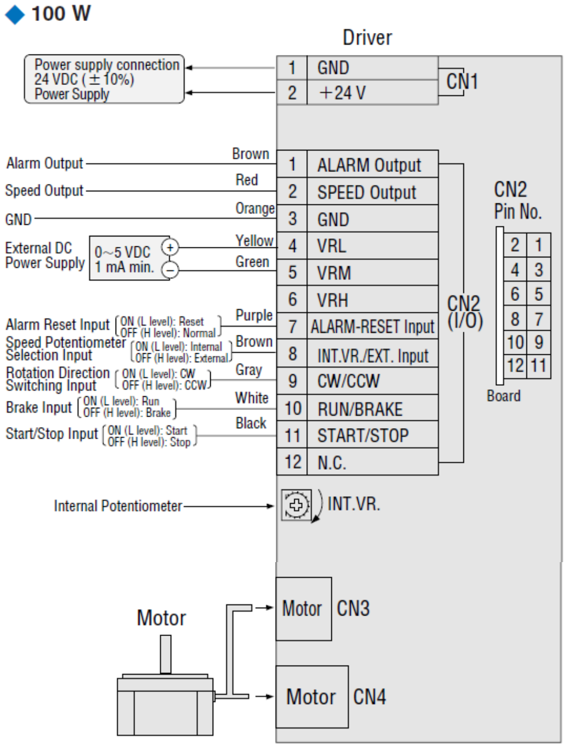
Connection and Operation

Connection Diagrams

15 W, 30 W, 50 W

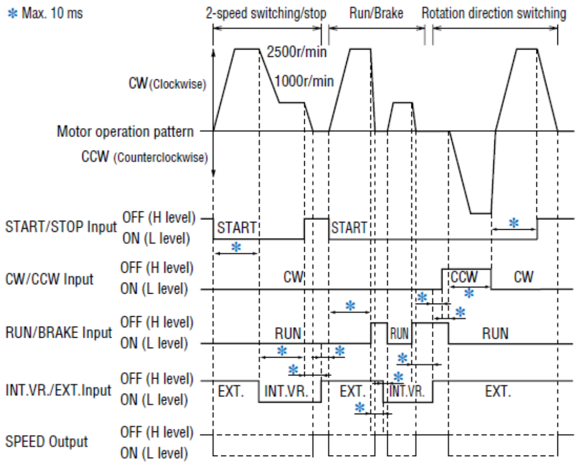


100 W



● When the motor cable needs to be extended, use an optional extension cable [sold separately, 4.9 ft. (1.5 m)].
Extension Cable→Page B-69

Timing Chart



- Run/stop, instantaneous stopping and rotation direction switching operations can all be controlled with the START/STOP, RUN/BRAKE and CW/CCW signals.
- If both the START/STOP signal and the RUN/BRAKE signal are set to ON (L level), the motor rotates. At this time, if the CW/CCW signal is set to ON (L level), then the motor rotates clockwise as seen from the motor shaft side; if the CW/CCW signal is set to OFF (H level), the motor rotates in the counterclockwise direction.
- If the RUN/BRAKE signal is set to OFF (H level) while the START/STOP signal is ON (L level), the motor stops instantaneously. If the START/STOP signal is set to OFF (H level) while the RUN/BRAKE signal is set to ON (L level), the motor stops naturally.
- Wait for 10 ms before switching the other input signals.
- Do not switch different input signals simultaneously. Wait for 10 ms before switching the other input signals.

Appendix 5: Arduino Project Programming Code (Transmitter part)

```
#include <NewPing.h>

#define SONAR_NUM     1 // Number or sensors.
#define MAX_DISTANCE 200 // Maximum distance (in cm) to ping.
#define PING_INTERVAL 33 // Milliseconds between sensor pings (29ms is about the min to avoid cross-sensor echo).

unsigned long pingTimer[SONAR_NUM]; // Holds the times when the next ping should happen for each sensor.
unsigned int cm[SONAR_NUM];         // Where the ping distances are stored.
uint8_t currentSensor = 0;          // Keeps track of which sensor is active.

NewPing sonar[SONAR_NUM] = {     // Sensor object array.
  NewPing(6, A0, MAX_DISTANCE) // Each sensor's trigger pin, echo pin, and max distance to ping.
};

void setup() {
  Serial.begin(115200);
  pingTimer[0] = millis() + 75;           // First ping starts at 75ms, gives time for the Arduino to chill before starting.
  for (uint8_t i = 1; i < SONAR_NUM; i++) // Set the starting time for each sensor.
    pingTimer[i] = pingTimer[i - 1] + PING_INTERVAL;
}

void loop() {
  for (uint8_t i = 0; i < SONAR_NUM; i++) { // Loop through all the sensors.
    if (millis() >= pingTimer[i]) {         // Is it this sensor's time to ping?
      pingTimer[i] += PING_INTERVAL * SONAR_NUM;  // Set next time this sensor will be pinged.
      if (i == 0 && currentSensor == SONAR_NUM - 1) oneSensorCycle(); // Sensor ping cycle complete, do something with the results.
      sonar[currentSensor].timer_stop();          // Make sure previous timer is canceled before starting a new ping (insurance).
      currentSensor = i;                          // Sensor being accessed.
      cm[currentSensor] = 0;                      // Make distance zero in case there's no ping echo for this sensor.
      sonar[currentSensor].ping_timer(echoCheck); // Do the ping (processing continues, interrupt will call echoCheck to look for echo).
    }
  }
  // The rest of your code would go here.
}

void echoCheck() { // If ping received, set the sensor distance to array.
  if (sonar[currentSensor].check_timer())
    cm[currentSensor] = sonar[currentSensor].ping_result / US_ROUNDTRIP_CM;
}
```

```
void oneSensorCycle() { // Sensor ping cycle complete, do something with the results.
  for (uint8_t i = 0; i < SONAR_NUM; i++) {
    Serial.print(i);
    Serial.print("=");
    Serial.print(cm[i]);
    Serial.print("cm ");
  }
  Serial.println();
}
```

Appendix 6: Arduino Project Programming Code (Shopping cart part)

```
#include <NewPing.h>

#define SONAR_NUM     2 // Number or sensors.
#define MAX_DISTANCE 200 // Maximum distance (in cm) to ping.
#define PING_INTERVAL 33 // Milliseconds between sensor pings (29ms is about the min to avoid cross-sensor echo).

unsigned long pingTimer[SONAR_NUM]; // Holds the times when the next ping should happen for each sensor.
unsigned int cm[SONAR_NUM];         // Where the ping distances are stored.
uint8_t currentSensor = 0;          // Keeps track of which sensor is active.


 int sensorValueUL=-1 ;      //Left ultrasonic receiver's status
 int sensorValueUR=-1;      //Right ultrasonic receiver's status
 int sensorValueUL_a=0 ;      //Left ultrasonic receiver's status
 int sensorValueUR_a=0 ;      //Right ultrasonic receiver's status

 int motorL1=2;               //pin-11 (Start/Stop)of Left Motor Driver is connected to Arduino digital pin2
 int motorL2=4;              //pin-10 (Run/Brake) of Left Motor Driver is connected to Arduino digital pin4
 int motorL3=5;               //pin-9 (CW/CCW) of Left Motor Driver is connected to Arduino digital pin5
 int motorL5=3;            //pin-5 (VRM) of Left Motor Driver is connected to Arduino digital pin3
 int motorR1=13;            //pin-11 (Start/Stop)of Right Motor Driver is connected to Arduino digital pin13
 int motorR2=12;           //pin-10 (Run/Brake) of Right Motor Driver is connected to Arduino digital pin12
 int motorR3=11;            //pin-9 (CW/CCW) of Right Motor Driver is connected to Arduino digital pin11
 int motorR5=8;          //pin-5 (VRM) of Right Motor Driver is connected to Arduino digital pin10
 int sensorL1=48;       //Front Left IR sensor is connected to Arduino digital pin48
 int sensorValueL1;
 int sensorL2=39;       //Left IR sensor is connected to Arduino digital pin39
 int sensorValueL2;
 int sensorR1=49;       //Front Right IR sensor is connected to Arduino digital pin49
 int sensorValueR1;
 int sensorR2=40;       //Right IR sensor is connected to Arduino digital pin40
 int sensorValueR2;
 int ground=24;
 int a;


NewPing sonar[SONAR_NUM] = {     // Sensor object array.
```

```
  NewPing(41, 42, MAX_DISTANCE), // Each sensor's trigger pin, echo pin, and max
distance to ping.
  NewPing(44, 45, MAX_DISTANCE)
};


 void setup()
 {
  pinMode(motorL1,OUTPUT);
  pinMode(motorL2,OUTPUT);
  pinMode(motorL3,OUTPUT);
  pinMode(motorL5,OUTPUT);
  pinMode(motorR1,OUTPUT);
  pinMode(motorR2,OUTPUT);
  pinMode(motorR3,OUTPUT);
  pinMode(motorR5,OUTPUT);
  pinMode(ground,OUTPUT);
  digitalWrite(ground,LOW);
  pinMode(sensorL1,INPUT);
  pinMode(sensorL2,INPUT);
  pinMode(sensorR1,INPUT);
  pinMode(sensorR2,INPUT);

  Serial.begin(115200);
  Serial.begin(115200);
  pingTimer[0] = millis() + 75;           // First ping starts at 75ms, gives time for the
Arduino to chill before starting.

   freewheel();
}
int count_hl=0,count_hr=0,count_ll=0,count_lr=0;
void oneSensorCycle() { // Sensor ping cycle complete, do something with the results.


  if(cm[0]  <= 105 && cm[0] != 0){ count_hl++;count_ll=0;}
   else{ count_ll++;count_hl=0;}

if(count_hl>=2){sensorValueUL = 1;}

if(count_ll>=7){sensorValueUL = 0;}


  if(cm[1] <= 105 && cm[1] != 0){ count_hr++;count_lr=0;}
  else{ count_lr++;count_hr=0;}

if(count_hr>=2){sensorValueUR = 1;}

if(count_lr>=7){sensorValueUR = 0;}

   Serial.print("1 = ");
```

```
  Serial.print("    ");
  Serial.print(sensorValueUL);
  Serial.print("    ");
  Serial.print(cm[0]);
 Serial.print("  2 = ");
 Serial.print("    ");
 Serial.print(sensorValueUR);
 Serial.print("    ");
 Serial.print(cm[1]);

 Serial.println();
}

void loop()

 {
 a=1;

   for (uint8_t i = 0; i < SONAR_NUM; i++) { // Loop through all the sensors.

     if (millis() >= pingTimer[i])
       {
       pingTimer[i] += PING_INTERVAL * SONAR_NUM;   // Set next time this sensor will be pinged.
      if (i == 0 && currentSensor == SONAR_NUM - 1) {oneSensorCycle();}

       sonar[currentSensor].timer_stop();          // Make sure previous timer is canceled before starting a new ping (insurance).
       currentSensor = i;                          // Sensor being accessed.
       cm[currentSensor] = 0;                      // Make distance zero in case there's no ping echo for this sensor.

       sonar[currentSensor].ping_timer(echoCheck); // Do the ping (processing continues, interrupt will call echoCheck to look for echo).
       }


 sensorValueL1 = digitalRead(sensorL1);
 sensorValueL2 = digitalRead(sensorL2);
 sensorValueR1 = digitalRead(sensorR1);
 sensorValueR2 = digitalRead(sensorR2);

  a=1;

 // Low = Missing
 if (sensorValueUL == 1 && sensorValueUR == 1)      // at mid
 {
   // High = missing
   if (sensorValueL1 == HIGH && sensorValueR1 == HIGH)  // path clear
   {
```

```
    forward();
   }
  else if (sensorValueL1 == LOW && sensorValueR1 == HIGH)  // left hit
  {
    if (sensorValueR2 == HIGH)
   { right();  }
  else { freewheel();}
    }
  else if (sensorValueL1 == HIGH && sensorValueR1 == LOW)  // right hit
  {
       if (sensorValueL2 == HIGH)
         { left();  }
      else { freewheel();}

  }
  else      // left n right hit OR front hit
  {
   freewheel();
  }
}


else if (sensorValueUL == 1 && sensorValueUR == 0)    // at left
{
  // High = missing
  if (sensorValueL1 == HIGH && sensorValueR1 == HIGH )  // front clear
  {
      if (sensorValueL2 == HIGH)
         { left();  }
      else { freewheel();}

  }
  else
  {
     freewheel();
  }
}

else if (sensorValueUL == 0 && sensorValueUR == 1)    // at right
{
  // High = missing
  if (sensorValueL1 == HIGH && sensorValueR1 == HIGH)  // front clear
  {
     if (sensorValueR2 == HIGH)
     { right();  }
    else { freewheel();}
  }
  else
  {
     freewheel();
```

```
    }
  }
  else if(sensorValueUL == 0 && sensorValueUR == 0)    // MISSING
  {
     freewheel();
  }

 }

 void forward()
 {
  digitalWrite(motorL1,LOW);        //
  digitalWrite(motorL2,LOW);        //  Move CW
  digitalWrite(motorL3,LOW);        //
  analogWrite(motorL5,200);         //

  digitalWrite(motorR1,LOW);        //
  digitalWrite(motorR2,LOW);        //  Move CCW
  digitalWrite(motorR3,HIGH);       //
  analogWrite(motorR5,90);          //
 }

 void left()
 {
  digitalWrite(motorL1,LOW);      //
  digitalWrite(motorL2,HIGH);     //  Brake
  digitalWrite(motorL3,LOW);      //
  analogWrite(motorL5,200);       //

  digitalWrite(motorR1,LOW);      //
  digitalWrite(motorR2,LOW);      //  Move CCW
  digitalWrite(motorR3,HIGH);     //
  analogWrite(motorR5,90);        //
 }

 void right()
 {
  digitalWrite(motorL1,LOW);      //
  digitalWrite(motorL2,LOW);      //  Move CW
  digitalWrite(motorL3,LOW);      //
  analogWrite(motorL5,200);       //

  digitalWrite(motorR1,LOW);      //
  digitalWrite(motorR2,HIGH);     //  Brake
  digitalWrite(motorR3,LOW);      //
  analogWrite(motorR5,90);        //
 }

void freewheel()
{
```

```
  digitalWrite(motorL1,HIGH);     //
  digitalWrite(motorL2,LOW);      //  Stop
  digitalWrite(motorL3,LOW);      //
  analogWrite(motorL5,100);       //

  digitalWrite(motorR1,HIGH);     //
  digitalWrite(motorR2,LOW);      //  Stop
  digitalWrite(motorR3,LOW);      //
  analogWrite(motorR5,100);       //
}

void brake()
{
  digitalWrite(motorL1,LOW);      //
  digitalWrite(motorL2,HIGH);     //  Brake
  digitalWrite(motorL3,LOW);      //
  analogWrite(motorL5,100);       //

  digitalWrite(motorR1,LOW);      //
  digitalWrite(motorR2,HIGH);     //  Brake
  digitalWrite(motorR3,LOW);      //
  analogWrite(motorR5,100);       //
}


void echoCheck() { // If ping received, set the sensor distance to array.
  if (sonar[currentSensor].check_timer())
   { cm[currentSensor] = sonar[currentSensor].ping_result / US_ROUNDTRIP_CM;
 }
}
```