Obstacle Avoiding Robot

AISR PROJECT REPORT

Submitted by

B.RAMA SAI SANTOSH [9920004285]

B.JATHIN KRISHNA [9920004284]

K.SRIKANTH [9920004308]

CH.SAI KIRAN [9920004753]

Under the guidance of

Dr.P.Mangalraj

(Assistant Professor, Department of Computer Science and Engineering)

In partial fulfillment for the award of the degree

Of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

COMPUTER SCIENCE AND ENGINEERING

Kalasalingam Academy of Research and Education

Anand Nagar, Krishnankovil – 626 126

Academic Year 2020-2021

DECLARATION

I/We affirm that the project work titled **"OBSTACLE AVOIDING ROBOT"** being submitted in partial fulfillment for the award of the degree of **Bachelor of Technology in Computer Science and Engineering** is the original work carried out by me/us. It has not formed the part of any other project work submitted for award of any degree or diploma, either in this or any other University.

B.RAMA SAI SANTOSH [9920004285]
B.JATHIN KRISHNA [9920004284]
K.SRIKANTH [9920004308]
CH.SAI KIRAN [9920004753]

BONAFIDE CERTIFICATE

Certified that this project report **"OBSTACLE AVOIDING ROBOT"** is the bonafide work of **"B.SANTOSH, K.SRIKANTH, B.JATHIN, CH.SAI KIRAN"** who carried out the project work under my supervision.

Dr. P.MANGALRAJ	**Dr. A. FRANCIS SAVIOUR DEVARAJ**
Associate Professor	Professor
(SUPERVISOR)	**HEAD OF THE DEPARTMENT**
Department of computer Science and Engineering,	**Department of computer Science and Engineering,**
Kalasalingam Academy of Research And Education (KARE)	**Kalasalingam Academy of Research And Education (KARE)**
Krishnankoil 626126	**Krishnankoil 626126**

Submitted for the Project Viva-voce examination held on......................................

Internal Examiner   External Examiner

ACKNOWLEDGEMENT

First and foremost, I wish to thank the Almighty God for his grace and benediction to complete this Project work successfully. I would like to convey my special thanks from the bottom of my heart to my dear Parents and affectionate Family members for their honest support for the completion of this Project work.

I express deep sense of gratitude to “Kalvivallal” Thiru. T. Kalasalingam B.com., Founder Chairman, “Ilayavallal” Dr.K.Sridharan Ph.D., Chancellor, Dr.S.Shasi Anand, Ph.D., Vice President (Academic), Mr.S.Arjun Kalasalingam M.S., Vice President (Administration) , Dr.R.Nagaraj, Vice- Chancellor, Dr.V.Vasudevan Ph.D., Registrar , Dr.P.Deepalakshmi M.E., Ph.D., Dean (School of Computing) . And also a special thanks to Dr. A. Francis Saviour Devaraj . Head, Department of CSE, Kalasalingam Academy of Research and Education for granting the permission and providing necessary facilities to carry out Project work.

I would like to express my special appreciation and profound thanks to my enthusiastic Project Supervisor **Dr.P.Mangalraj**, Associate Professor/CSE of Kalasalingam Academy of Research and Education [KARE] for his inspiring guidance, constant encouragement with my work during all stages. I am extremely glad that I had a chance to do my Project under my Guide, who truly practices and appreciates deep thinking. I will be forever indebted to my Guide for all the time he has spent with me in discussions. And during the most difficult times when writing this report, he gave me the moral support and the freedom I needed to move on.

Besides my Project guide, I would like to thank the rest of Class committee members and all faculty members and Non-Teaching staff for their insightful comments and encouragement. Finally, but by no means least, thanks go to all my school and college teachers, well wishers, friends for almost unbelievable support.

TABLE OF CONTENTS

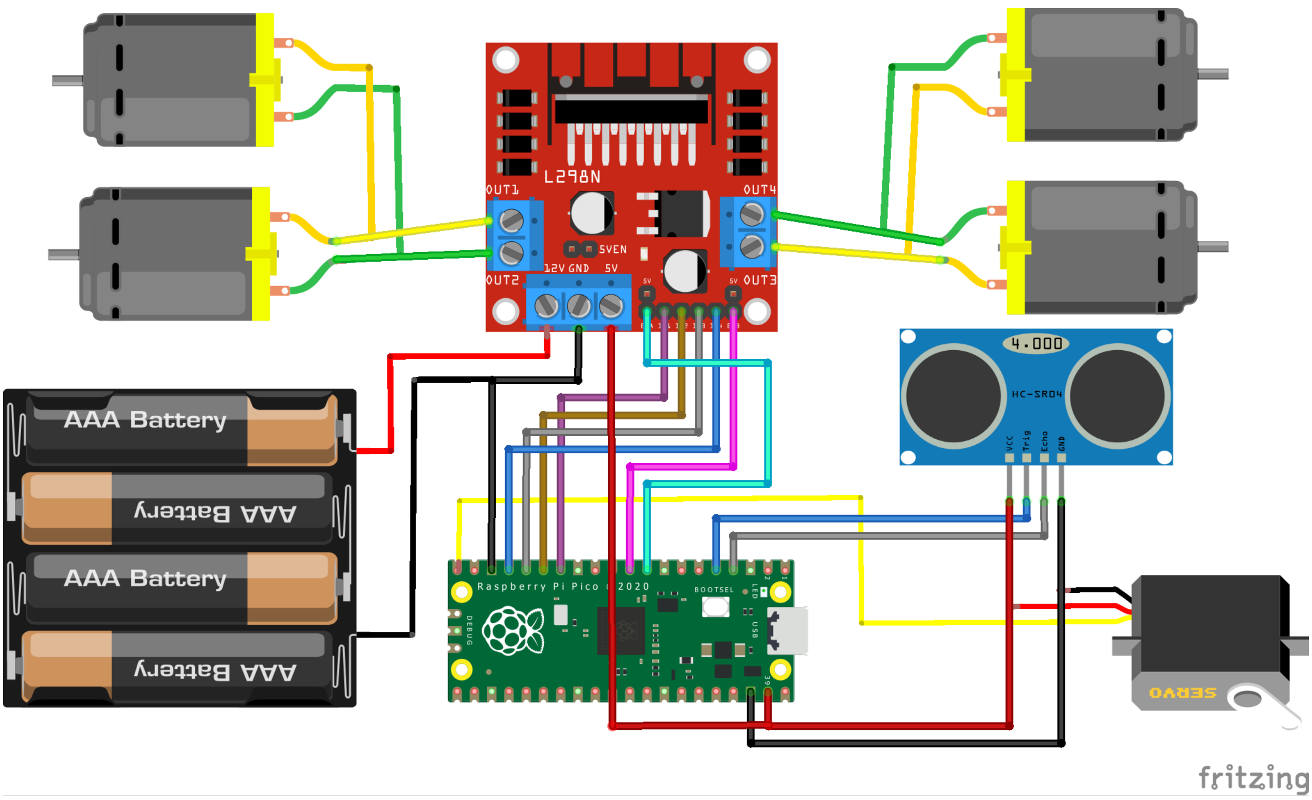
CHAPTER 1

INTRODUCTION

The project is design to build an obstacle avoidance robotic vehicle using ultrasonic sensors for its movement. A microcontroller (ATmega328) is used to achieve the desired operation. A robot is a machine that can perform task automatically or with guidance. The project proposes robotic vehicle that has an intelligence built in it such that it directs itself whenever an obstacle comes in its path. This robotic vehicle is built, using a micro-controller of raspberry pi pico family. An ultrasonic sensor is used to detect any obstacle ahead of it and sends a command to the micro-controller. Depending on the input signal received, the micro-controller redirects the robot to move in an alternate direction by actuating the motors which are interfaced to it through a motor driver. Some of the project is built with the IR sensors has its own application so in our project those application is not compactable so we are using ultrasonic sensor.

CHAPTER 2

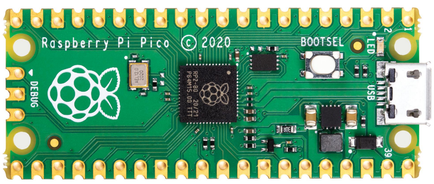
CIRCUIT DIAGRAM

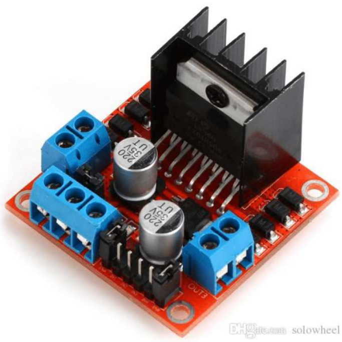
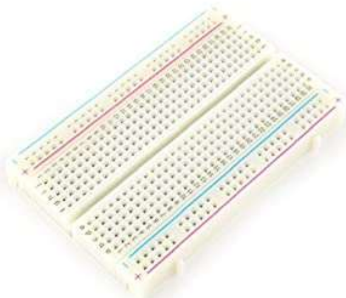


CHAPTER 3

MATERIALS REQUIRED

- Raspberry pi pico with soldered pins
- Micro servo, 60rpm motors and ultrasonic sensor
- Motor driver L298N
- Breadboard for pico connections
- Jumper wires
- Lithium-ion battery with 3 type battery holders
- Thonny IDE and USB cable



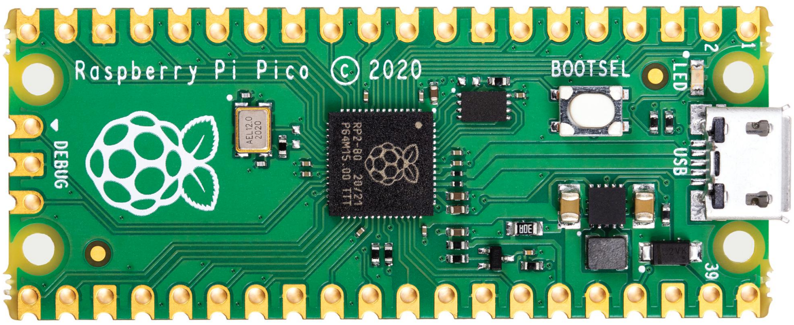






CHAPTER – 4

COMPONENT DESCRIPTION

Raspberry Pi Pico:



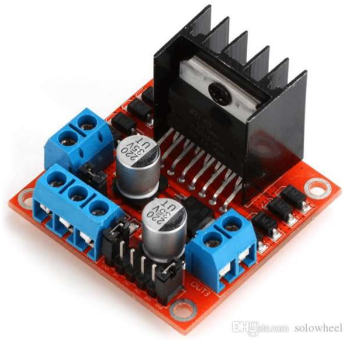
Raspberry Pi Pico is a low-cost, high-performance microcontroller board with flexible digital interfaces. Key features include:

- RP2040 microcontroller chip designed by Raspberry Pi in the United Kingdom
- Dual-core Arm Cortex M0+ processor, flexible clock running up to 133 MHz
- 264kB of SRAM, and 2MB of on-board flash memory
- USB 1.1 with device and host support
- Low-power sleep and dormant modes
- Drag-and-drop programming using mass storage over USB
- 26 × multi-function GPIO pins
- 2 × SPI, 2 × I2C, 2 × UART, 3 × 12-bit ADC, 16 × controllable PWM channels
- Accurate clock and timer on-chip
- Temperature sensor
- Accelerated floating-point libraries on-chip
- 8 × Programmable I/O (PIO) state machines for custom peripheral support

Micro servo:

The Tower Pro SG90 is a simple Servo Motor which can rotate 90 degrees in each direction (approximately 180 degrees in total).

L293D:



It is a motor driver which can provide bi-directional drive current for two motors.

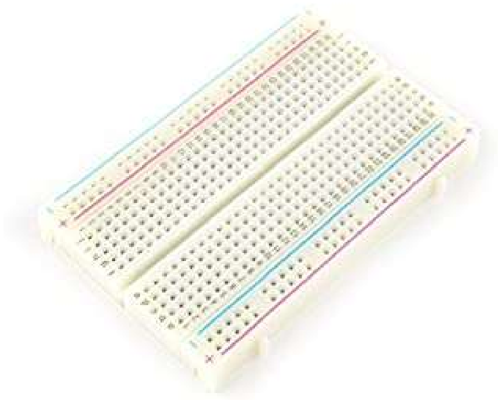
HC – SR04

It is an Ultrasonic Range Finder Sensor. It is a non-contact based distance measurement system and can measure distance of 2cm to 4m.

Jumper wires:

Jumper wires are simply wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering. Jumper wires are typically used with breadboards and other prototyping tools in order to make it easy to change a circuit as needed. Fairly simple. In fact, it doesn't get much more basic than jumper wires.

Breadboard:



A breadboard is a rectangular plastic board with a bunch of tiny holes in it. These holes let you easily insert electronic components to **prototype** (meaning to build and test an early version of) an electronic circuit, like this one with a battery, switch, resistor, and an LED (light-emitting diode). To learn more about individual electronic components, see our Electronics Primer.

CHAPTER-5

SOURCE CODE

```
from machine import Pin,PWM #importing PIN and PWM
import time #importing time
import utime

# Defining motor pins
motor1=Pin(10,Pin.OUT)
motor2=Pin(11,Pin.OUT)
motor3=Pin(12,Pin.OUT)
motor4=Pin(13,Pin.OUT)
# Defining enable pins and PWM object
enable1=PWM(Pin(6))
enable2=PWM(Pin(7))
```

```
# Defining  Trigger and Echo pins
trigger = Pin(3, Pin.OUT)
echo = Pin(2, Pin.IN)
```

```
# Defining  Servo pin and PWM object
servoPin = Pin(15)
servo = PWM(servoPin)
duty_cycle = 0 # Defining and initializing duty cycle PWM
```

```
# Defining frequency for servo and enable pins
servo.freq(50)
enable1.freq(1000)
enable2.freq(1000)
```

```
# Setting maximum duty cycle for maximum speed
enable1.duty_u16(65025)
enable2.duty_u16(65025)
```

```
# Forward
def move_forward():
    motor1.low()
    motor2.high()
    motor3.high()
    motor4.low()
# Backward
def move_backward():
    motor1.high()
    motor2.low()
    motor3.low()
    motor4.high()
```

```
#Turn Right
def turn_right():
    motor1.low()
    motor2.high()
    motor3.low()
    motor4.high()

#Turn Left
def turn_left():
    motor1.high()
    motor2.low()
    motor3.high()
    motor4.low()

#Stop
def stop():
    motor1.low()
    motor2.low()
    motor3.low()
    motor4.low()

# Defining function to get distance from ultrasonic sensor
def get_distance():
   trigger.low()
   utime.sleep_us(2)
   trigger.high()
   utime.sleep_us(5)
   trigger.low()
   while echo.value() == 0:
       signal1 = utime.ticks_us()
   while echo.value() == 1:
       signal2 = utime.ticks_us()
   timepassed = signal2 - signal1
   dist = (timepassed * 0.0343) / 2
   return dist
```

```
#Defining function to set servo angle
def setservo(angle):
    duty_cycle = int(angle*(7803-1950)/180) + 1950
    servo.duty_u16(duty_cycle)
```

```
setservo(90)
```

```
while True:
    distance=get_distance1() #Getting distance in cm

    #Defining direction based on conditions
    if distance < 15:
        stop()
        move_backward()
```

```
        time.sleep(1)
        stop()
        time.sleep(0.5)
        setservo(30) #Servo angle to 30 degree
        time.sleep(1)
        right_distance=get_distance()
        #print(right_distance)
        time.sleep(0.5)
        setservo(150) #Servo angle to 150 degree
        time.sleep(1)
        left_distance=get_distance()
        #print(left_distance)
        time.sleep(0.5)
        setservo(90)

        if right_distance > left_distance:
            turn_right()
            time.sleep(2)
            stop()
        else:
            turn_left()
            time.sleep(2)
            stop()
    else:
        move_forward()
```

```
    time.sleep(0.5)
```

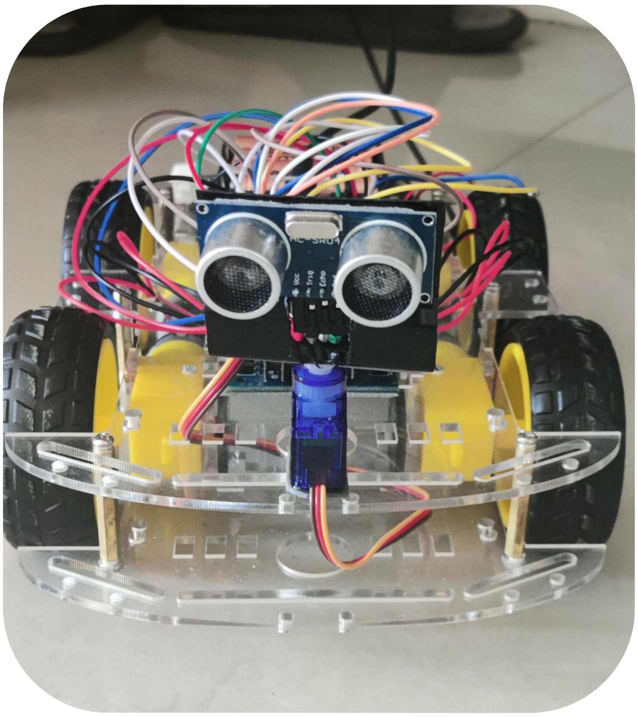
CHAPTER – 6

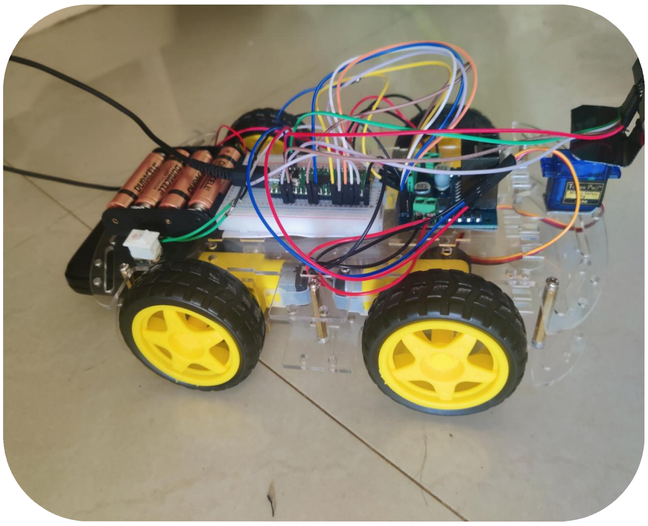
MODULE

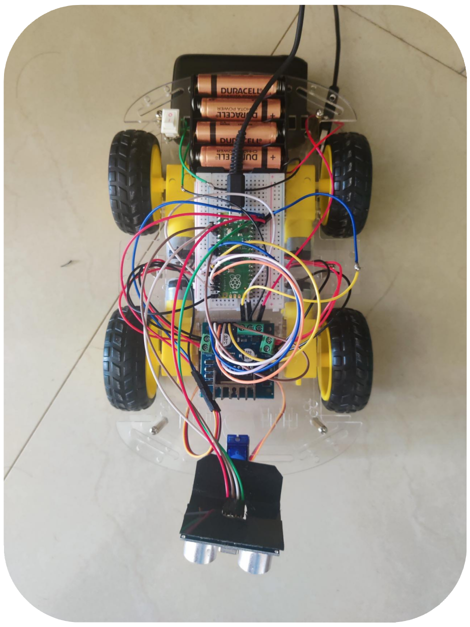
- Install Python and Thonny IDE into your system.
- Using micro USB connect pico to your system by pressing the boot button on pico board.
- You will get a notification once your pico is connected to computer.
- Now open Thonny IDE and in tools click on options and then in interpreter drop-down select MicroPython (Raspberry pi pico) option.
- In port leave it for default settings and then click ok, now you can install MicroPython.
- It immediately installs now click on ok.
- Check if everything is working properly by entering the test program in shell.
- Shell is terminal whihc can found below IDE
- Just type print("test")
- If the same appears below when you press enter you are good to go.
- Now from the menu, select file then open the program which was saved on desktop by clicking on this computer option.
- Now you have your program ready, Run the current script (green symbol on the menu section).
- Connect the battery and check if everything is working fine.
- Once it is verified go to file then save as
- Click on raspberry pi pico and save the file name as main.py
- If you are not doing thisprogram will not be saved on board.
- Now everything is complete, saving on board will execute program whenever the board is powered up.

CHAPTER – 7

FINAL HARDWARE







CHAPTER – 8

APPLICATIONS

- Obstacle avoiding robots can be used in almost all mobile robot navigation systems.
- They can be used for household work like automatic vacuum cleaning.
- They can also be used in dangerous environments, where human penetration could be fatal.

CHAPTER – 9

CONCLUSION

This project developed an obstacle avoiding robot to detect and avoid obstacles in its path. The robot is built on the raspberry pi pico platform for data processing and its software counterpart helped to communicate with the robot to send parameters for guiding movement. For obstacle detection,ultrasonic distance sensors were used that provided a wider field of detection. The robot is fully autonomous and after the initial loading of the code, it requires no user intervention during its operation. When placed in unknown environment with obstacles, it moved while avoiding all obstacles with considerable accuracy. In order to optimize the movement of the robot, we have many considerations for improvement. However, most of these ideas will cost more money and time as well. In future cameras can be used to detect the obstacle however, it is better to get CCD or industrial use ones to get clear and fast pictures. Even the ones we mentioned in the camera holder part will be better because of the special software.

CHAPTER – 10

REFERENCES

- https://www.ijert.org/research/obstacle-avoidance-robot-using-arduino-IJERTCONV6IS13197

- https://www.researchgate.net/publication/323393788_Development_of_an_Arduino-based_obstacle_avoidance_robotic_system_for_an_unmanned_vehicle#:~:text=In%20conclusion%2C%20an%20obstacle%20detection,the%20Arduino%20microcontroller%20which%20interprets