

2023

# ARDUINO IDE

bolashaq robototexnik sáttilik

Ali Talap

# Талап Ертаргынулы Али

# Arduino IDE

*[http://www.litres.ru/pages/biblio\\_book/?art=69596461](http://www.litres.ru/pages/biblio_book/?art=69596461)*

*SelfPub; 2023*

## **Аннотация**

Сәлем, менің есімім Талап. Қазақ тіліндегі робототехника саласындағы медиа және контент саласының қалаушысы. Егер сіз бұл кітапты оқитын болсаңыз, қазақ тілінің дамуына себепшісіз. Кітапша бастапқы Arduino саласының баспалдағы.

# Талап Али

## Arduino IDE

### КІРІСПЕ

#### СӘЛЕМ РОБОТОТЕХНИК!

Есіңіде болсын, Arduino IDE үйрену – бұл біртіндеп жүретін процесс. Шыдамды болыңыз, оқытудың итеративті сипатын қабылдаңыз және көмек сұраудан немесе жаңа идеяларды үйренуден тартынбаңыз. Саяхаттан ләззат алыңыз және Arduino-мен тәжірибе жасаудан ләззат алыңыз!

Мен бұл салаға ерте кіргенімен білетінім, білмейтінім өте көп. Себебі бұл сала әрдайым сізге жаңашылдылығымен ерекшеленеді. Ал сіз осы салада әрқашан ізденісте болуды үйретеді. Басында сізге сұрақтар пайда болуы мүмкін, қалай кодтауды үйренем, қалай ұмытып қалмауға болады немесе қалай тез үйрену керек деген сұрақтар болған бірақ оқи келе және іздене келе пікір өзгере береді. Сізге Arduino жөнінде шағын ақпаратпен бөлісем.

Arduino-Электроника, автоматика, процестерді автоматтандыру және робототехника саласындағы қарапайым жүйелерді, модельдер мен эксперименттерді құру және прототиптеу аппараттық-бағдарламалық құралдарының сауда белгісі.

Бағдарламалық бөлім бағдарламаларды жазуға, оларды құрастыруға және аппаратураны бағдарламалауға арналған ақысыз бағдарламалық қабықтан (IDE) тұрады. Аппараттық

құрал-бұл ресми өндіруші де, үшінші тарап өндірушілері де сататын монтаждалған ПХД жиынтығы. Жүйенің толық ашық архитектурасы еркін көшіруге мүмкіндік береді немесе Arduino өнім желісін толықтырады.

Ол дербес объектілерді құру үшін де, сымды және сымсыз интерфейстер арқылы Бағдарламалық жасақтамаға қосылу үшін де қолданылады. Электрониканы дамыту және бағдарламалау бойынша білімнің минималды кіріс шегі бар бастаушы пайдаланушылар үшін қолайлы.

Керемет! Мен сізге Arduino – да бағдарламалауды бастауға көмектесуге қуаныштымын. Arduino-с ПРОГРАММ бағдарламалау тілінің жеңілдетілген нұсқасын қолданатын ашық бастапқы электрондық платформа. Ол интерактивті жобаларды құру және әртүрлі электрондық құрылғыларды басқару үшін кеңінен қолданылады. Бағдарламалауға кіріспес бұрын, сізде қажетті параметрлер бар екеніне көз жеткізейік:

Жабдық: Arduino тақтасы (мысалы, Arduino Uno, Arduino Nano және т. б.)

Arduino тақтасын компьютерге қосуға арналған USB кабелі

Нан тақтасы және қосқыш сымдар (қосымша, күрделі жобалар үшін)

Бағдарламалық қамтамасыз ету: Arduino IDE (интеграцияланған даму ортасы): оны Arduino ресми сайтынан тегін жүк-

теп алуға болады (<https://www.arduino.cc/en/software> ). Ide-ді амалдық жүйеге сәйкес орнатыңыз.

Барлығы дайын болғаннан кейін, жұмысқа кірісу үшін мына қадамдарды орындаңыз:

1-қадам: Arduino IDE іске қосыңыз: Компьютерге орнатқан Arduino IDE ашыңыз.

2-қадам: Arduino тақтасын қосыңыз: Arduino тақтасын компьютерге USB кабелі арқылы қосыңыз.

Тақтаның дұрыс жалғанғанын және компьютермен анықталғанын тексеріңіз. Тақтаның танылғанын Arduino IDE-дегі "қызмет" мәзіріне қарап тексеруге болады.

3-қадам: Arduino тақтасы мен портын таңдаңыз: Arduino IDE-де "қызмет" мәзіріне өтіңіз.

"ТАҚТА" ішкі мәзірінен тақтаның дұрыс түрін таңдаңыз (мысалы, Arduino Uno, Arduino Nano).

Порт ішкі мәзірінен Arduino тақтасы қосылған портты таңдаңыз. Егер сенімді болмасаңыз, портты құрылғы менеджерінен немесе компьютердің жүйелік профилінен тексеруге болады.

4-қадам: Arduino үшін алғашқы бағдарламаңызды жазыңыз: Arduino IDE-де жаңа нобай терезесін ашатын бос нобай аймағын басыңыз.

Нобай терезесіне келесі кодты енгізіңіз:

```
void setup() {  
  // Code to run once at the beginning  
}  
void loop() {
```

```
// Code to repeat indefinitel
```

```
}
```

5-қадам: бағдарламаны Arduino тақтасына жүктеңіз: Arduino IDE құралдар тақтасындағы Жүктеу түймесін (оң жақ көрсеткі белгісі) басыңыз. Ide ортасы сіздің кодыңызды құрастырады және оны Arduino тақтасына жүктейді. Жүктеу барысын ide төменгі жағындағы күй жолағында көресіз.

6-қадам: кірістірілген жарықдиодты жыпылықтауға Назар аударыңыз: Arduino тақталарының көпшілігінде 13 істікшеге қосылған кіріктірілген жарық диоды бар. Біз жазған әдепкі бағдарлама сол жарық диодты жыпылықтайды. Жүктеу аяқталғаннан кейін сіз Arduino тақтасындағы кіріктірілген жарық диодының тұрақты аралықта қосылып, сөніп тұрғанын көруіңіз керек.

Құттықтаймыз! Сіз Arduino үшін алғашқы бағдарламаны сәтті жазып, жүктедіңіз. Енді сіз кодты өзгерту және әртүрлі компоненттер мен сенсорлармен тәжірибе жасау арқылы Arduino әлемін зерттей бастай аласыз.

Кейінгі сабақтарда біз сандық және аналогтық енгізу/шығару, сыртқы компоненттерді басқару және жетілдірілген бағдарламалау әдістері сияқты тақырыптарды қарастыра аламыз. Келесі жолы қандай нақты салаларға назар аударғыңыз келетінін маған хабарлаңыз, мен сізге одан әрі көмектесуге қуаныштымын!

Бұл оқулықта Arduino көмегімен сандық кірістер мен шы-

ғыстарды қарастырайық. Сандық кіріс түймені басу сияқты сандық сигнал күйін оқуды қамтиды, ал сандық Шығыс жарық диодтары сияқты сандық құрылғыларды басқаруды қамтиды. Сізге жан-жақты түсінік беру үшін екі аспектіні де қарастырамыз.

2-сабақ: сандық енгізу және шығару

1-қадам: схеманы орнату: Жарықдиодты Arduino тақтасына қосыңыз. Светодиодтың ұзын түйреуішін (анодты) сандық түйреуішке (мысалы, 13 түйреуішке), ал қысқа түйреуішті (катодты) Arduino тақтасындағы жерге қосу түйреуішіне (GND) қосыңыз. Шамадан тыс токтан қорғау үшін ток шектейтін резисторды (мысалы, 220-470 Ом) светодиодпен қатар қолданыңыз.

Түймені Arduino тақтасына қосыңыз. Түйменің бір аяғын сандық түйреуішке (мысалы, 2 түйреуіш), ал екінші аяғын Arduino тақтасындағы жерге қосу түйреуішіне (GND) жалғаңыз. Сондай-ақ, тұрақты оқуды қамтамасыз ету үшін түйме шығысы мен 5В шығысы немесе GND арасында жоғары немесе төмен резисторды (мысалы, 10к) қосқыңыз келуі мүмкін.

2-қадам: кодты жазыңыз: Arduino IDE-де бос нобай аймағын басу арқылы жаңа нобай жасаңыз.

Әдепкі кодты жойып, келесі кодты енгізіңіз:

```
// Constants for LED and button pin numbers
```

```
const int ledPin = 13;
```

```
const int buttonPin = 2;
```

```
// Variable to store the button state
int buttonState = 0;

void setup() {
  pinMode(ledPin, OUTPUT); // Set LED pin as output
  pinMode(buttonPin, INPUT); // Set button pin as input
}

void loop() {
  buttonState = digitalRead(buttonPin); // Read the button state

  if (buttonState == HIGH) { // If button is pressed
    digitalWrite(ledPin, HIGH); // Turn on the LED
  } else {
    digitalWrite(ledPin, LOW); // Turn off the LED
  }
}
```

3-қадам: кодты жүктеп алыңыз: Arduino тақтасын компьютерге қосыңыз.

Кодты Arduino тақтасына құрастыру және жүктеу үшін Arduino IDE ішіндегі "Жүктеу" түймесін басыңыз.

4-қадам: нәтижелерді Бақылаңыз: Код жүктелгеннен кейін түймені басқан кезде 13 түйреуішке қосылған жарық диоды жануы керек.

Түйме босатылған кезде жарық диоды сөнуі керек. Код-



та не болады? Біз сәйкесінше жарық диоды мен түйме үшін түйреуіш нөмірлерін сақтау үшін екі тұрақты, ledPin және buttonPin анықтаймыз.

Setup () функциясында біз ledPin-ді pinMode() және buttonPin-ді кіріс сигналы ретінде шығыс сигналы ретінде орнатамыз. Loop () функциясы digitalread() көмегімен түйменің күйін үздіксіз оқиды және оны buttonState айнымалысында сақтайды. Егер buttonState мәні жоғары болса (батырма басылған), индикатор жоғары мәні бар digitalWrite () функциясын қолдана отырып қосылады. Әйтпесе, жарық диоды төмен мәнді digitalWrite() функциясы арқылы өшеді. Қосымша светодиодтарды немесе түймелерді әртүрлі түйреуіштерге қосып, оларды сәйкесінше басқару арқылы кодты өзгертуге және одан әрі тәжірибе жасауға болады. Бұл Arduino көмегімен сандық енгізу және шығару негіздерін түсінуге көмектеседі.

Келесі сабақта біз айнымалы кернеу деңгейлерін қажет ететін сенсорлар мен басқару құрылғыларымен жұмыс істеуге мүмкіндік беретін аналогтық кірістер мен шығыстарды қарастырамыз. Егер сіз осы тақырыпты талқылауды жалғастырғыңыз келсе немесе білгіңіз келетін нақты нәрсе болса, маған хабарлаңыз!

Бұл оқулықта біз Arduino көмегімен аналогтық енгізу мен шығаруды қарастырамыз. Аналогтық сигналдар үздіксіз және мәндер ауқымына ие болуы мүмкін. Arduino аналог-

тық кірістерді оқу үшін аналогтық-цифрлық түрлендіруді (ADC) және аналогтық шығыс үшін импульстің ені модуляциясын (PWM) қолданады.

3-сабақ: аналогтық кіріс және шығыс

1-қадам: схеманы орнату:Потенциометрді Arduino тақтасына қосыңыз. Потенциометрдің бір ұшын Arduino тақтасындағы 5В түйреуішке, екінші ұшын жерге тұйықтау түйреуішіне (GND), ал шыны тазалағышты (ортаңғы түйреуіш) аналогтық кіріс түйреуішіне (мысалы, A0) қосыңыз.

Жарықдиодты Arduino тақтасындағы сандық Шығыс түйреуішіне (мысалы, 9 түйреуіш) қосыңыз. Ток шектейтін резисторды (мысалы, 220-470 Ом) жарықдиодты сериямен қосыңыз.

2-қадам: кодты жазыңыз:Arduino IDE-де бос нобай аймағын басу арқылы жаңа нобай жасаңыз.

Әдепкі кодты жойып, келесі кодты енгізіңіз:

```
// Constants for LED and potentiometer pin numbers
```

```
const int ledPin = 9;
```

```
const int potPin = A0;
```

```
void setup() {
```

```
  pinMode(ledPin, OUTPUT); // Set LED pin as output
```

```
  pinMode(potPin, INPUT); // Set potentiometer pin as input
```

```
}
```

```
void loop() {
```

```
int potValue = analogRead(potPin); // Read the potentiometer
value
int brightness = map(potValue, 0, 1023, 0, 255); // Map the
value to LED brightness range (0-255)
analogWrite(ledPin, brightness); // Set the LED brightness
using PWM
}
```

3-қадам: кодты жүктеп алыңыз: Arduino тақтасын компьютерге қосыңыз. Кодты Arduino тақтасына құрастыру және жүктеу үшін Arduino IDE ішіндегі "Жүктеу" түймесін басыңыз.

4-қадам: нәтижелерді Бақылаңыз: Код жүктелгеннен кейін потенциометрдің орналасуына байланысты 9-түйреуішке қосылған жарық диодының жарықтығы өзгереді. Потенциометрдің айналуы аналогтық кіріс мәнінің өзгеруіне әкеледі, ол map () функциясы арқылы жарық диодының жарықтығына сәйкес келеді. Потенциометрді айналдырған кезде жарық диоды сәйкесінше ашық немесе күңгірт болады.

Кодта не болады? Біз сәйкесінше жарық диоды мен потенциометрдің түйреуіш нөмірлерін сақтау үшін екі тұрақты, ledPin және potPin анықтаймыз. Setup () функциясында біз ledPin-ді pinMode() және potPin-ді кіріс сигналы ретінде шығыс сигналы ретінде орнатамыз. Loop () функциясы аналогтық кіріс мәнін потенциометрден analogRead() функциясын пайдаланып оқиды және оны potValue айнымалысында сақтайды. Содан кейін біз pot мәнін (0-ден 1023-ке дей-

in) analogWrite () функциясына сәйкес келетін 0-ден 255-ке дейінгі жарықтық мәніне түрлендіру үшін map () функциясын қолданамыз. Соңында, біз PWM көмегімен ledPin-ге қосылған жарық диодының жарықтығын орнату үшін analogWrite() қолданамыз. Кодты орнатып, эксперименттерді жалғастыра отырып, сіз басқа аналогтық сенсорларды, мысалы, жарық, температура және т.б. датчиктерді қосып, олардың мәндерін аналогтық Шығыс арқылы әртүрлі құрылғыларды басқару үшін пайдалана аласыз.

Келесі сабақта біз кітапханаларды пайдалану, сериялық байланыспен жұмыс істеу немесе серво басқару сияқты күрделі тақырыптарды зерттей аламыз. Келесі жолы қай салаға тереңірек барғыңыз келетінін немесе қабылдауда нақты тақырыптарыңыз болса, маған хабарлаңыз!

Бұл оқулықта Arduino-дағы кітапханалармен жұмыс істеуге тереңірек үңілейік. Кітапханалар алдын-ала жазылған функциялар мен кодты ұсынады, оларды Arduino жобаларының функционалдығын нөлден бастап жазбай-ақ кеңейтуге болады. Arduino-да әртүрлі сенсорларды, жетектерді, байланыс хаттамаларын және т.б. қамтитын кең кітапхана экожүйесі бар.

4-сабақ: кітапханалармен жұмыс

1-қадам: кітапханаларды тауып, орнатыңыз: Компьютерде Arduino IDE ашыңыз.

Нобай мәзіріне өтіп, кітапхананы қосу – > кітапхананы басқару тармағын таңдаңыз. Кітапханаларды шолуға және оларды іздеуге мүмкіндік беретін кітапхана менеджері терезесі ашылады. Сіз қалаған сенсорға немесе функционалдылыққа байланысты нақты кітапханаларды іздей аласыз. Пайдаланғыңыз келетін кітапхананы тапқаннан кейін оны нұқыңыз, содан кейін "Орнату" түймесін басыңыз. Кітапхана Arduino IDE әзірлеу ортасына жүктеледі және орнатылады.

2-қадам: нобайыңыздағы кітапханаларды импорттаңыз және пайдаланыңыз: Кітапхананы орнатқаннан кейін оның мүмкіндіктерін пайдалану үшін оны sketch-ке импорттау керек. Arduino IDE-де "эскиз" мәзіріне өтіп, "кітапхананы қосу" тармағын таңдаңыз. Орнатылған кітапханалардың тізімі пайда болады. Пайдаланғыңыз келетін кітапхананы таңдаңыз, ол сіздің эскизіңізге қосылады.

3-қадам: кодты жазыңыз: Енді сіз кітапхананы импорттағаннан кейін оның мүмкіндіктерін кодта пайдалана аласыз. Оның мүмкіндіктерін қалай пайдалану керектігін түсіну үшін кітапханамен бірге келген құжаттаманы немесе мысалдарды қараңыз.

Міне, сізге бастауға көмектесетін жалпы үлгі:

```
// Import the library
#include <LibraryName.h>
```

```
// Create any necessary objects or variables
```

```
void setup() {  
  // Initialize the library or any required settings  
}
```

```
void loop() {  
  // Use functions from the library to perform desired actions  
}
```

4-қадам: кодты жүктеп алыңыз және тексеріңіз: Сіз қолданатын кітапхана мен функционалдылыққа байланысты қажетті аппараттық компоненттерді қосыңыз. Arduino IDE ішіндегі "Жүктеу" түймесін басу арқылы Эскизіңізді Arduino тақтасына жүктеңіз. Нәтижелерді көру үшін дәйекті шығуды қадағалаңыз немесе қосылған компоненттердің әрекетін бақылаңыз. Пайдалану ерекшеліктерін және қол жетімді мүмкіндіктерді түсіну үшін сіз пайдаланатын кітапханамен бірге келген құжаттамалар мен мысалдарды тексеруді ұмытпаңыз. Кітапханалар кодтау процесін айтарлықтай жеңілдетеді және Arduino-да жобаларыңыздың мүмкіндіктерін кеңейтеді.

Келесі сабақта біз Arduino тақтасымен компьютерден немесе басқа құрылғыдан өзара әрекеттесуге мүмкіндік беретін сериялық байланысты қарастырамыз. Егер сіз осы тақырыпты талқылауды жалғастырғыңыз келсе немесе сіз білгіңіз келетін басқа нақты салаларыңыз болса, маған хабарлаңыз!

Бұл оқулықта Arduino-мен дәйекті байланысқа тереңірек үңілейік. Сериялық байланыс Arduino тақтасы мен компьютер немесе басқа құрылғылар арасында байланыс орнатуға мүмкіндік береді. Ол әдетте жөндеу, Arduino қашықтан басқару және сыртқы сенсорлармен немесе модульдермен өзара әрекеттесу үшін қолданылады.

5-сабақ: дәйекті байланыс

1-қадам: схеманы орнату: Бұл оқулық үшін арнайы схеманы орнату қажет емес. Біз дәйекті байланыстың бағдарламалық аспектісіне назар аударамыз.

2-қадам: кодты жазыңыз: Arduino IDE-де бос нобай аймағын басу арқылы жаңа нобай жасаңыз.

Әдепкі кодты жойып, келесі кодты енгізіңіз:

```
void setup() {  
  Serial.begin(9600); // Initialize serial communication with a  
  baud rate of 9600  
}
```

```
void loop() {  
  if (Serial.available()) { // Check if data is available to read  
    char data = Serial.read(); // Read the incoming data
```

```
// Perform actions based on the received data
```

```
if (data == 'A') {
```

```
// Do something when 'A' is received
```

```
} else if (data == 'B') {
```

```
// Do something when 'B' is received
```

```
}  
}  
}
```

3-қадам: кодты жүктеп алыңыз: Arduino тақтасын компьютерге қосыңыз. Кодты Arduino тақтасына құрастыру және жүктеу үшін Arduino IDE ішіндегі "Жүктеу" түймесін басыңыз.

4-қадам: сериялық мониторды ашыңыз: Сериялық байланысты көру үшін Arduino IDE ішіндегі қызмет мәзіріне өтіңіз.

"Сериялық мониторды" таңдаңыз (немесе Ctrl+Shift+M пернелерін басыңыз).

Кодқа сәйкес келу үшін сериялық монитордағы қораптардағы деректер жылдамдығын 9600-ге орнатыңыз.

Кіріс және шығыс сериялық деректерді көрсететін сериялық монитор ашылады.

5-қадам: сериялық байланысты тексеріңіз: Сериялық мониторда таңбаны енгізіңіз (мысалы, "A") және Enter пернесін басыңыз. Arduino тақтасы таңбаны алады және кодта тиісті әрекеттерді орындайды. Әр түрлі таңбалар үшін әр түрлі әрекеттерді анықтау үшін кодты өзгертуге болады.

Кодта не болады? Setup () функциясында біз сериялық байланысты инициализациялаймыз.begin () деректер жылдамдығы 9600 бод. Сериялық мониторда қораптарда бірдей деректерді беру жылдамдығы орнатылуы керек.



Loop () функциясы serial көмегімен сериялық байланыстан оқуға болатын деректердің бар-жоғын үздіксіз тексереді.available(). Егер қол жетімді деректер болса, біз кіріс деректерін Serial көмегімен оқимыз.read () және оларды data айнымалысында сақтаңыз. Алынған мәліметтерге сүйеніп отырып, біз белгілі бір әрекеттерді орындай аламыз немесе әртүрлі мінез-құлық үлгілерін орындай аламыз. Сериялық байланысты компьютерге сенсорлық деректерді жіберу, пайдаланушы интерфейсінен командалар алу немесе тіпті басқа микроконтроллерлермен немесе құрылғылармен өзара әрекеттесу сияқты әртүрлі тәсілдермен пайдалануға болады.

Келесі сабақта біз сервоны басқару, дисплеймен жұмыс істеу немесе сымсыз интеграция сияқты басқа тақырыптарды зерттей аламыз. Келесі жолы қай салаға тереңірек барғыңыз келетінін немесе қабылдауда нақты тақырыптарыңыз болса, маған хабарлаңыз!

Бұл оқулықта Arduino көмегімен сервомоторларды қалай басқаруға болатынын қарастырайық. Сервомоторлар робототехникада және бұрыштық позицияны дәл бақылауды қажет ететін басқа жобаларда кеңінен қолданылады. Arduino серво қозғалтқыштарын оңай басқаруға мүмкіндік беретін серво кітапханасын ұсынады.

6-сабақ: сервомоторларды басқару

1-қадам: схеманы орнату: Сервомоторды Arduino тақ-

тасына қосыңыз. Серво қозғалтқышының сигнал сымын Arduino тақтасындағы сандық PWM (импульстің ені модуляциясы) түйреуішіне қосыңыз (мысалы, 9 түйреуіш). Серво қозғалтқышының қуат сымын (әдетте қызыл) Arduino тақтасындағы 5в түйреуішке қосыңыз. Серво қозғалтқышының жерге қосу сымын (әдетте қара немесе қоңыр) Arduino тақтасындағы GND түйреуішіне қосыңыз.

2-қадам: кодты жазыңыз:Arduino IDE-де бос нобай аймағын басу арқылы жаңа нобай жасаңыз. Әдепкі кодты жойып, келесі кодты енгізіңіз:

```
#include <Servo.h> // Include the Servo library
```

```
Servo servo; // Create a servo object
```

```
void setup() {  
  servo.attach(9); // Attach the servo to pin 9  
}
```

```
void loop() {  
  servo.write(90); // Set the servo position to 90 degrees  
  delay(1000); // Wait for 1 second
```

```
  servo.write(0); // Set the servo position to 0 degrees  
  delay(1000); // Wait for 1 second
```

```
  servo.write(180); // Set the servo position to 180 degrees
```

```
delay(1000); // Wait for 1 second
```

```
}
```

3-қадам: кодты жүктеп алыңыз: Arduino тақтасын компьютерге қосыңыз. Кодты Arduino тақтасына құрастыру және жүктеу үшін Arduino IDE ішіндегі "Жүктеу" түймесін басыңыз.

4-қадам: нәтижелерді Бақылаңыз: Код жүктелгеннен кейін сервомотор 0 мен 180 градус арасында алға-артқа жылжи бастайды, әр позиция арасында 1 секунд кідіріс болады.

Кодта не болады? Біз серво қозғалтқыштарын басқаруға мүмкіндік беретін Servo кітапханасын қосамыз. Біз Servo деп аталатын Servo нысанын жасаймыз. Setup () функциясында біз Servo нысанын Servo көмегімен PWM 9 түйреуішіне қосамыз.attach(). Loop () функциясында біз servo қолданамыз.servo орнын белгілі бір бұрышқа орнату үшін жазу (). Содан кейін біз delay () бағдарламасын әр Позицияның өзгеруі арасында белгілі бір уақытқа (бұл жағдайда 1 секундқа) кідірту үшін қолданамыз. Сервомоторды басқару үшін кодты сенсор көрсеткіштері немесе сериялық байланыс арқылы алынған пайдаланушы пәрмендері сияқты сыртқы кірістерге негізделген өзгертуге болады.

Келесі сабақта біз дисплейлермен жұмыс істеу (мысалы, LCD немесе OLED), сымсыз интеграция немесе үзілістерді пайдалану сияқты жетілдірілген тақырыптар сияқты басқа тақырыптарды зерттей аламыз. Келесі жолы қандай саланы зерттегіңіз келетінін немесе қабылдауда нақты тақырыпта-

рыңыз болса, маған хабарлаңыз!

Бұл оқулықта Arduino-дағы дисплейлермен жұмыс істедуді қарастырайық. Дисплейлер әдетте Arduino жобаларында визуалды кері байланыс пен пайдаланушылардың өзара әрекеттесуін қамтамасыз ету үшін қолданылады. LCD (сұйық кристалды дисплей) және OLED (органикалық жарық диоды) сияқты дисплейлердің әртүрлі түрлері бар.

7-сабақ: дисплейлермен жұмыс

1-қадам: схеманы орнату: Дисплей модулін Arduino тақтасына оның нақты қосылу талаптарына сәйкес қосыңыз. Сымдардың дұрыс қосылғанына көз жеткізу үшін дисплей модулінің техникалық сипаттамасын немесе құжаттамасын қараңыз. Егер сіз сұйық кристалды дисплейді қолдансаңыз, контрастты реттеу үшін потенциометрді пайдалану қажет болуы мүмкін.

2-қадам: кітапхананы орнатыңыз: Кодты жазбас бұрын дисплей модулі үшін белгілі бір кітапхананы орнату қажет болуы мүмкін. Arduino IDE бағдарламасын ашып, Sketch мәзіріне өтіп, кітапхананы қосу –> кітапхананы басқару тармағын таңдаңыз. Дисплей модуліне сәйкес келетін кітапхананы тауып, оны орнатыңыз. Кітапхана құжаттамасындағы кез келген қосымша нұсқауларды орындаңыз.

3-қадам: кодты жазыңыз: Arduino IDE-де бос нобай аймағын басу арқылы жаңа нобай жасаңыз. Әдепкі кодты жойып, бастапқы нүкте ретінде келесі кодты енгізіңіз:

```
// Include the necessary libraries
```

```
#include <Library1.h>
```

```
#include <Library2.h>
```

```
// Create any necessary objects or variables
```

```
void setup() {
```

```
// Initialize the display or any required settings
```

```
}
```

```
void loop() {
```

```
// Update the display content or perform desired actions
```

```
}
```

4-қадам: дисплей кодын жазыңыз: Сіз қолданатын дисплей модулі мен кітапханаға байланысты нақты мүмкіндіктер мен пайдалану туралы ақпарат алу үшін кітапхана құжаттамасына жүгінуіңіз керек. Белгілі бір дисплей модулінде мәтінді, сандарды немесе графиканы көрсетуді бастау үшін кітапхана ұсынған мысалдарды немесе оқулықтарды қараңыз.

5-қадам: кодты жүктеп алыңыз және тексеріңіз: Қажетті аппараттық компоненттерді дұрыс жалғанғанына көз жеткізіңіз. Arduino IDE ішіндегі "Жүктеу" түймесін басу арқылы Эскизіңізді Arduino тақтасына жүктеңіз. Қажетті мазмұн Күткендей көрінетінін көру үшін Дисплейді бақылаңыз. Дисплей кітапханалары дисплейді басқаруға және мәтінді

көрсету, айналдыру, курсордың орнын орнату және графиканы көрсету сияқты әрекеттерді орындауға мүмкіндік береді. Әр кітапхананың өзіндік синтаксисі мен функциялары болуы мүмкін, сондықтан белгілі бір кітапхананың құжаттамасына жүгіну өте маңызды.

Келесі сабақта біз Arduino-мен сымсыз байланыс туралы біле аламыз, мысалы, сымсыз байланыс үшін Bluetooth немесе Wi-Fi модульдерін пайдалану. Егер сіз осы тақырыпты талқылауды жалғастырғыңыз келсе немесе сіз білгіңіз келетін басқа нақты салаларыңыз болса, маған хабарлаңыз!

Бұл оқулықта Bluetooth көмегімен Arduino-мен сымсыз байланысты қарастырайық. Bluetooth модульдері Arduino тақтасы мен Смартфондар, планшеттер немесе компьютерлер сияқты басқа құрылғылар арасында сымсыз байланыс орнатуға мүмкіндік береді. Bluetooth байланысын қашықтан басқару, деректерді тіркеу, сенсорларды біріктіру және т.б. үшін пайдалануға болады.

8-сабақ: Bluetooth арқылы сымсыз байланыс

1-қадам: схеманы орнату: Bluetooth модулін Arduino тақтасына қосыңыз. Нақты электр қосылымдары сіз қолданатын Bluetooth модуліне байланысты. Әдетте, Bluetooth Модулінің TX түйреуішін (беріліс) Arduino тақтасының RX түйреуішіне (қабылдауына) қосу керек және керісінше. Bluetooth Модулінің VCC түйреуішін Arduino тақтасындағы 5в түйре-

уішке, ал GND түйреуішін GND түйреуішке қосыңыз.

2-қадам: кітапхананы орнатыңыз: Кодты жазбас бұрын Arduino үшін Bluetooth кітапханасын орнату керек. "SoftwareSerial" сияқты әртүрлі Bluetooth кітапханалары немесе "Adafruit\_BluefruitLE\_nRF51" немесе "hc05" сияқты Bluetooth модульдеріне арналған арнайы кітапханалар бар. Сіз қолданатын Bluetooth модулі негізінде тиісті кітапхананы орнатыңыз. Орнату нұсқаулары кітапхана құжаттамасында берілген.

3-қадам: кодты жазыңыз: Arduino IDE-де бос нобай аймағын басу арқылы жаңа нобай жасаңыз. Әдепкі кодты жойып, бастапқы нүкте ретінде келесі кодты енгізіңіз:

```
// Include the necessary libraries
```

```
#include <Library1.h>
```

```
#include <Library2.h>
```

```
// Create any necessary objects or variables
```

```
void setup() {
```

```
// Initialize the Bluetooth module or any required settings
```

```
}
```

```
void loop() {
```

```
// Check for incoming Bluetooth data and perform desired actions
```

```
}
```

4-қадам: Bluetooth кодын жазыңыз: Сіз қолданатын Bluetooth модуліне және кітапханаға байланысты нақты мүмкіндіктер мен пайдалану туралы ақпарат алу үшін кітапхана құжаттамасына жүгінуіңіз керек. Bluetooth байланысын бастау үшін кітапхана ұсынған мысалдарды немесе оқулықтарды қараңыз. Әдетте, сіз Bluetooth модулін `setup()` функциясында инициализациялауыңыз керек, содан кейін `loop()` функциясындағы кіріс деректерін үздіксіз тексеріп отыруыңыз керек.

5-қадам: кодты жүктеп алыңыз және тексеріңіз: Қажетті аппараттық компоненттерді дұрыс жалғанғанына көз жеткізіңіз. Arduino IDE ішіндегі "Жүктеу" түймесін басу арқылы Эскизіңізді Arduino тақтасына жүктеңіз. Bluetooth модулін жұптастыру және қосу үшін Bluetooth қосылған құрылғыны (мысалы, смартфон немесе компьютер) пайдаланыңыз. Bluetooth байланысын орнатыңыз және Arduino тақтасы мен құрылғы арасында байланыс орнатыңыз. Жұптастыру және деректер пішімі сияқты Bluetooth байланысының ерекшеліктері сіз қолданатын Bluetooth модулі мен кітапханаға байланысты өзгеруі мүмкін. Нұсқаулық үшін кітапхана ұсынған құжаттар мен мысалдарды тексеріңіз.

Келесі сабақта біз сенсорларды біріктіру немесе деректерді протоколдаумен жұмыс істеу сияқты басқа тақырыптарды зерттей аламыз. Келесі жолы қандай саланы оқығыңыз келетінін немесе басқа нақты тақырыптар бар болса, маған хабарлаңыз!



Бұл оқулықта сенсорлардың Arduino-мен интеграциясын қарастырайық. Датчиктер көптеген Arduino жобаларында маңызды компоненттер болып табылады, өйткені олар тақтаға физикалық әлеммен өзара әрекеттесуге және қоршаған ортадан деректерді жинауға мүмкіндік береді. Температура, ылғалдылық, қозғалыс, Жарық, қашықтық және т.б. сияқты әртүрлі мақсаттарда қол жетімді сенсорлардың кең ауқымы бар.

## 9-сабақ: сенсорлық Интеграция

1-қадам: схеманы орнату: Біріктіргіңіз келетін нақты сенсорды анықтаңыз және қосылу нұсқаулары үшін оның техникалық сипаттамасына немесе құжаттамасына жүгініңіз. Ұсынылған қосылымдарға сүйене отырып, сенсорды Arduino тақтасына қосыңыз. Әдетте сенсорларда үш түйреуіш бар: VCC (қуат көзі), GND (жерге қосу) және оқылатын деректерді беретін Шығыс түйреуіш. Сенсордың VCC және GND түйреуіштері Arduino тақтасындағы тиісті қуат және жерге қосу түйреуіштеріне қосылғанына көз жеткізіңіз.

2-қадам: кітапхананы орнатыңыз (қажет болса): Сіз қолданатын сенсорға байланысты онымен әрекеттесу үшін белгілі бір кітапхананы орнату қажет болуы мүмкін. Arduino IDE бағдарламасын ашып, Sketch мәзіріне өтіп, кітапхананы қосу – > кітапхананы басқару тармағын таңдаңыз. Сенсорға сәйкес келетін кітапхананы тауып, оны орнатыңыз. Кітапхана құжаттамасындағы кез келген қосымша нұсқауларды

орындаңыз.

3-қадам: кодты жазыңыз: Arduino IDE-де бос нобай аймағын басу арқылы жаңа нобай жасаңыз. Әдепкі кодты жойып, бастапқы нүкте ретінде келесі кодты енгізіңіз:

```
// Include the necessary libraries
```

```
#include <Library1.h>
```

```
#include <Library2.h>
```

```
// Create any necessary objects or variables
```

```
void setup() {
```

```
// Initialize the sensor or any required settings
```

```
}
```

```
void loop() {
```

```
// Read sensor data and perform desired actions
```

```
}
```

4-қадам: сенсор кодын жазыңыз: Сіз қолданатын сенсор мен кітапханаға байланысты нақты мүмкіндіктер мен пайдалану туралы ақпарат алу үшін кітапхана құжаттамасына жүгінуіңіз керек. Сенсор деректерін оқуды бастау және оларды жобаңызда пайдалану үшін кітапхана ұсынған мысалдарды немесе оқулықтарды қараңыз. Көптеген сенсорлық кітапханалар сенсорды инициализациялау, сенсорлық деректерді оқу және нәтижелерді түсіндіру функцияларын ұсынады.

5-қадам: кодты жүктеп алыңыз және тексеріңіз: Қажет-

ті аппараттық компоненттерді дұрыс жалғанғанына көз жеткізіңіз. Arduino IDE ішіндегі "Жүктеу" түймесін басу арқылы Эскизіңізді Arduino тақтасына жүктеңіз. Сериялық шығуды басқарыңыз, дисплей модульдерін қолданыңыз немесе сенсор деректеріне негізделген кез келген қажетті әрекеттерді орындаңыз. Сенсорды біріктіру ерекшеліктері сенсор түріне және қолданылатын кітапханаға байланысты. Сенсор деректерін инициализациялау, оқу және түсіндіру бойынша ұсыныстар алу үшін кітапхана ұсынған құжаттар мен мысалдарды тексеріңіз.

Келесі сабақта біз SD картасы немесе EEPROM сияқты сыртқы сақтау құрылғысына сенсор көрсеткіштерін немесе басқа деректерді жазуға болатын Arduino көмегімен деректерді журналға жазуды қарастырамыз. Егер сіз осы тақырыпты талқылауды жалғастырғыңыз келсе немесе сіз білгіңіз келетін басқа нақты салаларыңыз болса, маған хабарлаңыз!

Бұл оқулықта Arduino көмегімен деректер журналын жүргізуді қарастырайық. Деректер журналын жүргізу уақыт өте келе сенсорлардың, өлшемдердің немесе басқа деректердің көрсеткіштерін жазуға мүмкіндік береді, бұл кейіннен жиналған деректерді талдауға және визуализациялауға мүмкіндік береді. Arduino деректерді SD картасы немесе EEPROM сияқты әртүрлі сыртқы сақтау құрылғыларына жаза алады (тек тұрақты пайдалану үшін электрмен өшірілетін бағдарламаланатын жад).

10-сабақ: деректерді тіркеу

1-қадам: схеманы орнату: Сіз қолданатын сақтау құрылғысының түріне байланысты (мысалы, SD картасы немесе EEPROM) нақты қосылу нұсқауларын орындаңыз. SD карта модулі әдетте Модульдің VCC түйреуішін Arduino тақтасындағы 5в түйреуішке, GND-ді GND-ге, ал Модульдің MISO, MOSI, SCK және CS түйреуіштерін Arduino тақтасындағы тиісті SPI түйреуіштеріне қосуды талап етеді. EEPROM үшін Қосылым сіз қолданатын нақты модульге байланысты өзгеруі мүмкін. Дұрыс қосылыстар туралы ақпарат алу үшін модульдің құжаттамасын қараңыз.

2-қадам: кітапхананы орнатыңыз (қажет болса): Сіз қолданатын сақтау құрылғысына байланысты сізге белгілі бір кітапхананы орнату қажет болуы мүмкін. SD картасында журнал жүргізу үшін Arduino IDE-мен бірге келетін "SD" кітапханасын пайдалануға болады. EEPROM журналын жүргізу үшін сізге EEPROM модуліне тән кітапхананы орнату қажет болуы мүмкін.

3-қадам: кодты жазыңыз: Arduino IDE-де бос нобай аймағын басу арқылы жаңа нобай жасаңыз.

Әдепкі кодты жойып, бастапқы нүкте ретінде келесі кодты енгізіңіз:

```
// Include the necessary libraries
```

```
#include <SD.h> // For SD card logging
```

```
#include <EEPROM.h> // For EEPROM logging
```

```
// Define constants and variables
```

```
void setup() {
```

```
// Initialize the data storage device or any required settings  
}
```

```
void loop() {
```

```
// Read sensor data or other values
```

```
// Log the data to the storage device
```

```
// Delay for a specific interval between readings
```

```
}
```

4-қадам: деректерді тіркеу кодын жазыңыз: Сіз қолданатын сақтау құрылғысына және кітапханаға байланысты нақты функциялар мен пайдалану туралы ақпарат алу үшін кітапхана құжаттамасына жүгінуіңіз керек. Деректерді белгілі бір сақтау құрылғысына жазуды бастау үшін кітапхана ұсынған мысалдарды немесе оқулықтарды қараңыз. Әдетте, сақтау құрылғысын `setup()` функциясында инициализациялау керек, содан кейін деректерді `loop()` функциясына үздіксіз тіркеу керек.

5-қадам: кодты жүктеп алыңыз және тексеріңіз: Қажетті аппараттық компоненттерді дұрыс жалғанғанына көз жеткізіңіз. Arduino IDE ішіндегі "Жүктеу" түймесін басу арқылы Эскизіңізді Arduino тақтасына жүктеңіз. Деректердің дұрыс тіркелгеніне көз жеткізу үшін сақтау құрылғысын (мысалы, SD картасы немесе EEPROM) бақылаңыз. Де-

рекерді тіркеу ерекшеліктері пайдаланылатын сақтау құрылғысының түріне және пайдаланылатын кітапханаға байланысты. Сақтау құрылғысынан деректерді инициализациялау, жазу және оқу бойынша ұсыныстар алу үшін кітапхана ұсынған құжаттар мен мысалдарды тексеріңіз.

Келесі сабақта біз қозғалтқышты басқару, жетілдірілген сенсорлармен жұмыс істеу немесе пайдаланушы интерфейсін құру сияқты басқа тақырыптарды зерттей аламыз. Келесі жолы қандай саланы оқығыңыз келетінін немесе басқа нақты тақырыптар бар болса, маған хабарлаңыз!

Контексте айтылған терминдердің қысқаша анықтамалары:

**SD картасы:** SD картасы (Secure Digital) – деректерді сақтау үшін қолданылатын жад картасының түрі. Ол ақпаратты сақтау және алу үшін Arduino-мен жұптастырылуы мүмкін, бұл оны деректерді тіркеу қолданбалары үшін пайдалы етеді.

**EEPROM:** EEPROM (тек тұрақты пайдалану үшін электрмен өшірілетін бағдарламаланатын жад) – қуат өшірілген кезде де деректерін сақтайтын тұрақты емес жад түрі. Arduino тақталарында көбінесе ендірілген EEPROM бар, ал қосымша сақтау үшін сыртқы EEPROM модульдерін қосуға болады.

**SPI:** SPI (сериялық перифериялық интерфейс) – микроконтроллерлерді сенсорлар, дисплейлер немесе жад модуль-

дері сияқты перифериялық құрылғыларға қосу үшін қолданылатын байланыс протоколы. Бұл деректерді беру үшін белгілі бір контактілерді, сағат сигналдарын және басқару сигналдарын қолдана отырып, сериялық интерфейс арқылы деректерді беруді қамтиды.

Кітапхана: Arduino бағдарламалау контекстіндегі Кітапхана-күрделі тапсырмаларды жеңілдету үшін мүмкіндіктерді қамтамасыз ететін алдын ала жазылған код жиынтығы. Кітапханаларды сенсорлар, дисплейлер және байланыс модульдері сияқты әртүрлі компоненттерді нөлден бастап толық кодты жазудың қажеті жоқ басқару үшін пайдалануға болады.

Деректерді тіркеу: деректерді тіркеу-бұл уақыт өте келе деректерді жазу процесі. Arduino жобаларында деректерді тіркеу Сенсорлардан немесе басқа көздерден ақпаратты SD картасы немесе EEPROM сияқты сыртқы сақтау құрылғысына кейінірек талдау үшін сақтауды қамтиды.

PWM (импульстің ені модуляциясы): PWM-сандық сигналдарды қолдана отырып, аналогтық кернеу деңгейлерін модельдеу үшін қолданылатын әдіс. Бұл қажетті аналогтық мәнге сәйкес келетін орташа кернеуге жету үшін әртүрлі жұмыс циклдары бар цифрлық сигналды жылдам қосу және өшіруді қамтиды. PWM көбінесе жарық диодтарын, қозғалтқыштарды және басқа құрылғыларды басқару үшін қолданылады.

Потенциометр: әдетте pot деп аталатын Потенциометр

үш полюсті айнымалы резистор болып табылады. Ол реттелетін қарсылықты қамтамасыз ету үшін электрондық схемаларда қолданылады. Arduino жобаларында потенциометрлер көбінесе жарықтық немесе позиция сияқты параметрлерді басқару үшін аналогтық енгізу құрылғылары ретінде қолданылады.

Жұмыс циклі: жұмыс циклі сандық сигналдың сигнал пішінінің бір кезеңінде жоғары (қосулы) болып қалатын уақыт пайызын білдіреді. PWM контекстінде жұмыс циклі орташа шығыс кернеуін, демек, басқарылатын құрылғылардың қарқындылығын немесе орналасуын анықтайды.

Bluetooth: Bluetooth-бұл құрылғылар арасында қысқа қашықтыққа деректерді бөлісу үшін қолданылатын сымсыз байланыс технологиясы. Arduino жобаларында Bluetooth модульдері Arduino тақтасы мен смартфондар немесе компьютерлер сияқты басқа құрылғылар арасында сымсыз байланыс орнатады.

SPI (сериялық перифериялық интерфейс): бұрын айтылғандай, SPI – микроконтроллерлер мен перифериялық құрылғылар арасында деректерді тасымалдау үшін қолданылатын синхронды сериялық байланыс протоколы. Бұл сенсорларды, дисплейлерді және жад модульдерін Arduino тақталарына қосудың кең таралған тәсілі.