

Circuits Project

IR Remote-controlled DC
motor

Presented by Abir, Lyndsay, Lily, Neelie

Purpose

- Design, simulate, and prototype a system to control a DC motor using an IR remote
- Gain hands-on experience with industry-relevant electrical components
- Learn microcontroller programming and signal interpretation.
- Understand the role of signal modulation in communication systems.
- Explore the significance of microcontrollers in robotics and automation

Components

- Arduino Uno
- IR remote and HX1838 IR receiver
- Breadboard
- Jumper wires
- H-bridge motor driver
- DC Motor

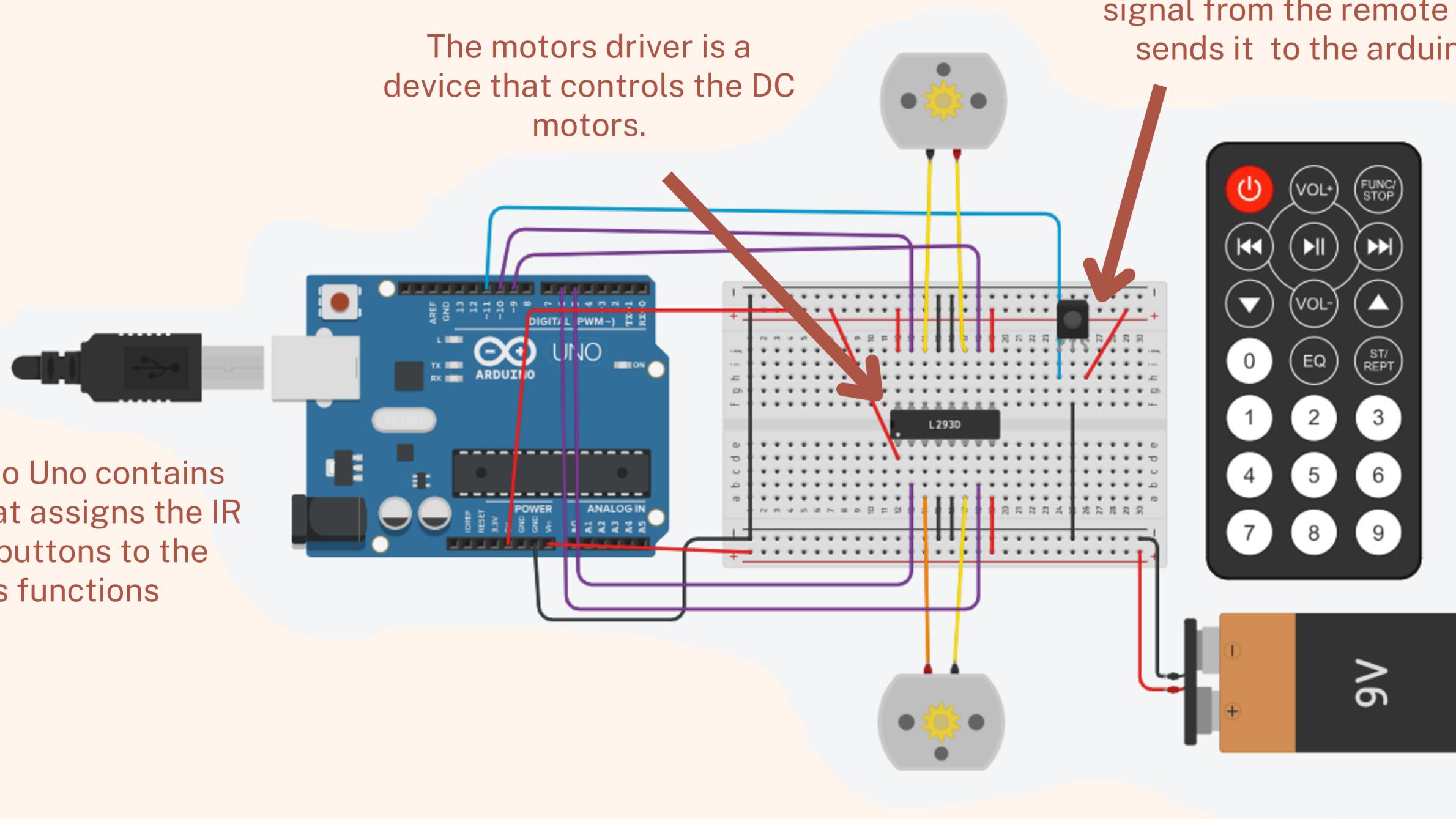
How it works

The Arduino Uno contains the code that assigns the IR remote's buttons to the motor's functions

The motors driver is a device that controls the DC motors.

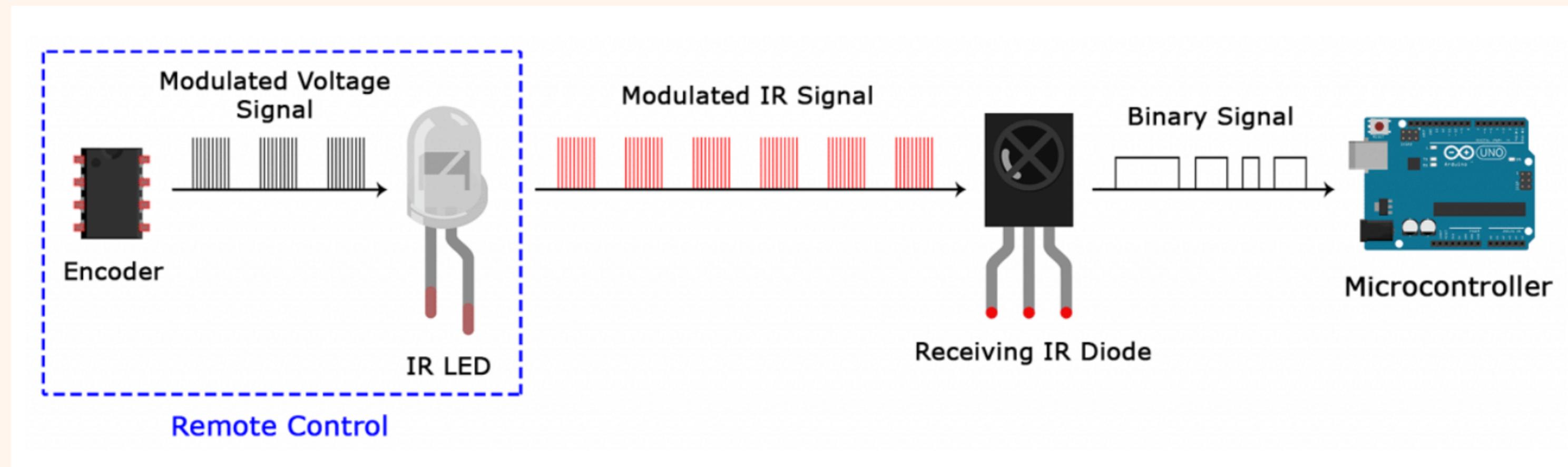
The IR receiver receives the signal from the remote and sends it to the arduino

The buttons on the IR remote are assigned to make the motor go forward, backwards, turn right, turn left and stop

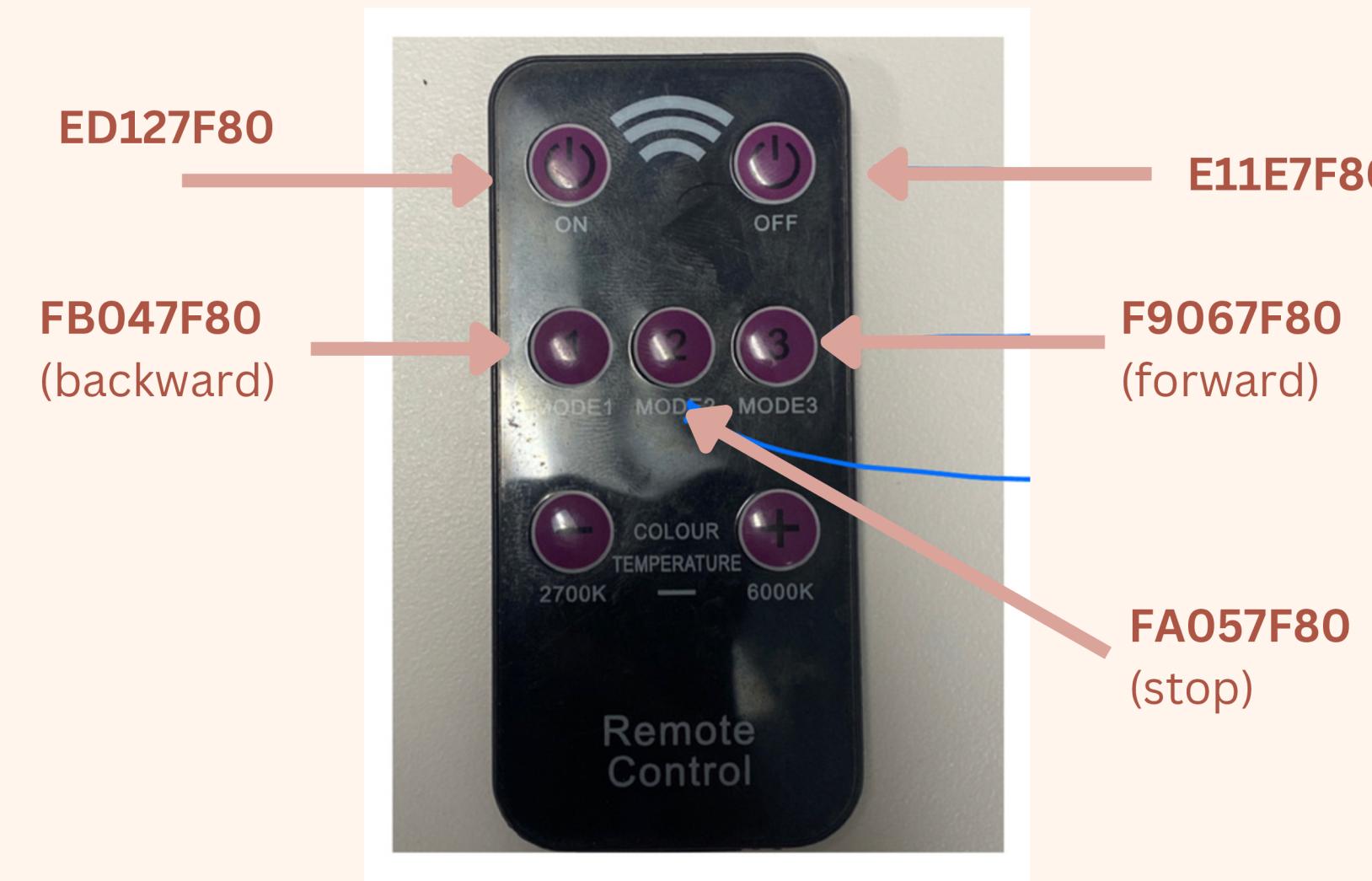


IR Signal transmission & reception

- Encoder (on the IR remote) converts a binary signal into a modulated voltage signal
- The modulated voltage signal is sent to the remote's LED, which converts the signal into a modulated IR signal
- The IR light signal is received by the receiving IR diode, which demodulates the IR signal back to binary
- The binary signal is read by the microcontroller



Decoding the buttons



IrButtonCodeFinder.ino

```
1 // C++ code
2
3 #include <IRremote.hpp>
4
5 #define IR_RECEIVE_PIN 11 //assigning pin
6
7 void setup()
8 {
9     Serial.begin(9600);
10    IrReceiver.begin(IR_RECEIVE_PIN, ENABLE_LED_FEEDBACK);
11    IrReceiver.blink13(true);
12 }
13
14 void loop()
15 {
16     if(IrReceiver.decode()){
17         Serial.println(IrReceiver.decodedIRData.decodedRawData, HEX);
18         IrReceiver.resume();
19     }
20 }
21
```

Output Serial Monitor X

Message (Enter to send message to 'Arduino Uno' on 'COM5')

```
Received value: ED127F80
Received value: 0
Received value: E11E7F80
Received value: FB047F80
Received value: 0
Received value: F9067F80
Received value: FA057F80
Received value: 0
Enabling IRin
Enabled IRin
```

Arduino Uno

RemoteControllingMotor.ino

```
1 #include <IRremote.hpp>
2 #define IR_RECEIVER_PIN 11;
3
4 void setup()
{
5     Serial.begin(9600);
6     // In case the interrupt driver crashes on setup, give a clue
7     // to the user what's going on.
8     Serial.println("Enabling IRin");
9
10    //irrecv.enableIRIn(); // Start the receiver
11    IrReceiver.begin(IR_RECEIVER_PIN, ENABLE_LED_FEEDBACK);
12    Serial.println("Enabled IRin");
13    pinMode(6, OUTPUT);
14    pinMode(5, OUTPUT);
15    pinMode(10, OUTPUT);
16    pinMode(9, OUTPUT);
17}
18
19 void loop()
20{
21
22    if (IrReceiver.decode()) {
23        Serial.print("Received value: ");
24        Serial.println(IrReceiver.decodedIRData.decodedRawData, HEX);
25        IrReceiver.resume(); // Receive the next value
26
27        if(IrReceiver.decodedIRData.decodedRawData == 0xF9067F80){
28            //forward// next
29            move(1,0,1,0);
30        }
31        else if (IrReceiver.decodedIRData.decodedRawData == 0xFFA857){
32            //right >>|
33            move(0,1,1,0);
34        }
35        else if (IrReceiver.decodedIRData.decodedRawData == 0xFB047F80){
```

```
36            //back CH-
37            move(0,1,0,1);
38        }
39        else if (IrReceiver.decodedIRData.decodedRawData == 0xFFE01F){
40            //left <<|
41            move(1,0,0,1);
42        }
43        else if (IrReceiver.decodedIRData.decodedRawData == 0xFA057F80){
44            //stop
45            move(0,0,0,0);
46        }
47    }
48    delay(100);
49}
50
51}
52
53 void move(int m1a, int m1b, int m2a, int m2b){
54     digitalWrite(6,m1a);
55     digitalWrite(5,m1b);
56     digitalWrite(10,m2a);
57     digitalWrite(9,m2b);
58}
```

Output

**Code for controlling
the motor**