OBSTACLE AVOIDANCE GROUND ROBOT

BRUTE FORCE

TEAM 11

BASIC IDEA BEHIND THE PROJECT

Obstacle Avoidance Robot is a device designed to automatically sense the obstacle in front of it and navigate in an unknown environment by avoiding collisions. This is a primary requirement for any autonomous mobile robot.

The robot developed in this project uses ultrasonic sensor to detect obstacles in real time and requires no path planning. Its processing unit is based on the Arduino platform

MOTIVATION

The motivation is to use the technology in an efficient manner. In today's world, the number of vehicles is increasing exponentially and with that, some problems are increasing too, the major one being the accidents of vehicles.In the field of robotics, it is necessary for any autonomous robot to be able to navigate by avoiding obstacles. Obstacle avoidance is essential for autonomous robot’s navigation in complex and unknown environments.

Robots guided with this technology can be put into diversified uses, e.g., surveying landscapes, driverless vehicles, autonomous cleaning, automated lawn mower and supervising robot in industries.

OBJECTIVES

The robot developed in this project is expected to fulfill the following objectives:

- The robot would have the capacity to detect obstacles in its path based on a predetermined threshold distance.

- After obstacle detection, the robot would change its course to a relatively open path by making autonomous decision.

- It can measure the distance between itself and the surrounding objects in real-time.

- It would be able to operate effectively in unknown environment.

IMPLEMENTATION

The proposed method uses a microcontroller of Arduino family. An ultrasonic sensor mounted on a servo motor at the front end of the chassis is used to detect any obstacles in its path and sends a command to the microcontroller.

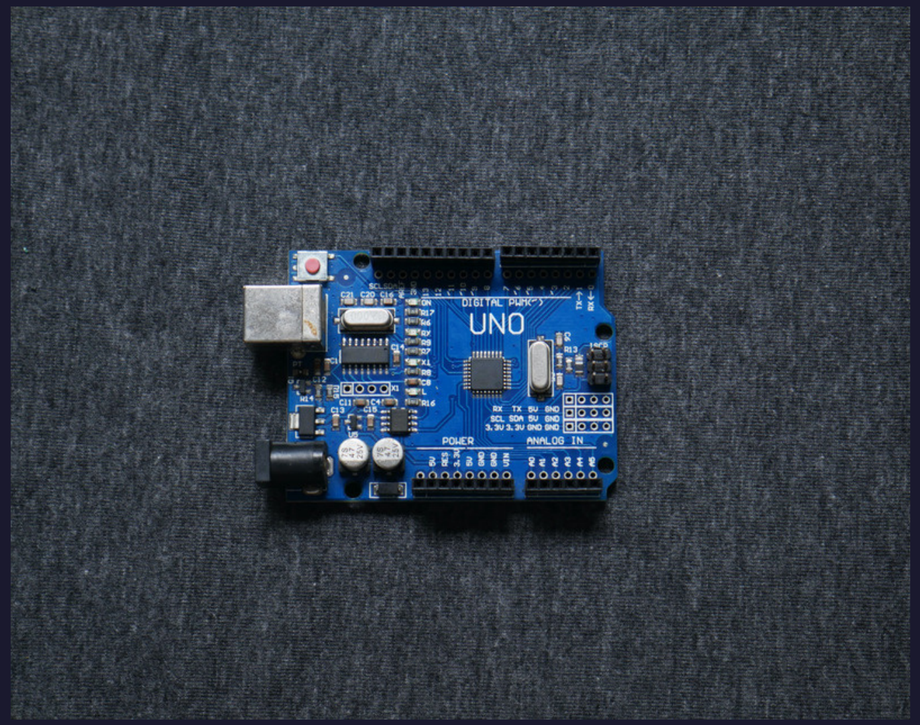
Depending on the input signal received, it redirects the robot to move in an alternate direction by actuating the motors which are interfaced with the motor driver. Arduino IDE (Integrated Development Environment) software is used to upload the program into the hardware.

COMPONENTS USED

- Arduino UNO Rev 2 :-

The microcontroller we used act as a central point of communication

.



- Ultrasonic sensor :-

use to find the distance between device and obstale

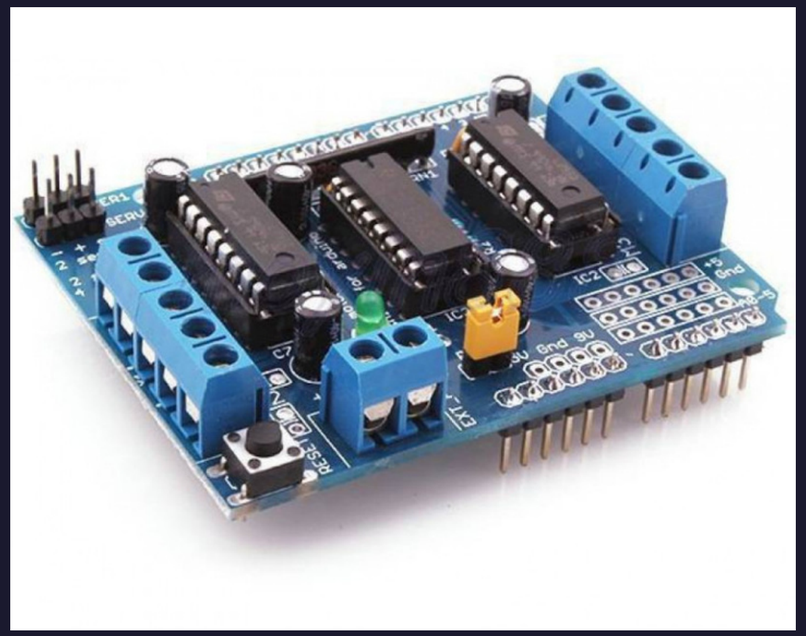
COMPONENTS USED

- BO Motor (x4)

use to rotate the wheels of the vehicles.

- Motor Driver (L293D)

act as a central point for the communication.



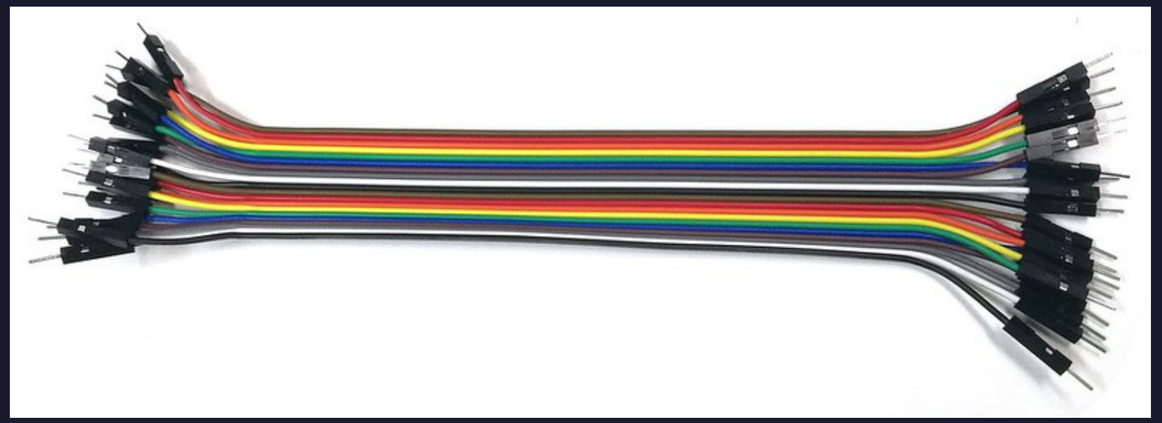
component used

Servo motor

used to rotate the ultrasonic in the range of 120 degree .

jumper wires

used to connect the different components .



component used

Robot Hardware - Chassis and wheels

Robot Chassis such as membrane, the motors to the Arduino supports, base plates, wheels, ball bearing etc. for building the robot structure and keeping all the components together.

component used

- Switch

used to connect and
disconnect the Powerbank .

- powerbank

to supply the external power to
the motors .

PSEUDOCODE

Step 0 : if the distance measured by left or right ultrasonic less then 20 cm , the vehicle will move in the opposite direction of the ultrasonic . Else, go to step 1.

Step 1: If both the distance are less than 20 cm, rotate and move backwards. Else, go to step 2

Step 2: Read forward distance. If greater than 30 cm, go to step 5. Else go to step 2.

Step 3: Stop, move backward, stop again.

Step 4: Look right, measure distance. Look left, measure distance.

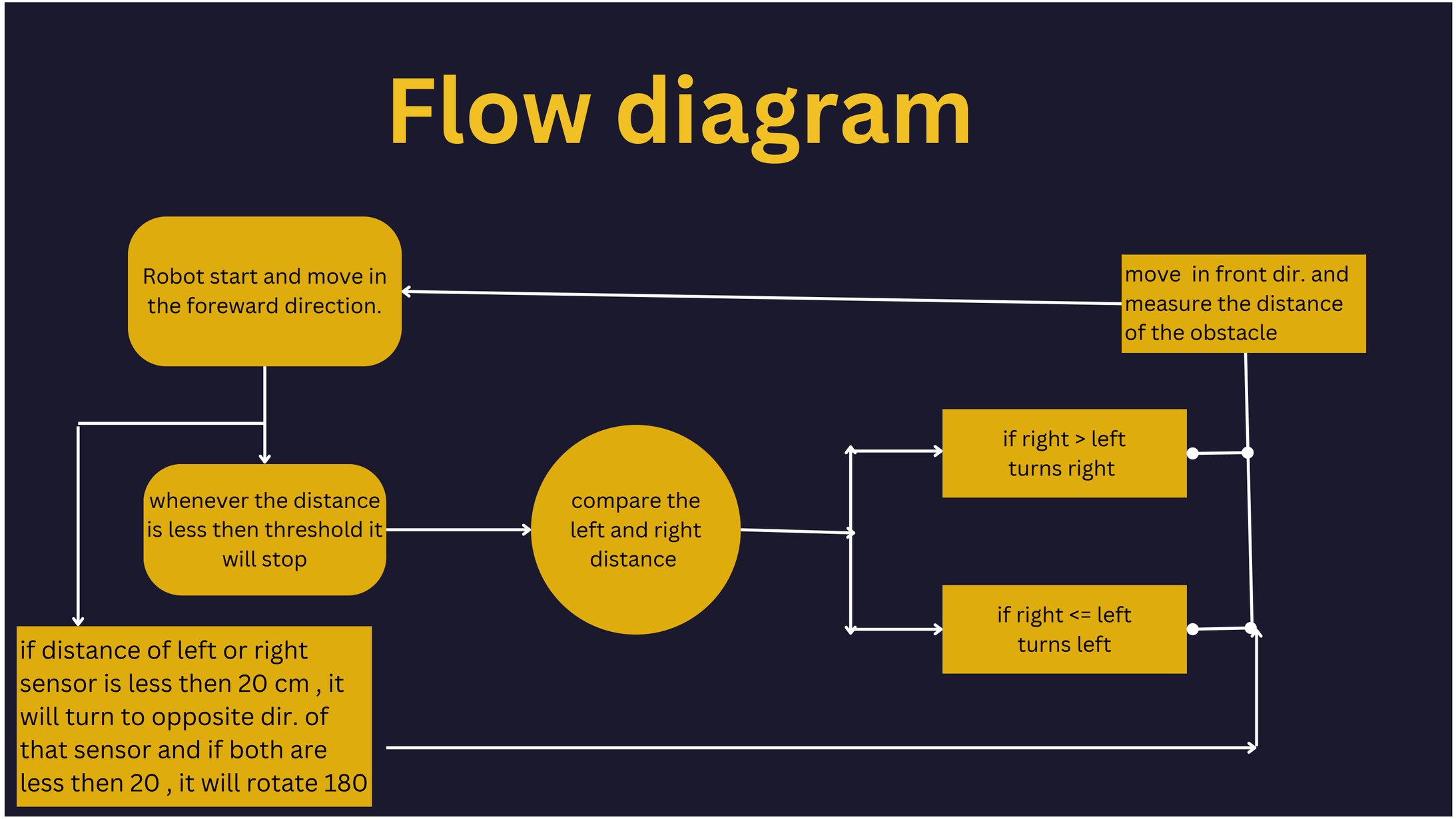
Pseudocode continue...

Step 5: If right distance is greater, turn right. Else if left distance is greater, turn left. Else, move backwards.

Step 6: Move forward.

Step 7: Go to step 1

Flow diagram



OM2M and thingspeak

- Method is secure and robust
- Implemented a sleep cycle like system
- Connections to OM2M and ThingSpeak servers are made after long delays to allow for the robot to move without interruptions for longer.
- Data aggregated each cycle and encoded into a JSON -
- Used the ThingSpeak bulk update API to push data to our ThingSpeak server 11
- JSON encrypted and stored in an OM2M container

DATA ANALYSIS WE TRIED TO CONDUCT

1. Proximity data is first collected from the ultrasonic sensor.
The data is then used to decide the direction of motion of the robot.

2. The expected x-y coordinates of the robot are calculated and a trajectory of the robot is plotted using these coordinates.

3. The encrypted data is sent to the OneM2M server.

CHALLENGES FACED

1. Difficulties in finding libraries compatible with Arduino. Eg. WifiNINA.h instead of WiFi.h
2. Problem in aligning the robot in a straight line due to one of the motors moving with less speed
3. Loose connections of wires in Arduino which got disconnected very frequently when testing code.
4. Detection of the slant is a major challenging for us.
5. After adding 3 ultrasonic , the soldering part is really challenging in such a smalll compact arduino.
6. The wheel often start fluctuating and this change the hardware functioning of the robot.

Limitation

- We tried to connect our vehicles to the phone with the help of bluetooth sensor and control it from there is not sufficient time that we can work on it .
- The wheels didn't rotate properly because of the weight of the chassis and powerbank .finding a light weight powerbank is too tough.
- Sending the data to cloud takes time which makes the process slow .

THANK YOU