

# ARDUINO TARAQQIYOTI

Ma'ruza #11

## ARDUINONING LCD DISPLEYLARI BILAN INTERFEYSI

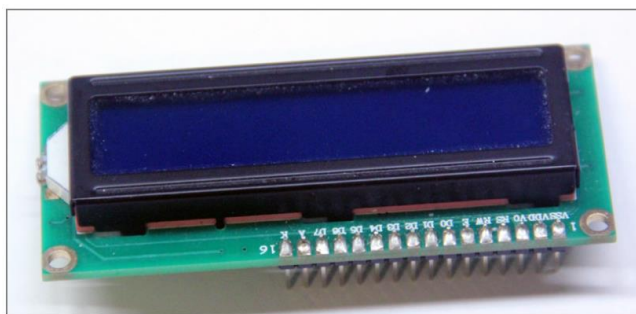
### Reja:

1. LCD displeyni sozlash
2. LCD displeyga yozish uchun LiquidCrystal kutubxonasidan foydalanish
3. Displeyga matn qo'shish
4. Maxsus belgilar va animatsiyalarni yaratish
5. I2C LCD displeylar

Foydalanilgan adabiyotlar

## LCD displeyni sozlash

Ushbu darsdagi misollarni bajarish uchun siz parallel LCD displeydan foydalanasiz. Ular juda keng tarqalgan va har xil shakl va o'lchamlarda bo'ladi. Eng keng tarqalgani 16 pinli bitta qatorli 16x2 belgili displey (agar u orqa yorug'lik bo'lmasa 14). Ushbu darsda siz jami 32 ta belgini (16 ta ustun va 2 qator) ko'rsatishi mumkin bo'lgan 16 pinli LCD displeydan foydalanasiz. Agar sizning displeyingiz allaqachon payvandlangan 16 pinli sarlavha bilan ta'minlanmagan bo'lsa, uni panelga osongina o'rnatishingiz uchun uni payvandlashingiz kerak. Header (terminal) muvaffaqiyatli payvandlangan bo'lsa, LCD displeyingiz 1-rasmda ko'rsatilgandek ko'rinishi kerak va siz uni bredbordga kiritishingiz mumkin.



1-rasm.

Keyinchalik, siz LCD displeyni panelga va Arduino-ga ulaysiz. Ushbu parallel LCD modullarning barchasi bir xil pin-outga ega va ikkita rejimdan birida simli ulanishi mumkin: 4-pinli yoki 8-pinli rejim. Aloqa uchun atigi 4 ta pin yordamida qilmoqchi bo'lgan hamma narsani amalga oshirishingiz mumkin; siz uni shunday bog'laysiz. Displeyni yoqish, displeyni buyruq rejimiga yoki belgilar rejimiga o'rnatish va uni o'qish/yoqish rejimiga o'rnatish uchun pinlar ham mavjud. 1-jadvalda ushbu pinlarning barchasi tasvirlangan.

Pin raqami	Pin nomi	Pin maqsadi
1	VSS	Manfiy qutb
2	VDD	+5V (Musbat qutb)
3	V0	Yorqinlik sozlagich (potensiometrغا ulanadi)
4	RS	Registr tanlash
5	RW	O'qish / Yoqish
6	EN	Ruxsat etish
7	D0	Ma'lumot liniyasi 0 (foydalanilmaydi)
8	D1	Ma'lumot liniyasi 1 (foydalanilmaydi)
9	D2	Ma'lumot liniyasi 2 (foydalanilmaydi)
10	D3	Ma'lumot liniyasi 3 (foydalanilmaydi)

11	D4	Ma'lumot liniyasi 4
12	D5	Ma'lumot liniyasi 5
13	D6	Ma'lumot liniyasi 6
14	D7	Ma'lumot liniyasi 7
15	A	Displey yoritgichining anod tarafi
16	K	Displey yoritgichining katod tarafi

1-jadval.

PIN ulanishlarining taqsimoti quyidagicha:

■Kontrastni sozlash pinasi displey qanchalik qorong'iligini o'zgartiradi. U potensiometrning markaziy piniga ulanadi.

■Registrni tanlash pinli LCD displeyni buyruq yoki belgilar rejimiga o'rnatadi, shuning uchun u ma'lumotlar liniyalari orqali uzatiladigan keyingi ma'lumotlar to'plamini qanday izohlashni biladi. Ushbu pin holatiga qarab, LCD displeyga yuborilgan ma'lumotlar buyruq (masalan, kursorni siljitish) yoki belgilar (masalan, a harfi) sifatida talqin qilinadi .

■Bu amalga oshirishda RW pin har doim yerga bog'langan, ya'ni siz faqat displeyga yozasiz va undan hech qachon o'qimaysiz.

■EN pin ma'lumotlar tayyor bo'lganda LCD displeyga xabar berish uchun ishlatiladi.

■Ma'lumotlar pinlari 4-7 ma'lumotlarni haqiqatda uzatish uchun ishlatiladi va 0-3 ma'lumotlar pinlari ulanmagan holda qoldiriladi.

■Agar siz orqa yorug'lik uchun o'rnatilgan rezistorli LCD displeydan foydalansangiz, anod pinini 5V ga va katod pinini yerga ulab, yorug'likni yoritishingiz mumkin. Agar yo'q bo'lsa, siz anod yoki katod piniga oqim cheklovchi rezistorni qo'yishingiz kerak. Qurilmangiz uchun ma'lumotlar jadvali odatda buni qilish kerakmi yoki yo'qligini aytadi.

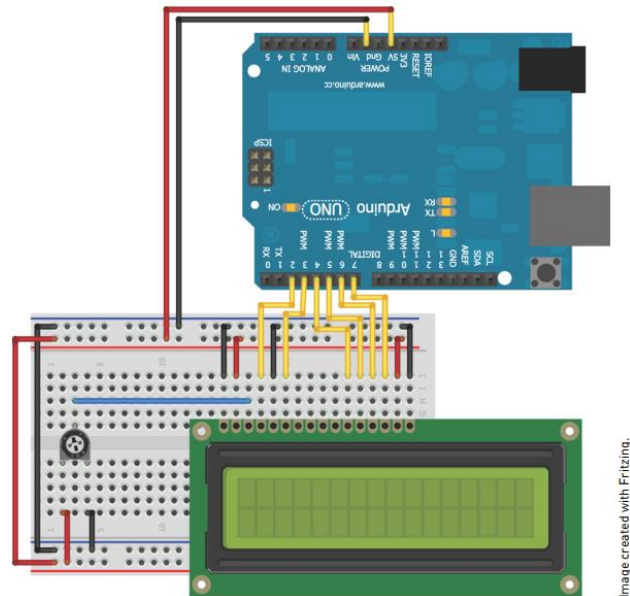
LCD displeyning aloqa pinlarini Arduino-dagi istalgan kiritish/chiqarish pinlariga ulashingiz mumkin. Ushbu darsda ular 2-jadvalda ko'rsatilgandek bog'langan.

<b>Displey pini</b>	<b>Arduino pini</b>
RS	Pin 2
EN	Pin 3
D4	Pin 4
D5	Pin 5
D6	Pin 6
D7	Pin 7

2-jadval.

2-rasmda ko'rsatilgan ulanish sxemasiga qarang va shunga mos ravishda LCD-ni ulang.

Endi LCD ekraningiz harakatga tayyor! Keyingi bo'limda yuklangan kodni olganingizdan so'ng, matnni ekranda ko'rsatishni boshlashingiz mumkin. Potansiyometr matn va ekranning fon rangi o'rtasidagi kontrastni sozlaydi.



2-rasm.

### **LCD displeyga yozish uchun LiquidCrystal kutubxonasidan foydalanish**

Arduino IDE LiquidCrystal kutubxonasini o'z ichiga oladi, bu siz foydalanayotgan parallel LCD bilan interfeysni juda osonlashtiradigan funktsiyalar to'plami. LiquidCrystal kutubxonasida kursorni miltillash, matnni avtomatik aylantirish, maxsus belgilar yaratish va matnni chop etish yo'nalishini o'zgartirish kabi ta'sirchan hajmdagi funktsiyalar mavjud. Ushbu dars har bir funktsiyani qamrab olmaydi, balki buning o'rniga sizga eng muhim funktsiyalardan foydalangan holda displey bilan interfeysni tushunishingiz kerak bo'lgan vositalarni beradi. Kutubxona funktsiyalarining tavsifi va ulardan foydalanishni ko'rsatadigan misollarni Arduino veb-saytida topishingiz mumkin: <http://arduino.cc/en/Reference/LiquidCrystal>

### **Displeyga matn qo'shish**

Ushbu birinchi misolda siz displeyga bir nechta matn va ortib boruvchi raqam qo'shasiz. Ushbu mashq displeyni ishga tushirish, matn yozish va kursorni qanday harakatlantirishni ko'rsatadi. Birinchidan, LiquidCrystal kutubxonasini qo'shing:

```
#include <LiquidCrystal.h>
```

Keyin LCD ob'yektini quyidagicha ishga tushiring:

```
LiquidCrystal lcd (2,3,4,5,6,7);
```

LCD-ni ishga tushirish uchun argumentlar RS, EN, D4, D5, D6 va D7 ga shu tartibda ulangan Arduino pinlarini ifodalaydi. O'rnatishda siz LCD displeyni belgilar o'lchami bilan sozlash uchun kutubxonaning begin() funksiyasini chaqirasiz. (Men foydalanayotgan displey 16x2 displey, lekin siz 20x4 kabi boshqa o'lchamdan foydalanayotgan bo'lishingiz mumkin.) Bu buyruq uchun argumentlar mos ravishda ustunlar va qatorlar sonini ifodalaydi:

```
lcd.begin(16, 2);
```

Buni qilgandan so'ng, displeyning ma'lum bir joyiga matnni chop etish uchun kutubxonaning print() va setCursor() buyruqlarini chaqirishingiz mumkin. Misol uchun, agar siz mening ismimni ikkinchi qatorga chop qilmoqchi bo'lsangiz, quyidagi buyruqlarni berasiz:

```
lcd.setCursor(0,1);  
lcd.print("Salom dunyo");
```

Ekrandagi pozitsiyalar yuqori chap holatda (0,0) dan boshlab indekslanadi. setCursor() ning birinchi argumenti qaysi ustun raqamini, ikkinchisi esa qaysi qator raqamini belgilaydi. Odatiy bo'lib, boshlang'ich joylashuvi (0,0). Shunday qilib, agar siz kursor o'rnini o'zgartirmasdan print() ni chaqirsangiz, matn yuqori chap burchakdan boshlanadi.

**OGOHLANTIRISH** Kutubxona juda uzun satrlarni tekshirmaydi. Shunday qilib, agar siz murojaat qilayotgan qatordagi belgilar sonidan uzunroq bo'lgan 0 pozitsiyasidan boshlangan satrni chop etishga harakat qilsangiz, g'alati xatti-harakatni ko'rishingiz mumkin. Siz chop etayotgan narsa displeyga mos kelishini tekshiring!

Ushbu bilimlardan foydalanib, endi siz birinchi qatorda ba'zi matnlarni ko'rsatadigan va ikkinchi qatorda har soniyada bir marta ko'payadigan hisoblagichni chop etadigan oddiy dastur yozishingiz mumkin. Quyidagi kod buni amalga oshirish uchun to'liq dasturni ko'rsatadi. Uni Arduino-ga yuklang va u kutilganidek ishlashini tasdiqlang. Hech narsa ko'rmasangiz, kontrastni potentsiometr bilan sozlang.

```
#include <LiquidCrystal.h>  
  
int time = 0;  
  
LiquidCrystal lcd(2, 3, 4, 5, 6, 7);  
void setup()  
{
```

```

lcd.begin(16, 2);

lcd.print("Salom dunyo");
}
void loop()
{

lcd.setCursor(0,1);

lcd.print(time);

delay(1000);

time++;
}

```

Ushbu dastur siz ilgari o'rgangan barcha qadamlarni birlashtiradi. Kutubxona birinchi navbatda dasturning yuqori qismida joylashgan. Vaqt o'zgaruvchisi 0 ga ishga tushiriladi, shuning uchun uni loop() davomida soniyada bir marta oshirish mumkin. lcd deb nomlangan LiquidCrystal ob'ekti siz allaqachon ulangan sxema asosida tayinlangan tegishli pinlar bilan yaratilgan. O'rnatishda LCD displey lcd.begin(16,2) ga qo'ng'iroq qilish orqali 16 ta ustun va 2 qatorga ega bo'lib sozlangan. Birinchi qator hech qachon o'zgarmasligi sababli, uni sozlashda yozish mumkin. Bu lcd.print() ga qo'ng'iroq qilish orqali amalga oshiriladi. E'tibor bering, avval kursor o'rnini o'rnatish shart emas, chunki siz matnni allaqachon standart boshlang'ich joy bo'lgan (0,0) holatiga chop etishni xohlaysiz. Siqilishda kursor har doim (0,1) holatiga qaytariladi, shunda har soniyada chop etadigan raqam oldingi raqamning ustiga yoziladi. Displey sekundiga bir marta oshirilgan vaqt qiymati bilan yangilanadi.

### **Maxsus belgilar va animatsiyalarni yaratish**

Oddiy matn yordamida ifodalab bo'lmaydigan ma'lumotlarni ko'rsatishni xohlasangiz nima bo'ladi? Balki siz yunoncha harf, daraja belgisi yoki ba'zi progress barlarini qo'shmoqchisiz. Yaxshiyamki, LiquidCrystal kutubxonasi displeyga yozilishi mumkin bo'lgan maxsus belgilar ta'rifini qo'llab-quvvatlaydi. Keyingi misolda siz ushbu imkoniyatdan displey bo'ylab aylanib yuradigan jonlantirilgan harakat panelini yaratish uchun foydalanasiz. Shundan so'ng, siz haroratni o'lchash va ko'rsatishda daraja belgisini qo'shish uchun maxsus belgilardan foydalanasiz.

Maxsus belgi yaratish juda oddiy. Agar siz LCD displeyni diqqat bilan ko'rib chiqsangiz, har bir belgi bloki aslida 5x8 piksellar panjarasidan iborat ekanligini ko'rasiz. Maxsus belgi yaratish uchun siz ushbu piksellarning har birining qiymatini belgilashingiz va ushbu ma'lumotni displeyga yuborishingiz kerak. Buni sinab ko'rish uchun siz displeyning ikkinchi qatorini jonlantirilgan jarayon paneli bilan to'ldiradigan bir qator belgilar

yaratasiz. Har bir belgi maydoni 5 piksel kengligida bo'lgani uchun jami beshta maxsus belgi bo'ladi: biri bitta ustun bilan to'ldirilgan, biri ikkita ustun bilan to'ldirilgan va hokazo.

Eskizning yuqori qismida maxsus belgilardan foydalanmoqchi bo'lgan joyda, yoqilgan piksellarni ifodalovchi 1 va o'chiriladigan piksellarni ifodalovchi 0 bilan bayt massivi yarating. Birinchi ustunni (yoki belgining birinchi 20% ni) to'ldiruvchi belgini ifodalovchi bayt massivi quyidagicha ko'rinadi:

```
bayt p20[8] = {  
  B10000,  
  B10000,  
  B10000,  
  B10000,  
  B10000,  
  B10000,  
  B10000,  
  B10000,  
};
```

Men bu bayt massivini p20 deb atashni tanladim, chunki u bitta belgi blokining 20 foizini to'ldiradi (p foizni bildiradi).

Setup() funksiyasida bayt massivini maxsus belgilar identifikatoriga belgilash uchun createChar() funksiyasini chaqiring. Maxsus belgilar identifikatorlari 0 dan boshlanadi va 7 gacha boradi, shuning uchun jami sakkizta maxsus belgilarga ega bo'lishingiz mumkin. 20% belgilar bayt massivini maxsus belgi 0 bilan solishtirish uchun setup() funksiyangizda quyidagilarni yozing:

```
lcd.createChar(0, p20);
```

Displayga maxsus belgi yozishga tayyor bo'lgach, kursorni to'g'ri joyga qo'ying va kutubxonaning write() funksiyasidan ID raqami bilan foydalaning:

```
lcd.write((bayt)0);
```

Oldingi satrda (bayt) 0 ni bayt qiymatiga aylantiradi yoki o'zgartiradi. Bu faqat to'g'ridan-to'g'ri 0 belgisini yozishda (0 ga belgilangan o'zgaruvchisiz) Arduino kompilyatorining o'zgaruvchi turi noaniq bo'lishidan kelib chiqadigan xatolikni oldini olish uchun kerak bo'ladi. "Bayt cast" ni olib tashlashga harakat qiling va Arduino IDE ko'rsatadigan xatoni ko'ring. Unsiz boshqa belgilar identifikatorlarini yozishingiz mumkin, masalan:

```
lcd.write(1);
```

Bularning barchasini jamlagan holda, siz qolgan belgilarni qo'shishingiz va progress satrini yangilashni boshqarish uchun dastur tsikliga ikkita ichki o'rnatilgan for() tsiklini qo'yishingiz mumkin. Tugallangan kod quyida ko'rsatilgan kodga o'xshaydi.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(2, 3, 4, 5, 6, 7);
```

```
byte p20[8] = {
  B10000,
  B10000,
  B10000,
  B10000,
  B10000,
  B10000,
  B10000,
  B10000,
};
```

```
byte p40[8] = {
  B11000,
  B11000,
  B11000,
  B11000,
  B11000,
  B11000,
  B11000,
  B11000,
};
```

```
byte p60[8] = {
  B11100,
  B11100,
  B11100,
  B11100,
  B11100,
  B11100,
  B11100,
  B11100,
};
```

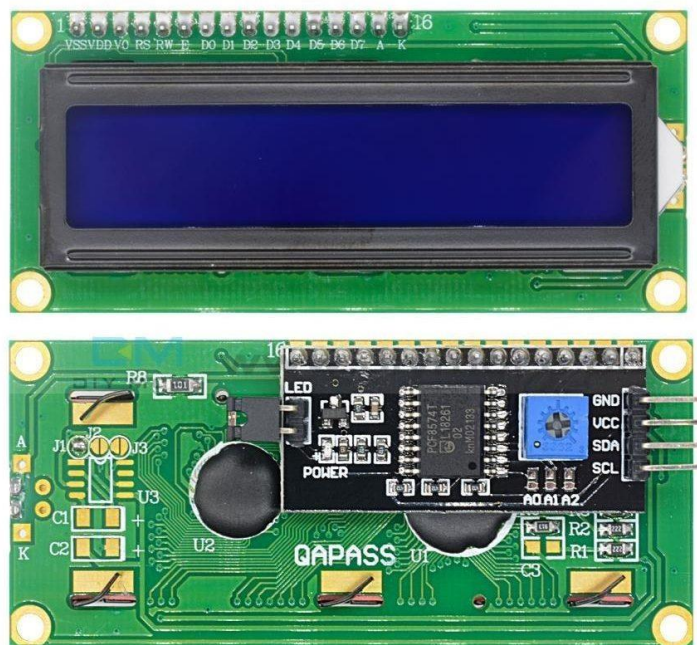
```
byte p80[8] = {
  B11110,
  B11110,
  B11110,
  B11110,
};
```

```
B11110,  
B11110,  
B11110,  
B11110,  
};  
byte p100[8] = {  
B11111,  
B11111,  
B11111,  
B11111,  
B11111,  
B11111,  
B11111,  
B11111,  
};  
void setup()  
{  
  
  lcd.begin(16, 2);  
  
  lcd.print("Jeremy's Display");  
  
  lcd.createChar(0, p20);  
  lcd.createChar(1, p40);  
  lcd.createChar(2, p60);  
  lcd.createChar(3, p80);  
  lcd.createChar(4, p100);  
}  
void loop()  
{  
  
  lcd.setCursor(0,1);  
  
  lcd.print(" ");  
  
  for (int i = 0; i<16; i++)  
  {  
  
    for (int j=0; j<5; j++)  
    {  
      lcd.setCursor(i, 1);  
      lcd.write(j);  
      delay(100);  
    }  
  }  
}
```

```
}  
}  
}
```

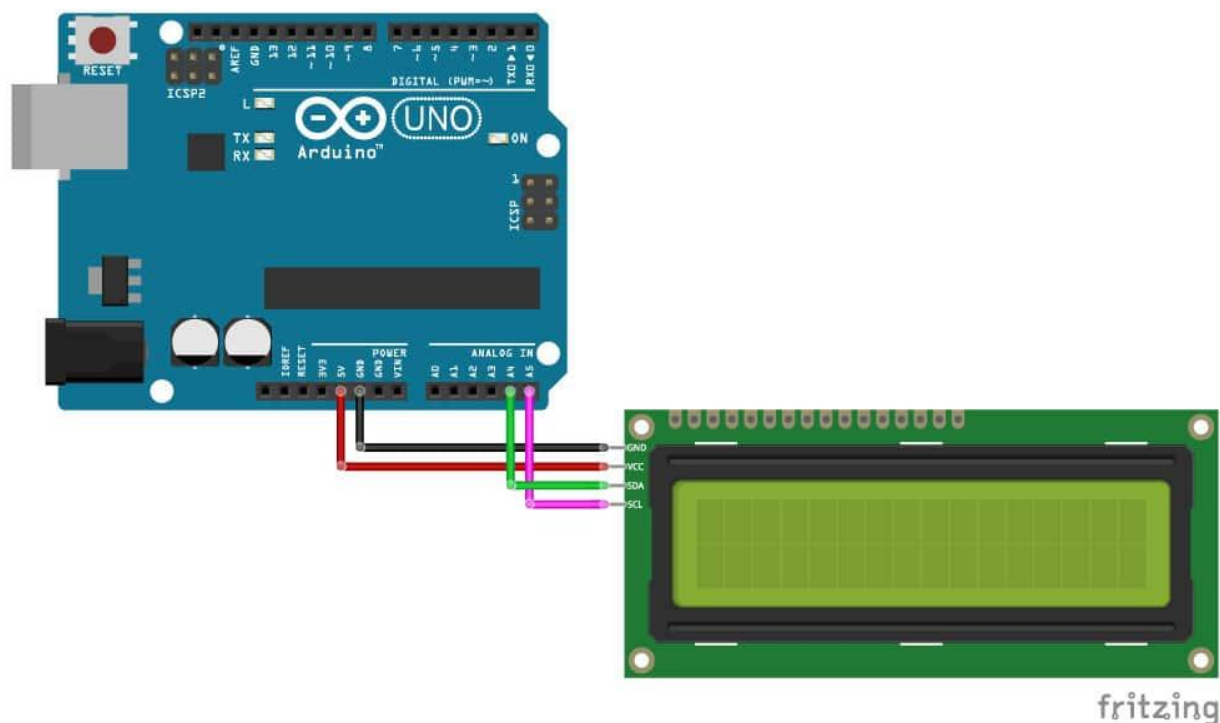
Har bir tsikldan o'tishning boshida displayga 16 belgidan iborat bo'shliqlar qatori yoziladi va u qaytadan boshlanishidan oldin jarayon satrini tozalaydi. Tashqi for() tsikli barcha 16 pozitsiyada takrorlanadi. Har bir belgi pozitsiyasida ichki for() tsikli kursorni o'sha erda ushlab turadi va o'sha joylashuvga ortib borayotgan progress satrining shaxsiy belgisini yozadi. Bu yerda baytni uzatish talab qilinmaydi, chunki ID 0 for() siklidagi j o'zgaruvchisi tomonidan aniqlanadi.

### I2C LCD displeylar.



5-rasm.

Quyidagi ulanish sxemasi sizga I2C LCD-ni Arduino-ga qanday ulashni ko'rsatadi. I2C LCD-ni ulash standart LCD-ni ulashdan ancha oson. 12 emas, balki faqat 4 ta pinni ulashingiz kerak.



Arduino ulanish diagrammasi bilan I2C LCD. Ulanishlar ham quyidagi jadvalda keltirilgan.

#### I2C LCD ulanishlari

I2C LCD displeyi	Arduino platformasi
GND	GND
VCC	5 V
SDA	A4
SCL	A5

Agar siz Arduino Uno dan foydalanmasangiz, SDA va SCL pinlari boshqa joyda bo'lishi mumkin. R3 tartibiga ega Arduino Uno (1.0 pinout) shuningdek, AREF piniga yaqin SDA (ma'lumotlar liniyasi) va SCL (soat chizig'i) pin sarlavhalariga ega ekanligini unutmang. Batafsil ma'lumot uchun quyidagi jadvalni tekshiring.

<b>Platformalar</b>	<b>SDA</b>	<b>SCL</b>
<u>Arduino Uno</u>	A4	A5
<u>Arduino Nano</u>	A4	A5
<u>Arduino Micro</u>	2	3
<u>Arduino Mega 2560</u>	20	21
<u>Arduino Leonardo</u>	2	3
<u>Arduino Muddati</u>	20	21

Turli Arduino platalarida SDA va SCL pinlarining joylashuvi.

LCD kontrastini sozlash

LCD displeyni ulaganingizdan so'ng, displeyning kontrastini sozlashingiz kerak bo'ladi. I2C modulida siz kichik tornavida bilan burishingiz mumkin bo'lgan potansiyometrni topasiz.

LCD displeyni yoqish uchun Arduino USB ulagichini ulang. Orqa yorug'lik yonayotganini ko'rishingiz kerak. Endi potensiyometrni bitta (16×2 LCD) yoki 2 qator (20×4 LCD) to'rtburchaklar paydo bo'lguncha aylantiring.

Agar kerak bo'lsa, kontrastni keyinroq sozlashingiz mumkin.

Buni tugatgandan so'ng, biz LCD-ni dasturlashni boshlashimiz mumkin.

### **I2C LCD uchun asosiy Arduino misol kodi.**

Quyidagi misol kodini Arduino IDE-dan foydalanib Arduinoga yuklashingiz mumkin.

Ushbu qo'llanma uchun men ushbu 16×2 I2C belgili LCD displeydan foydalandim , lekin siz turli o'lchamdagi boshqa I2C LCD displeylardan ham foydalanishingiz mumkin.

Ushbu eskiz namunasi klassik "Salom dunyo!"

```
#include <Wire.h> // I2C aloqasi uchun kutubxona
#include <LiquidCrystal_I2C.h> // LCD uchun kutubxona
// Simlar: SDA pin A4 ga va SCL pin A5 ga ulangan.
// LCD ga I2C orqali ulanish, standart manzil 0x27 (A0-A2 o'tkazilmagan)
LiquidCrystal_I2C lcd = LiquidCrystal_I2C (0x27, 16, 2);
void setup() {
// LCD displeyni ishga tushiring:
lcd.init();
lcd.backlight();
}
```

```
void loop() {  
  // “Salom dunyo!” deb chop eting. LCD displeyning birinchi qatorida:  
  lcd.setCursor(2, 0); // Kursorni uchinchi ustun va birinchi qatorga qo'ying.  
  lcd.print("Salom dunyo!"); // “Salom dunyo!” qatorini chop eting.  
  lcd.setCursor(2, 1); //Kursorni uchinchi ustun va ikkinchi qatorga qo'ying (hisoblash 0  
  dan boshlanadi!).  
  lcd.print("LCD darslik");  
}
```

### **Foydalanilgan adabiyotlar**

- Jeremy Blum, “Exploring arduino: Tools and techniques for engineering wizardry”, 2013
- Electronics: A Systems Approach (6th edition), Neil Storey, Pearson Education UK, 2017
- Arduino.cc veb sayti ma'lumotlari