Data Acquisition and Hardware Setup Development for Implementation of Feedback Control and Obstacle Detection for Car Type Robot

PRESENTED BY

AKSHAY ADINATH DHOTRE

M. TECH. ELECTRONICS, WALCHAND COLLEGE OF ENGINEERING, SANGLI.

International Conference on Advancements in Automation Robotics & Sensing June 23-24, 2016

In Association with ISA Bangalore Section

ICAARS 2016

make in India by robots

Outline

- Introduction
- Hardware and Software Implementation
- Demo Videos
- Conclusion and Future Scope
- References

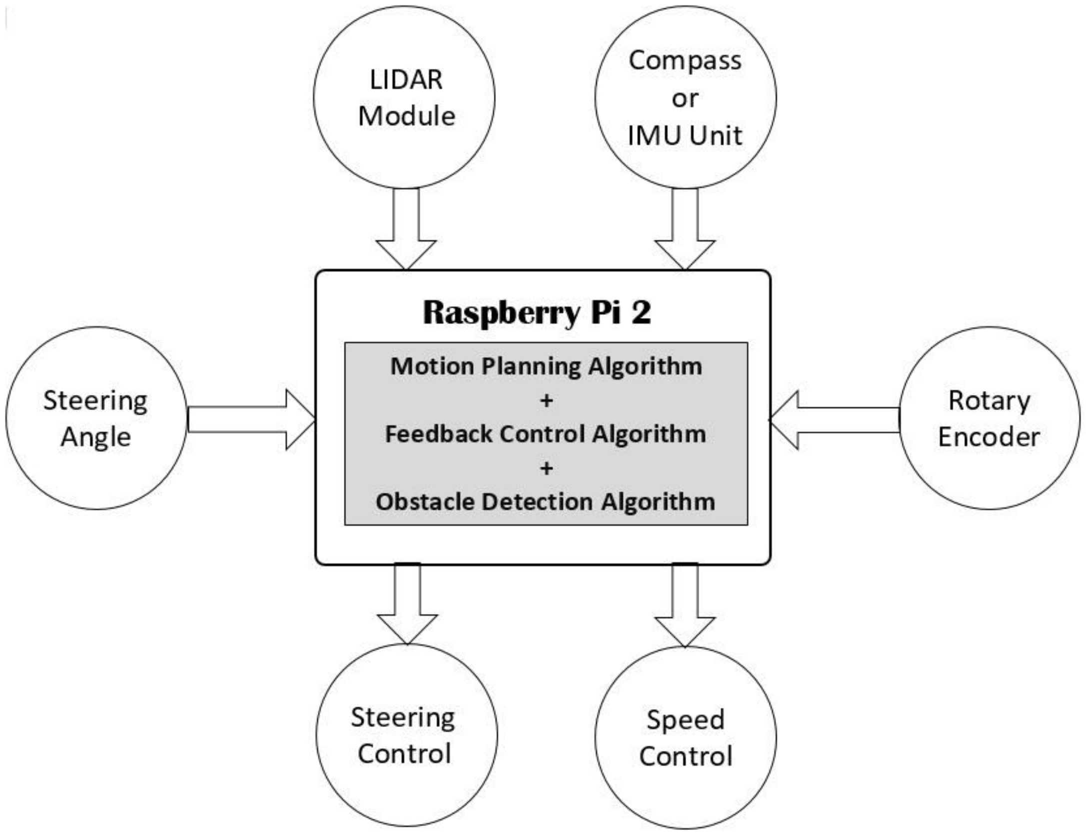
Introduction

- Modern cars are becoming intelligent.
- They contain many systems like ABS, Traction control
- In future auto driving , auto parking, obstacle avoidance will be included in them.
- For implementing these features in system, hardware needs to be powerful and reliable.
- It is a curtail part in design of system.

Objectives

- Interfacing Magnetometer, Optical Encoder, Potentiometer, LIDAR module, DC motor and Servo Motor with single Board Computer (SBC) and testing them.
- Implementing Open loop and closed loop control on hardware and testing it.
- Obtaining obstacle data around the robot and using that in motion planning algorithm.

Block Diagram



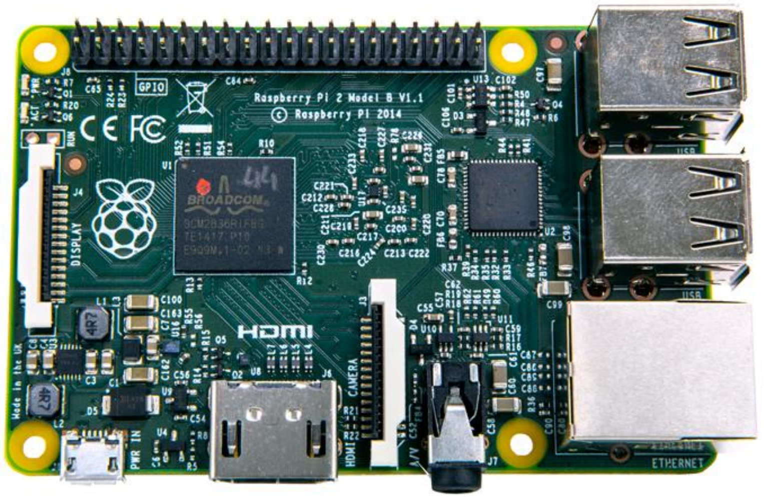
Hardware Implementation using Raspberry Pi

Raspberry Pi is used as main controller.

It has A 900MHz quad-core ARM Cortex-A7 CPU, 1GB RAM

It supports I2C, SPI and Searial Communication

Many libraries are avilable for implementing various functions

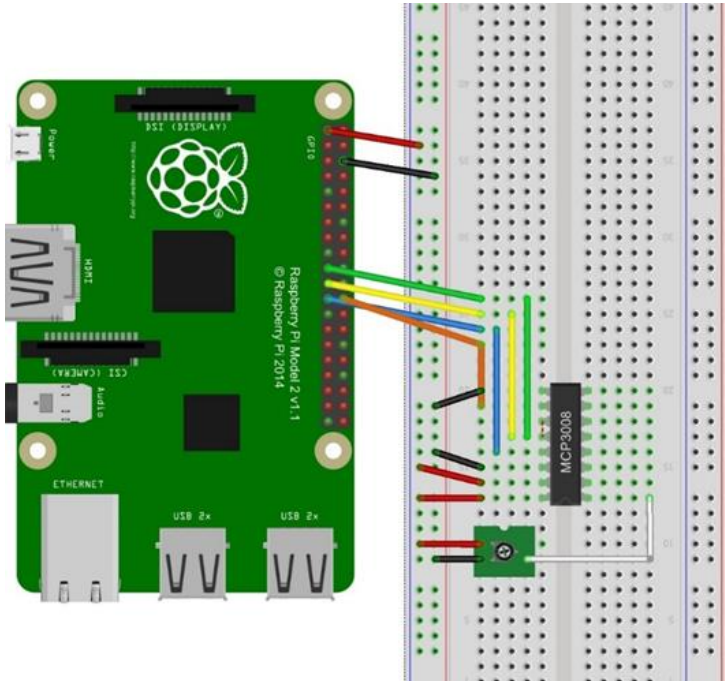


Implementation of Pulse Width Modulation

- RPi has both hardware and software PWM support
- Many Libraries support for motor control using software PWM
- WiringPi library is used for generating software PWM
- PWM signals are given to Steering Servo Motor and Driving DC Motor.
- Inputs:
  - Time of motion
  - Starting co-ordinates and End co-ordinates
  - Starting angle and End angle

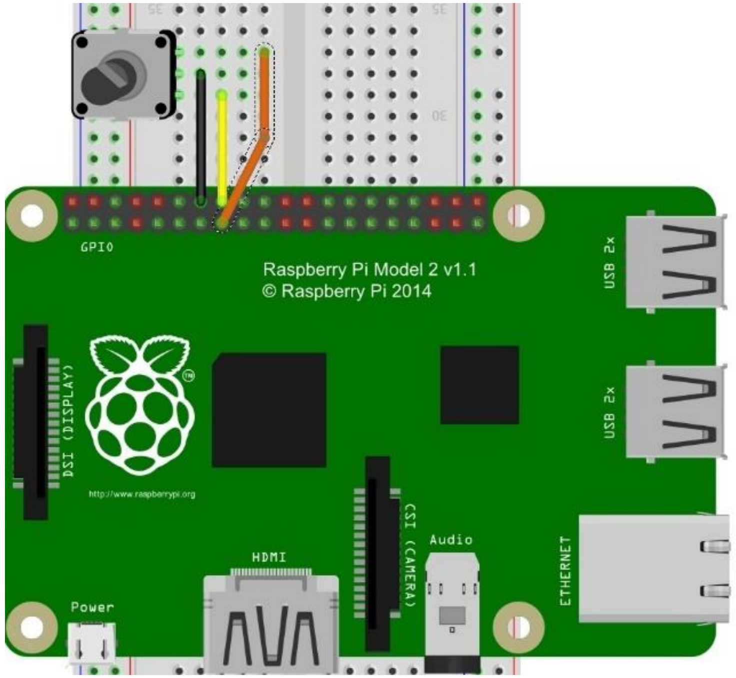
Reading Analog Values

- RPi doesn’t have analog reading capability so it requires external ADC.
- MCP 3008 is useful for this purpose which has 8 channels and 10 bit resolution.
- Communicates with RPi using full duplex SPI protocol.



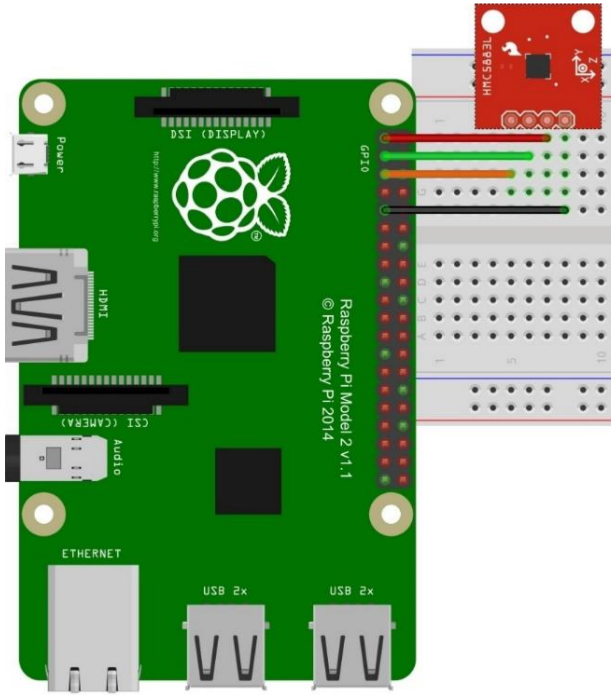
Interfacing Encoder

- Encoder is a device which is useful to find distance travelled and speed.
- As the encoder shaft is turned, the switches will alternatively open and close so that they progress through a series of states known as Gray Code.
- By reading any two successive states, one can tell which direction the encoder is turning.



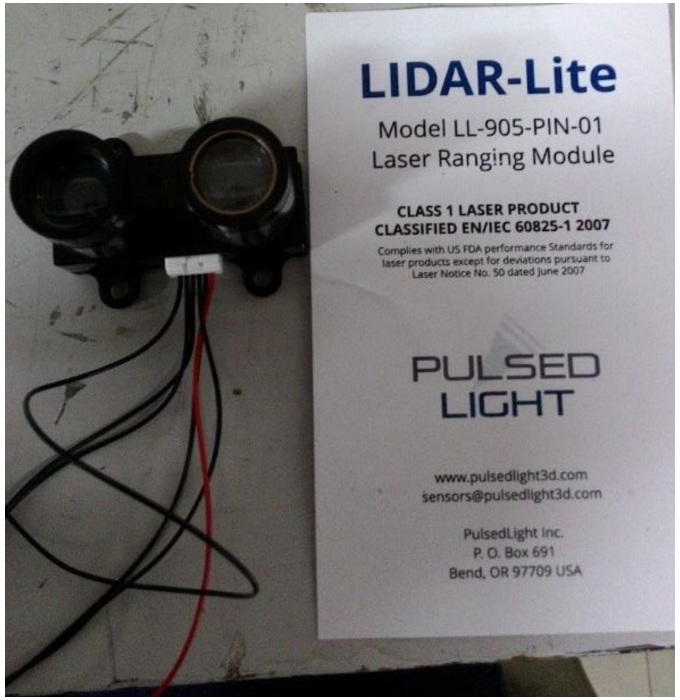
Interfacing Magnetometer

- Magnetometer module measures the earths magnetic field in three dimensions.
- Can measure raw magnetic strength of nearby magnetic source
- Magnetometer module HCM5883L can be connected to raspberry pi using I2C bus.
- It is used to get orientation angle of robot.



Interfacing LIDAR Module

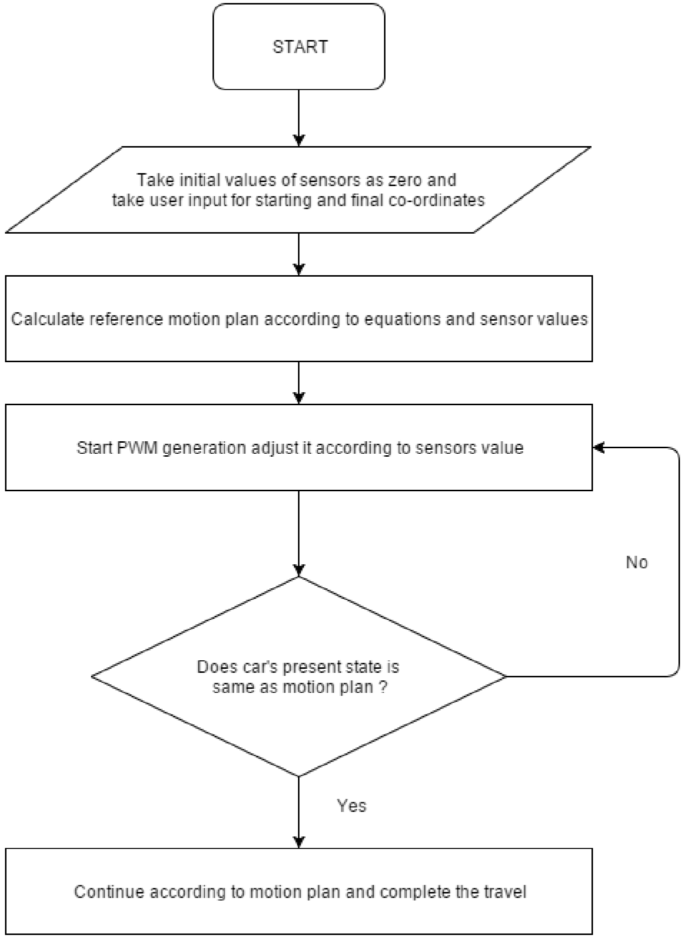
- To detect obstacles nearby, vision is needed.
- LIDAR is Light Detection and Ranging, similar to RADAR
- Use of laser for detection increases accuracy and speed.
- Lidar based module used is Pulsed-Light Lidar laser rangefinder.
- This module has I2C interface to communicate.



Accessing RPi remotely using Wi-Fi

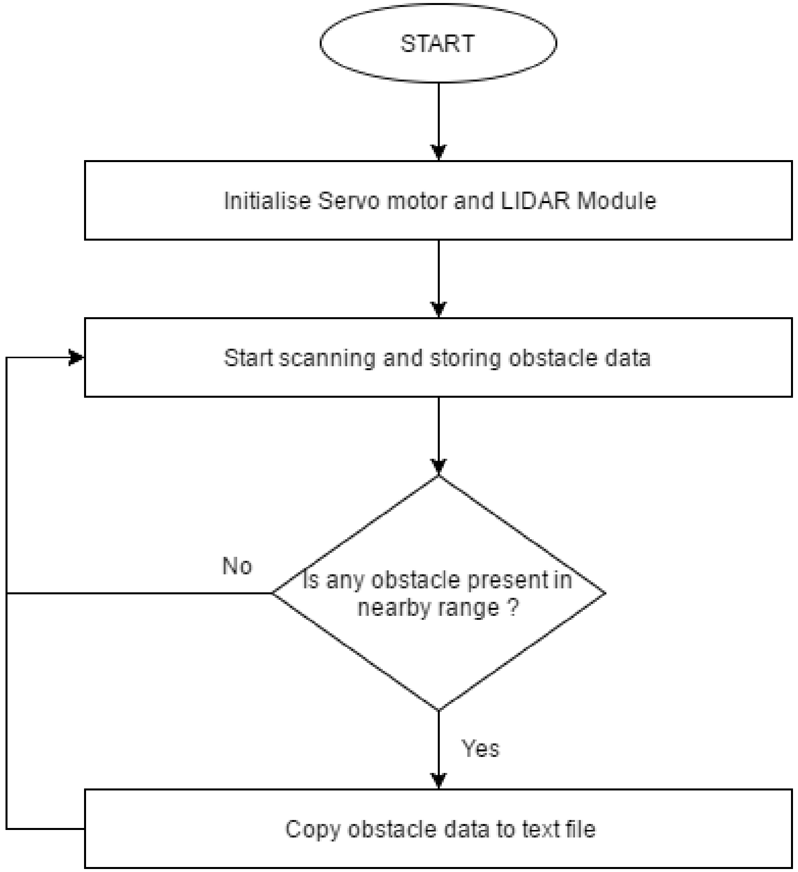
- One more feature of Raspberry Pi is accessibility from remote location.
- Here controlling system acts ac hotspot and RPi connects to it as client.
- This enables remote monitoring of robots, useful for troubleshooting and tracking.
- Various plug and play Wi-Fi modules are available for RPi and newer generation of RPi contains built in Wi-Fi module.

Feedback Control Implementation



- Separate function for data acquisition from each sensor.
- Called these functions from main program.
- Car’s state is calculated according to sensor data
  - Orientation angle from magnetometer
  - Speed and distance from encoder
  - Steering angle from potentiometer
- The present state is matched to state calculated by motion planner.

Obstacle Detection Implementation



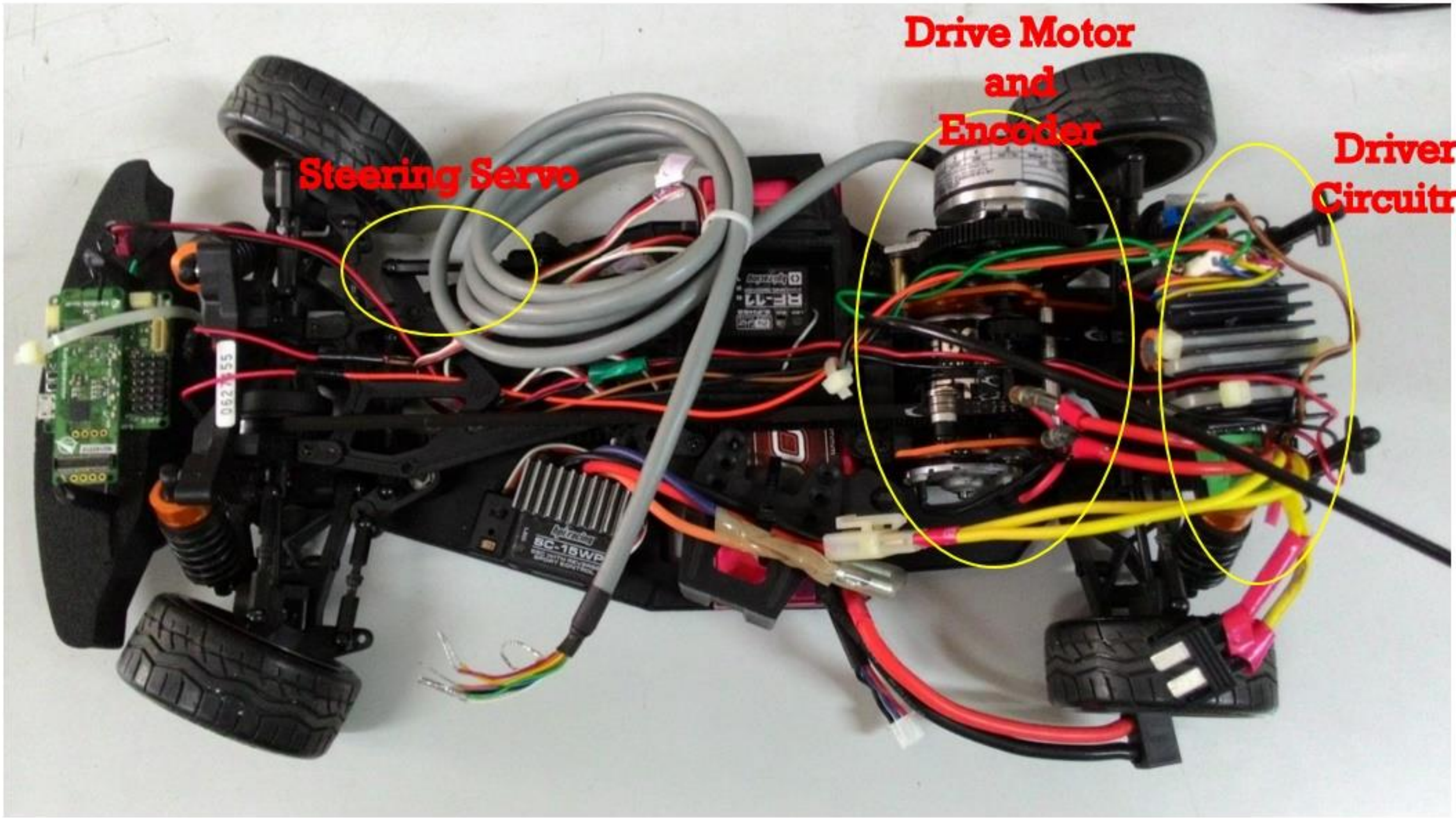
- For detection of obstacle nearby car, continuous scanning is necessary.
- This can be implemented using LIDAR mounted on Servo Motor.
- Threshold value is used to decide nearby obstacle.
- Distance and angle of nearby obstacle is stored in separate file for immediate processing.

Hardware Implementation

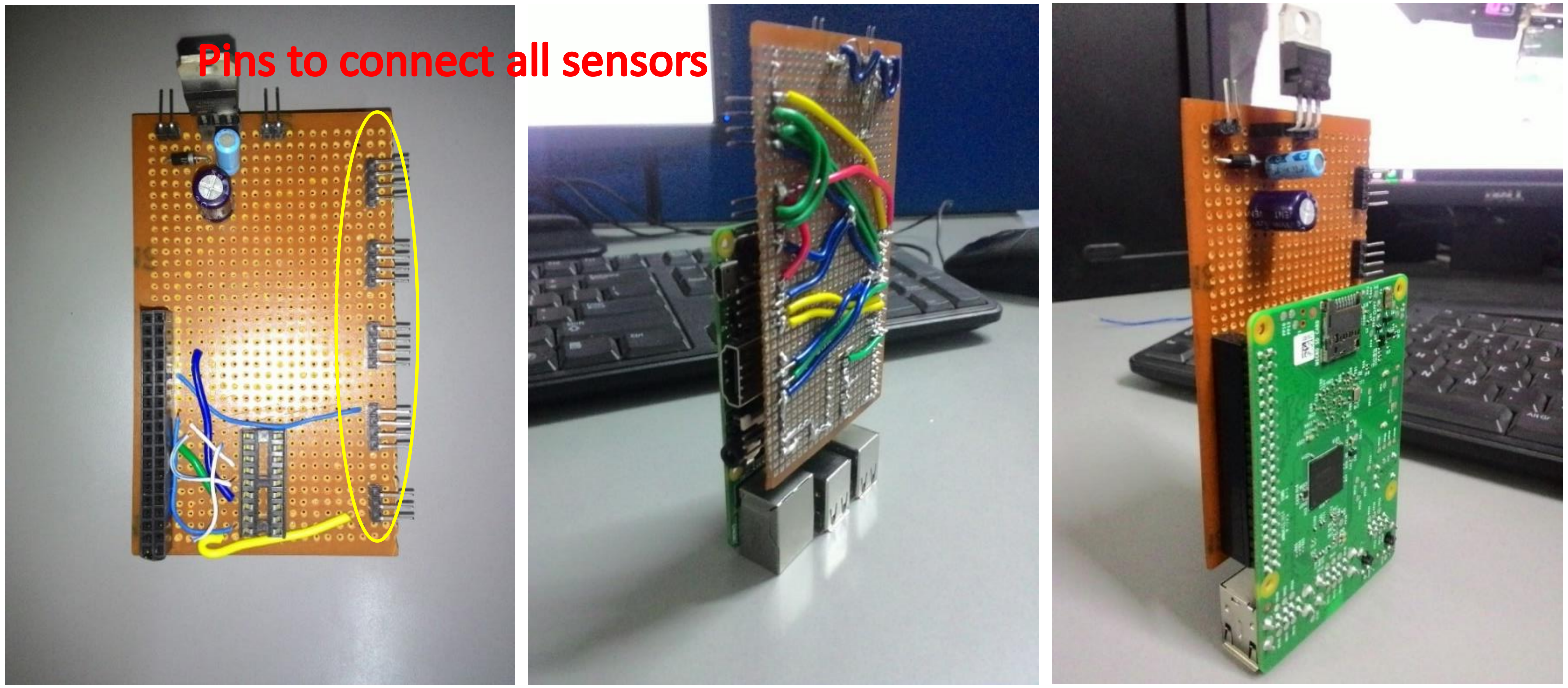
Prototype Model

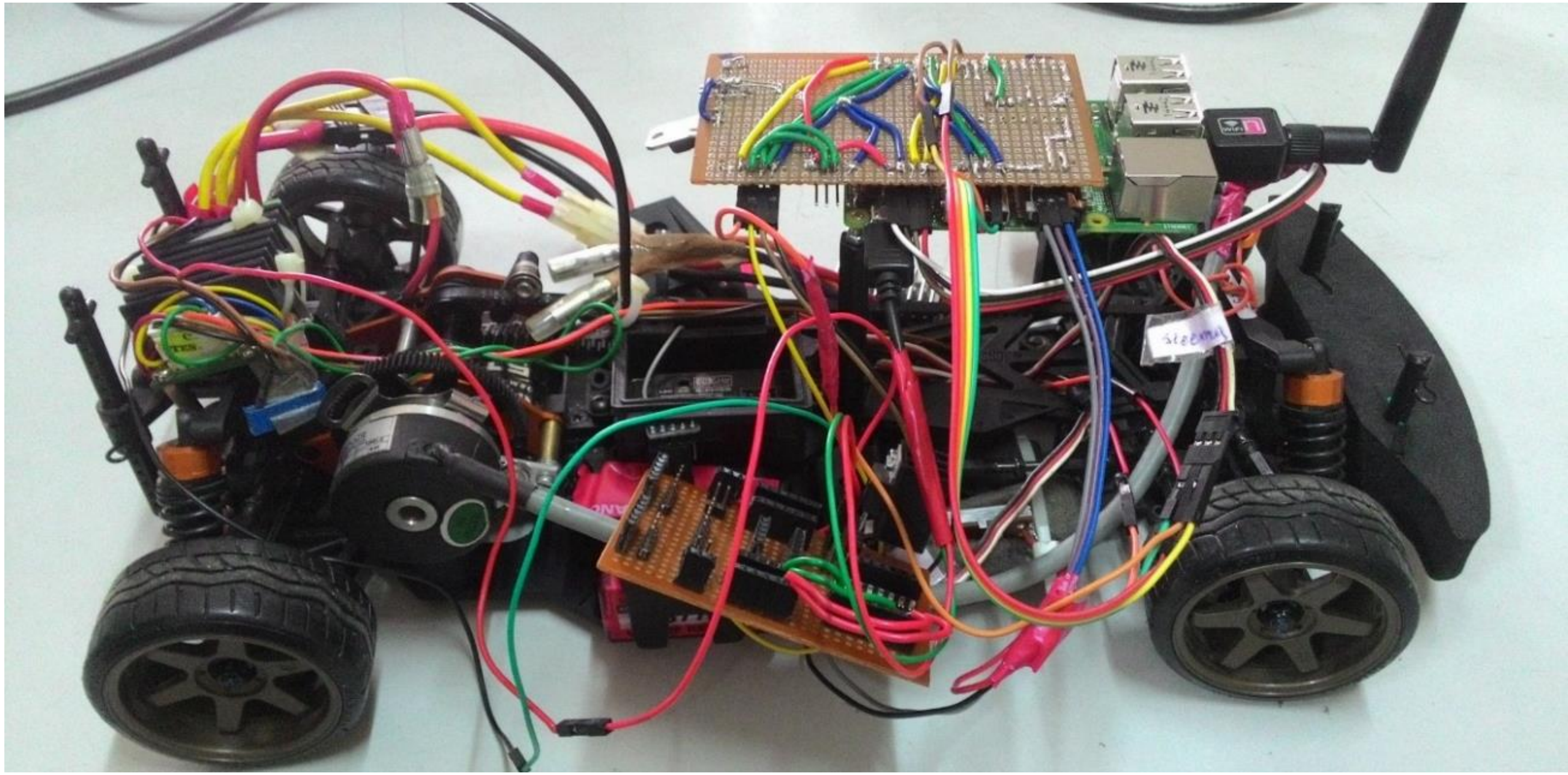
Car model with 1:10 scale of original car (Chevrolet Camaro) with exact mechanisms as original car like suspension, differential drive for wheels and steering mechanism

Car Model



Connecting All Sensors To RPi





Results

Magnetometer Data

```
****************************************
x=-4096, y=-786, z=-2
angle = -169.1


****************************************

****************************************
x=-4096, y=-786, z=1
angle = -169.1


****************************************
```

- To calculate orientation angle from x, y and z values, inverse tangent function can be used.

$$\text{Orientation Angle} = \frac{tan^{-1}\left(\frac{x}{y}\right)\times 360}{\pi}$$

- As car has movement is in x-y plane, no need to consider z axis value produced by magnetometer.

Encoder Data

```
Hello!
Distance Travelled = 0.000013 meters
Distance Travelled = 0.000026 meters
Distance Travelled = 0.000039 meters
Distance Travelled = 0.000052 meters
Distance Travelled = 0.000066 meters
Distance Travelled = 0.000079 meters
Distance Travelled = 0.000092 meters
Distance Travelled = 0.000105 meters
Distance Travelled = 0.000118 meters
Distance Travelled = 0.000131 meters
Distance Travelled = 0.000144 meters
Distance Travelled = 0.000157 meters
Distance Travelled = 0.000171 meters
Distance Travelled = 0.000184 meters
Distance Travelled = 0.000171 meters
```

- To calculate distance travelled using readings from encoder use formula:

$$\text{Distance Travelled} = \frac{(No.\ of\ encoder\ pulses) \times \pi \times (Diameter\ of\ Wheel)}{Number\ of\ pulses\ per\ rotation}$$

Potentiometer Data

```
**************************************
Voltage of Potentiometer:  1.801529 V
Steering angle from Potentiometer:  26.059483 steering_angle

**************************************

**************************************
Voltage of Potentiometer:  1.801819 V
Steering angle from Potentiometer:  26.070005 steering_angle

**************************************

**************************************
Voltage of Potentiometer:  1.801755 V
Steering angle from Potentiometer:  26.067665 steering_angle

**************************************

**************************************
Voltage of Potentiometer:  1.801465 V
Steering angle from Potentiometer:  26.057142 steering_angle

**************************************
```

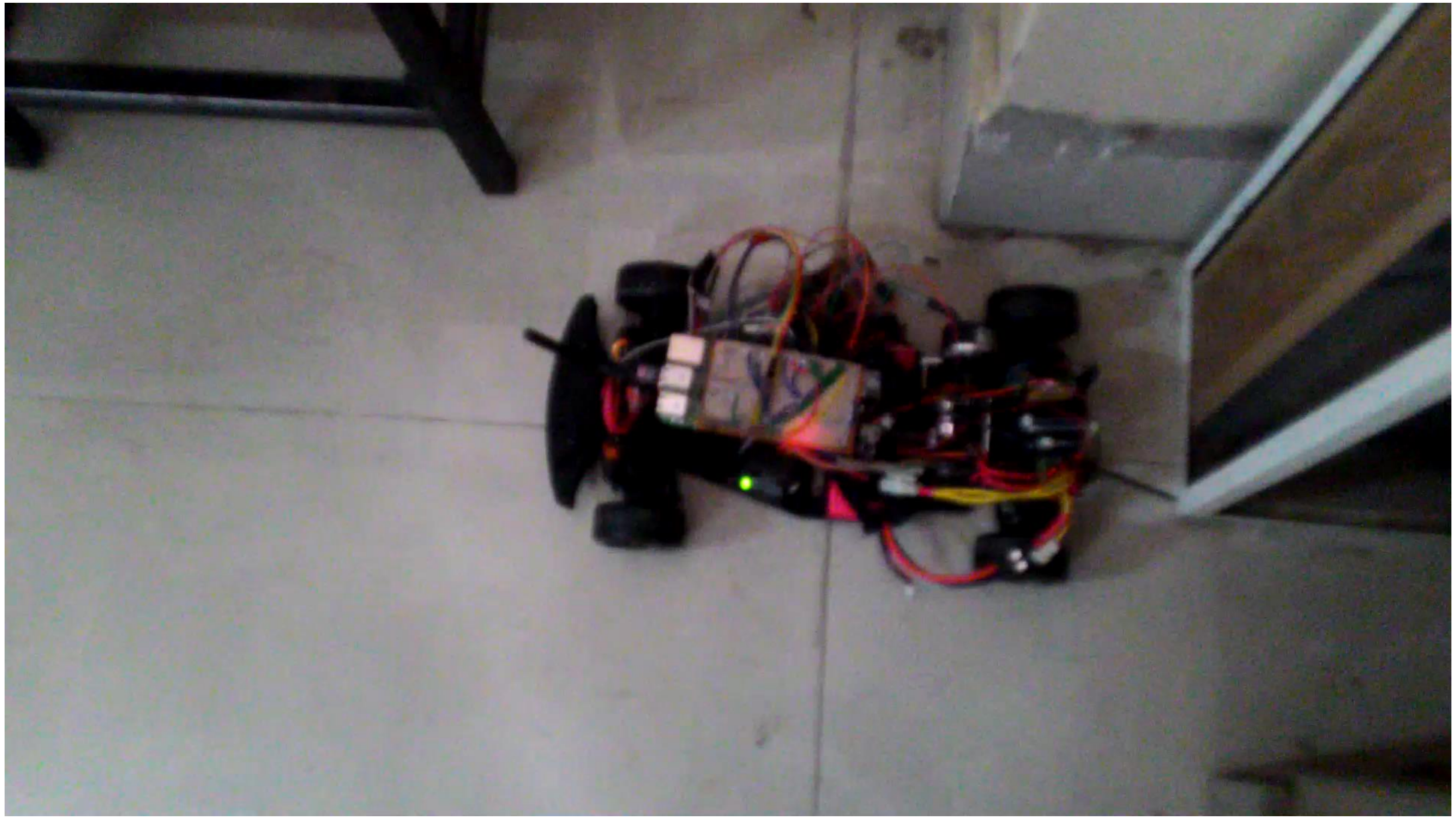
- To measure voltage of potentiometer we need to multiply ADC count output by ADC reference voltage and divide it by resolution.

Voltage =

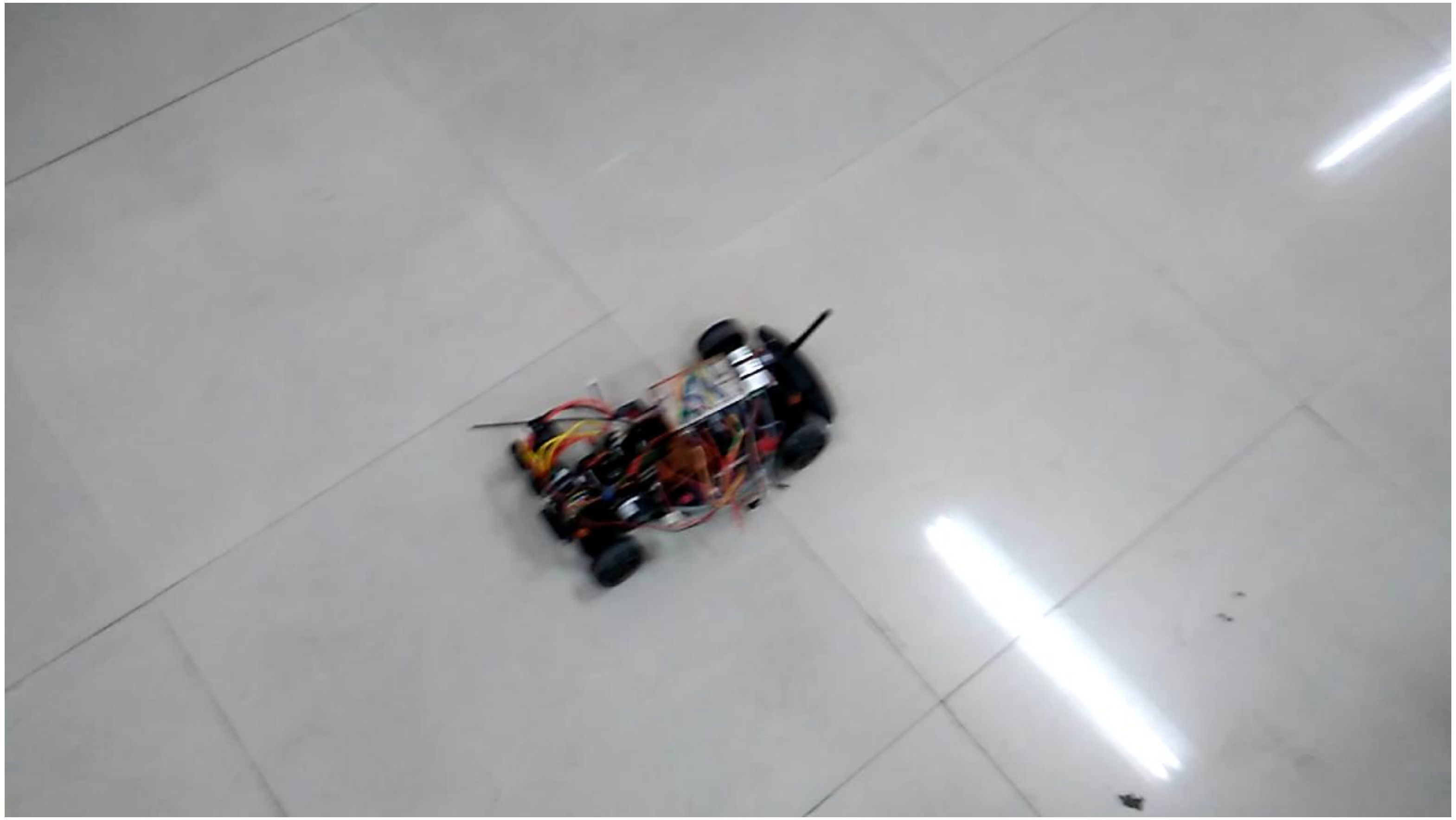
$$\frac{Count\ from\ ADC\ reading\ \times\ Reference\ voltage}{2^{Resolution\ of\ ADC\ in\ bits}}$$

Demo Videos

Open Loop Test



Closed Loop Test



Obstacle detection

References

- Agarwal N, Walambe R., Joshi V, Rao A. ’Integration of Grid Based Path Planning with a Differential Flatness Based Motion Planner in a Nonholonomic Car-type Mobile Robot’ in Institution of Engineers Journal, November 2015, Pune, India
- Walambe R.A, Agarwal N. , Kale S. Joshi V. ”Optimal Trajectory Generation for Car-Type Mobile Robot Using Spline Interpolation” in proceedings of ACODS 2016, Feb 2016, Trichy, India
- Jeffery Young, Milan Simic, ‘LIDAR and Monocular Based Overhanging Obstacle Detection’, *Knowledge-Based and Intelligent Information & Engineering Systems 19th Annual Conference, KES-2015, Singapore*, Volume 60, 2015, Pages 14231432, September 2015.
- LU Jiangzhou, Sepanta Sakhavat, XIE Ming and Christian Laugier, ‘Sliding Mode Control for Nonholonomic Mobile Robot’, *In Proc. Of the Int. Conf. on Control, Automation, Robotics and Vision, Singapore*, December 2000.
- A. De Luca, G. Oriolo and C. Samson, ‘Feedback Control of a Nonholonomic Car-Like Robot’, *Lectures Notes in Control and Information Sciences 229*. Springer, ISBN 3-540-76219-1, 1998, 343p.

References

- *http://www.instructables.com*
- *http://blog.oscarliang.net*
- *http://wiringpi.com*
- *http://www.modmypi.com*
- *http://theatticlight.net/posts/Reading-a-Rotary-Encoder-from-a-Raspberry-Pi*
- *https://github.com*
- *http://brundell.blogspot.in/2015/06/hmc5883l-magnetometer-toraspberry-pi.htm*
- *http://www.howtogeek.com/167425/how-to-setup-wi-fi-on-your-raspberry-pi-viathe-command-line/*

Acknowledgement

This project is funded under the WOS-A scheme (SR/ET/WOS 34/2013-14) of Dept. of Science and Technology, Govt. of India.

I am thankful to Dr. V. A. Joshi and Dr. Rahee Walambe (PVGCOET, Pune) for giving me the opportunity to work on this project.

Also I express my gratitude towards Dr. Mrs. S. S. Deshpande my project guide for encouraging me throughout the project work.

Questions ?

Thank You !!!