RISC-V Product Development Hackathon:

Stage 2-Product Idea Development

1. **Product Title**

   Voice Guided Motor Car using VSDSquadron board, Master-slave configured HC – 05 Bluetooth module, L298N Motor Driver and a Voice recognition module DF2301QG v1.0.

   Theme: IoT/IoE-based

2. **What does your product do?**
   - *Receive*
   - *Respond*
   - *Run*

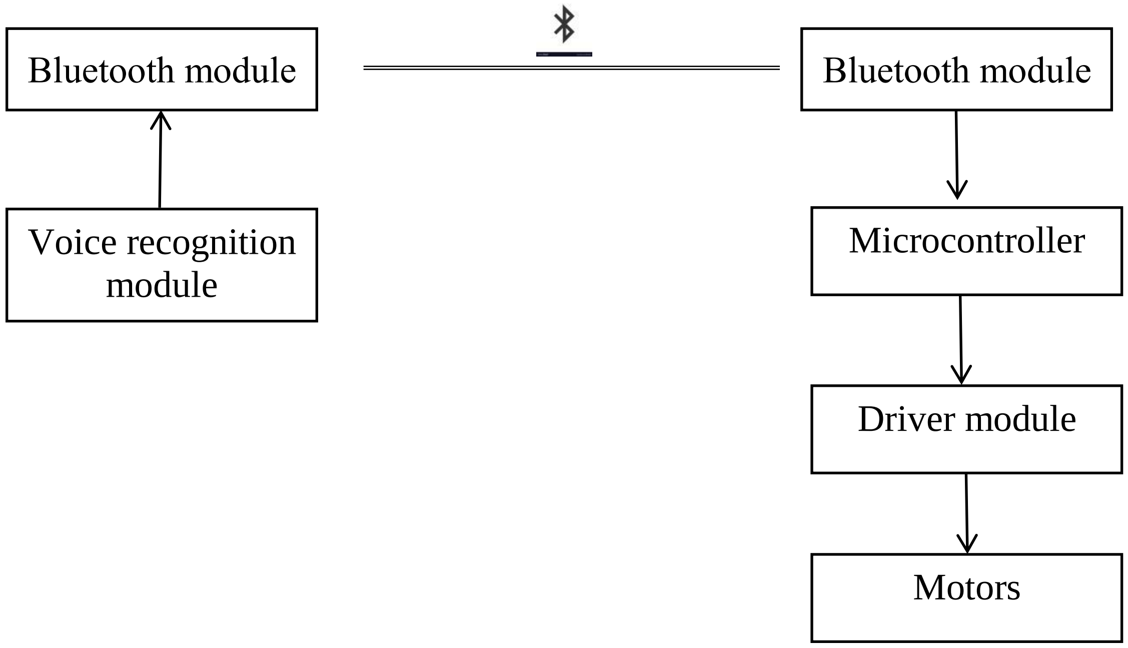
3. **What all interfaces of the board will used in the product?**

   We will be using both UART and I2C for communication between VSDS board and DF2301QG v1.0 for data transmission. GPIO pins for motor control and Bluetooth module.

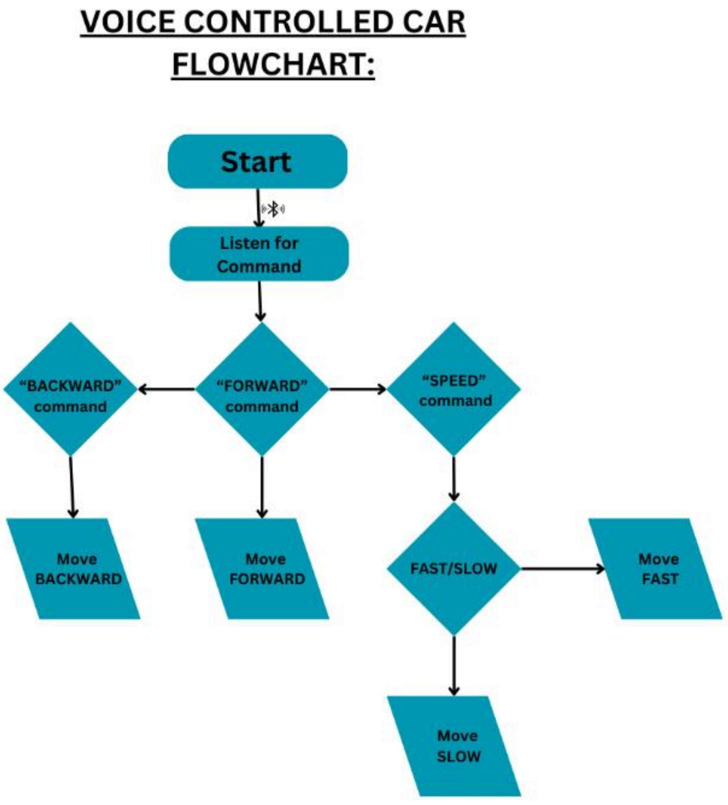
4. **Does the product utilise sensors?**
   - *Yes*

5. **If "Yes" for above question, then list your sensors here**
   - *HC - 05 Bluetooth Module*
   - *L298N Motor Driver*
   - *DF2301QG v1.0 Voice recognition module*

6. Draw a Block diagram of the product.



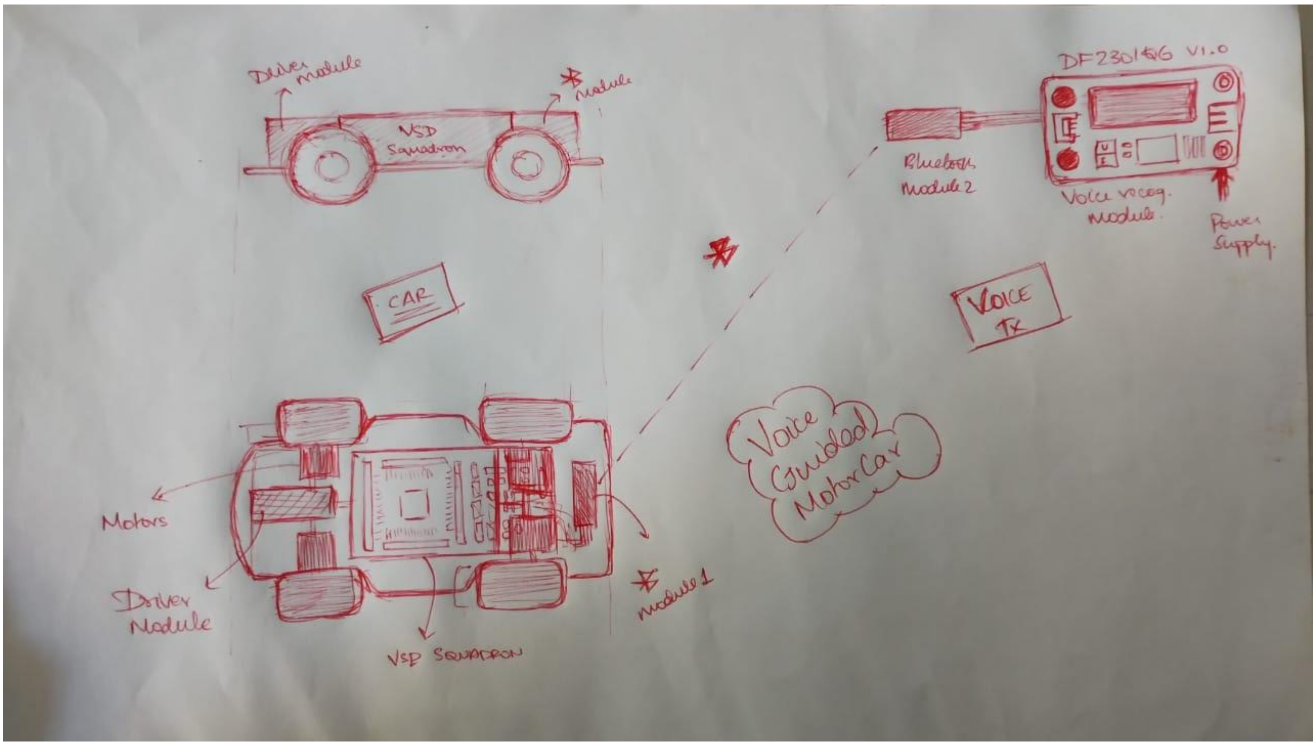
7. Upload the Algorithm flowchart of the product.



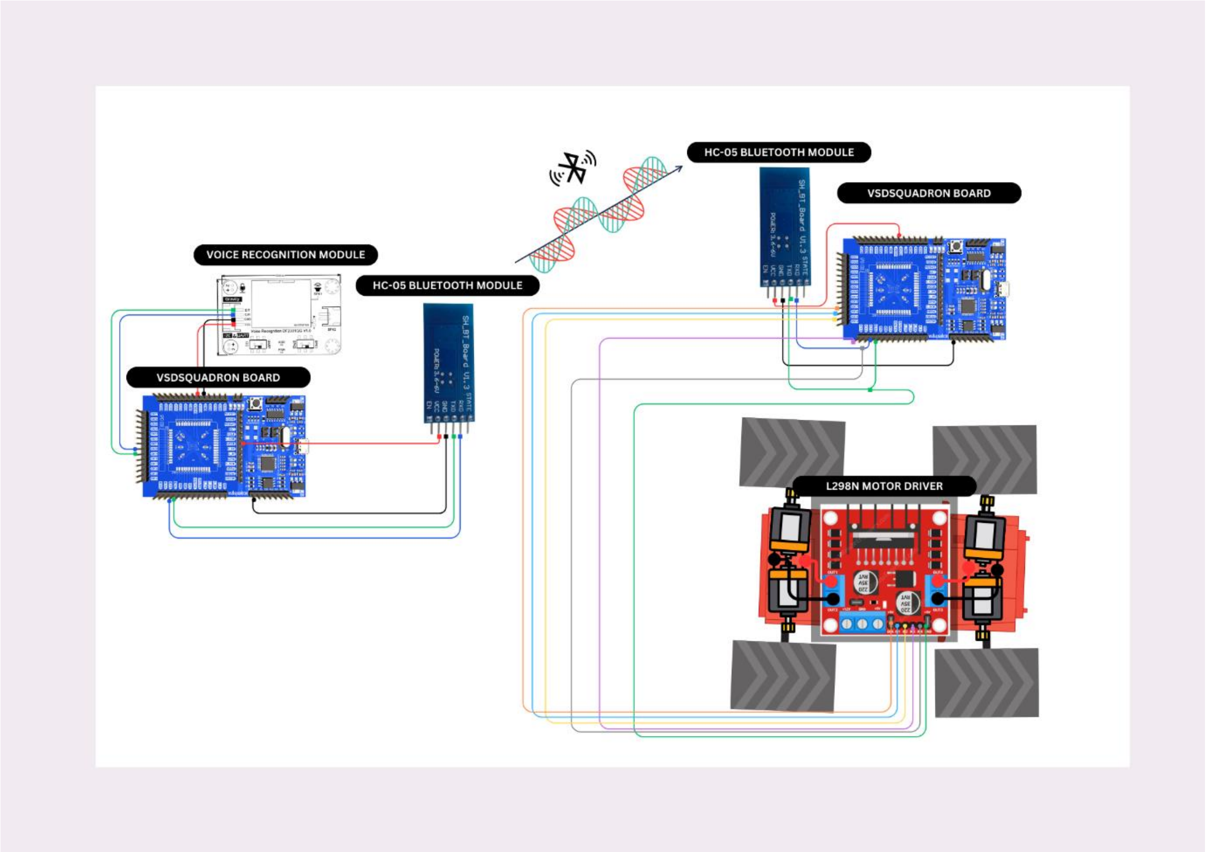
8. **Explain the algorithm of the product in bullet points.**

- *Start*
- *Listen for command*
- *If “FORWARD” command detected, move forward*
- *If “BACKWARD” command detected, move backward*
- *If “SPEED – SLOW/FAST” command detected, change the speed*

9. **Draw a Rough sketch of the final product.**



10.Upload the rough sketch of the Internal product (With all connection of components with the board and the product.).



11.BoM list (excluding the board) with cost.

Component name	Quantity Required	Unit price	Total Price (Unit price*Quantity)
HC-05 BLUETOOTH MODULE	2	440 (Appprox.)	880(Approx.)

L298N MOTOR DRIVER	1	180 (Approx.)	180 (Approx.)
Voice recognition module DF2301QG v1.0 (If available)	1	1500 (Approx.)	1500 (Approx)
12 V DC Motor	2	200	400(Approx)

12.Team details

Name	**University/Organisation**	**Age**	**Gender**	**Current Semester**	**Current Address**	**Do you need accommodation if the Demo is to done in Bangalore**	**Role in Product Development**
AKILESH S	Anna University, MIT Campus	20	Male	7	Chennai	Yes	Project Planning and Conceptualization
HARI KAILASH M M	Anna University, MIT Campus	20	Male	7	Chennai	Yes	Software Implementation

13.C Code:

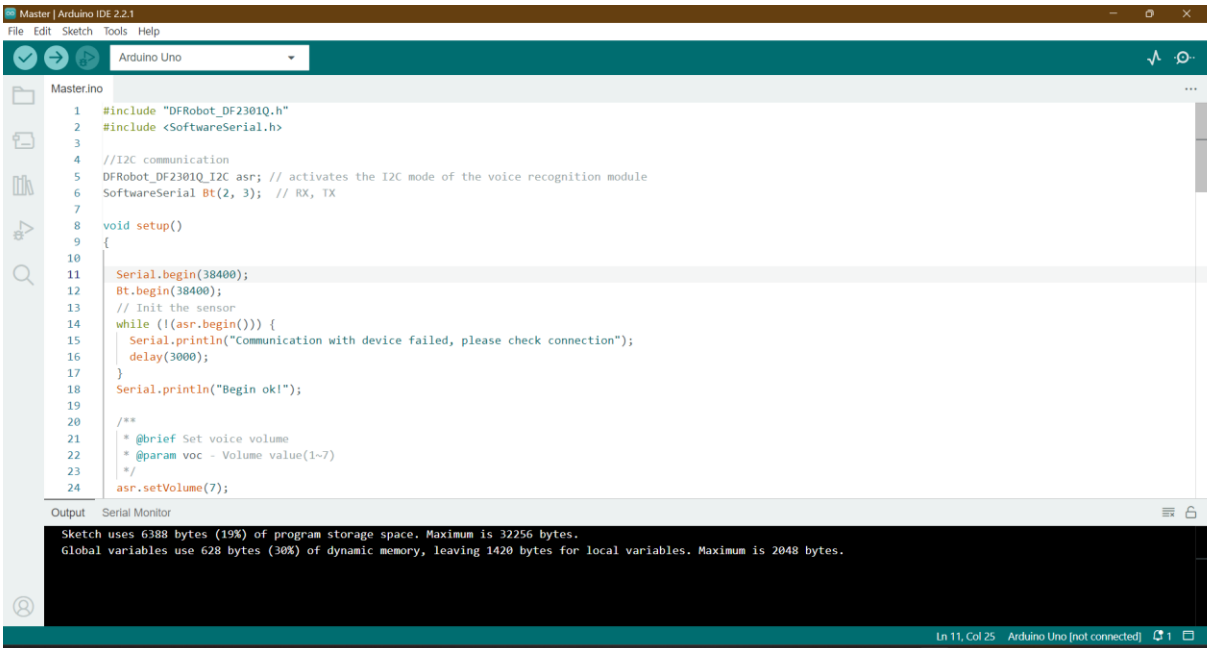
Master Side (Transmitter Side):

```
#include "DFRobot_DF2301Q.h"
#include <SoftwareSerial.h>

//I2C communication
DFRobot_DF2301Q_I2C asr; // activates the I2C mode of the voice recognition
module
SoftwareSerial Bt(2, 3);  // RX, TX

void setup()
{

  Serial.begin(38400);
  Bt.begin(38400);
  // Init the sensor
  while (!(asr.begin())) {
    Serial.println("Communication with device failed, please check
connection");
    delay(3000);
  }
  Serial.println("Begin ok!");

  /**
   * @brief Set voice volume
   * @param voc - Volume value(1~7)
   */
  asr.setVolume(7);

  /**
     @brief Set mute mode
     @param mode - Mute mode; set value 1: mute, 0: unmute
  */
  asr.setMuteMode(0);

  /**
     @brief Set wake-up duration
     @param wakeTime - Wake-up duration (0-255)
  */
  asr.setWakeTime(20);

  /**
     @brief Get wake-up duration
     @return The currently-set wake-up period
  */
  uint8_t wakeTime = 0;
  wakeTime = asr.getWakeTime();
  Serial.println("wakeTime = ");
  Serial.println(wakeTime);

  asr.playByCMDID(1);   // Wake-up command

  /**
     @brief Play the corresponding reply audio according to the ID
     @param CMDID - command word ID
  */
}

void loop()
{
  /**
     @brief Get the ID corresponding to the command word
     @return Return the obtained command word ID, returning 0 means no valid
ID is obtained
  */
  uint8_t CMDID = asr.getCMDID();
  Serial.println(CMDID);
  switch (CMDID)
  {
    // Custom commands in English

    case 5: // "FORWARD"
    Bt.write('F');
    break;

    case 6: // "BACKWARD"
    Bt.write('B');
    break;

    case 7: // "PARK"
    Bt.write('P');
    break;

    case 8: // "STEER LEFT"
    Bt.write('L');
    break;

    case 9: // "STEER RIGHT"
    Bt.write('R');
    break;

    case 10:  // "INCREASE THE SPEED"
    Bt.write('R');
```

```
      break;

      case 11:  // "DECREASE THE SPEED"
      Bt.write('S');
      break;

      case 12: // "MAINTAIN THE SPEED"
      Bt.write('M');
      break;


      default:
      Serial.println("Invalid Command");
      break;
    }
  delay(300);
}
```

Output:

Arduino Uno Compilation:

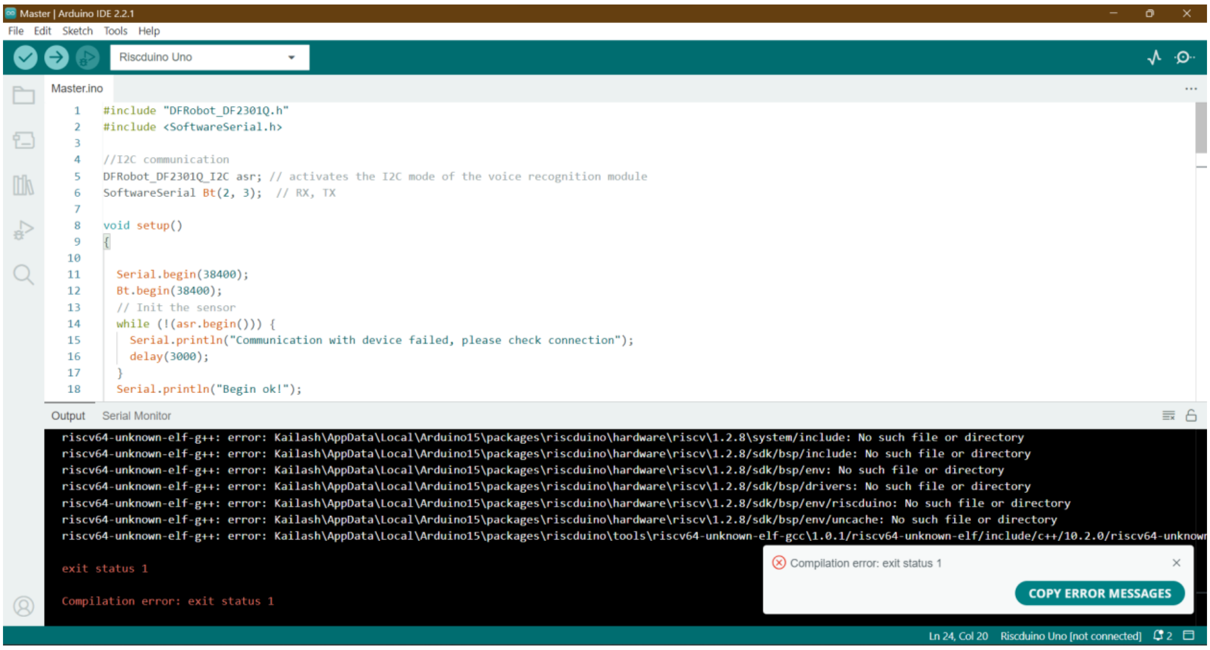


Master | Arduino IDE 2.2.1
File Edit Sketch Tools Help
Arduino Uno
Master.ino
Output Serial Monitor
Ln 11, Col 25 Arduino Uno [not connected] 1

```
#include "DFRobot_DF2301Q.h"
#include <SoftwareSerial.h>

//I2C communication
DFRobot_DF2301Q_I2C asr; // activates the I2C mode of the voice recognition module
SoftwareSerial Bt(2, 3);  // RX, TX

void setup()
{

  Serial.begin(38400);
  Bt.begin(38400);
  // Init the sensor
  while (!(asr.begin())) {
    Serial.println("Communication with device failed, please check connection");
    delay(3000);
  }
  Serial.println("Begin ok!");

  /**
   * @brief Set voice volume
   * @param voc - Volume value(1~7)
   */
  asr.setVolume(7);
```

```
Sketch uses 6388 bytes (19%) of program storage space. Maximum is 32256 bytes.
Global variables use 628 bytes (30%) of dynamic memory, leaving 1420 bytes for local variables. Maximum is 2048 bytes.
```

Riscduino Uno Compilation:


Master | Arduino IDE 2.2.1
File Edit Sketch Tools Help
Riscduino Uno
Master.ino
Output
Serial Monitor
Ln 24, Col 20
Riscduino Uno [not connected]
2

```
#include "DFRobot_DF2301Q.h"
#include <SoftwareSerial.h>

//I2C communication
DFRobot_DF2301Q_I2C asr; // activates the I2C mode of the voice recognition module
SoftwareSerial Bt(2, 3);  // RX, TX

void setup()
{

  Serial.begin(38400);
  Bt.begin(38400);
  // Init the sensor
  while (!(asr.begin())) {
    Serial.println("Communication with device failed, please check connection");
    delay(3000);
  }
  Serial.println("Begin ok!");
```

```
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\hardware\riscv\1.2.8\system/include: No such file or directory
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\hardware\riscv\1.2.8/sdk/bsp/include: No such file or directory
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\hardware\riscv\1.2.8/sdk/bsp/env: No such file or directory
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\hardware\riscv\1.2.8/sdk/bsp/drivers: No such file or directory
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\hardware\riscv\1.2.8/sdk/bsp/env/riscduino: No such file or directory
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\hardware\riscv\1.2.8/sdk/bsp/env/uncache: No such file or directory
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\tools\riscv64-unknown-elf-gcc\1.0.1/riscv64-unknown-elf/include/c++/10.2.0/riscv64-unknown

exit status 1

Compilation error: exit status 1
```

Compilation error: exit status 1

COPY ERROR MESSAGES

Slave Side (Receiver Side):

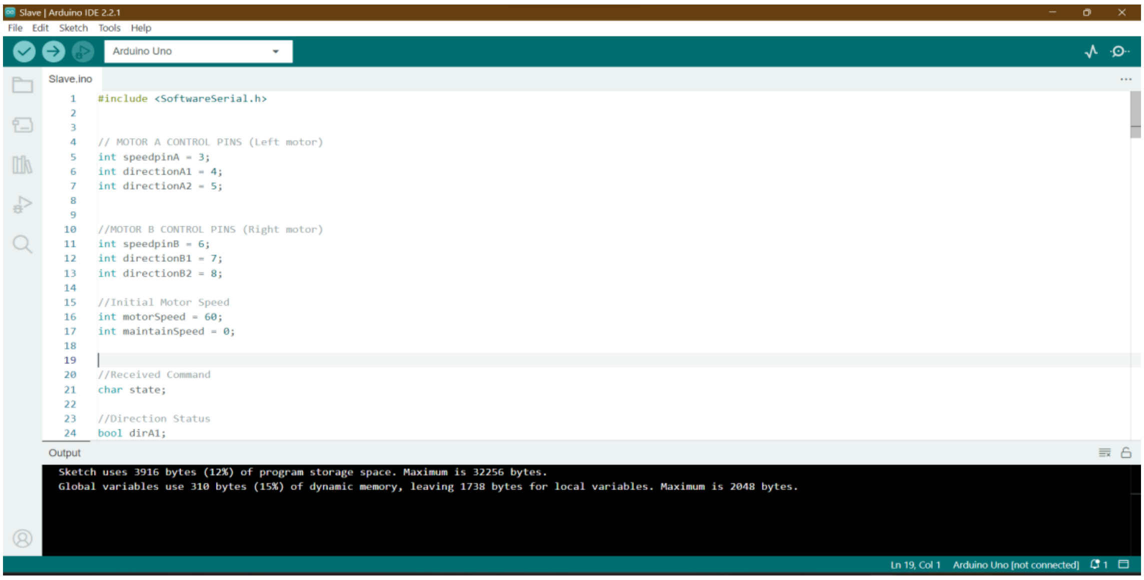
```
#include <SoftwareSerial.h>


// MOTOR A CONTROL PINS (Left motor)
int speedpinA = 3;
int directionA1 = 4;
int directionA2 = 5;


//MOTOR B CONTROL PINS (Right motor)
int speedpinB = 6;
int directionB1 = 7;
int directionB2 = 8;

//Initial Motor Speed
int motorSpeed = 60;
int maintainSpeed = 0;


//Received Command
char state;

//Direction Status
```

```
bool dirA1;
bool dirA2;
bool dirB1;
bool dirB2;

SoftwareSerial Bt(2, 3);  // RX, TX

void setup()
{
  // Bluetooth Initialisation
  Serial.begin(38400);  // Serial monitor for debugging
  Bt.begin(38400);  // Bluetooth module communication speed

  // Setting pin modes of the L298N Motor Driver
  pinMode(speedpinA, OUTPUT);
  pinMode(directionA1, OUTPUT);
  pinMode(directionA2, OUTPUT);
  pinMode(speedpinB, OUTPUT);
  pinMode(directionB1, OUTPUT);
  pinMode(directionB1, OUTPUT);

  //Standby mode of the car
  digitalWrite(directionA1,LOW);
  digitalWrite(directionA2,LOW);
  digitalWrite(directionB1,LOW);
  digitalWrite(directionB1,LOW);
}

void loop()
{
  while(Bt.available()) // Read data from the master device
  {
    state=Bt.read();
    switch(state)
    {

      case 'F':  // "FORWARD"
      dirA1=false;
      dirA2=true;
      dirB1=false;
      dirB2=true;
      controlDirection();
      break;


      case 'B': // "BACKWARD"
      dirA1=true;
      dirA2=false;
      dirB1=true;
      dirB2=false;
      controlDirection();
      break;


      case 'P':  //"PARK"
      dirA1=false;
      dirA2=false;
      dirB1=false;
      dirB2=false;
      controlDirection();
      break;

      case 'L':  //"STEER LEFT"
      dirA1=false;
      dirA2=false;
      dirB1=false;
      dirB2=true;
      controlDirection();
      break;

      case 'R':  //"STEER RIGHT"
      dirA1=false;
      dirA2=true;
      dirB1=false;
      dirB2=false;
      controlDirection();
      break;

      case 'I':  //"INCREASE THE SPEED"
      controlDirection();
      break;

      case 'D':  //"DECREASE THE SPEED"
      controlDirection();
      break;

      case 'M':  //"MAINTAIN THE SPEED"
      maintainSpeed =1;
      speedControl();
      break;



    }
  }

}
```

```
// Direction Control of the car
void controlDirection()
{
  digitalWrite(directionA1,dirA1);
  digitalWrite(directionA2,dirA2);
  digitalWrite(directionB1,dirB1);
  digitalWrite(directionB1,dirB2);
  speedControl();
}

// Speed control of the car
void speedControl()
{
  if(state=='I' ||  maintainSpeed == 1)
  {
    for(int i=motorSpeed; i<256; i++)
    {

      if( maintainSpeed == 1)
      {
        motorSpeed=i;
        maintainSpeed=0;
        break;
      }

      analogWrite(speedpinA,i);
      analogWrite(speedpinB,i);
      delay(20);
    }
  }


  if(state=='D' ||  maintainSpeed == 1)
  {
    for(int i=motorSpeed; i>0; i--)
    {

      if( maintainSpeed == 1)
      {
        motorSpeed=i;
        maintainSpeed=0;
        break;
      }


      analogWrite(speedpinA,i);
      analogWrite(speedpinB,i);
      delay(20);
    }
  }

  analogWrite(speedpinA,motorSpeed);
  analogWrite(speedpinB,motorSpeed);

}
```

Output:

Arduino Uno Compilation:

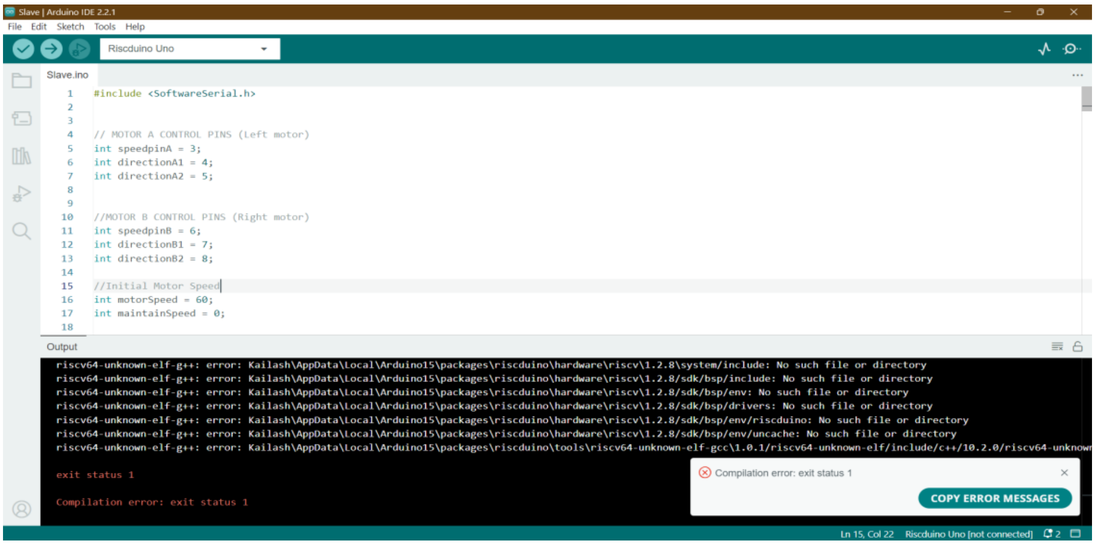


Slave | Arduino IDE 2.2.1
File Edit Sketch Tools Help
Arduino Uno
Slave.ino
Output
Ln 19, Col 1 Arduino Uno [not connected] 1

```
#include <SoftwareSerial.h>


// MOTOR A CONTROL PINS (Left motor)
int speedpinA = 3;
int directionA1 = 4;
int directionA2 = 5;


//MOTOR B CONTROL PINS (Right motor)
int speedpinB = 6;
int directionB1 = 7;
int directionB2 = 8;

//Initial Motor Speed
int motorSpeed = 60;
int maintainSpeed = 0;


//Received Command
char state;

//Direction Status
bool dirA1;
```

```
Sketch uses 3916 bytes (12%) of program storage space. Maximum is 32256 bytes.
Global variables use 310 bytes (15%) of dynamic memory, leaving 1738 bytes for local variables. Maximum is 2048 bytes.
```

Riscduino Uno Compilation:



Slave | Arduino IDE 2.2.1
File Edit Sketch Tools Help
Riscduino Uno
Slave.ino
Output
Compilation error: exit status 1
COPY ERROR MESSAGES
Ln 15, Col 22 Riscduino Uno [not connected] 2

```
#include <SoftwareSerial.h>


// MOTOR A CONTROL PINS (Left motor)
int speedpinA = 3;
int directionA1 = 4;
int directionA2 = 5;


//MOTOR B CONTROL PINS (Right motor)
int speedpinB = 6;
int directionB1 = 7;
int directionB2 = 8;

//Initial Motor Speed
int motorSpeed = 60;
int maintainSpeed = 0;

```

```
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\hardware\riscv\1.2.8\system/include: No such file or directory
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\hardware\riscv\1.2.8/sdk/bsp/include: No such file or directory
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\hardware\riscv\1.2.8/sdk/bsp/env: No such file or directory
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\hardware\riscv\1.2.8/sdk/bsp/drivers: No such file or directory
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\hardware\riscv\1.2.8/sdk/bsp/env/riscduino: No such file or directory
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\hardware\riscv\1.2.8/sdk/bsp/env/uncache: No such file or directory
riscv64-unknown-elf-g++: error: Kailash\AppData\Local\Arduino15\packages\riscduino\tools\riscv64-unknown-elf-gcc\1.0.1/riscv64-unknown-elf/include/c++/10.2.0/riscv64-unknown

exit status 1

Compilation error: exit status 1
```

Summary:

We've developed a Voice Guided Motor Car using the VSDSquadron board, with components including a Master-slave configured HC-05 Bluetooth module, an L298N Motor Driver, and a Voice recognition module DF2301QG v1.0. Our project involves using both UART and I2C for communication between the VSDS board and the DF2301QG v1.0 for data transmission, while GPIO pins control the motors and the Bluetooth module. We've written C code for both the transmitter and receiver sides of this project using the Arduino IDE, and the code successfully compiles for the Arduino Uno board. However, we're encountering difficulties when trying to compile the code for the riscdruino board. The problem lies in including the necessary library for the board, resulting in a "no files or directory found" error, despite following the tutorial instructions diligently.