Bluetooth Car Controlled Using Arduino

- Description for an Arduino-based RC Car :

The Arduino-based RC Car is a small, remote-controlled vehicle that is powered and controlled by an Arduino microcontroller. It is a fun and educational project that combines electronics, programming, and mechanics to create a customizable and interactive toy.

The basic components of the Arduino-based RC Car include:

1. Arduino Board: The heart of the project is an Arduino board (such as Arduino Uno or Arduino Nano), which serves as the brain of the RC car. It receives instructions from the remote control and sends signals to control the motors and other peripherals.

2. Motor Driver: To control the movement of the RC car, a motor driver is used. The motor driver acts as an interface between the Arduino and the DC motors, enabling the Arduino to control the speed and direction of the wheels.

3. DC Motors: The RC car typically consists of two or more DC motors that drive the wheels. These motors are connected to the motor driver, which receives signals from the Arduino to control their rotation and speed.

4. Remote Control: The remote control allows the user to wirelessly control the RC car's movement. It can be implemented using various methods, such as an RF module, Bluetooth, or even a smartphone app. The user's inputs are transmitted to the Arduino, which interprets them and adjusts the motor speed and direction accordingly.

5. Power Source: The RC car requires a power source, usually a rechargeable battery pack, to provide electrical energy to the Arduino, motor driver, and DC motors. The battery pack should have sufficient capacity to power the car for a reasonable amount of time.

6. Chassis and Wheels: The physical structure of the RC car includes a chassis to hold all the components and provide stability, as well as wheels to enable movement. The chassis can be constructed using materials like plastic, metal, or wood, depending on the desired strength and durability.

7. Optional Components: Additional components can be added to enhance the functionality and interactivity of the RC car. These may include sensors (such as ultrasonic sensors for obstacle avoidance), LED lights for visual effects, or even a camera for capturing images or video.

The Arduino-based RC Car operates by receiving commands from the remote control, which are wirelessly transmitted to the Arduino. The Arduino then processes these commands and sends appropriate signals to the motor driver, which controls the rotation and speed of the DC motors. The motors drive the wheels, allowing the RC car to move forward, backward, turn, and stop based on the user's input.

By programming the Arduino, users can customize the behavior of the RC car, implement different control algorithms, and add additional features according to their creativity and skill level. It serves as an excellent platform for learning about electronics, programming, and robotics, while providing an enjoyable and engaging experience for enthusiasts of all ages.

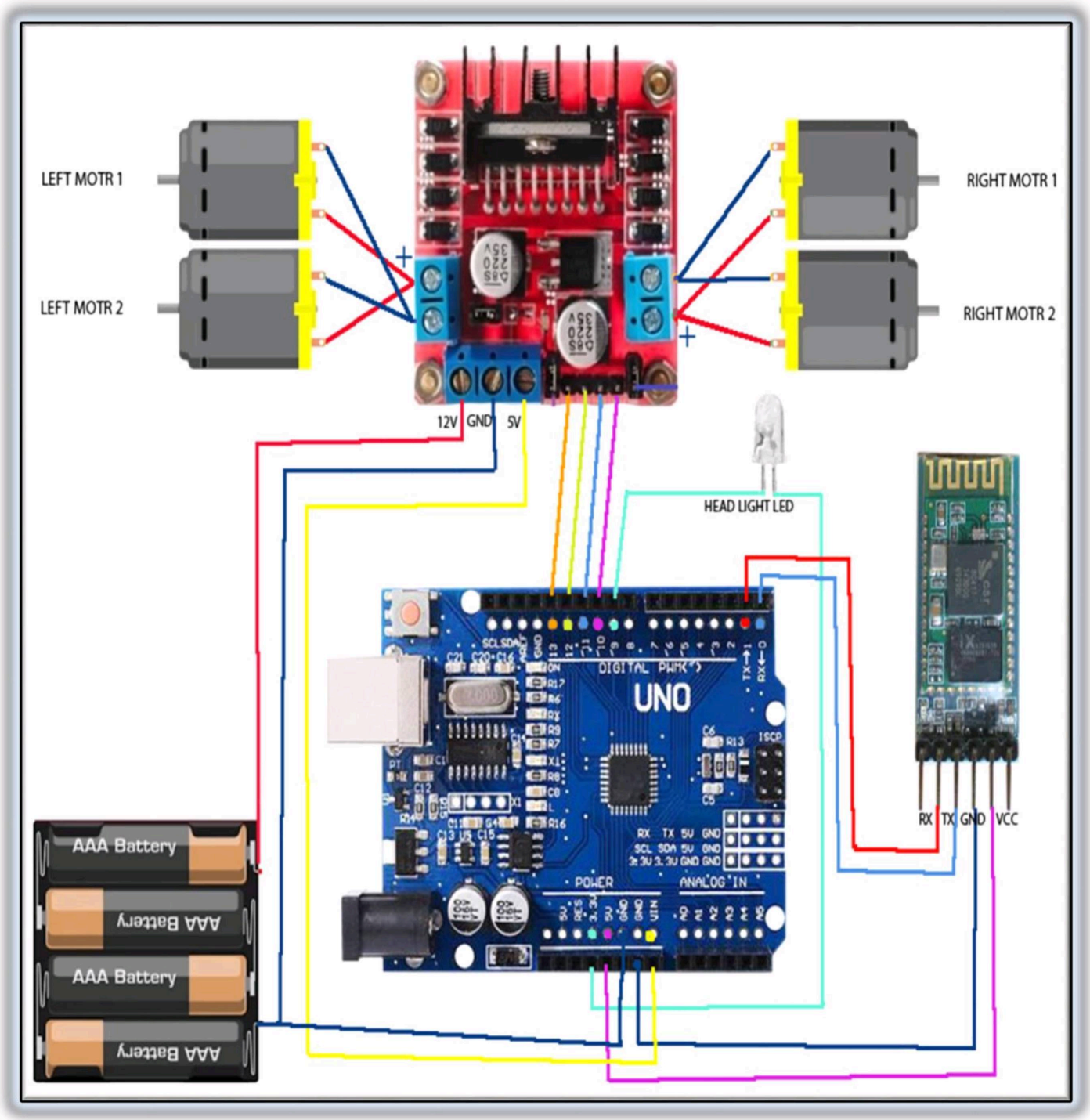
Bluetooth Car Controlled Using Arduino

Ardiuno Code :

```
char t;

void setup() {
pinMode(13,OUTPUT);   //left motors  forward
pinMode(12,OUTPUT);   //left motors reverse
pinMode(11,OUTPUT);   //right  motors forward
pinMode(10,OUTPUT);   //right motors reverse
pinMode(9,OUTPUT);   //Led
Serial.begin(9600);

}

void loop() {
if(Serial.available()){
  t = Serial.read();
  Serial.println(t);
}

if(t == 'F'){            //move  forward(all motors rotate in forward direction)
  digitalWrite(13,HIGH);
  digitalWrite(11,HIGH);
}

else if(t == 'B'){      //move reverse (all  motors rotate in reverse direction)
  digitalWrite(12,HIGH);
  digitalWrite(10,HIGH);
}

else if(t == 'L'){      //turn right (left side motors rotate in forward direction,  right side motors doesn't
rotate)
  digitalWrite(11,HIGH);
}

else  if(t == 'R'){      //turn left (right side motors rotate in forward direction, left  side motors doesn't rotate)
  digitalWrite(13,HIGH);
}

else if(t ==  'W'){     //turn led on or off)
  digitalWrite(9,HIGH);
}
else if(t == 'w'){
  digitalWrite(9,LOW);
}

else if(t == 'S'){      //STOP (all motors stop)
  digitalWrite(13,LOW);
  digitalWrite(12,LOW);
  digitalWrite(11,LOW);
  digitalWrite(10,LOW);
}
delay(100);
}
```

Bluetooth Car Controlled Using Arduino

Circuit Diagram :



Bluetooth Car Controlled Using Arduino

CAR Model :