|  |
| --- |
| /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | \* This sketch is for an Arduino 101 BLE Mars rover using Blynk and Adafruit Motor Shield V2 |
|  | \* Code was inspired from both Adafruit and Blynk librarie examples |
|  | \* Full documentation on building this rover yourself on Hackster.IO and electronhacks.com |
|  | \* Sketch is released under MIT license |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  |  |
|  | #define BLYNK\_PRINT Serial |
|  | #include <BlynkSimpleCurieBLE.h> |
|  | #include <CurieBLE.h> |
|  | #include <Wire.h> |
|  | #include <Adafruit\_MotorShield.h> |
|  | #include "utility/Adafruit\_MS\_PWMServoDriver.h" |
|  | #include <Servo.h> |
|  |  |
|  | //Blynk |
|  | char auth[] = "5c5778db5097425c971553510930ff64"; // You should get Auth Token in the Blynk App. |
|  | BLEPeripheral blePeripheral; |
|  |  |
|  | //Adafruit Motor Shield |
|  | Adafruit\_MotorShield AFMS = Adafruit\_MotorShield(); // Create the motor shield object with the default I2C address |
|  | Adafruit\_DCMotor \*motor1 = AFMS.getMotor(1); // Select which 'port' M1, M2, M3 or M4. |
|  | Adafruit\_DCMotor \*motor2 = AFMS.getMotor(2); |
|  | Adafruit\_DCMotor \*motor3 = AFMS.getMotor(3); |
|  | Adafruit\_DCMotor \*motor4 = AFMS.getMotor(4); |
|  |  |
|  | //Servo |
|  | Servo servo1; // create servo object to control a servo |
|  | Servo servo2; |
|  | Servo servo3; |
|  | Servo servo4; |
|  | int pos1 = 0; // variable to store the servo position |
|  | int pos2 = 0; |
|  | int pos3 = 0; |
|  | int pos4 = 0; |
|  | int vDirection = 1; |
|  | int vConfiguration = 1; |
|  | int vTurn = 0; |
|  | int vSpeed = 0; |
|  |  |
|  |  |
|  | //######### SETUP ###################################### |
|  | void setup() { |
|  | Serial.begin(9600); |
|  | delay(1000); |
|  |  |
|  | //Blynk BLE The name your bluetooth service will show up as, customize this if you have multiple devices |
|  | blePeripheral.setLocalName("EHMarsRover"); |
|  | blePeripheral.setDeviceName("EHMarsRover"); |
|  | blePeripheral.setAppearance(384); |
|  | Blynk.begin(auth, blePeripheral); |
|  | blePeripheral.begin(); |
|  | Serial.println("Waiting for connections..."); |
|  | Serial.println("Adafruit Motorshield v2 - DC Motor"); |
|  |  |
|  | //Adafruit Motor Shield |
|  | AFMS.begin(); // create with the default frequency 1.6KHz |
|  | motor1->setSpeed(255); // Set the speed to start, from 0 (off) to 255 (max speed) |
|  | motor2->setSpeed(255); |
|  | motor3->setSpeed(255); |
|  | motor4->setSpeed(255); |
|  |  |
|  | //Servo |
|  | servo1.attach(3); // attaches the servo on pin 3 to the servo object |
|  | servo2.attach(5); |
|  | servo3.attach(6); |
|  | servo4.attach(9); |
|  |  |
|  | } |
|  |  |
|  |  |
|  | //########## LOOP ###################################### |
|  | void loop() { |
|  | Blynk.run(); |
|  | blePeripheral.poll(); |
|  | } |
|  |  |
|  |  |
|  | //######### Subrutines ################################ |
|  |  |
|  | // This function will set the speed |
|  | BLYNK\_WRITE(V0) |
|  | { |
|  | vSpeed = param.asInt(); // assigning incoming value from pin V1 to a variable |
|  | motor1->setSpeed(vSpeed); |
|  | motor2->setSpeed(vSpeed); |
|  | motor3->setSpeed(vSpeed); |
|  | motor4->setSpeed(vSpeed); |
|  | Serial.print("V0... vSpeed: "); |
|  | Serial.print(vSpeed); |
|  | Serial.print(" vDirection: "); |
|  | Serial.print(vDirection); |
|  | Serial.print(" vTurn: "); |
|  | Serial.print(vTurn); |
|  | Serial.print(" vConfiguration: "); |
|  | Serial.println(vConfiguration); |
|  | //Serial.println("V0... vSpeed: " & vSpeed & " vDirection: " + vDirection + " vTurn: " + vTurn + " vConfiguration: " + vConfiguration); |
|  |  |
|  | motorStuff(); |
|  | } |
|  |  |
|  |  |
|  | // Direction |
|  | BLYNK\_WRITE(V1) |
|  | { |
|  |  |
|  | int pinValue = param.asInt(); // assigning incoming value from pin V1 to a variable |
|  | if(pinValue == 1) { |
|  | vDirection = 1; |
|  | } |
|  | if(pinValue == 0) { |
|  | vDirection = 0; |
|  | } |
|  | motorStuff(); |
|  | Serial.print(" vDirection: "); |
|  | Serial.println(vDirection); |
|  | } |
|  |  |
|  |  |
|  | // Configuration |
|  | BLYNK\_WRITE(V2) |
|  | { |
|  | int pinValue = param.asInt(); // assigning incoming value from pin V1 to a variable |
|  | if(pinValue == 0) { |
|  | vConfiguration = 0; |
|  | servoPosNormal(); |
|  | } |
|  | if(pinValue == 1) { |
|  | vConfiguration = 1; |
|  | servoPosStrafe(); |
|  | } |
|  | motorStuff(); |
|  | Serial.print(" vConfiguration: "); |
|  | Serial.println(vConfiguration); |
|  | } |
|  |  |
|  |  |
|  | // Turn |
|  | BLYNK\_WRITE(V3) |
|  | { |
|  | int pinValue = param.asInt(); // assigning incoming value from pin V1 to a variable |
|  | if(pinValue == 1) { |
|  | servoPosTurn(); |
|  | vTurn = 1; |
|  | } |
|  | if(pinValue == 0) { |
|  | if(vConfiguration == 0) { |
|  | servoPosNormal(); |
|  | } |
|  | else{ |
|  | servoPosStrafe(); |
|  | } |
|  | vTurn = 0; |
|  | } |
|  | motorStuff(); |
|  | Serial.print(" vTurn: "); |
|  | Serial.println(vTurn); |
|  | } |
|  |  |
|  |  |
|  |  |
|  | // This function will set the speed |
|  | BLYNK\_WRITE(V4) |
|  | { |
|  | int vStop = param.asInt(); // assigning incoming value from pin V1 to a variable |
|  |  |
|  | if(vStop == 1) { |
|  | motor1->run(RELEASE); |
|  | motor2->run(RELEASE); |
|  | motor3->run(RELEASE); |
|  | motor4->run(RELEASE); |
|  | } |
|  | } |
|  |  |
|  |  |
|  |  |
|  | void motorStuff() { |
|  |  |
|  | if(vSpeed == 0) { |
|  | motor1->run(RELEASE); |
|  | motor2->run(RELEASE); |
|  | motor3->run(RELEASE); |
|  | motor4->run(RELEASE); |
|  | } |
|  | else if(vTurn == 1) { |
|  | if(vDirection == 1) { |
|  | motor1->run(FORWARD); |
|  | motor2->run(BACKWARD); |
|  | motor3->run(FORWARD); |
|  | motor4->run(BACKWARD); |
|  | } |
|  | if(vDirection == 0) { |
|  | motor1->run(BACKWARD); |
|  | motor2->run(FORWARD); |
|  | motor3->run(BACKWARD); |
|  | motor4->run(FORWARD); |
|  | } |
|  | } |
|  | else if(vConfiguration == 1) { |
|  | if(vDirection == 1) { |
|  | motor1->run(FORWARD); |
|  | motor2->run(BACKWARD); |
|  | motor3->run(BACKWARD); |
|  | motor4->run(FORWARD); |
|  | } |
|  | if(vDirection == 0) { |
|  | motor1->run(BACKWARD); |
|  | motor2->run(FORWARD); |
|  | motor3->run(FORWARD); |
|  | motor4->run(BACKWARD); |
|  | } |
|  | } |
|  | else if(vConfiguration == 0) { |
|  | if(vDirection == 1) { |
|  | motor1->run(FORWARD); |
|  | motor2->run(FORWARD); |
|  | motor3->run(FORWARD); |
|  | motor4->run(FORWARD); |
|  | } |
|  | if(vDirection == 0) { |
|  | motor1->run(BACKWARD); |
|  | motor2->run(BACKWARD); |
|  | motor3->run(BACKWARD); |
|  | motor4->run(BACKWARD); |
|  | } |
|  | } |
|  | } |
|  |  |
|  | void servoPosNormal() { |
|  | servo1.write(178); |
|  | servo2.write(0); |
|  | servo3.write(7); |
|  | servo4.write(175); |
|  | } |
|  | void servoPosStrafe() { |
|  | servo1.write(75); |
|  | servo2.write(100); |
|  | servo3.write(108); |
|  | servo4.write(70); |
|  | } |
|  | void servoPosTurn() { |
|  | servo1.write(115); |
|  | servo2.write(65); |
|  | servo3.write(65); |
|  | servo4.write(115); |
|  | } |
|  |  |
|  | //Individual servo functions for calibration |
|  | // Servo1 |
|  | BLYNK\_WRITE(V10) |
|  | { |
|  | int pinValue = param.asInt(); // assigning incoming value from pin V1 to a variable |
|  | servo1.write(pinValue); |
|  | } |
|  |  |
|  | // Servo2 |
|  | BLYNK\_WRITE(V11) |
|  | { |
|  | int pinValue = param.asInt(); // assigning incoming value from pin V1 to a variable |
|  | servo2.write(pinValue); |
|  | } |
|  |  |
|  | // Servo3 |
|  | BLYNK\_WRITE(V12) |
|  | { |
|  | int pinValue = param.asInt(); // assigning incoming value from pin V1 to a variable |
|  | servo3.write(pinValue); |
|  | } |
|  |  |
|  | // Servo4 |
|  | BLYNK\_WRITE(V13) |
|  | { |
|  | int pinValue = param.asInt(); // assigning incoming value from pin V1 to a variable |
|  | servo4.write(pinValue); |
|  | } |