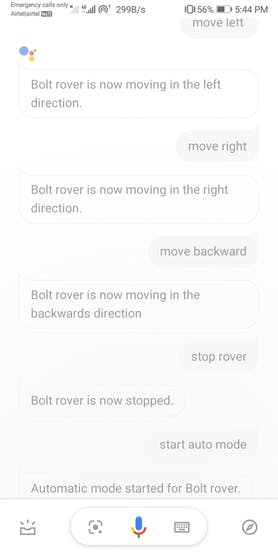
**1. Introduction**

This project is all about controlling a Robotic Car in a different way through the various devices, or by different means of controlling like voice command, mouse, keyboard, or mobile touch screen. Or by different platforms like a webpage, mobile app, and even though today's digital Assistant like 'Google Assistant'. When you say start robot it starts moving in a forwarding direction, when you say move left it starts moving in the left direction, When you say move Right it starts moving in the Right direction, When you say move backward it starts moving in a Backward direction, When you say stop it will be stoped. It is a simple example of the coming day of smart IoT, AI, and ML-enabled Vehicles. In these types of vehicles, there is no need for a driver to drive a car. These type of cars are self controllable with the help of new technology like 5G and IoT. These types of cars can make good use of Sensor data to detect Traffic other vehicles, Objects, Humans in their surroundings and is also capable of interact with other roadside vehicles to make a good decision for its preferred and the best route which are helpful in making driving hassle-free.

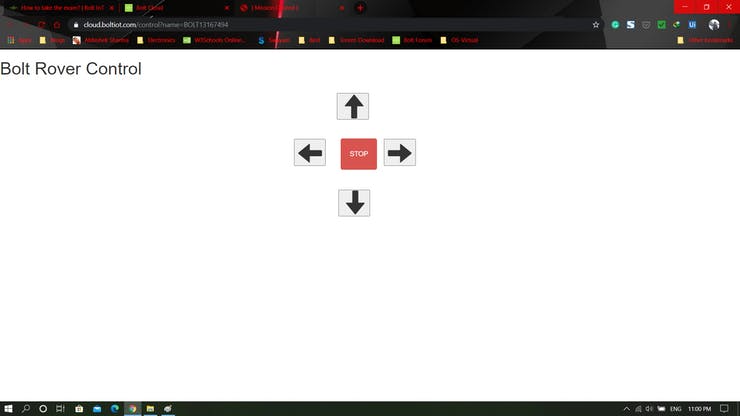
**3.Control Mode**

**3.1 Through Google Assistant**



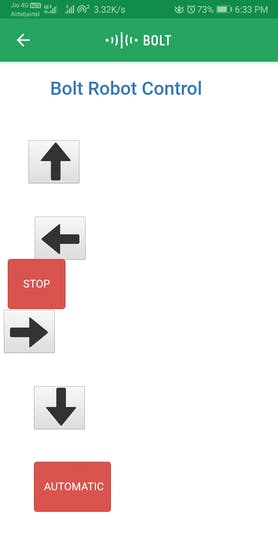
Screen Shots of Google Assistant for Controlling Robotic Car

**3.2 Through Webpage**



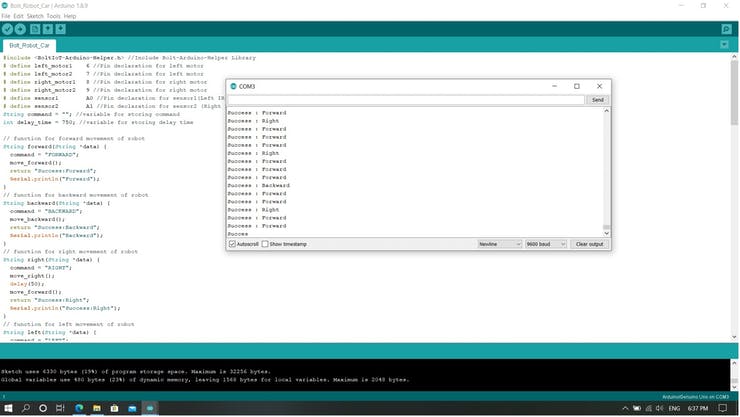
Controlling Robotic Car through Webpage

**3.3 Through Bolt IoT Android Application**



Control Interface for Controlling Robotic Car through Mobile App

**3.4 Through Arduino IDE Serial Monitor**



Robot Can be controlled through Arduino Serial Monitor by Sending Command

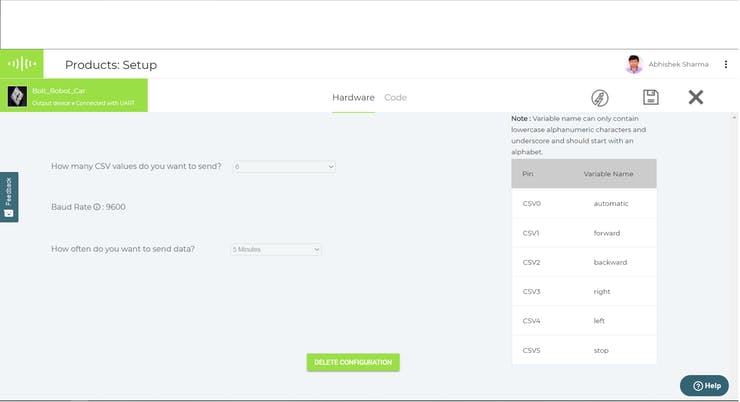
**3.5 Through Self Controlled Mode**

In this mode, it will be controlled by itself. When you say start robot it starts moving in a forwarding direction, when you say move left it starts moving in the left direction, When you say move Right it starts moving in the Right direction, When you say move backward it starts moving in a Backward direction, When you say stop it will be stopped.

**4.Steps for creating this Robotic Car Projects**

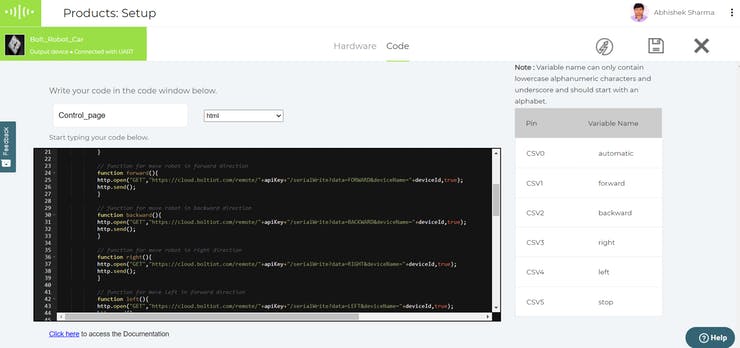
**4.1 Configure the product on Bolt Cloud**

4.1.1 Go to the Bolt Cloud site and log in to your account. Then go to the product page to make a new product give the name for this product as Bolt\_Robot\_Car, Device Type INPUT, Device Pin Communication selects UART. Then in the Hardware section select 6 CSV Values, Baud Rate 9600. Give a variable name for Pin from CSV0 to CSV5 as automatic, forward, backward, right, left, and stop respectively.

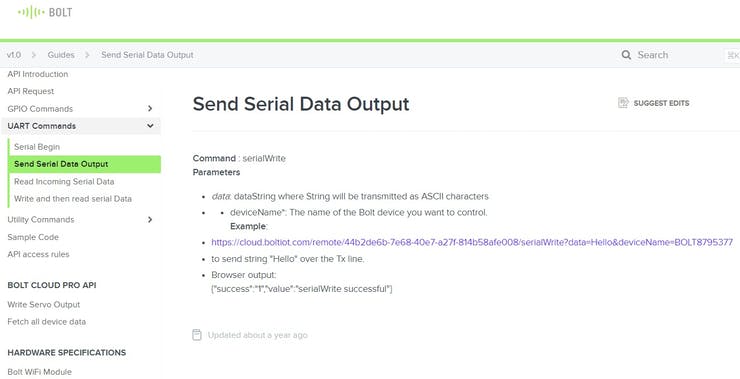


Product Hardwre Configuration

4.1.2 Now go to the Code section, Here you have to write HTML code to design an Interface webpage which contains a control Button for Robot movement control. Here we also use Javascript and Bolt Cloud UART Serial Data Transmitter API for sending the required data to Bolt Wi-Fi module TX Pin, Which is then Received by Arduino RX Pin. You need your BOLT CLOUD API KEY And DEVICE ID here.



you can find the required HTML code for this in the Code Section with an explanation. Here we Use Bolt UART API for 'Sending Serial Data Output'



https://docs.boltiot.com/docs/send-serial-data-output

we have to make 6 URLs for controlling our device for different work. URL looks as <https://cloud.boltiot.com/remote/44b2de6b-7e68-40e7-a27f-814b58afe008/serialWrite?data=Hello&deviceName=BOLT8795377> Here *data*: data string where String will be transmitted as ASCII characters like in our case it is AUTOMODE, STOP, LEFT, RIGHT, BACKWARD, and FORWARD.

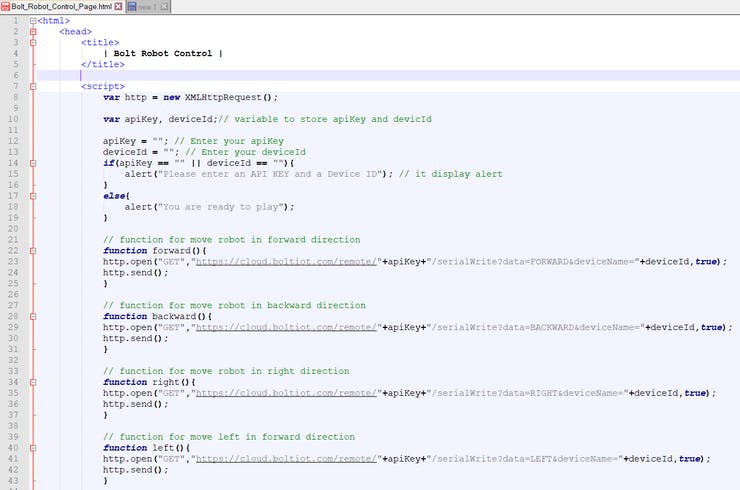
deviceName\*: The name of the Bolt device you want to control (Device ID)

when we click Buton from Webpage, Bolt app, or giving a voice command an HTTP request is sent as http.open("GET", "https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=FORWARD&deviceName="+deviceId, true);

apiKey: your ApiKey.

4.1.3 HTML Code for BOLT CLOUD

[4.1.3.1](http://4.1.3.1/) Javascript Part



Javascript Code for Controlling Robotic Car

here we make 6 functions namely forward(), backward(), right(), left(), stop(), automatic()

// function for move robot in forward direction

function forward(){

http.open("GET", "https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=FORWARD&deviceName="+deviceId, true);

http.send();

}

// function for move robot in backward direction

function backward(){

http.open("GET", "https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=BACKWARD&deviceName="+deviceId, true);

http.send();

}

// function for move robot in right direction

function right(){

http.open("GET", "https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=RIGHT&deviceName="+deviceId, true);

http.send();

}

// function for move left in forward direction

function left(){

http.open("GET", "https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=LEFT&deviceName="+deviceId, true);

http.send();

}

// function for stop robot

function stop(){

http.open("GET", "https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=STOP&deviceName="+deviceId, true);

http.send();

}

function automatic(){

http.open("GET", "https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=AUTOMATIC&deviceName="+deviceId, true);

http.send();

}

[4.1.3.2](http://4.1.3.2/) HTML Part

we make 6 button for controlling the all function of Robotic Car. These Button are shown in Interface page. forward(), backward(), right(), left(), stop(), automatic()

<button class="glyphicon glyphicon-arrow-up" onclick="forward()"></button>

<button class="glyphicon glyphicon-arrow-up" onclick="backward()"></button>

<button class="glyphicon glyphicon-arrow-up" onclick="right()"></button>

<button class="glyphicon glyphicon-arrow-up" onclick="left()"></button>

<button class="glyphicon glyphicon-arrow-up" onclick="stop()"></button>

<button class="glyphicon glyphicon-arrow-up" onclick="automatic()"></button>

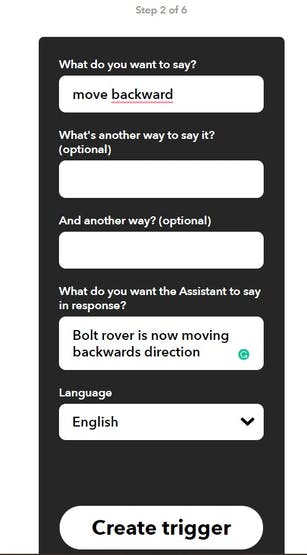
After writing this code save the configuration. Link product to the device and deploy this configuration on the Bolt Device.

**4.2 Configure IFTTT for Controlling Robot through Google Assistant**

4.2.1 Open the IFTTT site by open [https://ifttt.com](https://ifttt.com/) make your account and login. Click on Create and Select Applets.

STEP 1. Choose a Service i.e Google Assistant.

STEP 2. Choose Trigger i.e Say a simple Phrase, and type required data as



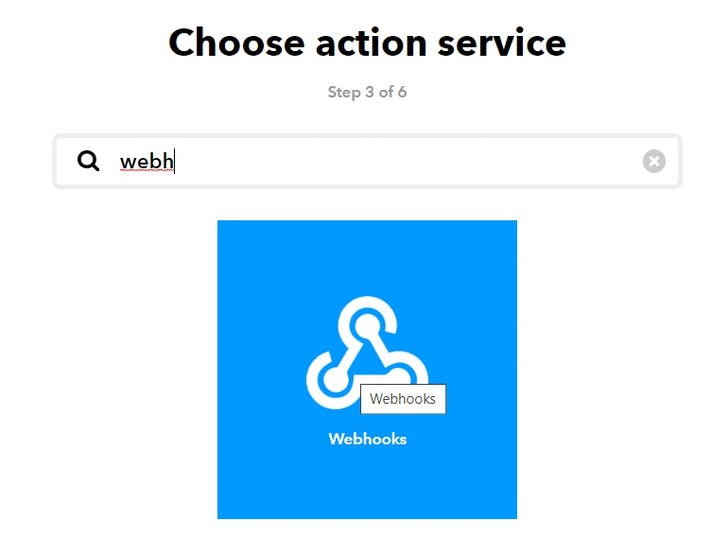
IFTTT Trigger

STEP 3. NOW you see A Statement Like This



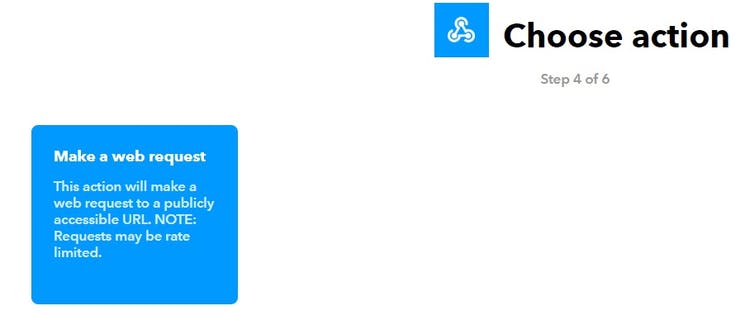
Step 3

Click on PLUS '+' Sign and proceed further and choose action service as Webhooks

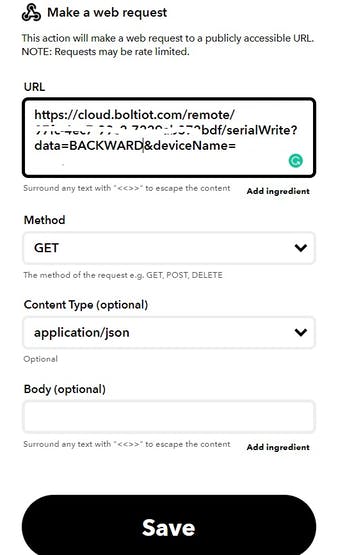


step 3

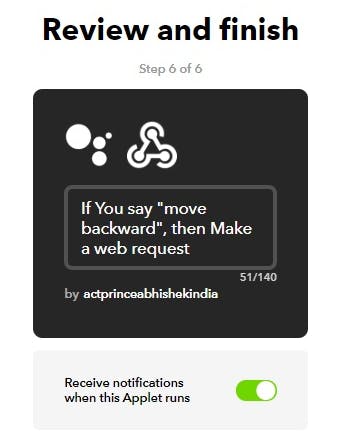
STEP 4. Make a Web request



Step 5. Fill URL which is discussed above in Section 4.1.2.



STEP 6. Review and Finish



step 6

STEP 7. Repeat STEP 1 to 6 for all the 6 functions i.e AUTOMODE, STOP, LEFT, RIGHT, BACKWARD, and FORWARD.

https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=FORWARD&deviceName="+deviceId

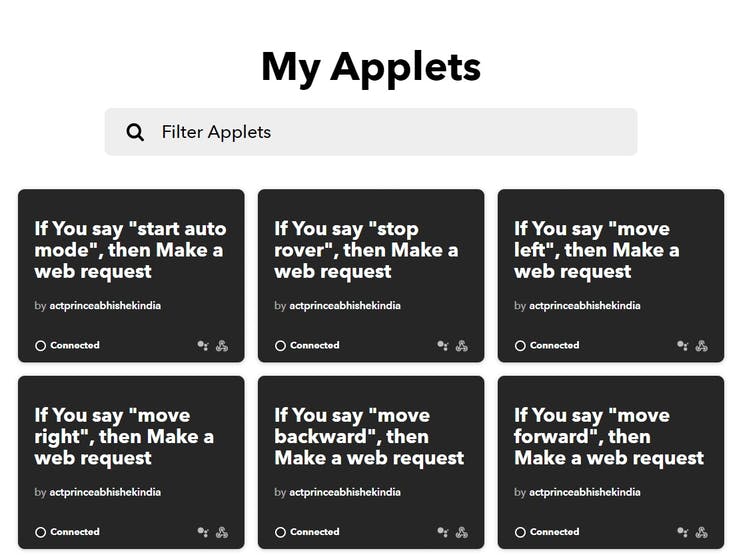
https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=BACKWARD&deviceName="+deviceId

https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=RIGHT&deviceName="+deviceId

https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=LEFT&deviceName="+deviceId

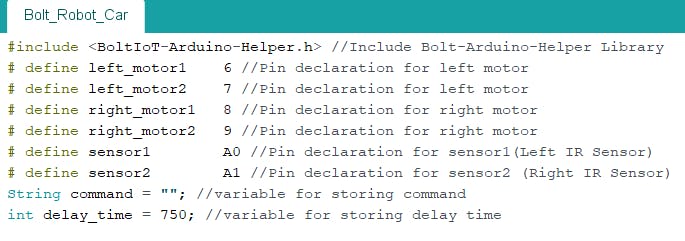
https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=STOP&deviceName="+deviceId

https://cloud.boltiot.com/remote/"+apiKey+"/serialWrite?data=AUTOMATIC&deviceName="+deviceId



**4.3 Arduino UNO Code Configuration**

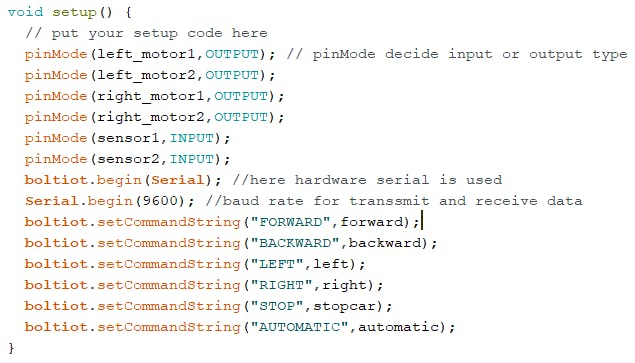
4.3.1 PIN Initialization and Library Part



Arduinno Code Intialization

Here left\_motor1, left\_motor2 are two-terminal of DC motor. Connect all PINS according to this code.

4.3.2 Setup Part

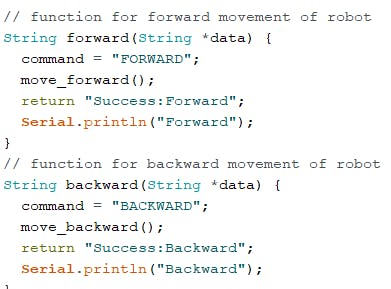


Arduino Code Setup

boltiot.setCommandString("FORWARD", forward);

This function has two arguments 1st is the received command from Bolt Cloud server and 2nd argument for calling the command handling function.

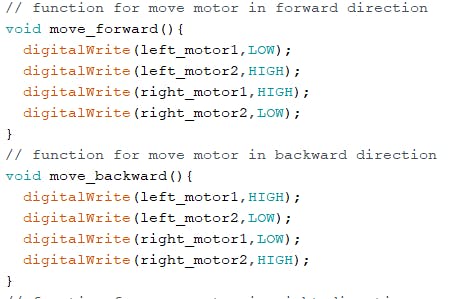
4.3.3 Command Handling Function



Arduino Code Command Handling Function

Here command function whenever received specified command it calls particular Motor Control Function like if it received the command "FORWARD" it then call move\_forward() function and return "Success:FORWARD" message and also print this message on the serial monitor.

4.3.4 Motor Control Function



Arduino Code Motor Control Function

4.3.5 Loop Control - Main Program



Arduino Code Main Program

In this part when command AUTOMATIC is received then it executes if block of codes in which continuously check the both IR sensor if it detects an object than the required function is called for motor control. But when no AUTOMATIC command is received then it executes else block of code.