19L620 – INNOVATION PRACTICES

HANDS FREE TROLLEY

R AKASH	**(21L401)**
R SURESHKUMAR	**(20L147)**
S A S VISHNUPRIYA	**(20L153)**
K V SANKAR SUBRAMANIUM	**(20L157)**

Project report submitted in partial fulfillment of the requirements for the degree of

BACHELOR OF ENGINEERING

Branch: ELECTRONICS AND COMMUNICATION ENGINEERING

Of Anna University

MARCH 2023

PSG COLLEGE OF TECHNOLOGY

(Autonomous Institution)

COIMBATORE– 641004.

PSG COLLEGE OF TECHNOLOGY

(Autonomous Institution)

COIMBATORE– 641004.

HANDS FREE TROLLEY

Bonafide record of work done by

R AKASH	**(21L401)**
R SURESHKUMAR	**(20L147)**
S A S VISHNUPRIYA	**(20L153)**
K V SANKAR SUBRAMANIUM	**(20L157)**

Dissertation submitted in partial fulfillment of the requirements for the course of

19L620 – Innovation Practices

BACHELOR OF ENGINEERING

Branch: ELECTRONICS AND COMMUNICATION ENGINEERING

Of Anna University

MARCH 2023

………………………… …………………………

Dr.V.Krishnaveni

(Faculty In-charge) (Head of the department)

Certified that the candidate was examined on

………………………… …………………………

(Internal Examiner) (External Examiner)

CONTENTS

ACKNOWLEDGEMENT

We would like to extend our sincere thanks to **Dr.K.Prakasan**, Principal, PSG College of Technology, for his kind patronage.

We also wish to express our sincere thanks to **Dr. V. Krishnaveni**, Professor and Head-of-the Department, Electronics and Communication Engineering.

We are thankful to **Dr. V.K. Manoharan**, Professor, Department of Electronics and Communication Engineering.

We are grateful for the support extended by our Programme Coordinator, **Dr. K. V. Anusuya**, Associate Professor (CAS), Department of Electronics and Communication Engineering.

We thank our Tutor, **Dr. S. Hemachithra**, Associate Professor, Department of Electronics and Communication Engineering.

We thank all the teaching and non-teaching staff members of the Department of Electronics and Communication Engineering for their support.

Finally, we would like to thank all my student colleagues who have helped us in completing this project without any hassles.

SYNOPSIS

Shopping is one of our regular tasks for buying the necessary things. In Supermarkets, the customer has to push the heavily loaded cart and proceed to checkout. It is also a matter of inconvenience for elderly people and physically challenged people to push the trolley manually. So many have started to purchase online which is much easier and more flexible. Hands-free Trolley will pave the way to make shopping easier in offline shopping. It follows the customer while purchasing items and it maintains a safe distance between the customer and itself.

Hands-free Trolley consists of Raspberry Pi Pico H as a master controller which is installed on the cart and connected to four motors through motor driver, one ultrasonic sensor and two IR sensors. The motor driver assists in controlling the motor's speed and direction, which can be either clockwise or anticlockwise. Ultrasonic sensor maintains the distance between cart and the customer, who is accessing the cart. Based on distance between the cart and customer, microcontroller takes the choice of speed of motor. It always tries to maintain the nominal distance from the user. IR sensors takes care of the left and right turns of the user.

Outcomes are better when applied to a single user scenario. Nonetheless, this project still needs to be improved further in order to increase accuracy and withstand the hectic market conditions. This report also contains some of the methods that were attempted for following verified users. The best remedy for that issue must also be found in future advancements. Mecanum wheels can be extended farther, aiding the trolley's appropriate and more effective navigation. This project can be expanded to the general-purpose applications where it is required to handle more weight. For instance, a luggage trolley might be developed further by incorporating this technology.

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 NEED FOR THE WORK

Hands-free Trolley is a necessary product where there is a demand to carry more weight. Now a days, purchasing and shopping at big malls is becoming a daily activity in metro cities. On weekends and holidays, there is a noticeable rush at the supermarkets, this rush can also be heightened by any special discounts or offers. It is too hectic to push the trolley till entire shopping is done. To overcome these issues or to ease this our project is proposed, in which customer need not to concentrate on the movement of trolley as Trolley will move automatically behind the user. The use of this method will influence the modern shoppers. Using the same technology, it can be further expanded to general purpose usage also. For example, this product may be very useful in our local markets as well. It will be very helpful for handicapped and older people as this will reduce their burden.

1.2 OBJECTIVES FOR THE WORK

The objective of the work is,

- To help physically challenged people in carrying weights.
- To provide more comfort for modern shoppers.
- To make trolley to move based on user’s movement without any constraints.
- To maintain the distance between customer and trolley by using ultrasonic sensor.
- To sense the direction of movement of customer by using IR sensor.
- To move the direction of motor as desired.
- To follow the consumer automatically without any assistance from a person.

1.3 ORGANIZATION OF REPORT

The report is structured as follows. In Chapter 2, the literature review is discussed which outlines the various approaches used in trolley automation. Chapter 3 comprises the hardware modules involved in this project and principle of working of suggested technique. Then Results are shown and analyzed in Chapter 4. Outcomes and Future scope are concluded in Chapter 5.

CHAPTER 2

LITERATURE SURVEY

The project comprises of many challenges. Reviewing previous works in this area is necessary for choosing suitable hardware and reliable and trustful software algorithms. Autonomous navigation is an important part of this project which requires reviewing several literatures to use best algorithms for error minimization. In [1] cart is implemented to track and follow the target in the unstructured environment. Here microcontroller and ultrasonic sensor are used to identify and follow a targeted person. In [2] a robot automatically follows the particular person as a unique tag is placed on the person that should be followed by a robot. In [3] trolley can be controlled by using the android app. It gives commands to the trolley to move upward, backward, left, right. As per signal generated by the android app, command is given to the raspberry pi, and controlling process is done. In [4] a trolley follows using line following technique with carrying goods and also the bar code reader is placed in a trolley to save the time of billing. In [5] and [6], Kinect sensor is used which has human tracking capabilities from a static position that can provide full-body motion capture. Computer processes the information from the Kinect sensor and converts it into a skeletal image then calculates angle between joints to move a Servo-Motor to a specific angle.

CHAPTER 3

PROPOSED WORK

3.1 BLOCK DIAGRAM

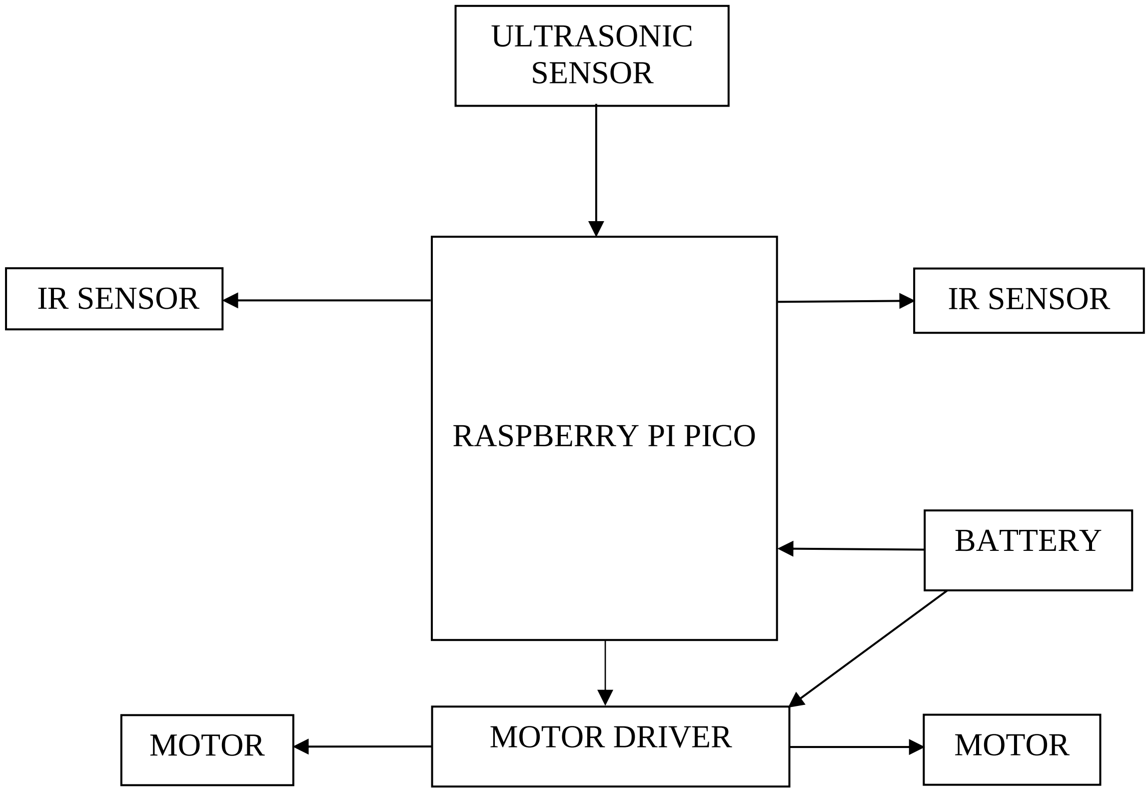

Fig 3.1 – Block Diagram

3.2 FLOW CHART

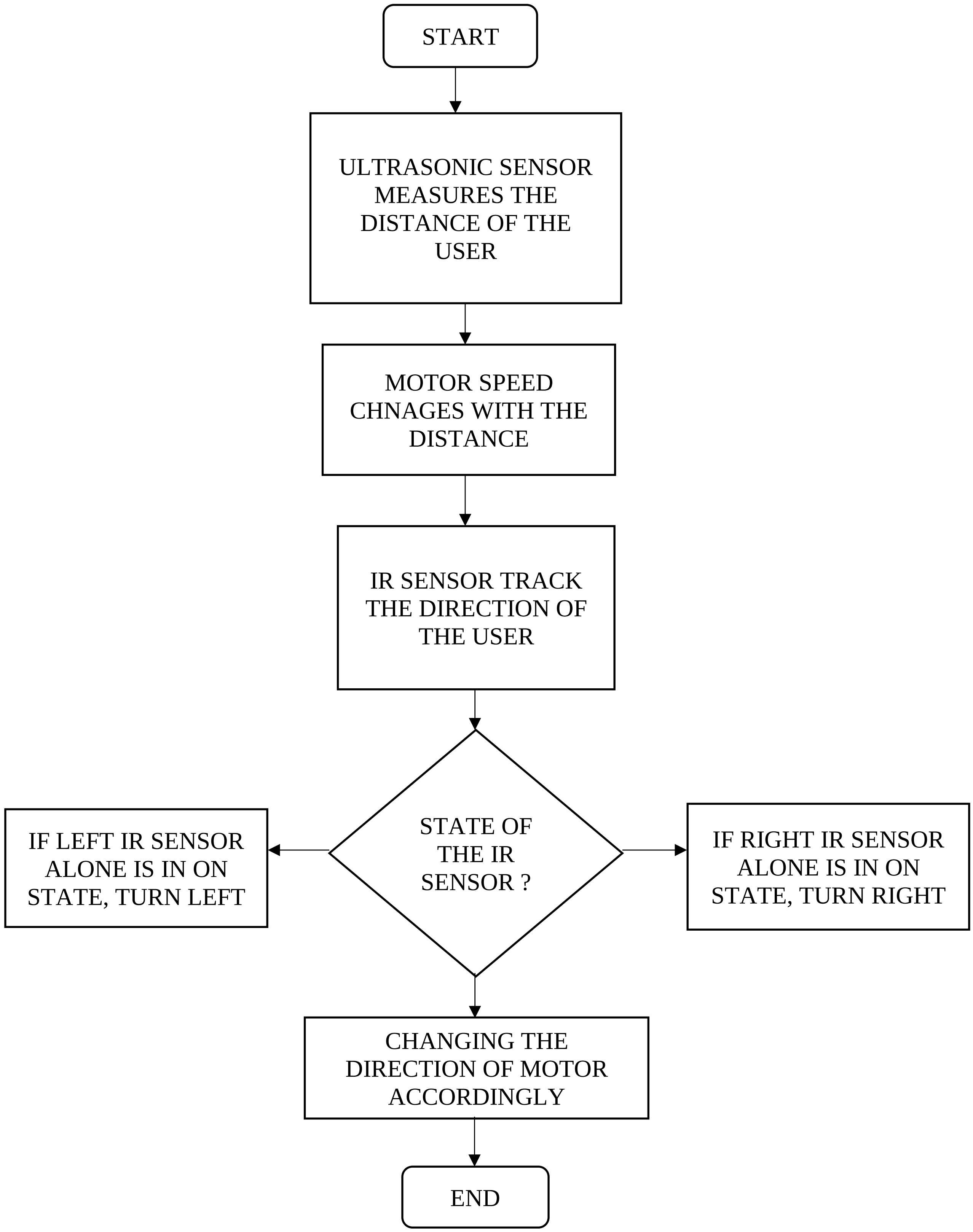

Fig 3.2 – Flowchart

3.3 SOFTWARE MODULES

The Software Simulation for this project is done through Proteus 8 Professional Tool.

3.3.1 SIMULATION

Referring to brake, MOTOR 1 and MOTOR 2 stops for distance values greater than 200cm

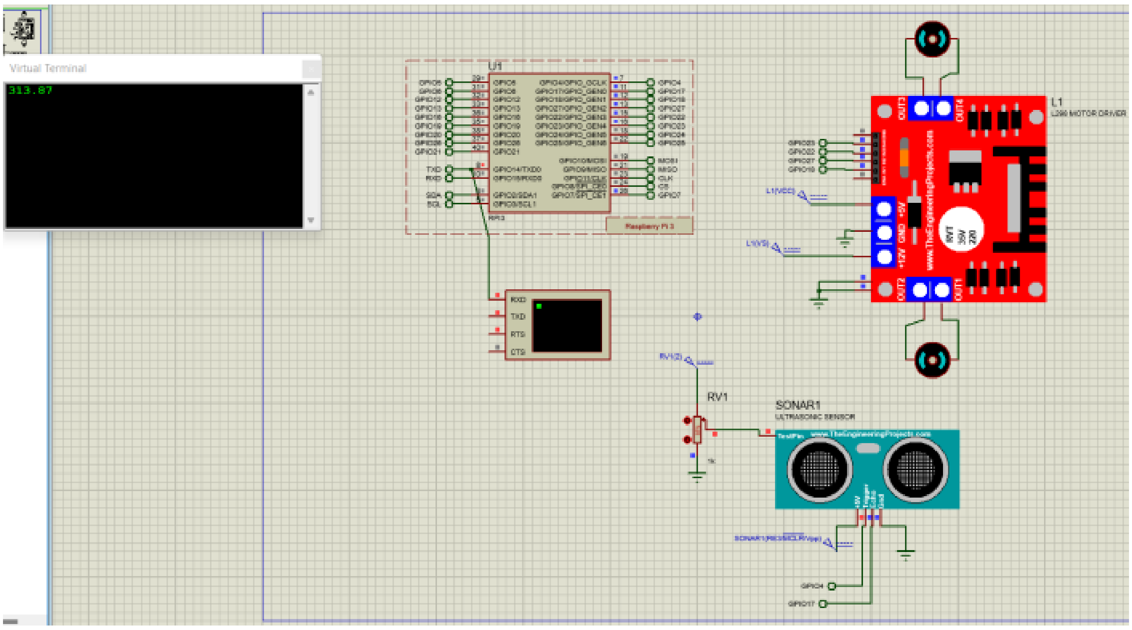

Fig 3.3.1(i) – Brake condition 1

Referring to brake, MOTOR 1 and MOTOR 2 stops for distance values lesser than 50cm

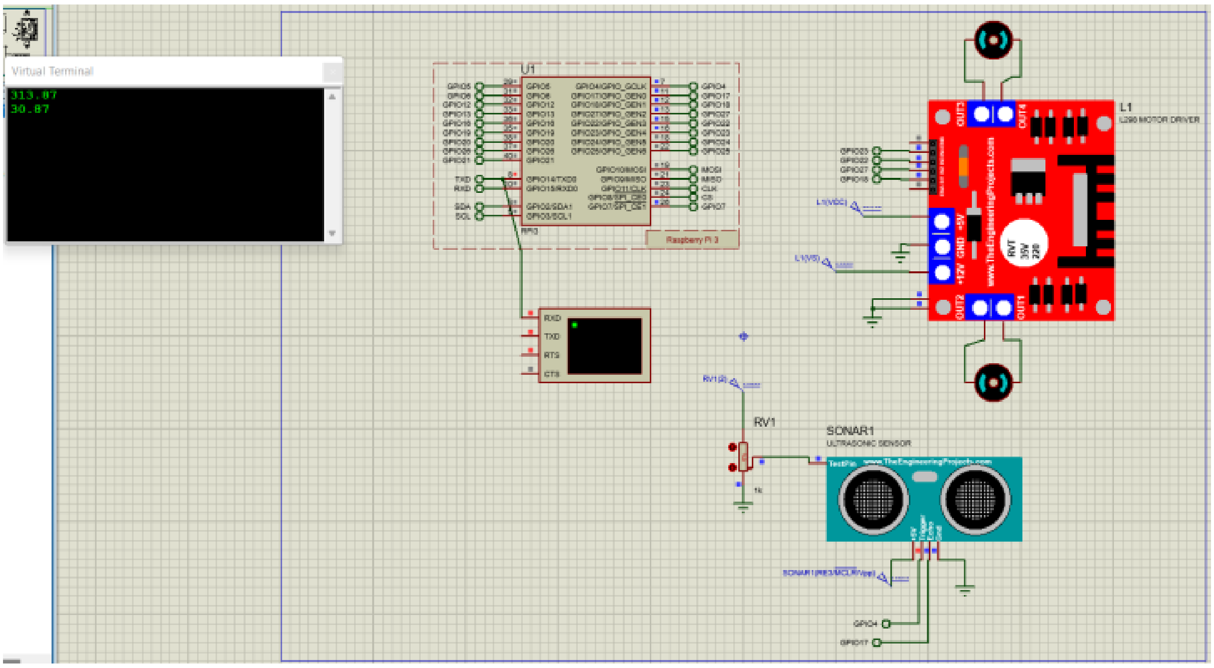

Fig 3.3.1(ii) – Brake condition 2

For distance values greater than 150cm and lesser than 300cm, MOTOR 1 and MOTOR 2 runs at high speed.

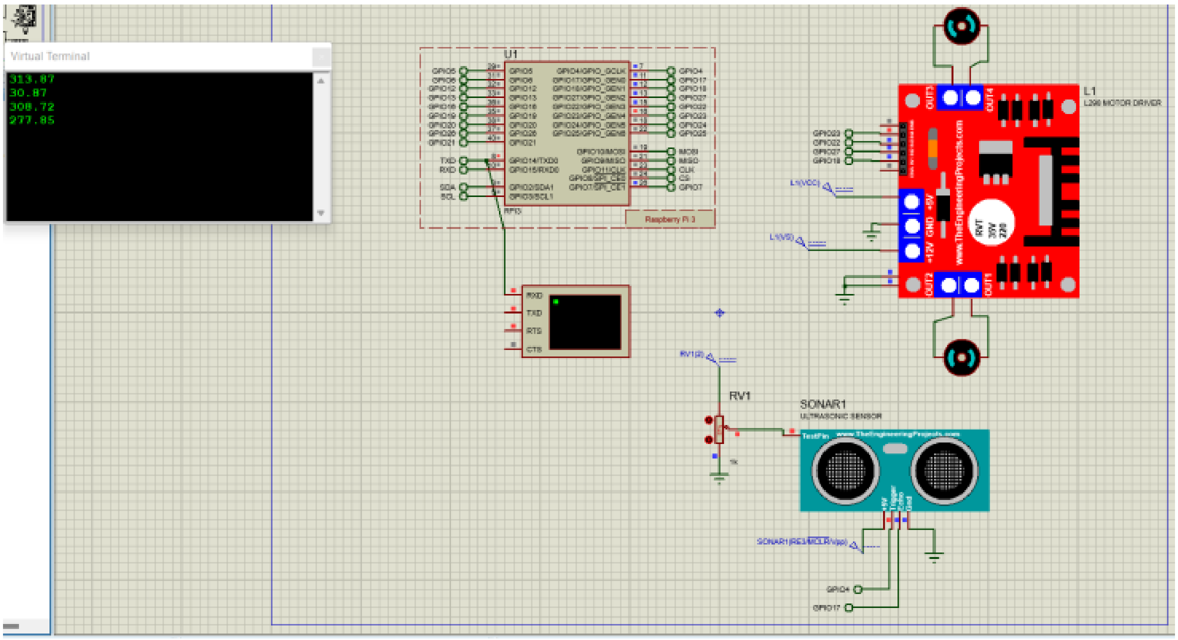

Fig 3.3.1(iii) – High speed condition

For distance values greater than 50cm and lesser than 100cm, MOTOR 1 and MOTOR 2 runs at nominal speed.

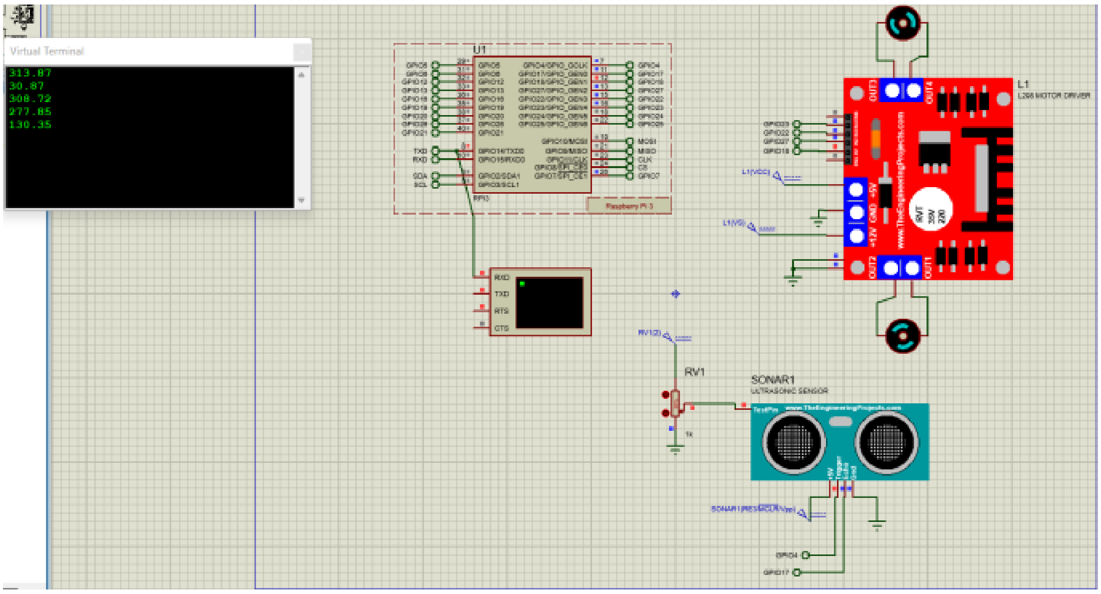

Fig 3.3.1(iv) – Nominal speed condition

3.4 HARDWARE MODULES

CHASSIS DOMAIN

- 12V DC Motor with 500 RPM – 4 units
- L298N Motor Driver
- 2 parallel units of three 3.7V Li-ion cell (3 x 3.7 = 11.1V)

CONTROLLER DOMAIN

- Raspberry PI PICO H
- Ultrasonic sensor HC-SR04
- IR sensor HW-201 – 2 units

3.4.1 RASPBERRY PI PICO H

Raspberry Pi PICO is a powerful, low-cost microcontroller development board.

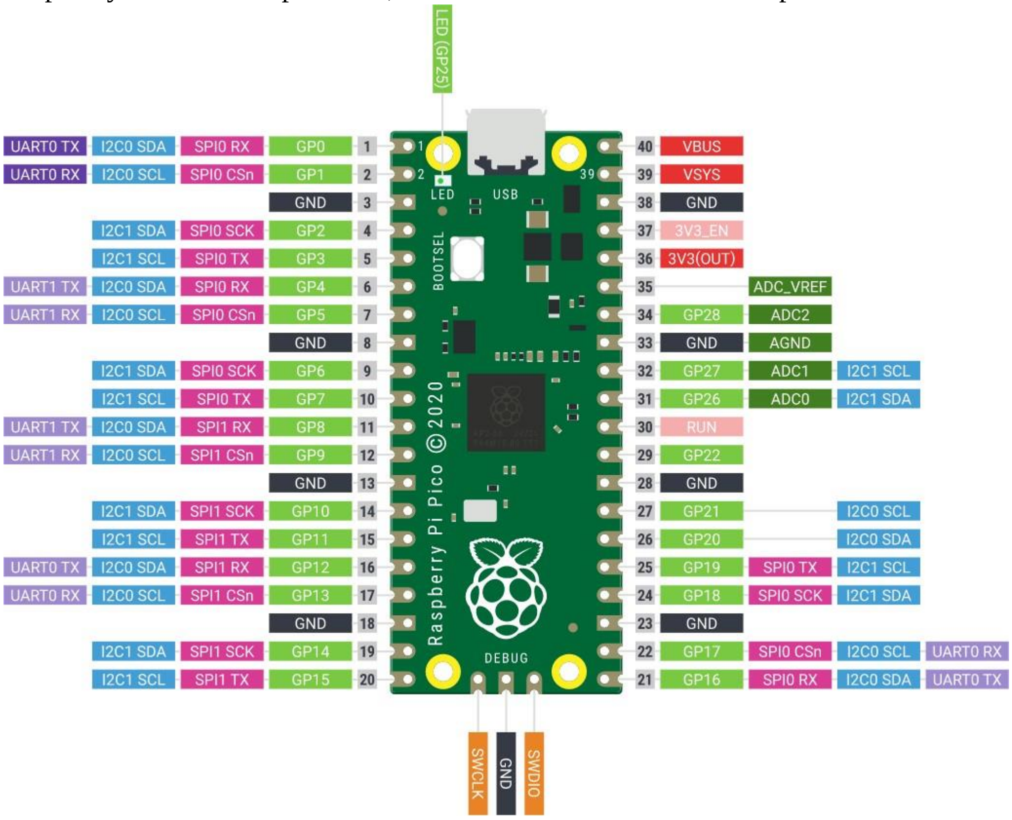

Fig 3.4.1 – Raspberry Pi Pico H

SPECIFICATION

- RP2040 SoC based on dual-core Arm Cortex-M0+ processor, clock running up to 133 MHz
- 264kB on-chip SRAM
- 2MB on-board QSPI flash
- 26 multifunction GPIO pins, including 3 analogue inputs with 3.3V.
- 2 × SPI, 2 × I2C, 2 × UART, 3 × 12-bit ADC, 16 × controllable PWM channels
- Onboard temperature sensor

3.4.2 ULTRASONIC SENSOR

HC-SR04 module includes an ultrasonic transmitter, a receiver and a control circuit. It uses ultrasound waves.

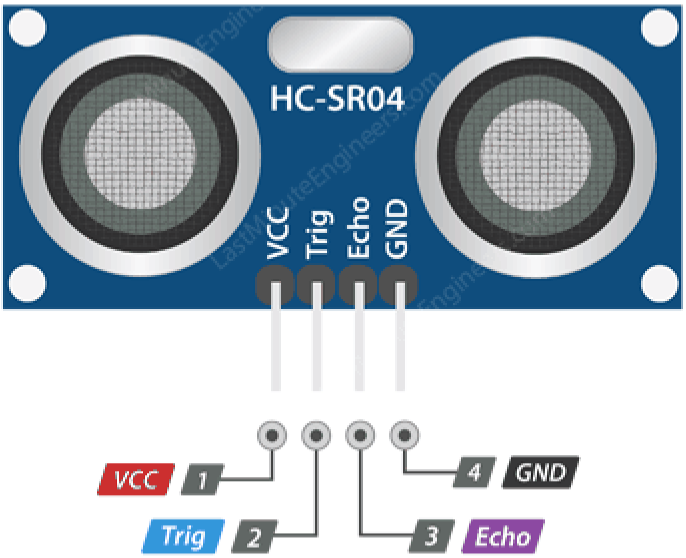

Fig 3.4.2 – Ultrasonic Sensor

Supply voltage	5v
Current Consumption	15 mA
Ultrasonic Frequency	40kHz
Maximal Range	400 cm
Minimal Range	2 cm
Trigger Pulse Width	10 µs

Table 3.4.2 – Specifications of Ultrasonic Sensor

WORKING PRINCIPLE

1. Send a short, but long enough 10µs pulse on the trigger pin from PICO and set the timer.
2. Wait for the echo line to go high.
3. Note the start time and not the end time when it goes low.
4. Estimate the time duration and by using speed = 343m/s (Speed of sound), distance can be calculated as distance = speed × time.
5. This obtained distance is twice the distance of obstacle from trolley since wave has travelled back and forth, so finally distance value is divided by 2.

3.4.3 IR SENSOR

It consists of two main components: the first is the IR transmitter section and the second is the IR receiver section. In the transmitter section, It works on the basic principle of reflecting light. The IR light from the IR transmitter is emitted, and once it hits a surface and bounces back, the Photodiode detects the IR light and in this way, sensors get to know there is an object in front of it.

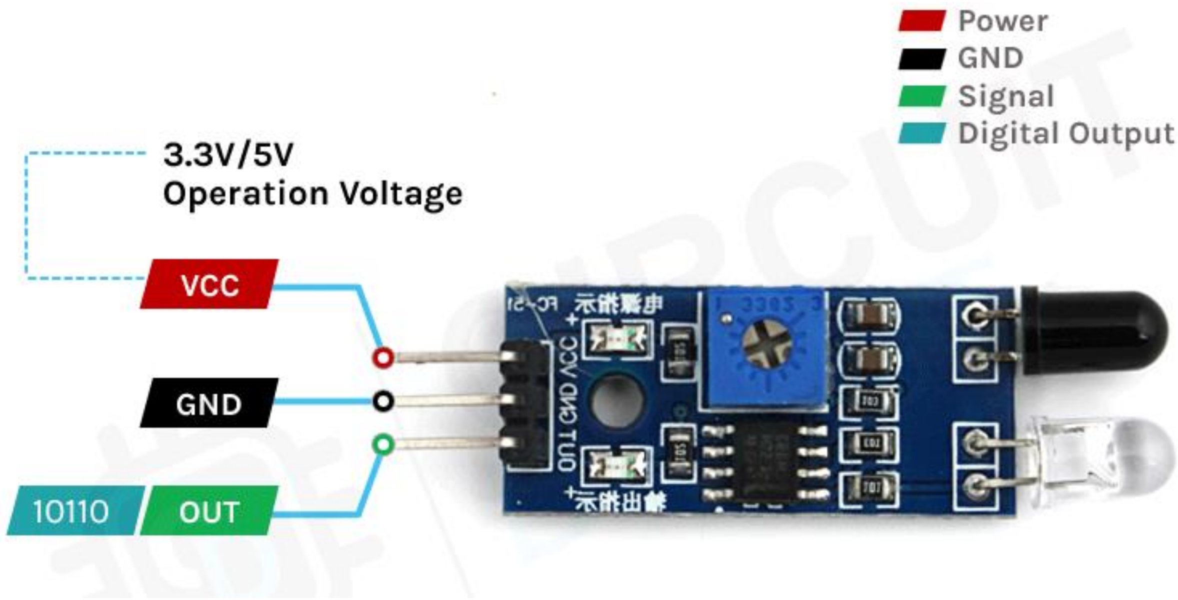

Fig 3.4.3 – IR sensor

Supply voltage	3.3v to 5v
Range	Adjustable upto 20 cm

Table 3.4.3 – Specifications of IR Sensor

3.4.4 MOTOR DRIVER (L298N)

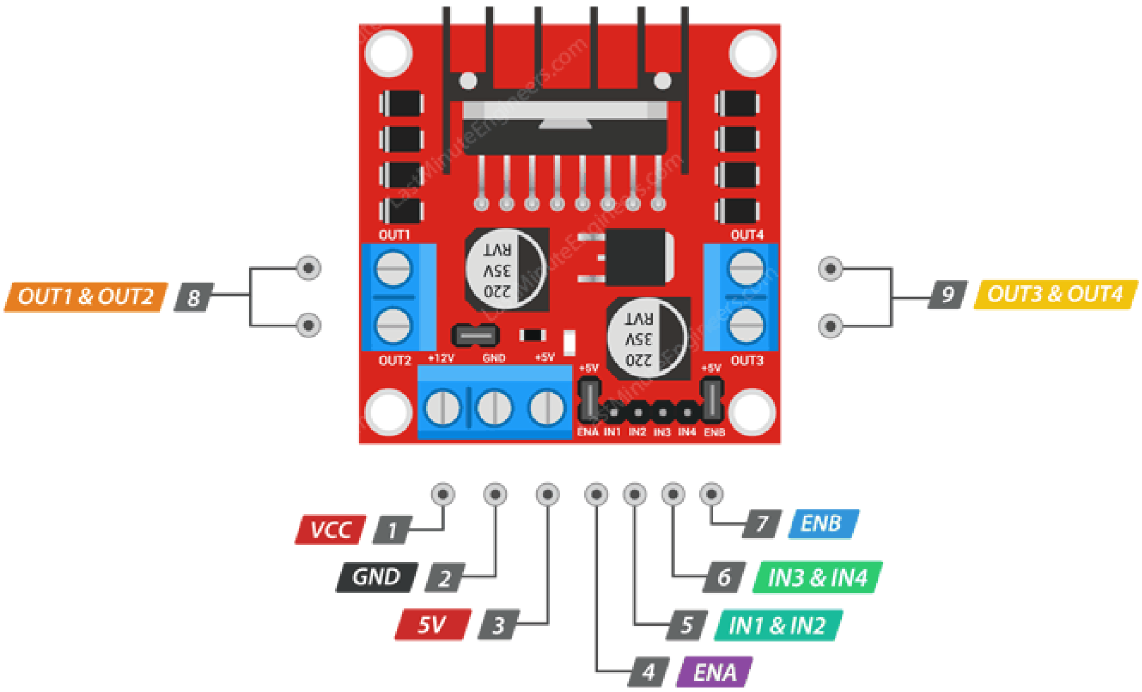

Fig 3.4.4 – Motor driver (L298N)

Motor Supply voltage	46V (maximum)
Motor output voltage	5V-35V
Continuous current per channel	2A
Maximum Power Dissipation	25W

Table 3.4.4(i) – Specifications of Motor driver

PWM signals are used to control the speed of the motors using ENA and ENB. The spinning direction of the motor 1 can be controlled by applying logic HIGH (5V) or logic LOW (Ground) to IN1, IN2 inputs. Similarly motor 2 can be controlled by IN3, IN4 inputs.

Input1	**Input2**	**Spinning Direction**
Low (0)	Low (0)	Motor OFF
High (1)	Low (0)	Forward
Low (0)	High (1)	Backward
High (1)	High (1)	Motor OFF

Table 3.4.4(ii) – Direction control of Motor driver

3.4.5 DC MOTOR

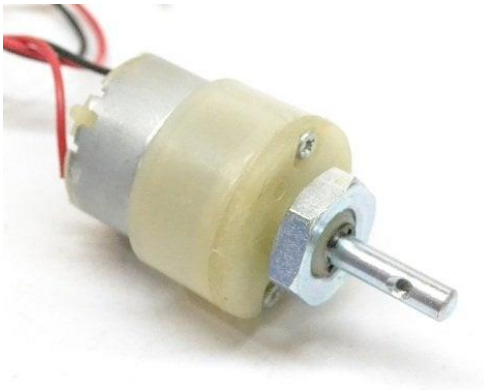

Fig 3.4.5 – DC Motor

Operating Voltage	12V
Rated speed	500 RPM
Rated Torque	0.7 kg-cm
Stall torque	3 kg-cm
Current	0.06A (no load), 0.3A (load)

Table 3.4.5 – Specifications of DC Motor

3.4.6 18650 LI-ION CELLS

Fig 3.4.6 – 18650 Li-ion cells

Higher the current, higher the torque. So two parallel units of 11.1V battery packs are used in this project to drive the motors with high torque. Each unit consists of three 18650 Li-ion cells (3 x 3.7V = 11.1V).

Voltage	3.7V
Nominal capacity	2000mAh

Table 3.4.6 – Specifications of Li-ion cells

3.4.7 7805 VOLTAGE REGULATOR

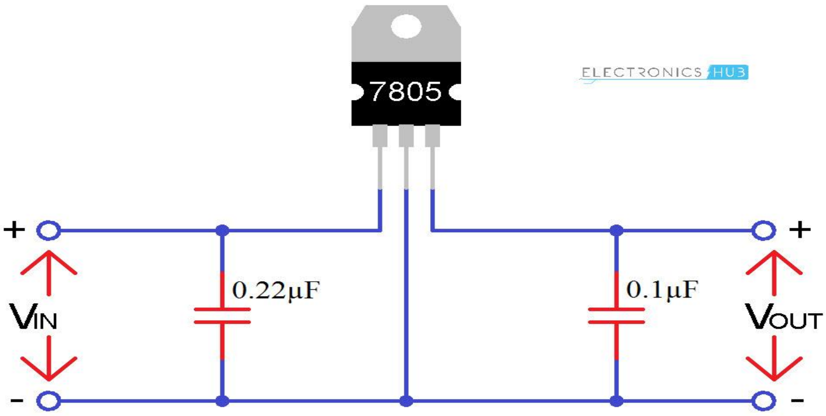

Fig 3.4.7 – 7805 Voltage regulator

Input Voltage	7 – 25V
Operating Current	5mA
Output Voltage	5V

Table 3.4.7 – Specifications of Voltage regulator

3.5 PRINCIPLE OF WORKING

This project works on the principle of the distance estimation and speed adjustment. It is like a closed loop system. The speed of the motor is adjusted each time when the distance is obtained from the ultrasonic sensor which acts like feedback. Trolley is designed in such a way that it always tries to maintain the nominal distance of 30 cm behind the user. If the distance is less than 30cm then the trolley would stop moving further. If the distance is between 30 to 50cm then the trolley would move slowly and the speed of the motor increases according to the distance between the trolley and the user.

Each IR sensor is placed 45^{o} to the trolley on the left and right sides. Using the state of IR sensors, the left and right turns can be achieved through PICO and motor driver. By moving one side of the wheels in clockwise and the other side of wheels in anticlockwise, the left and right turns can be done and using the delay, precision of turning can be controlled.

User's state	Left IR sensor state	Right IR sensor state
Forward	OFF	OFF
Right	OFF	ON
Left	ON	OFF
Stop	ON	ON

Table 3.5 – Working of IR sensors

Since speed is concentrated than torque for functionality verification, low cost 500 rpm motor is used to move the trolley with high speed. As a trade-off only limited amount of weight (approximately 1kg) can be carried now. But in a real time implementation, high cost motors with high speed and high torque can be used.

3.6 INFERENCE

- The main concern of the work is that, how the trolley will follow the verified user alone. To make that possible, certain things were tried.

- Firstly, RFID tags have been tried. RFID tags are the unique tags which has a unique code that can be scanned and read by RFID reader if tag is available within the range. So idea is to provide RFID tags to user and placing the RFID scanner at trolley. By continuous checking, trolley can verify whether the user is in the range or not. But the problem is economical high range RFIDs are not available in the market.

- Next, it has been planned to use WiFi's RSSI (Received Signal Strength Indicator) value to check the user's availability in the range. But the problem is WiFi's signal strength is not constant always. It is changing rapidly.

- Next, it has been planned to implement BLE beacons concept. A beacon is a small Bluetooth radio transmitter, powered by batteries. Beacons are similar to a lighthouse in functionality. These small hardware devices incessantly transmit Bluetooth Low Energy (BLE) signals. The devices which has greater than Bluetooth v4.0 can be used as a BLE beacons. Here ESP 32 CAM (Bluetooth v4.2) is tried as BLE beacon which is placed in trolley and apps like NRF connect and BLE scanner are provided to user which scans for beacons and gets paired with the targeted device. Here the signal strength is almost stable. But the problem comes when distance is tried to calculate from RSSI. There is no fixed formula for it. Proportionality is tried to obtain from various cases. Since that is not stable, range cannot be properly determined.

- Next, it has been planned to stick unique tags with alphanumeric characters at user's backside. Then using ESP 32 CAM module, availability of tag can be checked at each frame. If the tag is available in frame, then it can be confirmed that verified user is before the trolley. Trolley can start tracking of the tag and it can follow the user even without IR sensors. But the problem comes in image recognition, it couldn't recognise the tag properly due to low quality camera and also it takes delay upto 20 seconds which is definitely not tolerable.

- Next, Raspberry Pi with Pi camera has been tried. Now the quality of camera is fixed. But here also, the time taken for image recognition is 10 to 15 seconds which is not tolerable.

- With the same Raspberry Pi and Pi camera, finally text recognition using pytesseract module has been done. Here also delay is nearly 10 seconds.

- With all these considerations and the feedback obtained from expo, it is planned to do shape and colour recognition to follow the verified user in the future development of trolley.

- Instead of normal wheels Mecannum wheels can be used. Since Mecannum wheels are omni-directional, smooth turns and drive can be achieved.

- To achieve more precision in drive speed, Hall effect magnetic encoders can be used. They are the contactless rotary encoders that provides information on the direction and number of rotations made by the wheels.

- To control the speed based on the load condition, weight sensor can be incorporated in future development.

- To make it more smarter, RFID tags can be used for product verification and bill calculation instead of physical bill counters. In this way, many money can be saved which are spent for human powers in bill counters.

- People also can save the time instead of waiting in bill counter queues.

- By adding security features using NodeMCU which can able to alert the concerned person through IoT mobile applications, project can be made more practical and more smarter.

- Through the lessons learnt in doing the project and feedbacks obtained, it is decided to improve the trolley with precise calculations and by adding more smart features, it will be definitely tried in future to make the optimal product for the practical scenarios.

CHAPTER 4

RESULTS

4.1 OUTPUT

Fig 4.1(i) – Trolley working video

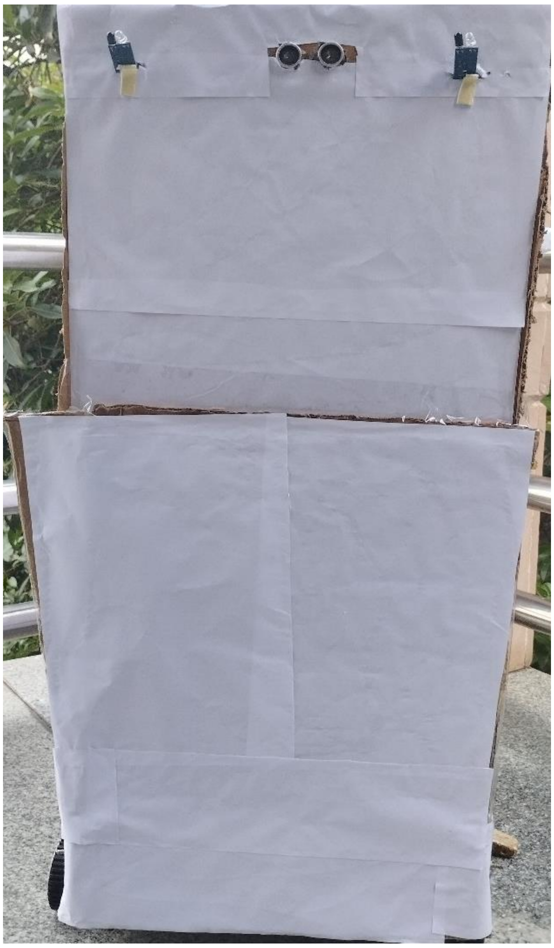

Fig 4.1(ii) – Front view of the trolley

Fig 4.1(iii) – Side view of the trolley

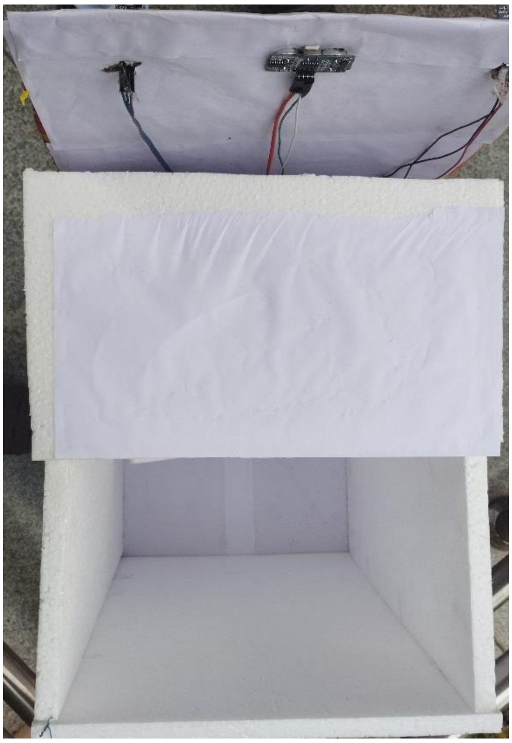

Fig 4.1(iv) – Top view of the trolley

4.2 RESULT ANALYSIS

S.NO	DUTY CYCLE	DISTANCE (cm) T=5 sec	SPEED
1	65535	445	99cm\s
2	60000	395	79 cm\s
3	55000	300	60 cm\s
4	50000	235	47 cm\s
5	45000	175	35 cm\s
6	40000	120	24 cm\s
7	35000	48	10 cm\s
8	30000	NIL	NIL

Table 4.2 – Result analysis

The above table value is obtained by the trial-and-error method by repeating the function of the trolley again and again. Duty cycle vs speed graph is plotted and shown below.

X axis: Duty cycle
Y axis: Speed (cm / s)

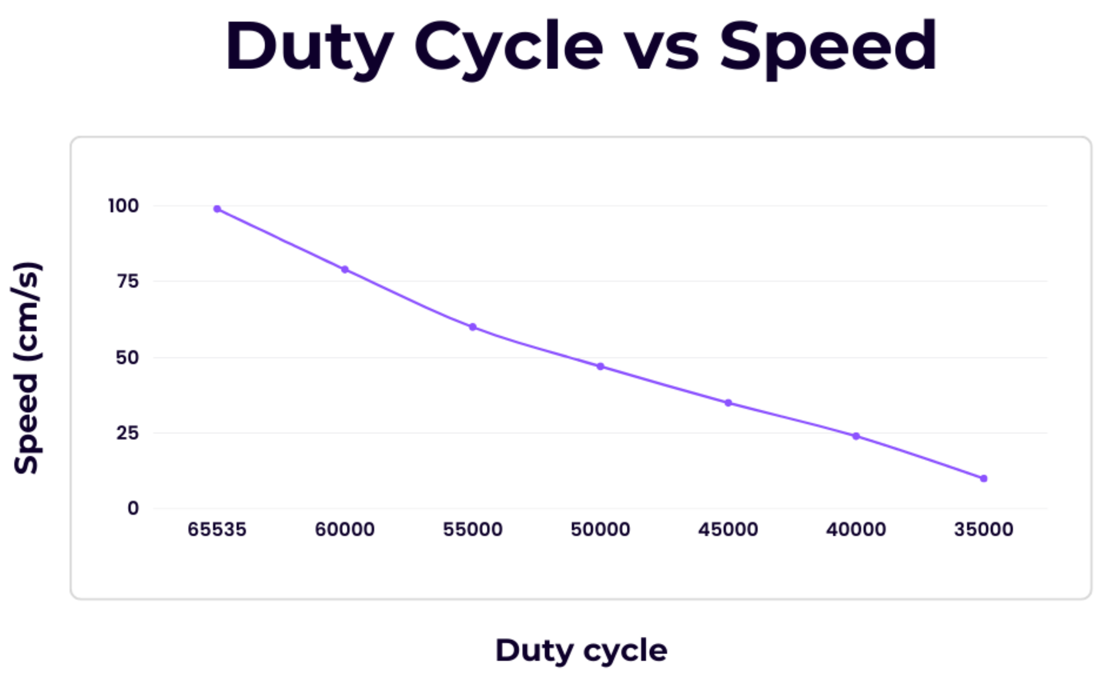

Fig 4.2 – Duty cycle vs Speed

CHAPTER 5

CONCLUSION

In the proposed problem statement, movement of the trolley without any assistance from user and following the verified user is concentrated. Trolley's movement is controlled using ultrasonic sensors and IR sensors. To follow the verified user, certain techniques tried couldn't bring the expected results. So algorithms have to be developed further to make trolley to follow the verified user alone. Those algorithms need to be integrated in near future to make it feasible. If required, some smart features like bill counting and security features can be added. After this, giving the trolley a good physic would yield a complete product.

APPENDIX 1 – CODE

1.1 Code for Simulation

```
# !/usr/bin/env python3
#!/usr/bin/env python3
# Main.py file generated by New Project wizard
# Created:   Sun Jan 22 2023
# Processor: RPI3
# Compiler:  Python 3 (Proteus)

# Modules
from goto import *
import RPi.GPIO as GPIO
import time
import var
import pio
import resource

# Peripheral Configuration Code (do not edit)
#---CONFIG_BEGIN---
import cpu
import FileStore
import VFP
import timer
import Ports

def peripheral_setup () :
# Peripheral Constructors
        pio.cpu=cpu.CPU ()
        pio.storage=FileStore.FileStore ()
        pio.server=VFP.VfpServer ()
        pio.timer=timer.Timer ()
        pio.uart=Ports.UART ()
        pio.storage.begin ()
        pio.server.begin (0)
        # Install interrupt handlers
        def peripheral_loop () :
        pio.server.poll ()
        pio.timer.poll ()

#---CONFIG_END---

# Main function
def main () :
# Setup
        peripheral_setup()
        pio.uart.setup(9600)
        GPIO.setmode(GPIO.BCM)
        trigger=4
        echo=17
        in1=18
        in2=27
        in3=22
```

```
    in4=23

    GPIO.setup(in1,GPIO.OUT)
    GPIO.setup(in2,GPIO.OUT)
    GPIO.setup(in3,GPIO.OUT)
    GPIO.setup(in4,GPIO.OUT)
    GPIO.setup(trigger,GPIO.OUT)
    GPIO.setup(echo,GPIO.IN)
    GPIO.setup(7,GPIO.OUT)
    GPIO.output(trigger,False)
    time.sleep(2)

    # Infinite loop
    while 1 :
        peripheral_loop()
        GPIO.output(trigger,True)
        time.sleep(0.00001)
        GPIO.output(trigger,False)
        start=time.time()
        stop=time.time()
        while(GPIO.input(echo)==0):
            start=time.time()
        while(GPIO.input(echo)==1):
            stop=time.time()
        time_elapsed=stop-start
        distance=((time_elapsed)*3430)/2
        d=distance*1.143
        pio.uart.println(str(round(d,2)))
        time.sleep(1)
        if(d<100):
            GPIO.output(in1,GPIO.LOW)
            GPIO.output(in2,GPIO.LOW)
            time.sleep(5)
        elif(d>=100 and d<250):
            GPIO.output(in1,GPIO.HIGH)
            GPIO.output(in2,GPIO.LOW)
            time.sleep(5)
            GPIO.output(in1,GPIO.LOW)
            GPIO.output(in2,GPIO.LOW)
        else:
            GPIO.output(in3,GPIO.LOW)
            GPIO.output(in4,GPIO.HIGH)
            time.sleep(5)
            GPIO.output(in3,GPIO.LOW)
            GPIO.output(in4,GPIO.LOW)
        #print("distance :",d,"cm")
        pass
# Command line execution
if _name_ == '_main_':
    main()
```

1.2 Code for Hardware

```
from machine import Pin, PWM
from time import sleep
import utime

IN1 = Pin(1, Pin.OUT)
IN2 = Pin(2, Pin.OUT)
IN3 = Pin(3, Pin.OUT)
IN4 = Pin(4, Pin.OUT)

ena = PWM(Pin(0))
enb = PWM(Pin(5))
ena.freq(1000)
enb.freq(1000)
speed=65535
trigger = Pin(17, Pin.OUT)
echo = Pin(16, Pin.IN)
sensor1 = Pin(12,Pin.IN)
sensor2 = Pin(15,Pin.IN)
def ultra():
	trigger.low()
	utime.sleep_us(2)
	trigger.high()
	utime.sleep_us(5)
	trigger.low()
	while echo.value() == 0:
		signaloff = utime.ticks_us()
	while echo.value() == 1:
		signalon = utime.ticks_us()
	timepassed = signalon - signaloff
	distance = (timepassed * 0.0343) / 2
	print("The distance from object is ",distance,"cm")
	return distance

def clk(en,in1,in2,duty):
	en.duty_u16(duty)
	in1.low() #spin forward
	in2.high()

def aclk(en,in1,in2,duty):
	en.duty_u16(duty)
	in1.high() #spin backward
	in2.low()

def brk(en,in1,in2,duty):
	en.duty_u16(duty)
	in1.low() #brake
	in2.low()

def fwd():
	clk(ena,IN1,IN2,speed)
	clk(enb,IN3,IN4,speed)
```

```
while True:
	d=ultra()
	#utime.sleep(0.5)
	#print(sensor1.value(),sensor2.value())
	if(sensor1.value() and sensor2.value()):
		if(d<=30):
			#print("fwd")
			speed=0
		elif(d>30 and d<=50):
			speed=45000
		elif(d>50 and d<=60):
			speed=50000
		elif(d>60 and d<=70):
			speed=53000
		elif(d>70 and d<=80):
			speed=55000
		elif(d>80 and d<=100):
			speed=57000
		elif(d>100 and d<=250):
			speed=60000
		else:
			speed=0
		fwd()


	elif(sensor1.value() and not(sensor2.value())):
		clk(ena,IN1,IN2,65535)
		aclk(enb,IN3,IN4,65535)
		sleep(0.8)

	elif(sensor2.value() and not(sensor1.value())):
		aclk(ena,IN1,IN2,65535)
		clk(enb,IN3,IN4,65535)
		sleep(0.8)

	else:
		brk(ena,IN1,IN2,0)
		brk(enb,IN3,IN4,0)
```

APPENDIX 2 – DATASHEET

DC Motor Datasheets

https://robu.in/wp-content/uploads/2019/02/test-report-PG45775126000-19.2K.pdf
https://robu.in/wp-content/uploads/2019/02/PG45775126000-19.2K.pdf

Raspberry Pi Pico

https://datasheets.raspberrypi.com/pico/pico-datasheet.pdf

Ultrasonic sensor

https://datasheetspdf.com/pdf-file/1380136/ETC/HC-SR04/1

IR sensor

https://components101.com/sites/default/files/component_datasheet/Datasheet%20of%20IR%20%20Sensor.pdf

BIBLIOGRAPHY

[1] Akshay Chalke, Ashwini Bhor, Pradnya Bhoite, Uddhav Jadhav, Follower robotic cart using the ultrasonic sensor, Published in (IJESC), 2017

[2] Muhammad Sarmad Hassan, Mafaz Wali Khan, Ali Fahim Khan, Design and Development of Human Following Robot published in Student Research Paper Conference Vol-2, No-15, July 2015.

[3] L.S.Y. Dehigaspege, M.K.C. Liyanage, N.A.M. Liyanage, M.I. Marzook and Dilshan Dhammearatchi, Follow Me Multifunctional Automated Trolley, published in International Journal of Engineering Research & Technology (IJERT) Vol. 6 Issue 07, July – 2017

[4] Jayant Nivritti Patil, H. K. Bhangale, and Android-based robot using Raspberry Pi, Published in International Journal of Science and Research (IJSR) 2015 International Conference on DOI: 10.1109/ICAEE.2015.7506791, 11 July 2016.

[5] Georgios Th. Papadopoulos, Apostolos Axenopoulos and Petros Daras. "Real-Time Skeleton-Tracking-Based Human Action Recognition Using Kinect Data" LNCS, volume 8325

[6] B. Ilias, S.A. Abdul Shukor, S. Yaacob, A.H. Adom and M.H. Mohd Razali. A Nurse Following Robot with High speed Kinect Sensor. ARPN Journal of Engineering and Applied Sciences, vol. 9. No 12, December 2014, pp. 2454-2459.