### ARDUINO PROGRAMMING

#include <SoftwareSerial.h>

#include <Servo.h>

#include <Stepper.h>

Servo servo01;

Servo servo02;

Servo servo03;

Servo servo04;

Servo servo05;

SoftwareSerial Bluetooth(0, 1); // Arduino(RX, TX) - HC-06 Bluetooth (TX, RX)

int servo1Pos, servo2Pos, servo3Pos, servo4Pos, servo5Pos; // current position

int servo1PPos, servo2PPos, servo3PPos, servo4PPos, servo5PPos; // previous position

int index = 0;

String dataIn = "";

void setup() {

servo01.attach(8);

servo02.attach(9);

servo03.attach(10);

servo04.attach(11);

servo05.attach(12);

Bluetooth.begin(38400); // Default baud rate of the Bluetooth module

Bluetooth.setTimeout(1);

delay(20);

// Robot arm initial position

servo1PPos = 90;

servo01.write(servo1PPos);

servo2PPos = 150;

servo02.write(servo2PPos);

servo3PPos = 35;

servo03.write(servo3PPos);

servo4PPos = 140;

servo04.write(servo4PPos);

servo5PPos = 85;

servo05.write(servo5PPos);

}

void loop() {

// Check for incoming data

if (Bluetooth.available() > 0) {

dataIn = Bluetooth.readString(); // Read the data as string

// If "Waist" slider has changed value - Move Servo 1 to position

if (dataIn.startsWith("s1")) {

String dataInS = dataIn.substring(2, dataIn.length()); // Extract only the number. E.g. from "s1120" to "120"

servo1Pos = dataInS.toInt(); // Convert the string into integer

// We use for loops so we can control the speed of the servo

// If previous position is bigger then current position

if (servo1PPos > servo1Pos) {

for ( int j = servo1PPos; j >= servo1Pos; j--) { // Run servo down

servo01.write(j);

delay(20); // defines the speed at which the servo rotates

}

}

// If previous position is smaller then current position

if (servo1PPos < servo1Pos) {

for ( int j = servo1PPos; j <= servo1Pos; j++) { // Run servo up

servo01.write(j);

delay(20);

}

}

servo1PPos = servo1Pos; // set current position as previous position

}

// Move Servo 2

if (dataIn.startsWith("s2")) {

String dataInS = dataIn.substring(2, dataIn.length());

servo2Pos = dataInS.toInt();

if (servo2PPos > servo2Pos) {

for ( int j = servo2PPos; j >= servo2Pos; j--) {

servo02.write(j);

delay(50);

}

}

if (servo2PPos < servo2Pos) {

for ( int j = servo2PPos; j <= servo2Pos; j++) {

servo02.write(j);

delay(50);

}

}

servo2PPos = servo2Pos;

}

// Move Servo 3

if (dataIn.startsWith("s3")) {

String dataInS = dataIn.substring(2, dataIn.length());

servo3Pos = dataInS.toInt();

if (servo3PPos > servo3Pos) {

for ( int j = servo3PPos; j >= servo3Pos; j--) {

servo03.write(j);

delay(30);

}

}

if (servo3PPos < servo3Pos) {

for ( int j = servo3PPos; j <= servo3Pos; j++) {

servo03.write(j);

delay(30);

}

}

servo3PPos = servo3Pos;

}

// Move Servo 4

if (dataIn.startsWith("s4")) {

String dataInS = dataIn.substring(2, dataIn.length());

servo4Pos = dataInS.toInt();

if (servo4PPos > servo4Pos) {

for ( int j = servo4PPos; j >= servo4Pos; j--) {

servo04.write(j);

delay(30);

}

}

if (servo4PPos < servo4Pos) {

for ( int j = servo4PPos; j <= servo4Pos; j++) {

servo04.write(j);

delay(30);

}

}

servo4PPos = servo4Pos;

}

// Move Servo 5

if (dataIn.startsWith("s5")) {

String dataInS = dataIn.substring(2, dataIn.length());

servo5Pos = dataInS.toInt();

if (servo5PPos > servo5Pos) {

for ( int j = servo5PPos; j >= servo5Pos; j--) {

servo05.write(j);

delay(30);

}

}

if (servo5PPos < servo5Pos) {

for ( int j = servo5PPos; j <= servo5Pos; j++) {

servo05.write(j);

delay(30);

}

}

servo5PPos = servo5Pos;

}

}

//Stepper Motor

int step\_pin\_1 = 8;

int step\_pin\_2 = 9;

int step\_pin\_3 = 10;

int step\_pin\_4 = 11;

float delay\_time;

int value\_bluetooth = 0;

String readString;

void setup()

{

Serial.begin(9600);

}

void loop()

{

while (Serial.available()){

char c= Serial.read();

readString+=c;

}

if(readString.length() >0) {

value\_bluetooth =readString.toInt();

if (value\_bluetooth > 0 ){

delay\_time = 15;

}

else

{

delay\_time=0;

}

Serial.println(value\_bluetooth);

digitalWrite(8, HIGH); digitalWrite(step\_pin\_2, HIGH); digitalWrite(step\_pin\_3, LOW); digitalWrite(step\_pin\_4, LOW);

delay(delay\_time);

digitalWrite(step\_pin\_1, LOW); digitalWrite(step\_pin\_2, HIGH); digitalWrite(step\_pin\_3, HIGH); digitalWrite(step\_pin\_4, LOW);

delay(delay\_time);

digitalWrite(step\_pin\_1, LOW); digitalWrite(step\_pin\_2, LOW); digitalWrite(step\_pin\_3, HIGH); digitalWrite(step\_pin\_4, HIGH);

delay(delay\_time);

digitalWrite(step\_pin\_1, HIGH); digitalWrite(step\_pin\_2, LOW); digitalWrite(step\_pin\_3, LOW); digitalWrite(step\_pin\_4, HIGH);

delay(delay\_time);

readString="";

}

}

}