#define in1 5 //L298n Motor Driver pins.

#define in2 6

#define in3 10

#define in4 11

#define LED 13

int command; //Int to store app command state.

int Speed;

int Speedsec;

int Turnradius = 200; // This controls the radius of a turn, the higher the smaller the turn. This should not exceed

// 255

void setup() {

 pinMode(in1, OUTPUT);

 pinMode(in2, OUTPUT);

 pinMode(in3, OUTPUT);

 pinMode(in4, OUTPUT);

 pinMode(LED, OUTPUT); //Set the LED pin.

 Serial.begin(9600); //Set the baud rate to your Bluetooth module.

}

void loop() {

 if (Serial.available() > 0) {

 command = Serial.read();

 Stop(); //Initialize with motors stoped.

 switch (command) {

 case 'F':

 forward();

 break;

 case 'B':

 back();

 break;

 case 'L':

 left();

 break;

 case 'R':

 right();

 break;

 case 'G':

 forwardleft();

 break;

 case 'I':

 forwardright();

 break;

 case 'H':

 backleft();

 break;

 case 'J':

 backright();

 break;

 case '0':

 Speed = 128;

 break;

 case '1':

 Speed = 140;

 break;

 case '2':

 Speed = 153;

 break;

 case '3':

 Speed = 165;

 break;

 case '4':

 Speed = 178;

 break;

 case '5':

 Speed = 191;

 break;

 case '6':

 Speed = 204;

 break;

 case '7':

 Speed = 216;

 break;

 case '8':

 Speed = 229;

 break;

 case '9':

 Speed = 242;

 break;

 case 'q':

 Speed = 255;

 break;

 case 'W': //LED pin on or off.

 digitalWrite(LED, HIGH);

 break;

 case 'w':

 digitalWrite(LED, LOW);

 break;

 case 'X':

 Auto();

 break;

 case 'x':

 Manual();

 break;

 }

 Speedsec = Speed - Turnradius;

 }

}

void forward() {

 analogWrite(in1, Speed);

 analogWrite(in3, Speed);

}

void back() {

 analogWrite(in2, Speed);

 analogWrite(in4, Speed);

}

void left() {

 analogWrite(in3, Speed);

 analogWrite(in2, Speed);

}

void right() {

 analogWrite(in4, Speed);

 analogWrite(in1, Speed);

}

void forwardleft() {

 analogWrite(in1, Speedsec);

 analogWrite(in3, Speed);

}

void forwardright() {

 analogWrite(in1, Speed);

 analogWrite(in3, Speedsec);

}

void backright() {

 analogWrite(in2, Speed);

 analogWrite(in4, Speedsec);

}

void backleft() {

 analogWrite(in2, Speed);

 analogWrite(in4, Speedsec);

}

void Stop() {

 analogWrite(in1, 0);

 analogWrite(in2, 0);

 analogWrite(in3, 0);

 analogWrite(in4, 0);

}

void Auto() {

 //Reserved for future use

}

void Manual() {

 //Reserved for future use

}

