

№9. AMALIY MASHG'ULOT

ARDUINO-GA LCD DISPLAY VA YETTI SEGMENTLI INDIKATORNI ULASH.

Ishning maqsadi: Arduino UNO ga LCD display va 7 segmentli indikatorni ulash uchun dasturiy va apparat modellarini o'rganish . Amaldagi tizimning tarkibiy qismlari haqida texnik xususiyatlar va umumiy ma'lumotlarni o'rganish.

Amaliy mashg'ulot № 9.1. LCD displeyni Arduino -ga ulash .

Mashg'ulotning maqsadi: Arduino LCD displeyi sensorlardan olingan ma'lumotlarni vizual tarzda ko'rsatish imkonini beradi. QAPASS LCD modulini I2C orqali Arduino -ga qanday qilib to'g'ri ulash kerakligini aytib beramiz va LCD 1602-ning asosiy ishga tushirish va boshqarish buyruqlarini ko'rib chiqamiz. Shuningdek, displayda matn ma'lumotlarini ko'rsatish uchun C ++ dasturlash tilidagi turli funksiyalarni ko'rib chiqamiz, ko'pincha Arduino loyihalarida foydalanish uchun talab qilinadi .

Umumiy ish rejasi:

1. Arduino -ga i2csiz qanday ulash mumkin
2. Arduino uchun kod va i2csiz display
3. LCDni ulash uchun i2c interfeysi sxemasi
4. LCD displeyni i2c orqali Arduinoga qanday ulash mumkin
5. Bir nechta displeylarni i2c avtobusiga ulash
6. LCD 1602 uchun LiquidCrystal_I2C.h kutubxonasi
7. LiquidCrystal_I2C ning foydali funksiyalari



9.1-rasm. LCD displeyni Arduino -ga ulash .

Arduino LCD -ga qanday ulanish mumkin

16 × 2 matnli ekran sensorlardan ma'lumotlarni ko'rsatish, menyularni yoki takliflarni ko'rsatish uchun ishlatiladi. Ekranda 5×8 piksel o'lchamdagi qora belgilar ko'rsatiladi. O'rnatilgan orqa yorug'lik modul pinlarini quvvat bilan ta'minlash orqali yoqiladi. IIC modulisiz 16×2 matnli displey mikrokontrollerga 16 pin orqali ulangan. Ulanish misoli bilan ekranning pinouti quyida joylashgan.

Ushbu mashg'ulot uchun sizga kerak bo'ladi:

- Arduino Uno / Arduino Nano / Arduino Mega;
- LCD monitor 1602 i2c;
- «папа-мама».simlari.



9.2-rasm. 16 × 2 matnli ekranni Arduino -ga ulash sxemasi

LCD 1602 i2c	Arduino Uno	Arduino Nano	Arduino Mega
GND	GND	GND	GND
VCC	5V	5V	5V
SDA	A4	A4	yigirma
SCL	A5	A5	21

Suyuq kristall displeyda 16 ta belgidan iborat 2 qator bor, shuning uchun uning nomi LCD 1602. Qurilma xotirasiga 192 ta belgi o'rnatilgan, yana 8 ta belgini foydalanuvchining o'zi aniqlashi mumkin. Displeyni IIC modulisiz ulashda siz Arduino mikrokontrollerida quvvatni hisobga olmaganda 6 ta umumiy maqsadli portdan foydalanishingiz kerak bo'ladi. Yuqoridagi rasmdagidek sxemani yig'ing va quyidagi dasturni doskaga yuklang.

Eskiz . Arduino va LCD 1602 I2C modulisiz

```
// ekran bilan ishlash uchun kutubxonani ulang
#include <LiquidCrystal.h>
// ob'ektni e'lon qilish, displeyni boshqarish uchun pinlarni belgilang
LiquidCrystal LCD(13, 12, 11, 10, 9, 8);
void setup() {
    LCD.begin(16, 2); // satr va ustunlar sonini belgilang
    LCD.setCursor(1, 0); // kursorni birinchi qatorning 1 belgisiga qo'ying
    LCD.print("Men sevaman"); // xabarni birinchi qatorga chop eting
    LCD.setCursor(8, 1); // kursorni ikkinchi qatorning 1 belgisiga qo'ying
    LCD.print("ARDUINO"); // xabarni ikkinchi qatorga chop eting
}
void loop() {
}
```

Kod uchun tushuntirishlar:

1. QAPASS uchun LiquidCrystal.h standart kutubxonasidan foydalaniladi, u kirill alifbosini qo'llab-quvvatlamaydi;
2. sxemani soddalashtirish va ko'p sonli mikrokontroller pinlarini ishlatmaslik uchun siz I2C modulli displeydan foydalanishingiz kerak.

Arduino I2C interfeysining ishlash printsipi

I2C elektron qurilmalar ichidagi integral mikrosxemalarni ulash uchun ketma-ket ikki simli avtobus bo'lib, I²C yoki IIC (inglizcha Inter-Integrated Circuit) deb nomlanadi.



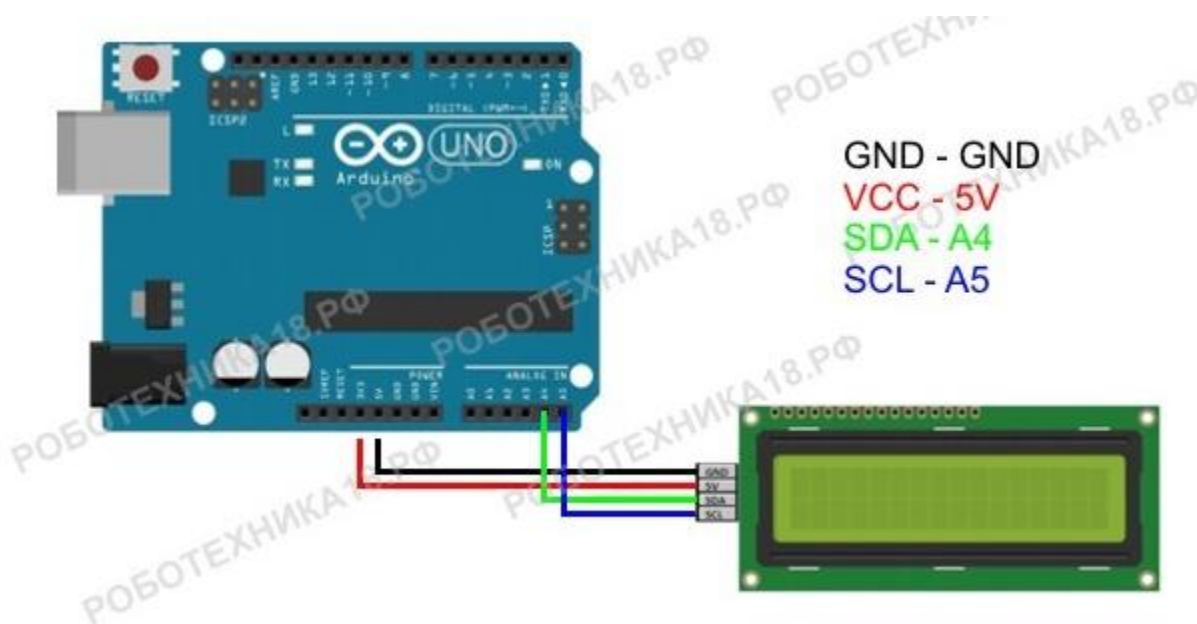
9.3-rasm. lcd16x2 ni ulash uchun i2c interfeysi diagrammasi

I²C 1980-yillarning boshlarida Flibs tomonidan nazorat elektronikasidagi kontaktlarning zanglashiga olib (masalan, anakart kompyuterlari, mobil telefonlar va boshqalar) o'rtasida ichki aloqa uchun oddiy 8-bitli avtobus sifatida ishlab chiqilgan.

Oddiy I²C tizimida bir nechta tobe qurilmalar va ma'lumotlarni uzatishni boshlaydigan va signalni soatlab beradigan bitta asosiy qurilma bo'lishi mumkin. SDA (ma'lumotlar liniyasi) va SCL (soat liniyasi) liniyalariga bir nechta tobe qurilmalar ulanishi mumkin. Ko'pincha asosiy qurilma Arduino boshqaruvchisi, yordamchi qurilmalar esa real vaqt soati yoki LCD displaydir.

Arduino -ga qanday ulash mumkin

I2C modulli 1602 LCD display Arduino platasiga atigi 4 ta sim – 2 ta ma'lumot simi va 2 ta quvvat simi bilan ulangan. QAPASS 1602a-ni Arduino -ga ulash I2C avtobusi uchun standart sifatida amalga oshiriladi: **SDA chiqishi** A4 portiga, **SCL chiqishi** A5 portiga ulangan. LCD display +5V portdan quvvatlanadi. Qo'shimcha ma'lumot olish uchun quyidagi fotosuratda 1602 LCD monitor ulanish sxemasiga qarang.



9.4-rasm. LCD 16x2 modulini I²C orqali Arduino UNO ga ulash

LCD monitorni I2C orqali Arduino -ga **ulaganingizdan so'ng, I2C interfeysi va Wire.h kutubxonasi** (standart Arduino IDE dasturida mavjud) orqali LCD display bilan ishlash uchun LiquidCrystal_I2C.h kutubxonasini o'rnatishingiz kerak bo'ladi. Siz bizning veb-saytimizdagi Arduino kutubxonalari sahifasidagi I2C modulli LCD 1602 uchun LiquidCrystal_I2C.h ishchi kutubxonasini Google Drive-dan to'g'ridan-to'g'ri havola orqali yuklab olishingiz mumkin.

Eskiz uchun LCD display QAPASS 1602 I2C

```
#include < Wire .h > // I2C orqali qurilmalarni boshqarish kutubxonasi
#include < LiquidCrystal_I2C .h > // QAPASS 1602 kutubxonasini qo'shing
LiquidCrystal_I2C LCD (0x27,16,2) ; // displey uchun LCD nomini tayinlang
void setup() {
    LCD. init (); // LCD displeyni ishga tushirish
    LCD. backlight (); // displeyning orqa yorug'ligini yoqish
    LCD. setCursor (1, 0) ; // kursorni birinchi qatorning 1 belgisiga qo'ying
    LCD. print ( "Men sevaman" ) ; // xabarni birinchi qatorga chop eting
    LCD. setCursor (8, 1) ; // kursorni ikkinchi qatorning 1 belgisiga qo'ying
    LCD. print ( "ARDUINO" ) ; // xabarni ikkinchi qatorga chop eting
}
void loop() {
    LCD. noDisplay (); // displeydagi yozuvni o'chiring va yoqing
    delay ( 1000);
    LCD. display ();
    delay ( 1000);
}
```

Kod uchun tushuntirishlar:

1. displeyda ma'lumotni ko'rsatishdan oldin kursor o'rnini `setCursor (0,1)` buyrug'i bilan o'rnatishingiz kerak , bu erda 0 - qatordagi belgilar raqami, 1 - qator raqami;
2. `clear_()` dan farqli o'laroq - `noDisplay ()` buyrug'i yozuvni olib tashlamaydi, lekin uni displeyda ko'rsatishni o'chiradi va uni yana ko'rsatish mumkin.

I2C orqali ikkita displeyni ulash

Odatiy bo'lib, I2C modul manziliga ega barcha 1602 displeylar "0x27" dir, ammo siz matn ekranining manzilini o'zgartirishingiz va uni iic avtobus skaneri orqali topishingiz mumkin.

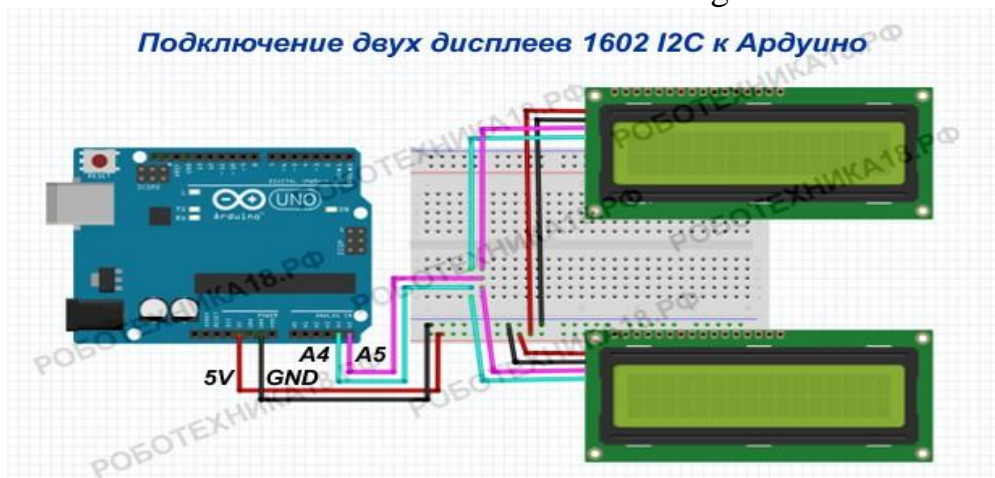


9.5-rasm. I2C modulli 16×2 matnli displey

Shunday qilib, agar siz bir nechta 1602 displeyni bitta mikrokontrollerga ulashingiz kerak bo'lsa, unda mos kelmaslik uchun qurilma manzillarini o'zgartirishingiz kerak. Keling, suyuq kristall displeyning IIC manzilini qanday o'zgartirishni ko'rib chiqaylik.

Agar siz displeyni o'girib, IIC moduliga qarasangiz (yuqoridagi rasmga qarang), "A0 ", " A1" va "A2" deb belgilangan kontaktlarni ko'rishingiz mumkin. Agar sukut bo'yicha LCD displeyda IIC shinasida "0x27" manzili bo'lsa, u holda "A0" jumperini yopish displey manzilini "0x26" ga o'zgartiradi. Shunday qilib, bir nechta displeylar bitta avtobusga ulanishi mumkin, ularning manzillarini eskizda ko'rsatishni unutmang - quyidagi kod misoliga qarang.

Eskiz . Ulanish bir nechta LCD 1602 dan i2c avtobusiga



9.6-rasm. Ikkita 16x2 displeyni I2C orqali Arduino -ga ulash

Keyingi eskizni yuklashdan oldin, avval dual display sxemasini quring va IIC avtobusini skanerlang. Buni Arduino platasi avtobusdagi ikkala qurilmani ham "ko'rishini" ta'minlash uchun qilish kerak . Shuningdek, manzillarning to'g'riligini ikki marta tekshiring. Shundan so'ng, siz quyidagi kodni yuklab olishingiz mumkin, bu sizga bitta Arduino mikrokontrolleridan bir vaqtning o'zida IIC modulli ikkita displeyni boshqarish imkonini beradi. Uno .

```
#include < Wire .h > // I2C avtobus kutubxonasi  
#include < LiquidCrystal_I2C .h > // 16x2 I2C uchun kutubxona
```

```
LCD1 (0x27, 16, 2); // birinchi displeyga nom bering  
LCD2 (0x26, 16, 2); // ikkinchi displeyga nom bering  
bekor sozlash ( ) {  
LCD1. init (); // birinchi displeyni ishga tushirish  
LCD2. init (); // ikkinchi displeyni ishga tushirish  
LCD1. backlight (); // orqa yorug'likni yoqing  
LCD2. backlight (); // orqa yorug'likni yoqing
```

```

}
void setup() {
  // birinchi displeydagi yorliqni aylantiring
  LCD1.setCursor (1, 0);
  LCD1.print ( "Men ARDUINONI SEVAMAN" );
  LCD1.scrollDisplayLeft ();
  // ikkinchi displeyda yorliqni aylantiring
  LCD2.setCursor (1, 0);
  LCD2.print ( "HELLO WORLD" );
  LCD2.scrollDisplayRight ();
  delay ( 300);
}

```

Tushuntirishlar uchun kod :

1. dasturda har bir displey o'ziga xos nomga ega bo'lishi kerak.

LCD displey uchun LiquidCrystal_I2C.h kutubxonasi

Quyidagi misolda biz LiquidCrystal_I2C Arduino kutubxonasi taqdim etadigan bir nechta imkoniyatlarni bir vaqtning o'zida tahlil qilamiz (aslida biz buyruqlarni standart kutubxonadan oldik). Biz miltillovchi kursorning chiqishini, yozuvni o'chirmasdan bir muddat ekrandan qanday olib tashlashni va dastur kodidan displeyning orqa yorug'ligini qanday boshqarishni ko'rsatamiz. Buning uchun quyidagi dastur kodini Arduino ga yuklang.

Eskiz . Tavsif LiquidCrystal_I2C.h kutubxonalari

```

#include < Wire .h > // I2C orqali qurilmalarni boshqarish kutubxonasi
#include < LiquidCrystal_I2C .h > // QAPASS 16x2 kutubxonasini qo'shish

LiquidCrystal_I2C LCD (0x27,16,2) ; // displey uchun LCD nomini tayinlang

void setup() {
  LCD.init ( ) ; // LCD displeyni ishga tushirish
  LCD.backlight (); // displeyning orqa yorug'ligini yoqish

  LCD.blink (); // miltillovchi kursorni yoqish
  delay ( 2000);
  LCD.noBlink ();

  LCD.setCursor (1, 0 ) ; // kursorni birinchi qatorning 1 belgisiga qo'ying
  LCD.print ( "Men sevaman" ); // xabarni birinchi qatorga chop eting
  LCD.setCursor (8, 1 ) ; // kursorni ikkinchi qatorning 1 belgisiga qo'ying
  LCD.print ( "ARDUINO" ); // xabarni ikkinchi qatorga chop eting
  delay ( 2000);
}

```

```

LCD.cursor(); // joriy kursor o'rnini yoqish
delay ( 2000);
LCD.noCursor ();
}

bekor loop ( ) {
  LCD.noBacklight (); // o'chiring va ekranning orqa yorug'ligini yoqing
  delay ( 1000);
  LCD.backlight ();
  delay ( 1000);
}

```

Kod uchun tushuntirishlar:

1. LiquidCrystal_I2C.h kutubxonasi LCD displeyni I²C avtobusi orqali boshqarish uchun o'rnatilgan buyruqlarni o'z ichiga oladi va eskizni sezilarli darajada soddalashtirishga imkon beradi.
2. Arduino displeyida chiziqni avtomatik aylantirish imkoniyati haqida ham o'qing .

Xulosa: Ushbu postdagi lcd 1602a uchun Arduino dasturlari va lcd 1602-ni i2c orqali Arduino -ga ulash sxemasidan foydalanib, siz ushbu displeyni Arduino loyihalarida LCD va turli misollarda ishlatishingiz mumkin. Agar sizda 1602 i2c Arduino monitoridan foydalanish bo'yicha savollaringiz bo'lsa , ularni ushbu sahifadagi izohlarda qoldiring. Ehtimol , siz qidirayotgan narsa allaqachon hal qilingan va javoblarda.

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:

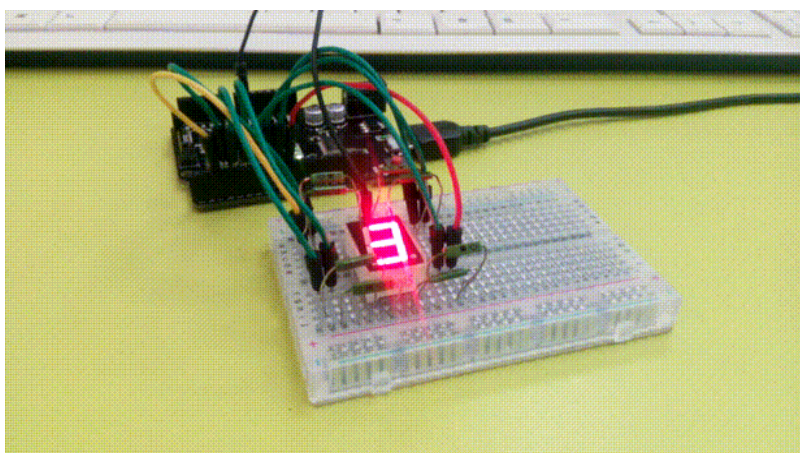
1. Amaliyot nomlari va ishning maqsadi.
2. Breadboard va tadqiqot tushunchasi.
3. Tadqiqotda foydalaniladigan texnik vositalarning xarakteristikalar.
4. Dastur kodini va kompilyatsiya skrinshotlarini yozish.
5. Bajirilgan vazifa bo'yicha foto / video hisobot.
6. Nazorat savollariga javob yozing (qo'lda).

№ 9.2. Amaliy mashg'ulot Arduino 7 segmentli LED .

Mashg'ulotning maqsadi: Bir xonali etti segmentli indikatorni Arduino -ga non doskasi orqali ulashingiz mumkin, eng muhimi , Arduino -dan segmentlarni boshqarish uchun pinout (indikator pinout) ni bilishdir. Uno . Etti segmentli indikatorni Arduino - ga qanday ulash va oddiy taymer dasturini yaratishni ko'rib chiqing. Biz soat tugmasi yordamida indikatorni to'g'ridan-to'g'ri mikrokontrollerdan boshqaramiz.

Umumiy ish rejasi:

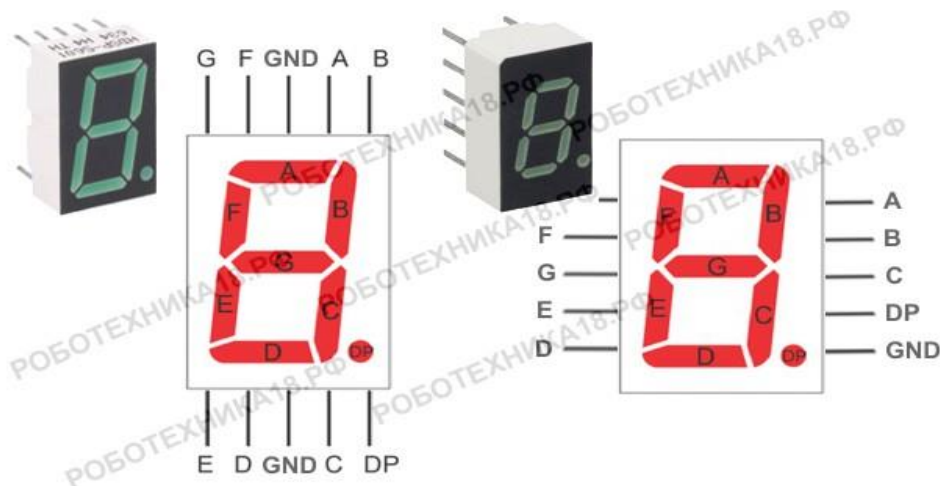
1. Pinout etti segmentli ko'rsatkich Arduino segmentli Arduino indikatorini qanday ulash mumkin
3. Yetti segmentli indikator va Arduino kodi
4. 1-raqamli yetti segmentli indikatorni boshqarish
5. Yetti segmentli indikator va tugma uchun kod



9.7-rasm. Yetti segmentli ko'rsatkich Arduino

Rasmda (9.8-rasm) umumiy katodli (minus) bir xonali etti segmentli indikatorning pinouti ko'rsatilgan. Modul kichik led ettita LED mavjud bo'lgan indikator (shu sababli indikator o'z nomini oldi) va sakkizinchi LED nuqta shaklida. Arduino Uno dan boshqa ketma-ketlikda LEDlarni yoqish orqali siz turli raqamlarni ko'rsatishingiz mumkin.

E'tibor bering, panelda rezistorlar yo'q, shuning uchun LEDlarni ulashda tashqi rezistorlardan foydalaning. Agar pinout bo'lsa umumiy anodli etti segmentli indikator, siz tushunmaysiz, keyin quvvatni turli chiqishlarga ulash orqali empirik ravishda pinoutni o'rnatishingiz mumkin.



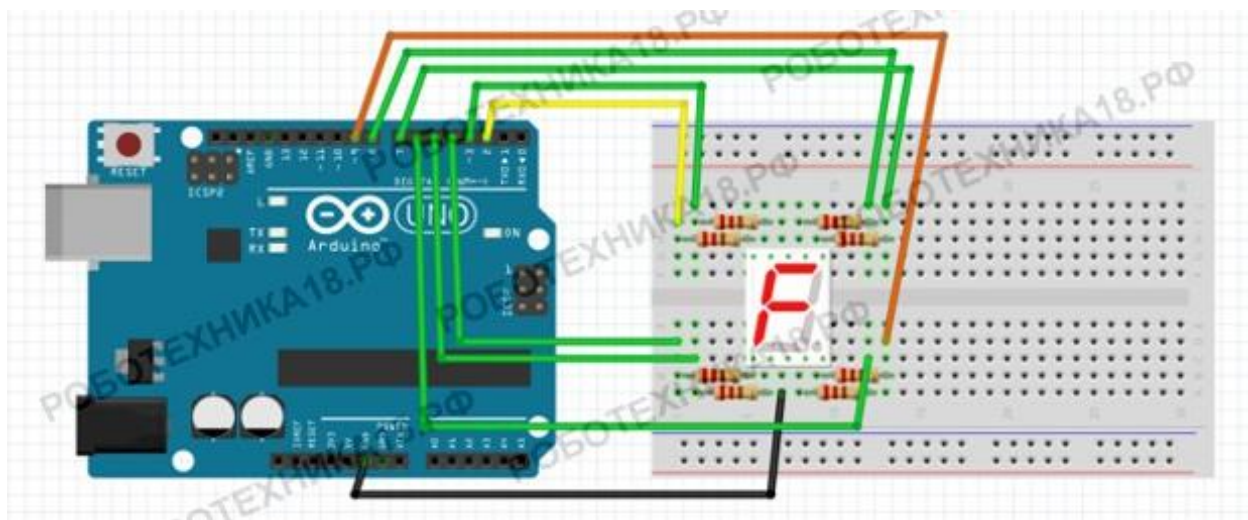
9.8-rasm. hdsp 7503 / 5161as 1 xonali etti segmentli indikatorning pinouti

Agar siz uni noto'g'ri yoqsangiz, hech qanday yomon narsa bo'lmaydi, lekin qarshiliksiz LEDlar yonib ketishi mumkin.

segmentli indikatorni qanday ulash mumkin

Ushbu mashg'ulot uchun sizga kerak bo'ladi:

- Arduino Uno / Arduino Nano / Arduino Mega;
- bitta raqamli etti segmentli ko'rsatkich 5161 sifatida / hdsp 7503;
- soat tugmasi;
- rezistorlar 220 Ohm;
- taxtasi (nonboard);
- erkak-erkak simlar.



9.9-rasm. Arduino -ga yetti segmentli indikator uchun ulanish sxemasi

5161as / hdsp 7503 modulidan siljish registrsiz foydalanish uchun siz ko'p sonli pinlardan foydalanishingiz kerak bo'ladi. Arduino Indikatoridagi LEDlarni yoqish uchun Uno . Birinchi misolda biz paneldagi turli raqamlarni ko'rsatish uchun LEDlarni navbat bilan yoqamiz / miltillaymiz . Yuqoridagi rasmdagi kabi sxemani yig'ing va mikrokontrollerga quyidagi eskizni yuklang.

Eskiz . Chiqish raqamlar 7 segmentli displeyga Arduino

```
#define A 8
#define B 7
#define C 6
#define D 5
#define E 4
#define F 3
#define G 2
void setup() {
  pinMode(A, OUTPUT);
  pinMode(B, OUTPUT);
  pinMode(C, OUTPUT);
  pinMode(D, OUTPUT);
  pinMode(E, OUTPUT);
  pinMode(F, OUTPUT);
  pinMode(G, OUTPUT);
}
void loop() {
  digitalWrite(A, LOW); // birinchi raqam
  digitalWrite(B, HIGH);
  digitalWrite(C, HIGH);
  digitalWrite(D, LOW);
  digitalWrite(E, LOW);
  digitalWrite(F, LOW);
  digitalWrite(G, LOW);
  delay(1000);
  digitalWrite(A, HIGH); // ikkinchi raqam
  digitalWrite(B, HIGH);
  digitalWrite(C, LOW);
  digitalWrite(D, HIGH);
  digitalWrite(E, HIGH);
  digitalWrite(F, LOW);
  digitalWrite(G, HIGH);
  delay(1000);
  digitalWrite(A, HIGH); // uchinchi raqam
  digitalWrite(B, HIGH);
  digitalWrite(C, HIGH);
  digitalWrite(D, HIGH);
```

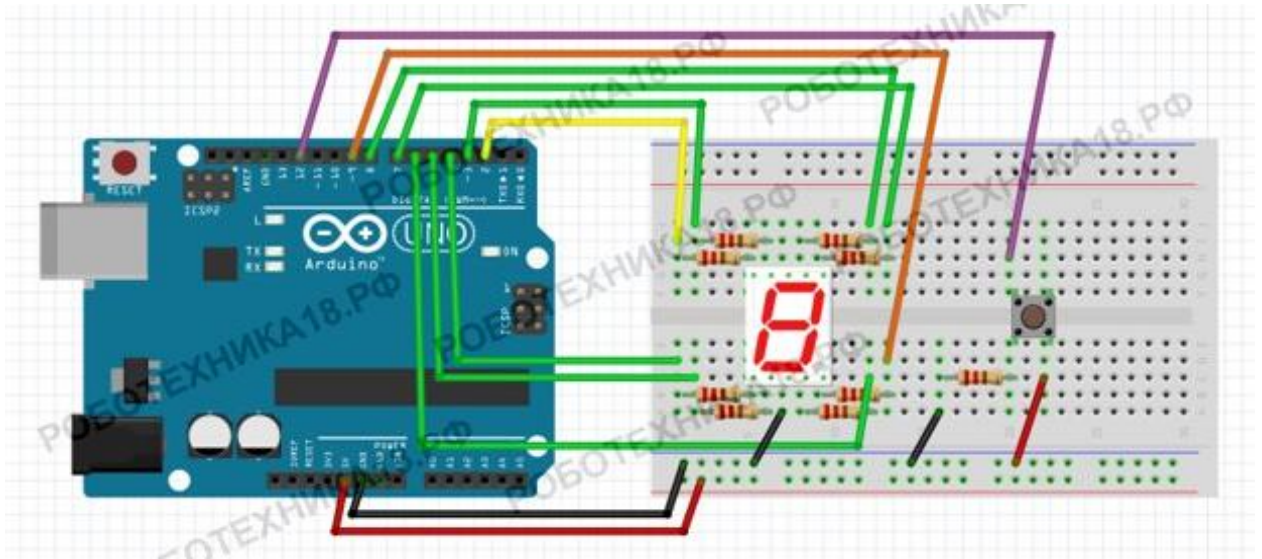
```

digitalWrite(E, LOW);
digitalWrite(F, LOW);
digitalWrite(G, HIGH);
delay(1000);
}

```

Kod uchun tushuntirishlar:

1. eskiz katta bo'lib chiqadi, shuning uchun biz o'zimizni uchta raqam bilan chekladik. Arduino etti segmentli indikatorga boshqa raqamlarni chiqarish qiyin emas.



9.10-rasm. Etti segmentli indikatorning tugma bilan ulanish diagrammasi

Quyidagi misolda indikatoragi raqamlar faqat takt tugmasi bosilganda o'zgaradi. 3 raqamiga erishgandan so'ng, taymer qayta tiklanadi va tugma yana bosilishini kutadi. Bular Arduino uchun juda oddiy dasturlar va yetti segmentli indikator; yanada murakkab va qiziqarli dasturlar uchun Arduino uchun 74hc595 smenali registrdan allaqachon foydalanishingiz kerak .

Eskiz . Yagona raqam etti segmentli indikator va tugma

```

#define A 8
#define B 7
#define C 6
#define D 5
#define E 4
#define F 3
#define G 2
#define BUTTON 12
byte v = 0;
void setup() {
  pinMode(A, OUTPUT);

```

```

pinMode(B, OUTPUT);
pinMode(C, OUTPUT);
pinMode(D, OUTPUT);
pinMode(E, OUTPUT);
pinMode(F, OUTPUT);
pinMode(G, OUTPUT);
pinMode(BUTTON, INPUT);
}
void loop()
{
  digitalWrite(A, HIGH); //цифра ноль
  digitalWrite(B, HIGH);
  digitalWrite(C, HIGH);
  digitalWrite(D, HIGH);
  digitalWrite(E, HIGH);
  digitalWrite(F, HIGH);
  digitalWrite(G, LOW);
  if (digitalRead(BUTTON) == HIGH)
  { delay(500); v = 1; }
  while (v == 1)
  {
    digitalWrite(A, LOW); //цифра один
    digitalWrite(B, HIGH);
    digitalWrite(C, HIGH);
    digitalWrite(D, LOW);
    digitalWrite(E, LOW);
    digitalWrite(F, LOW);
    digitalWrite(G, LOW);
    if (digitalRead(BUTTON) == HIGH) { delay(500); v = 2; }
  }
  while (v == 2)
  {
    digitalWrite(A, HIGH); //цифра два
    digitalWrite(B, HIGH);
    digitalWrite(C, LOW);
    digitalWrite(D, HIGH);
    digitalWrite(E, HIGH);
    digitalWrite(F, LOW);
    digitalWrite(G, HIGH);
    if (digitalRead(BUTTON) == HIGH)
    { delay(500); v = 3; }
  }
  while (v == 3) {
    digitalWrite(A, HIGH); //цифра три

```



```
digitalWrite(B, HIGH);  
digitalWrite(C, HIGH);  
digitalWrite(D, HIGH);  
digitalWrite(E, LOW);  
digitalWrite(F, LOW);  
digitalWrite(G, HIGH);  
if (digitalRead(BUTTON) == HIGH) { delay(500); v = 0; }  
} }
```

Kod uchun tushuntirishlar:

1. o'zgaruvchan bayt $v = 0$; dasturda bir while tsiklidan ikkinchisiga o'tish uchun ishlatiladi. Tugma bosilganda v o'zgaruvchining qiymati o'zgaradi;
2. while siklidan ikkinchisiga tez o'tishdan himoya qilish uchun har bir holatda biroz kechikishga ega .

Xulosa _ Biz ushbu modul va undan Arduino platasi bilan foydalanish bilan tanishish bilan cheklandik . Bir nechta panellar yoki to'rt xonali etti segmentli indikatorlardan foydalanib, siz allaqachon to'liq huquqli Arduino taymerini yoki real vaqt soatini yaratishingiz mumkin . Biz ushbu sxemalarni yangi boshlanuvchilar uchun Arduino loyihalari bo'limiga joylashtirdik, bu yerda har kim mikrokontrollerda o'z xohishiga ko'ra loyihani topishi mumkin.

Tarkib haqida xabar berish.

Hisobotda quyidagilar bo'lishi kerak:

1. Amaliyot nomlari va ishning maqsadi.
2. Breadboard va tadqiqot tushunchasi.
3. Tadqiqotda foydalaniladigan texnik vositalarning xarakteristikalari.
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. LCD displey nima?
2. LCD16x2 ni ulash uchun I2C interfeys diagrammasini tushuntiring.
3. segmentli indikator tushunchasini aniqlang .
4. Arduino LCD- ga I2Csiz qanday ulanish mumkin?
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. Arduino ga ulash sxemasini chizing .
9. Ushbu tadqiqotning sxematik sxemasini tuzing.
10. O'rganish sxemasida datchiklar parametrlariga qanday talablar qo'yiladi?

