Automatic Floor Cleaner Navigation Algorithm

First Review

Date: 7/9/2020(DD/MM/YYYY)

MADHUKAR TEMBA

19BAI1161

INTRODUCTION

The aim is to develop an algorithm for a floor cleaning robot which allows it to return to its charging station.

OBJECTIVE

We need to develop a way through which the robot can know the position of the charging station with high accuracy.

ABSTRACT

For that I decided to use IR sensors to figure out the location.

Some changes are made due to….

- The current robot cleaner does clean well but repeatability of its steps is low. That means the robot cannot precisely execute an action every time.

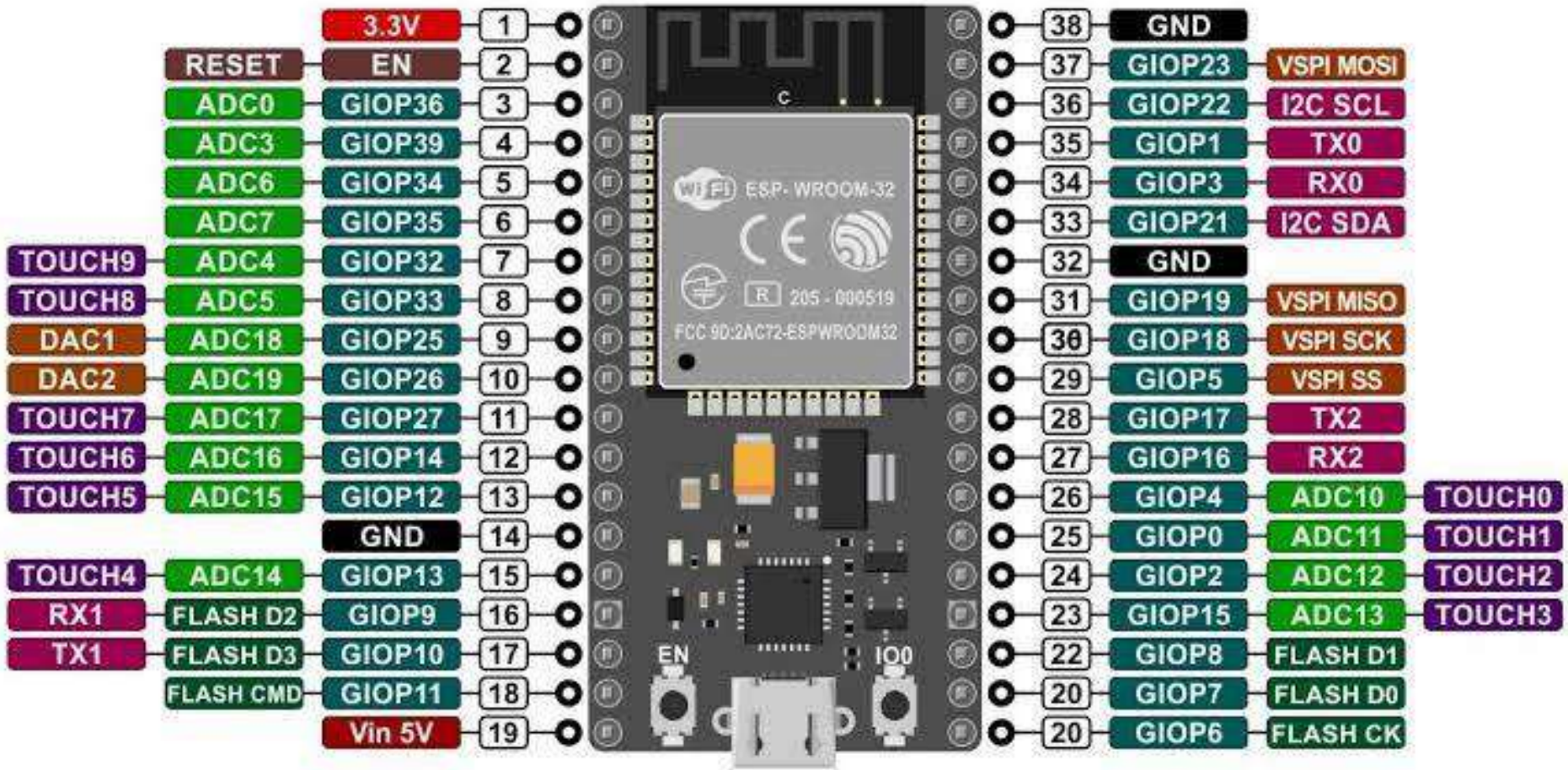
- The gyroscope sensor being tested suddenly stops responding due to unknown reason. Connected to main controller by I2C bus.

- The robot tends to get stuck at platforms elevated at 2.25cm.

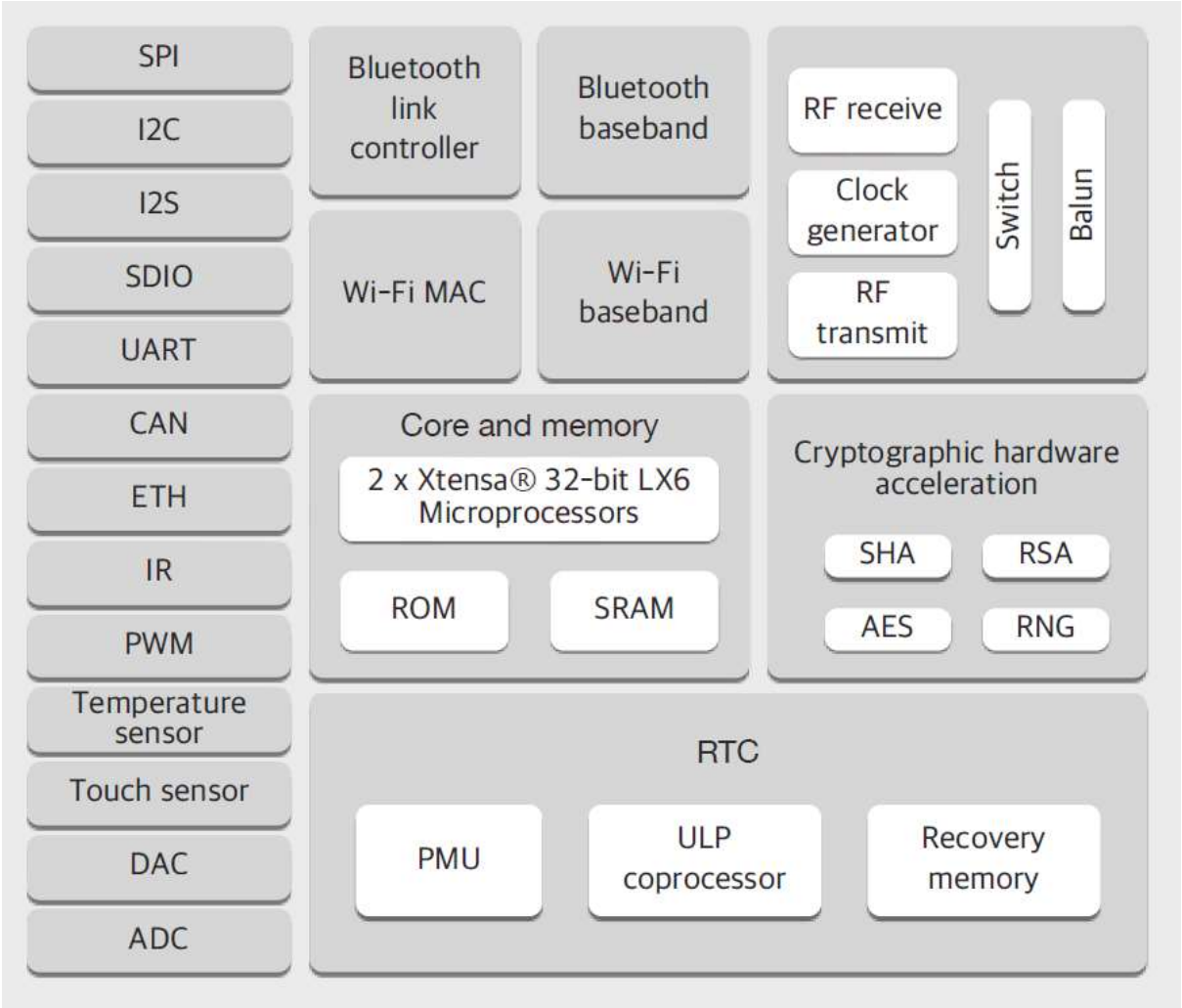
Changes are:

- Decided to redesign addressing all the issues as I still want to use the current robot.
- Using stepper motors with 2048 steps per revolution this time.
- Making it low profile with spring loaded tires and using a different microcontroller.

ESP 32


3.3V 1
RESET EN 2
ADC0 GIOP36 3
ADC3 GIOP39 4
ADC6 GIOP34 5
ADC7 GIOP35 6
TOUCH9 ADC4 GIOP32 7
TOUCH8 ADC5 GIOP33 8
DAC1 ADC18 GIOP25 9
DAC2 ADC19 GIOP26 10
TOUCH7 ADC17 GIOP27 11
TOUCH6 ADC16 GIOP14 12
TOUCH5 ADC15 GIOP12 13
GND 14
TOUCH4 ADC14 GIOP13 15
RX1 FLASH D2 GIOP9 16
TX1 FLASH D3 GIOP10 17
FLASH CMD GIOP11 18
Vin 5V 19
38 GND
37 GIOP23 VSPI MOSI
36 GIOP22 I2C SCL
35 GIOP1 TX0
34 GIOP3 RX0
33 GIOP21 I2C SDA
32 GND
31 GIOP19 VSPI MISO
30 GIOP18 VSPI SCK
29 GIOP5 VSPI SS
28 GIOP17 TX2
27 GIOP16 RX2
26 GIOP4 ADC10 TOUCH0
25 GIOP0 ADC11 TOUCH1
24 GIOP2 ADC12 TOUCH2
23 GIOP15 ADC13 TOUCH3
22 GIOP8 FLASH D1
20 GIOP7 FLASH D0
20 GIOP6 FLASH CK
ESP-WROOM-32
EN
IO0

BLOCK DIAGRAM



EXISTING SYSTEMS

- LIDAR sensors
- Ultrasonic sensors
- 360 degree infrared sensors
- IR proximity sensors.

LIMITATIONS OF EXISTING SYSTEMS

- LIDAR sensors are expensive and also need a fast processor to process all the data.

- Ultrasonic sensors may not detect an obstacle at a specific angle and are slow as compared to light based sensors. Also modules available in the market have prebuilt ICs which give fixed output making custom implementation hard.

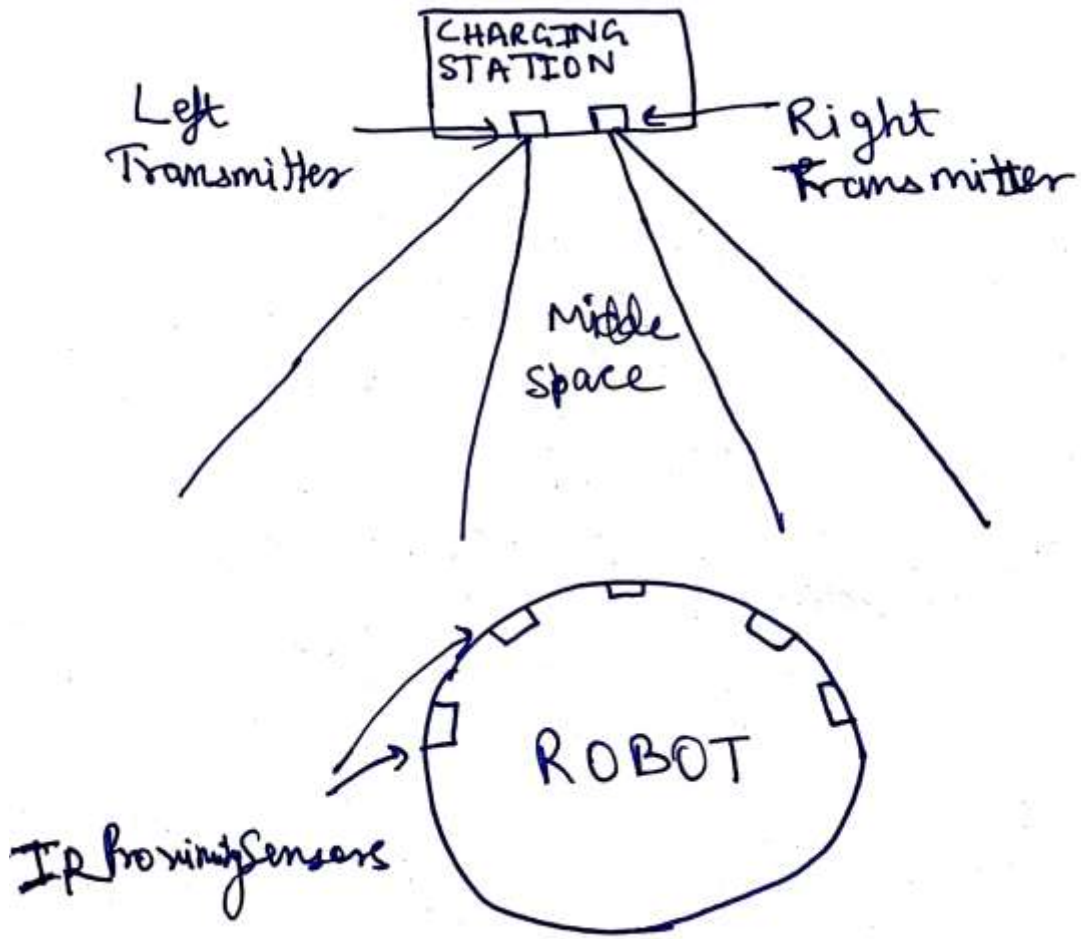
- IR proximity sensors can be affected by sunlight or other devices such as remotes etc.

PROPOSED SYSTEM

- I am using IR proximity sensors to detect obstacles but with a custom code allowing them to transmit and receive data also.

- There are 5 sensors which will be placed in front of the cleaner and each sensor will be able to receive a specific code from the charging station.

WORKING


CHARGING STATION
Left Transmitter
Right Transmitter
Middle space
ROBOT
IR Proximity Sensors

WORKING

- When the robot reaches near the base station by backtracking it will turn off its IR LEDs and checks each sensor for the transmitted code.
- If the robot receives the LEFT transmitter’s code only then it will turn right a bit. It will turn left if it receives the right transmitters code.
- If it receives both transmitter’s code or nothing then it will move forward.
- The robot will stop when it detects that it is charging.

CUSTOM CODE:

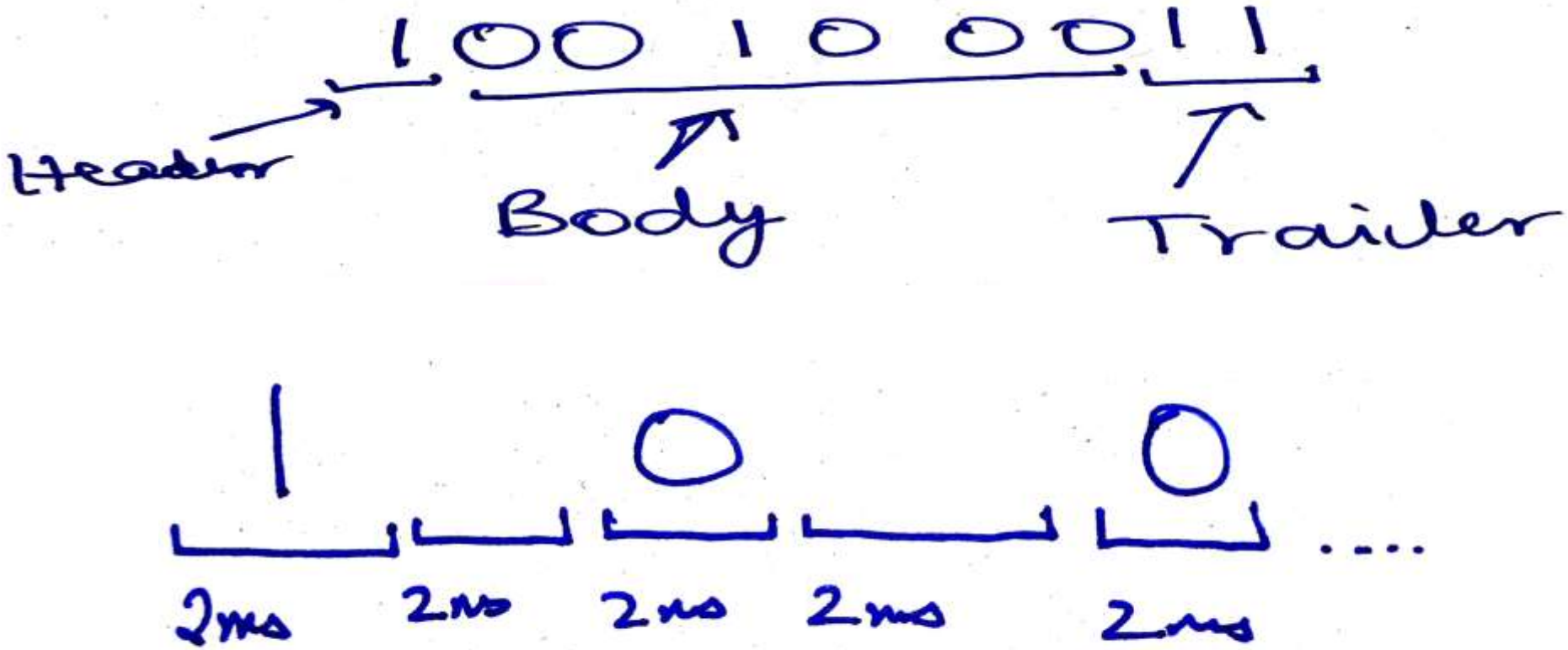
- The left and right transmitter will send an 9 bit code. The code will include a header and trailer.

LEFT Transmitter will send: 100100011

RIGHT Transmitter will send: 100010011

The robot when in the centre will receive 100110011 or nothing.

CODE INCLUDES



WORKING

As soon as the robot receives an high signal it starts its timing and stores the 8-bit code in an array.

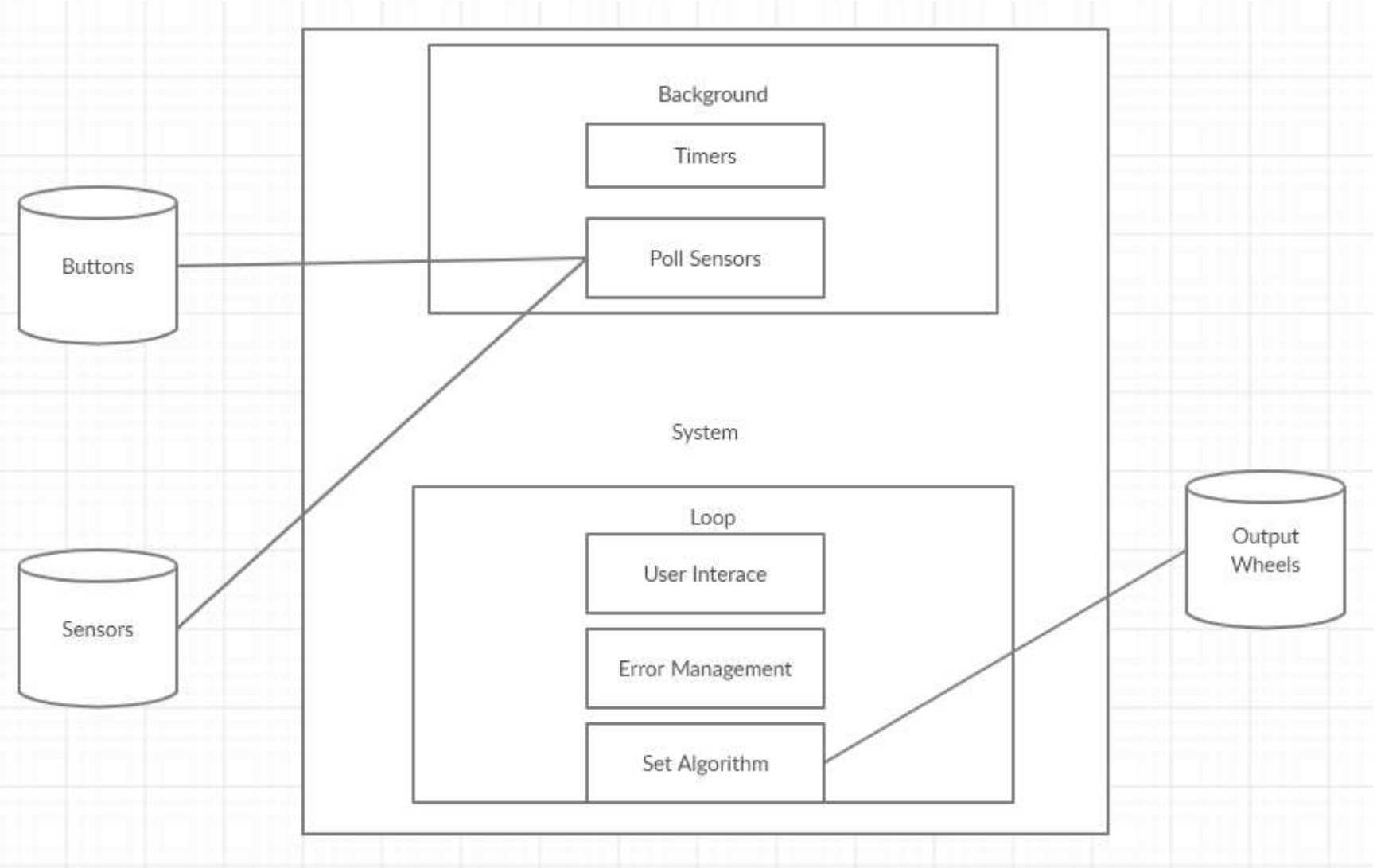
Then it checks if the trailer is correct or not. If not then it rejects the code.

ADVANTAGES OF PROPOSED SYSTEM

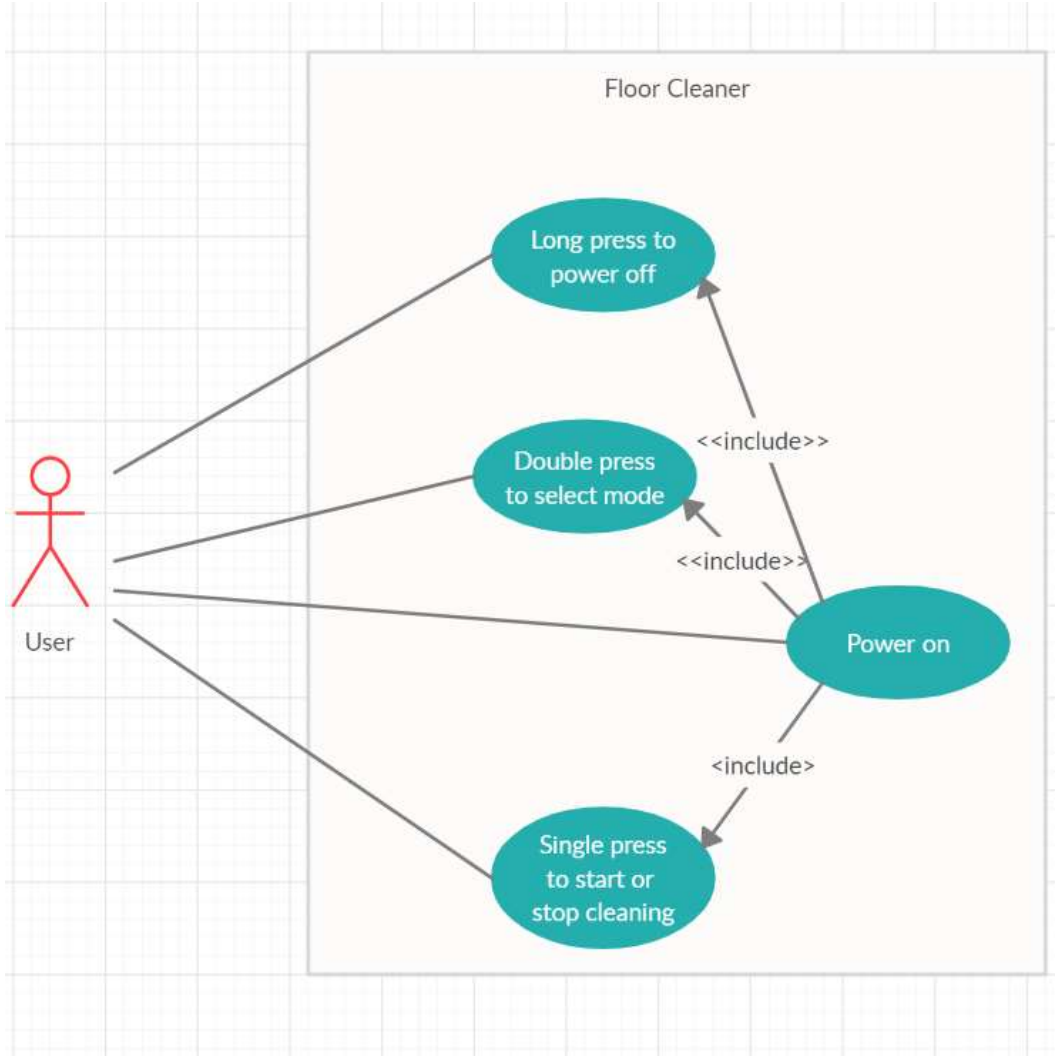
The same sensor performs 2 tasks that is obstacle detection and data receiving.

The sensors cost less in comparison to other methods.

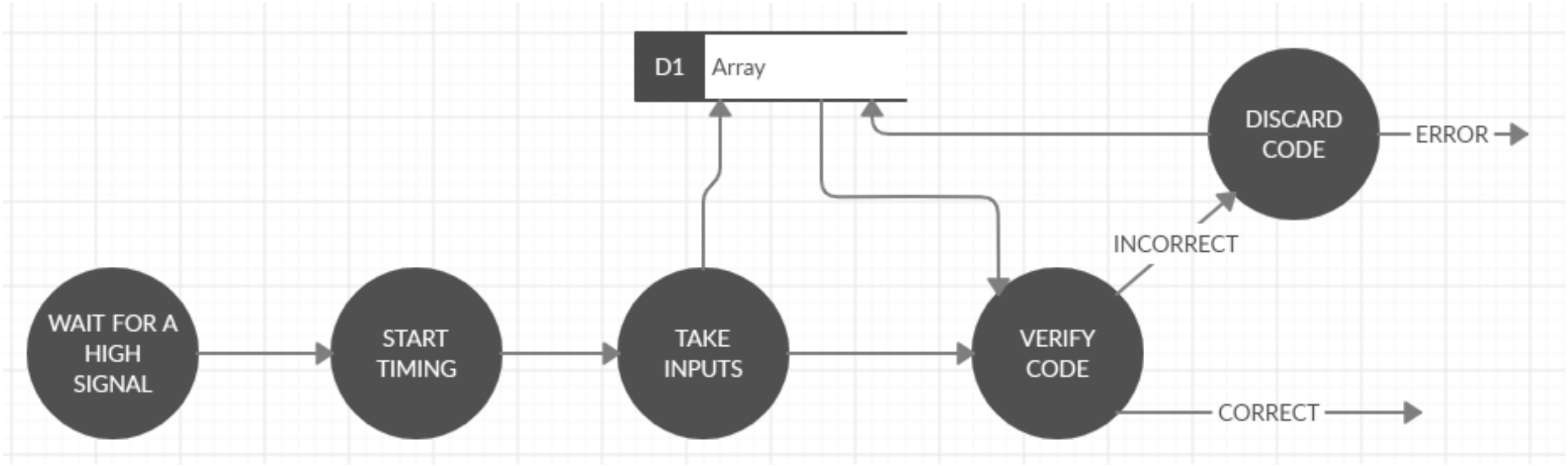
SYSTEM ARCHITECHTURE



USE CASE DIAGRAM



DATA FLOW DIAGRAM(ONLY FOR IR SENSOR)



IMPLEMENTATION

Video has been removed due to file size limitation. Please go to the link given in review 3 to view all the videos.

CONCLUSION

Implemented the IR transmission and receiving function of the code.

REFERENCES

- https://www.youtube.com/user/EEVblog

- https://www.youtube.com/channel/UCu7_D0o48KbfhpEohoP7YSQ

Videos:

#171 Arduino Guide to Infrared (IR) Communication also for ESP32 and ESP8266: https://www.youtube.com/watch?v=gADIb1Xw8PE

EEVblog #980 - RoboMaid Automated Vacuum Cleaner Teardown:

https://www.youtube.com/watch?v=NJvBQoIb5lg