

№10 AMALIY MASHG'ULOT.

AQLLI SHAHARNING HARAKATINI AVTOMATIK BOSHQARISH UCHUN INTELEKTUAL TIZIMLARNI ISHLAB CHIQISH.

Ishning maqsadi: Arduino UNO ga servo va step motorini ulash uchun apparat va dasturiy ta'minot modellarini o'rganish. Amaldagi tizimning tarkibiy qismlari haqida texnik xususiyatlar va umumiy ma'lumotlarni o'rganish.

№ 10.1. Amaliy mashg'ulot Arduino-ga servo ulanish.

Mashg'ulotning maqsadi: Ushbu mashg'ulotda biz servo uzatmalarining qurilmasi va ishlash printsipini ko'rib chiqamiz. Arduino potansiyometri yordamida SG90 servosini boshqarish uchun ikkita oddiy eskizni tahlil qilaylik. Shuningdek, biz C++ dasturlash tilidagi yangi buyruqlarni o'rganamiz - **servo.write** , **servo.read** , **servo.attach** va Arduino orqali servo va boshqa qurilmalarni boshqarish uchun eskizlardagi kutubxonani qanday ulashni o'rganamiz.

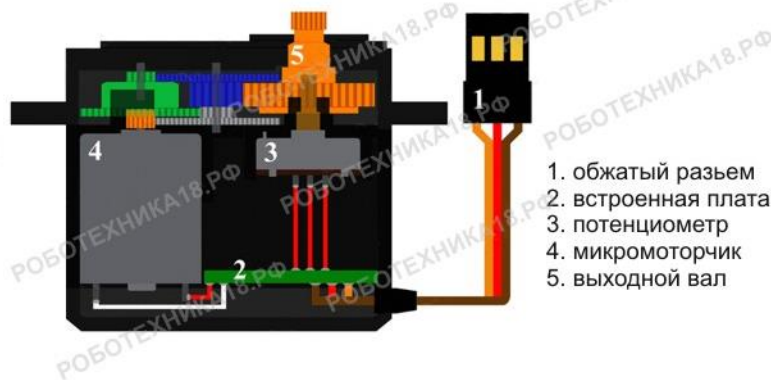
Umumiy ish rejasi:

1. Arduino mikro servo vosita qurilmasi
2. Arduino servo ulanish sxemasi
3. Arduino Uno mikro servo eskizi
4. Arduino potansiyometridan servo boshqaruv



10.1-rasm. Servo vosita qurilmasi (servo) Arduino

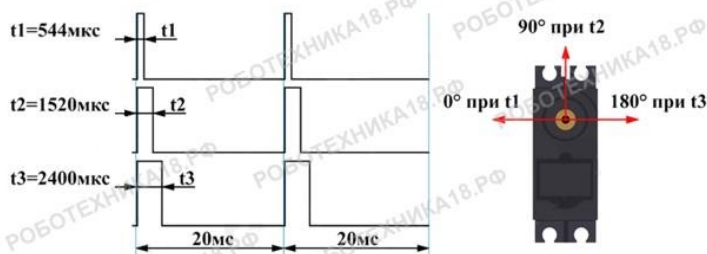
Servo haydovchi (servo vosita) turli xil robotlar va mexanizmlarni loyihalashda muhim element hisoblanadi. Bu mexanizmlarning harakatini aniq boshqarish imkonini beruvchi fikr-mulohazalarga ega bo'lgan aniq ijrochi . Boshqacha qilib aytganda, kirishda boshqaruv signalining qiymatini qabul qilib, servomotor o'z aktuatorining chiqishida ushbu qiymatni saqlab qolishga intiladi.



10.2-rasm. Servo qurilma diagrammasi

Servo drayvlar robotlarning mexanik harakatlarini taqlid qilish uchun keng qo'llaniladi. Servo haydovchi sensordan (tezlik, joylashuv va boshqalar), mexanik tizimdan qo'zg'alish boshqaruv bloki va elektron sxemadan iborat. Qurilmaning reduktorlari (tishli) metall, uglerod tolasi yoki plastmassadan tayyorlangan. Servomotorning plastik viteslari og'ir yuklarga va zarbalarga bardosh bermaydi.

Зависимость положения вала от ширины импульсов



Изменяя ширину импульсов в данных пределах мы сможем точно задавать угол поворота выводного вала

10.3-rasm. Puls kengligi modulyatsiyasi bilan servo boshqaruv

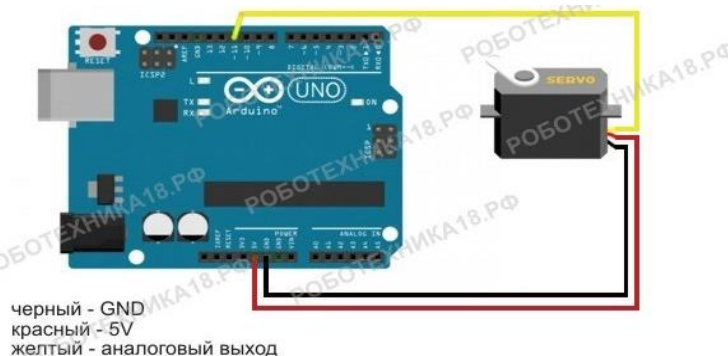
Servomotorlarda chiqish miliga ulangan o'rnatilgan potansiyometr mavjud. Milni aylantirib, servo potansiyometrda kuchlanish qiymatini o'zgartiradi. Kengash kirish signalining kuchlanishini tahlil qiladi va uni potansiyometrda kuchlanish bilan taqqoslaydi, farqga asoslanib, vosita chiqish va potansiyometrda kuchlanishni tenglashtirmaguncha silliqlik aylanadi.

Servoni Arduino-ga qanday ulash mumkin

Ushbu mashg'ulot uchun sizga kerak bo'ladi:

- Arduino Uno / Arduino Nano / Arduino Mega;
- Servo SG90, MG995, MG996;
- potansiyometr (o'zgaruvchan qarshilik);
- plata;
- «папа-папа» va «папа-мама».

Servoni Arduino-ga ulash sxemasi odatda quyidagicha: qora simni GND ga ulang, qizil simni 5V ga, to'q sariq / sariq simni umumiy maqsadli raqamli pinga ulang. Arduino-da servo nazorat qilish juda oddiy, ammo servolar aylanish burchaklari bo'yicha 180 ° va 360 ° ga bo'linadi, ular servo yordamida loyihalarni ishlab chiqishda hisobga olinishi kerak.



10.4-rasm. Arduino UNO uchun servo ulanish diagrammasi

Servo	Arduino Uno	Arduino Nano	Arduino Mega
qora sim	GND	GND	GND
qizil sim	5V	5V	5V
sariq sim	o'n bir	o'n bir	o'n bir

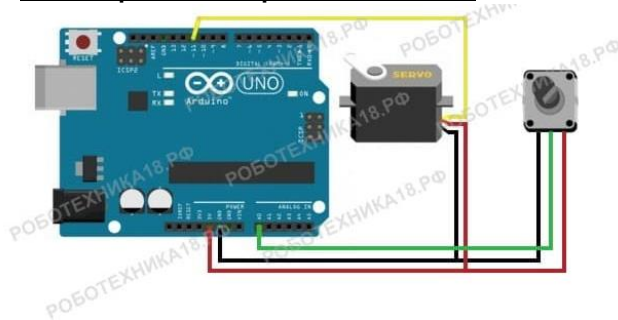
Birinchi eskizda `myservo.write(0)` buyrug'i yordamida Arduino-da servoni qanday boshqarishni ko'rib chiqamiz. Bundan tashqari, biz `Servo.h` standart kutubxonasidan foydalanamiz. Yuqoridagi rasmdagi diagrammaga muvofiq servoni Arduino platasiga ulang va tayyor eskizni yuklang. `Void loop()` protsedurasida biz shunchaki servo uchun kerakli burilish burchagini va keyingi aylanishgacha kutish vaqtini o'rnatamiz.

Arduino servo eskizi

```
#include < Servo .h> // servo bilan ishlash uchun kutubxonani qo'shing
Servo servo1; // "servo1" tipidagi servo o'zgaruvchini e'lon qilish
void setup() {
    servo1.attach(11); // servoni analog chiqish 11 ga ulash
}
void loop() {
    servo1.write(0); // burilish burchagini 0 ga o'rnatish
    delay (2000); // 2 soniya kuting
    servo1.write (90); // burilish burchagini 90 ga o'rnatish
    delay (2000); // 2 soniya kuting
    servo1.write (180); // burilish burchagini 180 ga o'rnatish
    delay (2000); // 2 soniya kuting
}
```

Kod uchun tushuntirishlar:

1. Servo o'zgaruvchisi bir nechta servolarni Arduino-ga ulashda chalkashmaslik uchun kerak. Biz har bir haydovchiga o'z nomini beramiz;
2. `servo1.attach (10)` drayverni raqamli chiqish 10 ga bog'laydi.
3. dasturda biz haydovchini 0-90-180 gradusga aylantiramiz va dastlabki holatga qaytamiz, chunki bo'sh pastadir protsedurasi tsiklik takrorlanadi.



10.5-rasm. Servo va potansiyometrni Arduino Uno-ga ulash

Arduino nafaqat nazorat qilish, balki servo ko'rsatkichlarini o'qish imkonini beradi. `MyServo.read (0)` buyrug'i joriy servo mil burchagini o'qiydi va biz uni port monitorida ko'rishimiz mumkin. Keling, Arduino-da potentsiometr yordamida servo boshqarishning yanada murakkab misolini keltiraylik. Potansiyometr bilan sxemani yarating va servoni boshqarish uchun eskizni Arduino-ga yuklang.

Potansiyometrli servo uchun eskiz

```
#include < Servo .h> // servo bilan ishlash uchun kutubxonani qo'shing

Servo servo // "servo" tipidagi servo o'zgaruvchini e'lon qilish

void setup() {
  servo.attach(10); // servoni analog chiqish 10 ga ulash
  pinMode (A0, INPUT); // potansiyometrni analog kirish A0 ga ulang
  Serial . boshlash (9600); // port monitorini ulash
}

void loop() {
  servo.write(analogRead(A0)/4); // servo mil uchun qiymatlarni o'tkazadi

  Serial.println (analogRead(A0)); // monitorda potansiyometr ko'rsatkichlarini
ko'rsatish
  Serial . println ( analogRead (A0)/4); // servoga qo'llaniladigan signalni chiqaring

  Serial . println (); // port monitoriga bo'sh qator chiqarish

  delay (1000); // bir soniya kechikish
}
```

Kod uchun tushuntirishlar:

1. bu safar biz eskizdagi servoni servo deb nomladik ;
2. `servo.write (analogRead(A0)/4)` buyrug'i servo mil uchun qiymatlarni uzatadi - biz potansiyometrdan olingan kuchlanishni to'rtga bo'lamiz va bu qiymatni SG90, MG995, MG996 Arduino servomotoriga yuboramiz.
3. `Serial.println (servo.read(10))` buyrug'i servo mil burchagi qiymatini o'qiydi va uni port monitoriga yuboradi.

Xulosa _ Servo motorlar ko'pincha turli funktsiyalar uchun turli Arduino loyihalarida qo'llaniladi : burilish tuzilmalari, mexanizmlarning harakatlanuvchi qismlari. Arduino servo doimiy ravishda ma'lum bir aylanish burchagini saqlashga intilayotganligi sababli, quvvat sarfini oshirishga tayyor bo'ling. Bu, ayniqsa, batareyalar yoki batareyalar bilan ishlaydigan avtonom robotlarda sezgir bo'ladi.

№10.2. Amaliy mashg'ulot Step motorni Arduino-ga ulash.

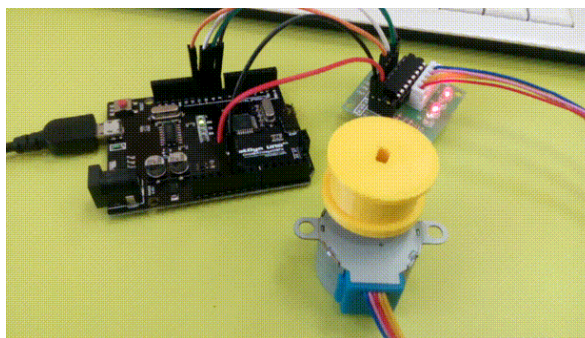
Mashg'ulotning maqsadi: Bosqichli dvigatel ob'ektni ma'lum miqdordagi mil qadamlari bilan aniq joylashtirish yoki harakatlantirish uchun mo'ljallangan. Arduino taxtasi haydovchi va **stepper.h** yoki **accelstepper.h** kutubxonasi yordamida step motorini boshqarishi mumkin. Arduino Uno / Nano-ga step motorining ishlash printsipli va ulanish sxemasini ko'rib chiqing, shuningdek, step motorini boshqarish uchun eskizni tahlil qiling.

Umumiy ish rejasi:

1. Arduino step motor qurilmasi
2. Arduino step motori uchun haydovchi
3. Arduino-ga step motorini qanday ulash mumkin
4. Arduino step motorini boshqarish
5. Accelstepper Arduino kutubxonasining tavsifi



10.6-rasm. Arduino step motor qurilmasi



10.7-rasm. Step motorni Arduino-ga ulash

Step motor qanday ishlaydi

Dizaynga qarab, bugungi kunda uch turdagi step motorlar qo'llaniladi: doimiy magnit, o'zgaruvchan istaksiz va gibrid motorlar. Doimiy magnitlangan motorlar uchun milning aylanishida qadamlar soni 48 ga etadi, ya'ni bir qadam milning $7,5^\circ$ aylanishiga to'g'ri keladi. Gibrid motorlar aylanish uchun kamida 400 qadamni ta'minlaydi (qadam burchagi $0,9^\circ$).



10.8-rasm. Bosqichli dvigatelning kesma ko'rinishi

Qabul qilingan qadamlar sonini hisoblab, rotorning burilish burchagini aniq aniqlash mumkin. Shunday qilib, step motor bugungi kunda 3D printerlar, CNC mashinalari va boshqa sanoat uskunalarida ideal haydovchi hisoblanadi. Bu qurilmaning qisqacha ko'rinishi va step motorining ishlash printsipli, biz Arduino yordamida step motorini qanday boshqarishga ko'proq qiziqamiz.

Arduino step motor haydovchi

Bosqichli vosita cho'tkasiz sinxron vosita bo'lib, barcha motorlar kabi elektr energiyasini mexanik energiyaga aylantiradi. Mil aylanadigan doimiy dvigateldan farqli o'laroq, step motorlarining mili diskret harakatlar qiladi, ya'ni u doimiy ravishda emas, balki qadamlar bilan aylanadi. Milning (rotor) har bir qadami to'liq inqilobning bir qismidir.

Dvigatel milining aylanishi haydovchi statoridagi sariqlarning magnit maydonini boshqaruvchi signal yordamida amalga oshiriladi. Signal step motor drayveri tomonidan ishlab chiqariladi. Stator sariqlarida elektr tokining o'tishi natijasida hosil bo'lgan magnit maydon magnitlar o'rnatilgan milning aylanishiga olib keladi. Qadamlar soni dasturda Arduino IDE kutubxonasi yordamida o'rnatiladi.



10.9-rasm. Step motorni boshqarish uchun drayverlarning turlari

ULN2003 drayveri orqali 28BYJ-48 step motorini Arduino Uno-ga ulash diagrammasi quyidagi rasmda ko'rsatilgan. 28BYJ-48 dvigatelining asosiy xarakteristikalar: 5 yoki 12 volt, 4 fazali vosita, qadam burchagi $5,625^\circ$ tomonidan quvvatlanadi. IN1 - IN4 haydovchi portlari Arduino Mega yoki Nano platadagi istalgan raqamli pinlarga ulanadi. Modulda LEDlar vosita bobinlarining faollashishini ko'rsatish uchun xizmat qiladi.

Step motorni arduinoga qanday ulash mumkin

Ushbu mashg'ulot uchun sizga kerak bo'ladi:

- Arduino Uno / Arduino Nano / Arduino Mega;
- step motor drayveri ULN2003;
- step motor 28BYJ-48;



10.10-rasm. Arduino UNO uchun step motorini ulash diagrammasi

Step vosita Arduino orqali ma'lum bir ketma-ketlikda vosita o'rashlariga impulslarni qo'llash orqali boshqariladi. Step motorni boshqarishni osonlashtirish uchun stepper.h va accelstepper.h maxsus kutubxonalari yaratilgan, ammo siz standart kutubxonalarsiz motor milini aylantirishingiz mumkin. Yuqoridagi diagrammadagi kabi qadam motorini mikrokontrollerga ulang va quyidagi eskizni yuklang.

Step motorni boshqarish uchun eskiz

```
// ULN2003 modulini Arduino-ga ulash uchun portlar
```

```
#define in1 8
```

```
#define in2 9
```

```
#define in3 10
```

```
#define in4 11
```

```
void setup() {
```

```
    pinMode(in1, OUTPUT);
```

```
    pinMode(in2, OUTPUT);
```

```
    pinMode(in3, OUTPUT);
```

```
    pinMode(in4, OUTPUT);
```

```
}
```

```
void loop() {
```

```
    digitalWrite(in1, HIGH);
```

```
    digitalWrite(in2, LOW);
```

```
    digitalWrite(in3, LOW);
```

```
    digitalWrite(in4, HIGH);
```

```
    delay(dl);
```

```
    digitalWrite(in1, HIGH);
```

```
    digitalWrite(in2, HIGH);
```

```
    digitalWrite(in3, LOW);
```

```
    digitalWrite(in4, LOW);
```

```
    delay(dl);
```

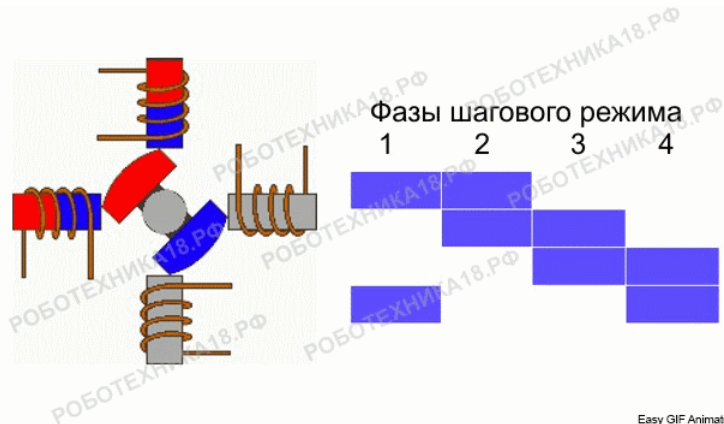
```

digitalWrite(in1, LOW);
digitalWrite(in2, HIGH);
digitalWrite(in3, HIGH);
digitalWrite(in4, LOW);
delay(dl);
digitalWrite(in1, LOW);
digitalWrite(in2, LOW);
digitalWrite(in3, HIGH);
digitalWrite(in4, HIGH);
delay(dl);
}

```

Kod uchun tushuntirishlar:

1. 8,9,10,11 portlari o'rniga istalgan raqamli portlardan foydalanishingiz mumkin;
2. kechikish vaqti millisekundlarda `int dl = 5;` o'zgartirilishi mumkin, dasturda kechikish qanchalik kichik bo'lsa, vosita mili tezroq aylanadi;
3. dasturning algoritmi quyidagi rasmda ko'rsatilgan.



10.11-rasm. Step motorning sariqlarini yoqish ketma-ketligi

Arduino bilan step motorini boshqarish

Keling, standart `stepper.h` kutubxonasidan foydalangan holda va yanada qulayroq va ommabop `accelstepper.h` kutubxonasidan foydalangan holda step motorini boshqarishni ko'rib chiqaylik, bu erda yana ko'plab qo'shimcha buyruqlar va funktsiyalar amalga oshiriladi. Dvigatelning ulanish sxemasi o'zgarishsiz qoladi. Birozdan keyin biz ushbu kutubxonalar bilan dasturda foydalanish mumkin bo'lgan buyruqlarni sanab o'tdik.

Arduino-da step motorining eskizi (`Stepper.h`)

```

#include <Stepper.h> // step motor kutubxonasi
// aylanish uchun qadamlar soni, dvigatelningiz qiymatini o'zgartiring
const int stepPerRevolution = 200;

```



```
// drayverlarga ulanish uchun portlarni o'rnatish
Stepper myStepper(stepsPerRevolution, 8, 9, 10, 11);
void setup() {
  myStepper.setSpeed(60); // tezlikni 60 rpm ga o'rnatish
}
void loop() {
  // rotni soat yo'nalishi bo'yicha aylantirish
  myStepper.step(stepsPerRevolution);
  delay(500);
  // rotni soat miliga teskari burang
  myStepper.step(-stepsPerRevolution);
  delay(500);
}
```

Tushuntirishlar uchun kod :

1. 8,9,10,11 portlari o'rniga istalgan raqamli portlardan foydalanishingiz mumkin;
2. Stepper.h kutubxonasi kam funktsionallikka ega, shuning uchun u faqat step motorini sinab ko'rish va haydovchi ulanishini tekshirish uchun javob beradi.
AccelStepper.h kutubxonasi bilan step motorining eskizini

```
#include <AccelStepper.h> // step motor kutubxonasi
```

Kod uchun tushuntirishlar:

1. Motor Shield L293D ga step motorini ulashda AccelStepper.h kutubxonasidan foydalanish mumkin ;
2. dasturda siz nafaqat maksimal tezlikni, balki step vosita milining tezlashishi va sekinlashishini ham o'rnatishingiz mumkin.

AccelStepper.h kutubxonasi buyruqlarining tavsifi

AccelStepper mystepper (DRIVER , qadam , yo'nalish); // Kengash tomonidan boshqariladigan step motor

AccelStepper mystepper (FULL 2 WIRE , pinA , pinB); // H-ko'priqli step motor

AccelStepper mystepper (FULL4WIRE, pinA1, pinA2, pinB1, pinB2); // Bir qutbli dvigatel

mystepper.setMaxSpeed(stepsPerSecond); // RPM tezligini o'rnatish

mystepper.setSpeed(stepsPerSecond); // Tezlikni soniyasiga qadamlar bilan belgilang

mystepper.setAcceleration(stepsPerSecondSquared); // Tezlashtirishni o'rnatish

mystepper.currentPosition(); // Joriy pozitsiyani bosqichma-bosqich qaytaradi

mystepper.setCurrentPosition(uzoq pozitsiya); Qayta tiklash joriy pozitsiya oldin nol

mystepper.targetPosition(); // Yakuniy pozitsiyani bosqichma-bosqich

mystepper.distanceToGo(); // Belgilangan joyga masofani qaytarish

mystepper.moveTo(uzun mutlaq); // Mutlaq belgilangan joyga o'tish

`mystepper.harakat (uzoq qarindosh); // Nisbatan belgilangan joyga o'tish`
`mystepper.run(); // Tezlashtirish bilan harakatni boshlang, funksiya yana`
chaqirilishi kerak
`mystepper.runToPosition(); // Belgilangan nuqtaga o'tishni boshlang`
`mystepper.runToNewPosition(); // Berilgan pozitsiyaga tezlanish bilan harakat`
qilishni boshlang
`mystepper.stop(); // mumkin bo'lgan eng tez to'xtash (sekinlashuv yo'q)`
`mystepper.runSpeed(); // Berilgan tezlikda silliq harakatni boshlang`
`mystepper.runSpeedToPosition(); // Bir tekis tezlashmasdan, pozitsiyaga harakat`
qilishni boshlang
`mystepper.disableOutputs(); // Zaxiralangan pinlarni o'chiradi va ularni LOW`
rejimiga o'rnatadi. Dvigatel sariqlaridan kuchlanishni yo'qotadi, energiyani tejaydi
`mystepper.enableOutputs(); // Zaxiralangan pinlarni faollashtiradi va ularni`
OUTPUT rejimiga o'rnatadi. Dvigatel ishga tushirilganda avtomatik ravishda
chaqiriladi

Dastur yozish algoritmi:

1. Muammoni oydinlashtirish;
2. Nazariy manbalar va foydali materiallarni izlash;
3. Zarur bo'lganda kutubxonani ulash;
4. Dastur kodini yozish;
5. Kompilyatsiya.

Hisobotda quyidagilar bo'lishi kerak:

1. Amaliyot nomlari va ishning maqsadi.
2. Breadboard va tadqiqot tushunchasi.
3. Tadqiqotda foydalaniladigan texnik vositalarning xarakteristikalar.
4. Dastur kodini va kompilyatsiya skrinshotlarini yozish.
5. Bajarilgan vazifa bo'yicha foto / video hisobot.
6. Nazorat savollariga javob yozing (qo'lda)

Nazorat savollari

1. Servo vosita qurilmasi (servo) nima?
2. Servo qurilmaning sxemasini tushuntiring.
3. Arduino step motor qurilmasi atamasini aniqlang.
4. Step motorni boshqarish uchun drayverlarning turlari.
5. Ushbu loyihaning dasturiy ta'minoti uchun qanday kutubxonalardan foydalandingiz?
6. Loyihani ishlab chiqish uchun zarur bo'lgan apparat komponentlari ro'yxatini tuzing.
7. Ushbu loyihadan qanday maqsadlarda foydalanish mumkin?
8. Datchiklarni Arduino ga ulash sxemasini chizing.
9. Ushbu tadqiqotning sxematik sxemasini tuzing.
10. O'rganish sxemasida datchiklar parametrlariga qanday talablar qo'yiladi?