ДОДАТОК А

Програмний код керування модульною робототехнічною системою

```
#include <SoftwareSerial.h>
#include <Servo.h>

Servo servo01;
Servo servo02;
Servo servo03;
Servo servo04;
Servo servo05;
Servo servo06;

SoftwareSerial Bluetooth(3, 4); // Arduino(RX, TX) - HC-05 Bluetooth (TX, RX)

int servo1Pos, servo2Pos, servo3Pos, servo4Pos, servo5Pos, servo6Pos;
// current position
int servo1PPos, servo2PPos, servo3PPos, servo4PPos, servo5PPos, servo6PPos; //
previous position
int servo01SP[50], servo02SP[50], servo03SP[50], servo04SP[50], servo05SP[50],
servo06SP[50]; // for storing positions/steps
int speedDelay = 20;
int index = 0;
String dataIn = "";

void setup() {
  servo01.attach(5);
  servo02.attach(6);
  servo03.attach(7);
  servo04.attach(8);
  servo05.attach(9);
  servo06.attach(10);
  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);
  servo6PPos = 80;
  servo06.write(servo6PPos);
}

void loop() {
  // Check for incoming data
  if (Bluetooth.available() > 0) {
    dataIn = Bluetooth.readString();
    if (dataIn.startsWith("s1")) {
      String dataInS = dataIn.substring(2);
```

```
  servo1Pos = dataInS.toInt();
  if (servo1PPos > servo1Pos) {
    for (int j = servo1PPos; j >= servo1Pos; j--) {
      servo01.write(j);
      delay(20);
    }
  }
  if (servo1PPos < servo1Pos) {
    for (int j = servo1PPos; j <= servo1Pos; j++) {
      servo01.write(j);
      delay(20);
    }
  }
  servo1PPos = servo1Pos;
}

if (dataIn.startsWith("s2")) {
  String dataInS = dataIn.substring(2);
  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;
}

if (dataIn.startsWith("s3")) {
  String dataInS = dataIn.substring(2);
  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;
}

if (dataIn.startsWith("s4")) {
  String dataInS = dataIn.substring(2);
  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;
}

if (dataIn.startsWith("s5")) {
  String dataInS = dataIn.substring(2);
  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;
}

if (dataIn.startsWith("s6")) {
  String dataInS = dataIn.substring(2);
  servo6Pos = dataInS.toInt();
  if (servo6PPos > servo6Pos) {
    for (int j = servo6PPos; j >= servo6Pos; j--) {
      servo06.write(j);
      delay(30);
    }
  }
  if (servo6PPos < servo6Pos) {
    for (int j = servo6PPos; j <= servo6Pos; j++) {
      servo06.write(j);
      delay(30);
    }
  }
  servo6PPos = servo6Pos;
}

if (dataIn.startsWith("SAVE")) {
  servo01SP[index] = servo1PPos;
  servo02SP[index] = servo2PPos;
  servo03SP[index] = servo3PPos;
  servo04SP[index] = servo4PPos;
  servo05SP[index] = servo5PPos;
  servo06SP[index] = servo6PPos;
  index++;
}

if (dataIn.startsWith("RUN")) {
  runservo();
}
```

```
    if (dataIn == "RESET") {
      memset(servo01SP, 0, sizeof(servo01SP));
      memset(servo02SP, 0, sizeof(servo02SP));
      memset(servo03SP, 0, sizeof(servo03SP));
      memset(servo04SP, 0, sizeof(servo04SP));
      memset(servo05SP, 0, sizeof(servo05SP));
      memset(servo06SP, 0, sizeof(servo06SP));
      index = 0;
    }
  }
}

void runservo() {
  while (dataIn != "RESET") {
    for (int i = 0; i <= index - 2; i++) {
      if (Bluetooth.available() > 0) {
        dataIn = Bluetooth.readString();
        if (dataIn == "PAUSE") {
          while (dataIn != "RUN") {
            if (Bluetooth.available() > 0) {
              dataIn = Bluetooth.readString();
              if (dataIn == "RESET") {
                break;
              }
            }
          }
        }
        if (dataIn.startsWith("ss")) {
          String dataInS = dataIn.substring(2);
          speedDelay = dataInS.toInt();
        }
      }

      // Servo movement logic for each of 6 servos from [i] to [i+1]
      // Servo 1
      if (servo01SP[i] > servo01SP[i + 1]) {
        for (int j = servo01SP[i]; j >= servo01SP[i + 1]; j--) {
          servo01.write(j);
          delay(speedDelay);
        }
      } else if (servo01SP[i] < servo01SP[i + 1]) {
        for (int j = servo01SP[i]; j <= servo01SP[i + 1]; j++) {
          servo01.write(j);
          delay(speedDelay);
        }
      }

      // Servo 2
      if (servo02SP[i] > servo02SP[i + 1]) {
        for (int j = servo02SP[i]; j >= servo02SP[i + 1]; j--) {
          servo02.write(j);
          delay(speedDelay);
        }
      } else if (servo02SP[i] < servo02SP[i + 1]) {
        for (int j = servo02SP[i]; j <= servo02SP[i + 1]; j++) {
          servo02.write(j);
          delay(speedDelay);
        }
      }
```

```
      // Servo 3
      if (servo03SP[i] > servo03SP[i + 1]) {
        for (int j = servo03SP[i]; j >= servo03SP[i + 1]; j--) {
          servo03.write(j);
          delay(speedDelay);
        }
      } else if (servo03SP[i] < servo03SP[i + 1]) {
        for (int j = servo03SP[i]; j <= servo03SP[i + 1]; j++) {
          servo03.write(j);
          delay(speedDelay);
        }
      }

      // Servo 4
      if (servo04SP[i] > servo04SP[i + 1]) {
        for (int j = servo04SP[i]; j >= servo04SP[i + 1]; j--) {
          servo04.write(j);
          delay(speedDelay);
        }
      } else if (servo04SP[i] < servo04SP[i + 1]) {
        for (int j = servo04SP[i]; j <= servo04SP[i + 1]; j++) {
          servo04.write(j);
          delay(speedDelay);
        }
      }

      // Servo 5
      if (servo05SP[i] > servo05SP[i + 1]) {
        for (int j = servo05SP[i]; j >= servo05SP[i + 1]; j--) {
          servo05.write(j);
          delay(speedDelay);
        }
      } else if (servo05SP[i] < servo05SP[i + 1]) {
        for (int j = servo05SP[i]; j <= servo05SP[i + 1]; j++) {
          servo05.write(j);
          delay(speedDelay);
        }
      }

      // Servo 6
      if (servo06SP[i] > servo06SP[i + 1]) {
        for (int j = servo06SP[i]; j >= servo06SP[i + 1]; j--) {
          servo06.write(j);
          delay(speedDelay);
        }
      } else if (servo06SP[i] < servo06SP[i + 1]) {
        for (int j = servo06SP[i]; j <= servo06SP[i + 1]; j++) {
          servo06.write(j);
          delay(speedDelay);
        }
      }
    }
  }
}
```

ДОДАТОК Б

Демонстраційний матеріал

№ докумен-та	Позначення	Найменування	Додаткові відомості
		Текстові документи	
1	ГЮИК.ХХХХХХ.038ПЗ	Пояснювальна записка	А4, 62 с.
		Додаткові матеріали	
2		Повний програмний код для керування модульною робототехнічною системою	А4, 5 с.
3		Демонстраційний матеріал	А4, 11 с.

Змін.	Арк.	Номер докум.	Підпис	Дата	ГЮИК.ХХХХХХ.038ВД			
Розроб.		Стрельченко М.Д.			Розроблення робототехної системи виробничого призначення за принципом модульності	Літера	Аркуш	Аркушів
Перевір.		Демська Н.П.				Н	1	1
Н.контр.		Демська Н.П.				Кафедра КІТАР ХНУРЕ		
Затв.		Невлюдов І.Ш.						