

Logo

Arduino Training Board

MUNDARIJA

1-DARS. ARDUINO IDE DASTURINI O'RNATISH.....	1
1.1 Kirish.....	1
1.2 Dasturini o'rnatish.....	3
1.3 Kutubxonalarini o'rnatish.....	6
1.4 Arduino.....	9
1.5 Arduino Uno haqida ma'lumot.....	10
2-DARS LED YORITGICH.....	12
2.1 Umumiylar.....	12
2.2 Ishlash prinsipi.....	12
2.3 Ulanish tavsifi.....	13
2.4 Dasturiy taminot qismi.....	13
2.5 Kodni yuklash.....	14
3-DARS TUGMA YORDAMIDA LEDNI BOSHQARISH.....	16
3.1 Umumiylar.....	16
3.2 Ulanish tavsifi.....	16
3.3 Dasturiy taminot qismi.....	17
4-DARS AKTIV BUZZER.....	18
4.1 Umumiylar.....	18
4.2 Ulanish tavsifi.....	18
4.3 Dasturiy taminot qismi.....	18
5-DARS SVETAFOR YASASH	19
5.1 Umumiylar.....	19
5.2 Ishlash prinsipi.....	19
5.3 Ulanish tavsifi.....	20
5.4 Dasturiy taminot qismi.....	20
6-DARS FOTOREZISTOR	22
6.1 Umumiylar.....	22
6.2 Ishlash prinsipi.....	23
6.3 Ulanish tavsifi.....	23
6.4 Dasturiy taminot qismi.....	24
7-DARS IR SENSOR ORQALI LED YORITGICHNI BOSHQARISH.....	25
7.1 Umumiylar.....	25
7.2 Ulanish tavsifi.....	25
7.3 Dasturiy taminot qismi.....	25

8-DARS. RGB LED MODULINI BOSHQARISH	27
8.1 Umumiy ma'lumot.....	27
8.2 Ulanish tavsifi.....	27
8.3 Dasturiy taminot qismi.....	27
9-DARS SERVO MATOR BOSHQARISH.....	29
9.1 Umumiy ma'lumot.....	29
9.2 Ulanish tavsifi.....	29
9.3 Dasturiy taminot qismi.....	30
10-DARS AYLANADIGAN ENKODER.....	30
10.1 Umumiy ma'lumot.....	30
10.2 Ulanish tavsifi.....	31
10.3 Dasturiy taminot qismi.....	31
11-DARS. LED MATRITSAGA MATN YOZISH.....	33
11.1 Umumiy ma'lumot.....	33
11.2 Ulanish tavsifi.....	34
11.3 Dasturiy taminot qismi.....	34
12-DARS 7 SEGMENTLI DISPLAYDAN SANAGICH YASASH.....	35
12.1 Umumiy ma'lumot.....	35
12.2 Ulanish tavsifi.....	36
12.3 Dasturiy taminot qismi.....	36
13-DARS. HARORAT VA NAMLIKNI ANIQLASH.....	38
13.1 Umumiy ma'lumot.....	38
13.2 Ulanish tavsifi.....	39
13.3 Dasturiy taminot qismi.....	39
14-DARS. ULTIRASONIK ORQALI MASOFANI O'LCHASH.....	41
14.1 Umumiy ma'lumot.....	41
14.2 Ulanish tavsifi.....	42
14.3 Dasturiy taminot qismi.....	42
15-DARS VAQT MODULI ORQALI SOAT YASASH.....	44
15.1 Umumiy ma'lumot.....	44
15.2 Ulanish tavsifi.....	45
15.3 Dasturiy taminot qismi.....	45
16-DARS KEYPAD ORQALI EKRANGA YOZISH.....	48
16.1 Umumiy ma'lumot.....	48
16.2 Ulanish tavsifi.....	49
16.3 Dasturiy taminot qismi.....	49

17-DARS. NRF24L01 MODULDAN FOYDALANISH.....	51
17.1 Umumiy ma'lumot.....	51
17.2 Ulanish tavsifi.....	51
17.3 Dasturiy taminot qismi.....	52
18-DARS. BLUETOOTH ORQALI RGB LEDNI BOSHQARISH.....	54
18.1 Umumiy ma'lumot.....	54
18.2 Ulanish tavsifi.....	54
18.3 Dasturiy taminot qismi.....	54
18.4 Ilovani sozlash.....	56
19-DARS ESP8266 WI-FI MODULI.....	57
19.1 Umumiy ma'lumot.....	57
19.2 ESP8266 texnik xususiyatlari.....	57
19.3 ESP8266 versiyasi.....	58
19.4 NODEMCU ESP8266 pinlar.....	59
20-DARS ARDUINO IDE DASTURI YORDAMIDA ESP8266 MODULIGA DASTUR YUKLASH.....	60
20.1 Arduino IDEga ESP8266 pluginini o'rnatish.....	60
20.2 Sinov uchun sodda loyiha.....	61
21-DARS ESP8266 DAN FOYDALANIB WI-FI ORQALI TRAFFIC LIGHT MODULINI BOSHQARISH.....	63
21.1 Asinxron tarmoq server.....	64
21.2 Ulanish tavsif.....	64
21.3 Dasturiy taminot qismi.....	65

1-DARS. ARDUINO IDE DASTURINI O'R NATISH

1.1. Kirish

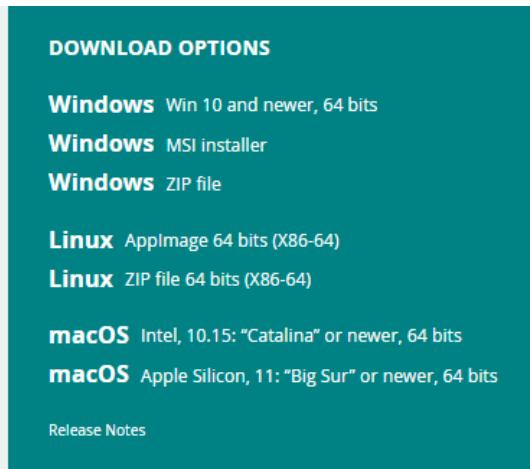
Arduino IDE dasturi Arduino platformasining dasturiy ta'minot qismidir. Ushbu dastur orqali loyihalaringizning dasturiy taminot qismini ishlab chiqishingiz mumkun. Ushbu darsda siz kompyuteringizga Arduino IDE dasturini qanday o'rnatish va uni qanday sozlash kerakligini bilib olasiz. Windows, Mac va Linux tizimlariga mo'ljallangan Arduino IDE dasturlari mavjud. O'rnatish jarayoni barcha uchta tizim uchun farq qiladi va dasturiy ta'minotni o'rnatish bir nechta qadamlarni o'z ichiga oladi.

1-QADAM: <https://www.arduino.cc/en/software/> saytiga o'ting.

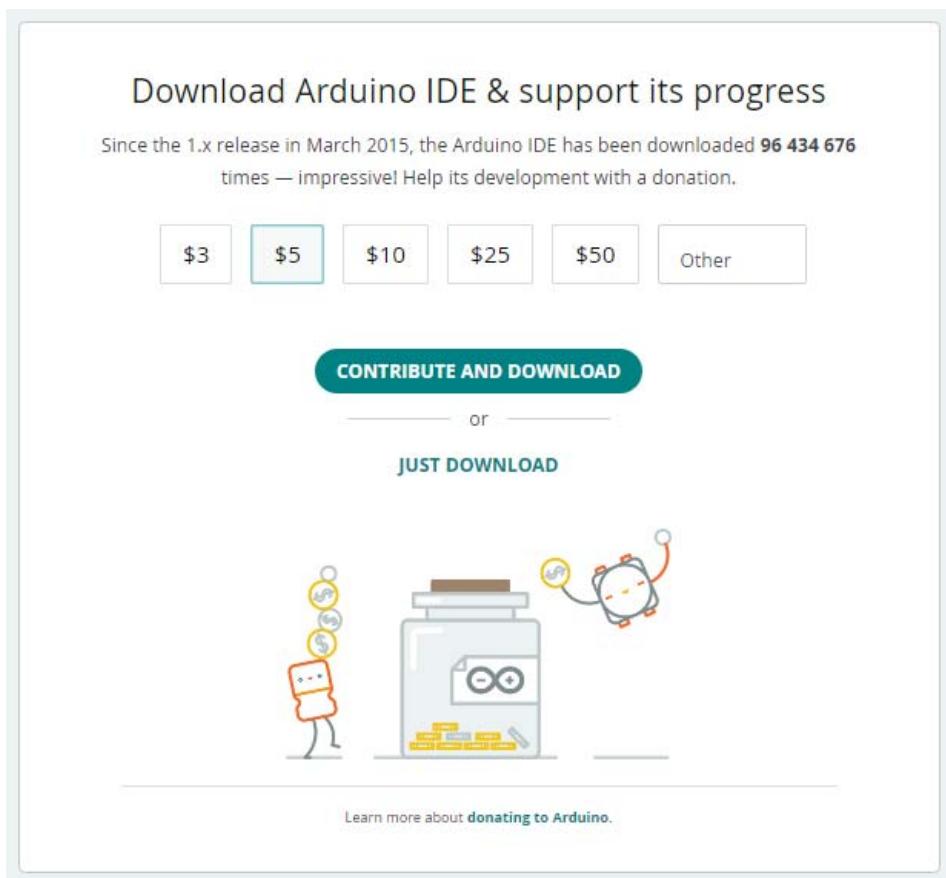
Ushbu veb-saytda mavjud bo'lgan versiya odatda eng so'nggi versiya hisoblanadi, va siz saytga kirganingizda rasmda ko'rsatilgan versiyadan yangiroq versiyani ko'rishingiz mumkin.

2-BOSQICH: Kompyuteringiz operatsion tizimiga mos bo'lgan dasturiy ta'minotni yuklab oling. Bu yerda Windows misol sifatida keltirilgan. Chunki Windows operatsion tizimi dunyo bo'yicha eng ko'p foydalaniladigan operatsion tizim hisoblanadi.

Veb-saytning quyidagi qismidan ‘Windows Win 10 and newer, 64 bits’ ni bosing.



Keyin quyidagi oyna ochiladi.



‘JUST DOWNLOAD’ ni bosing va siz dasturni yuklab olasiz.

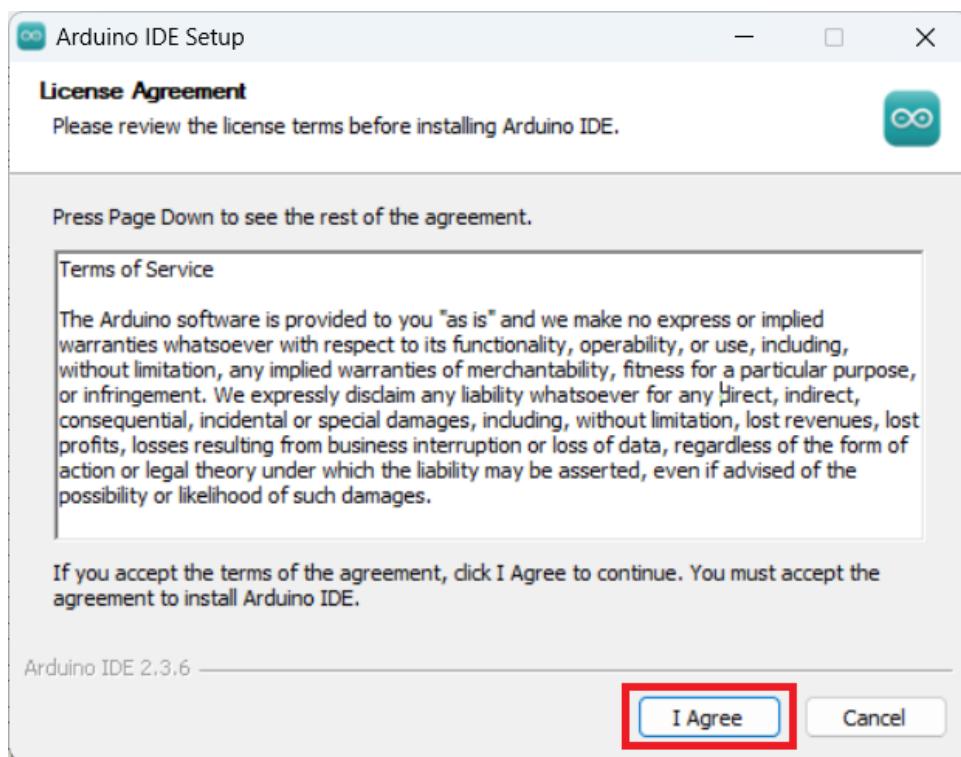
Kurs tayyorlangan vaqtida barcha materiallarimizning versiyalari eng so‘nggi holatda bo‘lgan.

1.2 Dasturini o‘rnatish

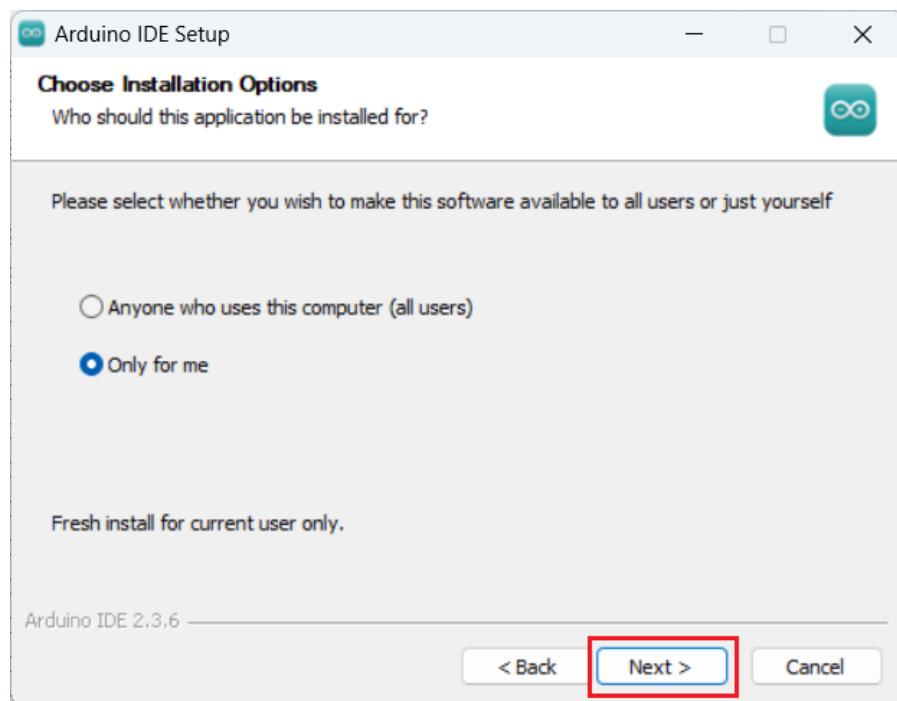
Dasturni o‘rnatish uchun kompyuteringizdan yuklab olingan dasturni toping.

Name	Date modified	Type	Size
▼ Today			
arduino-ide_2.3.6_Windows_64bit.exe	7/13/2025 10:2...	Application	153,806 KB
▼ Last week			

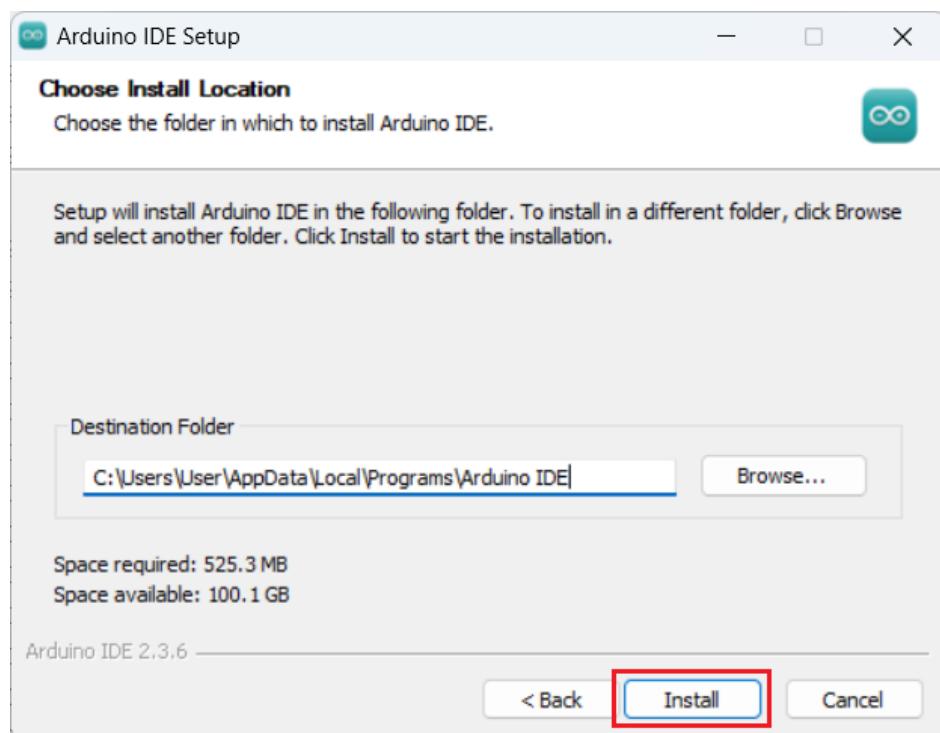
Yuqorida yuklab olgan .exe farmatdagi fayilni ishga tushuramiz.



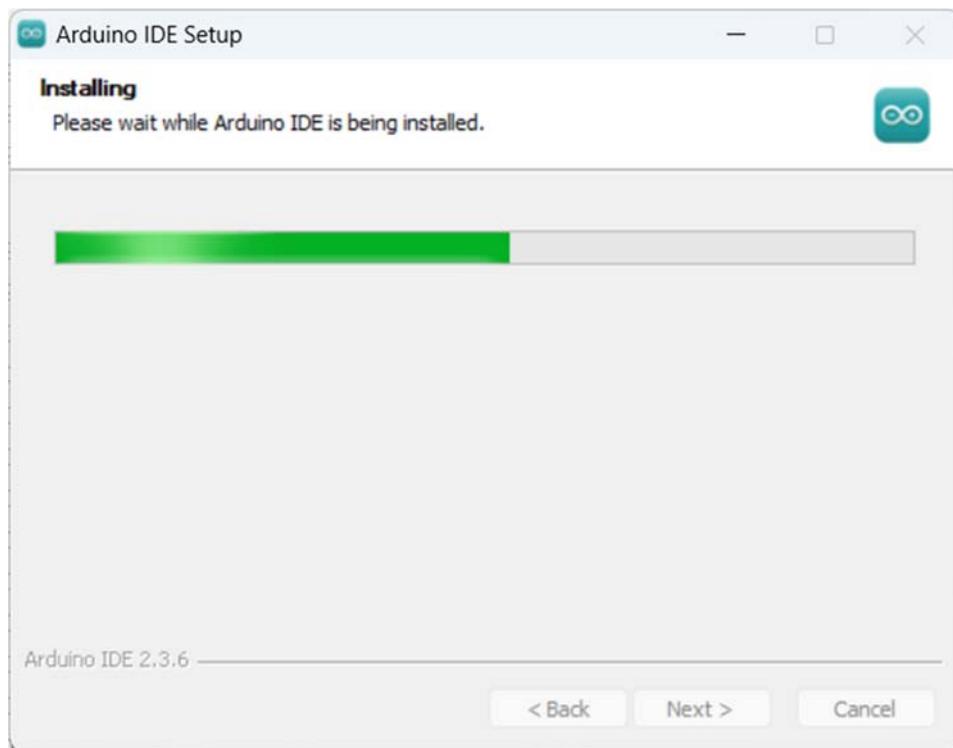
‘I Agree’ tugmasini bosing



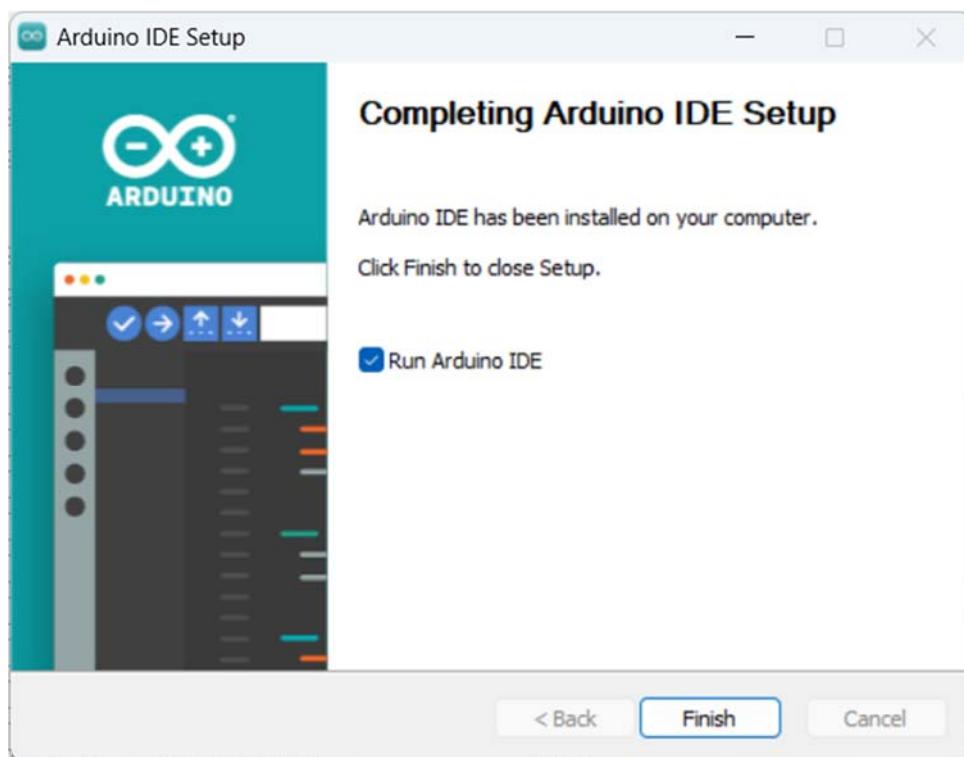
‘Next >’ tugmasini bosing.



Bu yerda dasturni o‘rnatish joyini o‘zgartitishingiz mumkun. Keyin esa ‘Install’ tugmasini bosing.



O‘rnatish yakunlangandan so‘ng quyidagi oyna ochiladi.



‘Finish’ tugmasini bosganingizdan so‘ng o‘rnatish yakunlanadi va Arduino IDE dasturi ishga tushadi.

Arduinoni o‘rnatish (Mac OS) uchun Zip faylni yuklab oling va arxivdan chiqaring, so‘ng Arduino.app fayliga ikki marta bosib, Arduino IDE dasturiga kiring; agar kompyuteringizda Java ishgatushurish kutubxonasi (Java runtime library) mavjud bo‘lmasa, tizim uni o‘rnatishni so‘raydi. O‘rnatish tugallangach, Arduino IDE dasturidan foydalanishingiz mumkin.

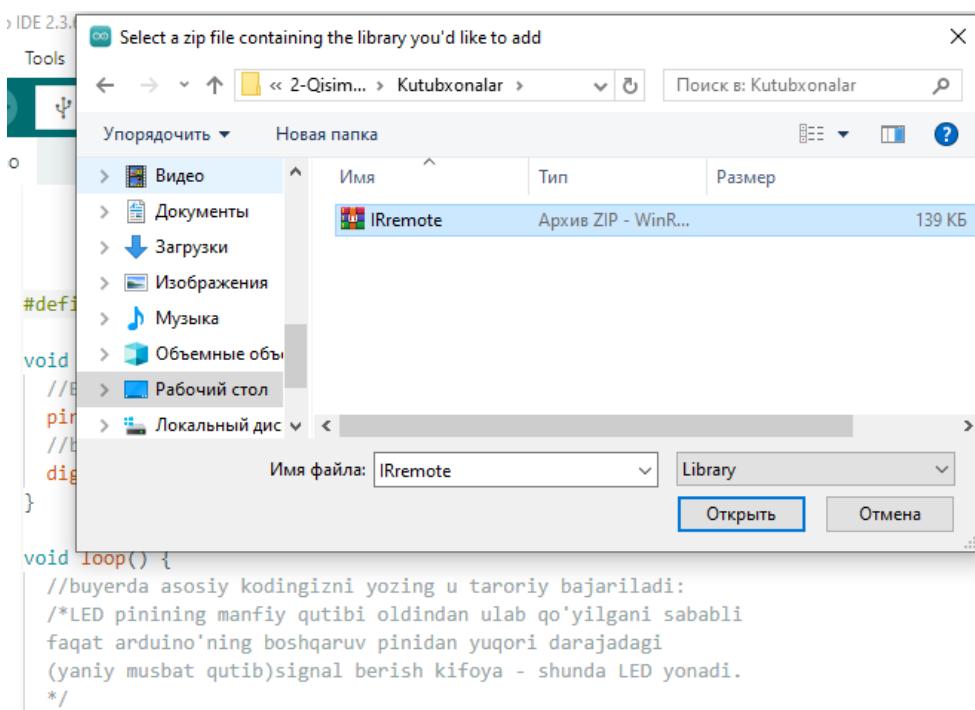
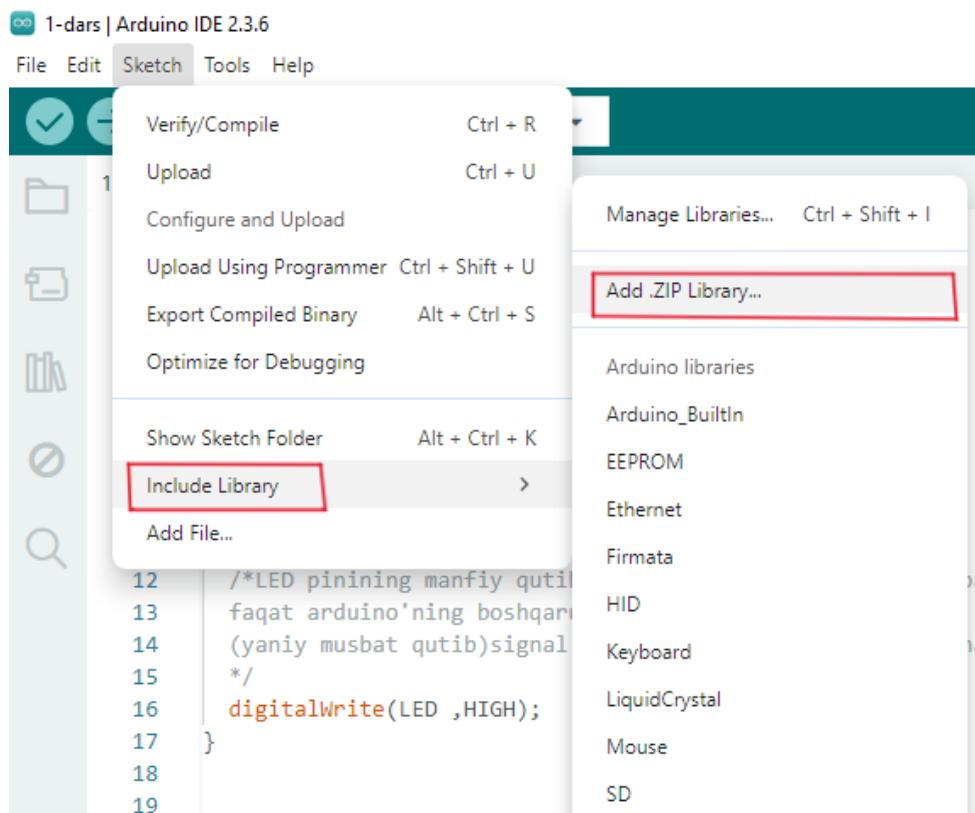
Arduinoni o‘rnatish (Linux) uchun make install buyrug‘idan foydalanishingiz kerak bo‘ladi. Agar Ubuntu tizimidan foydalanayotgan bo‘lsangiz, Arduino IDE dasturini Ubuntu dastur markazi orqali o‘rnatish tavsiya etiladi.

Qo‘srimcha Arduino kutubxonalarini o‘rnatish. Agar siz Arduino dasturiy ta’minoti bilan ishlashda va ichki funksiyalardan foydalanishda yetarli darajada tajribaga ega bo‘lsangiz, o‘zingizning Arduino qurilmangiz imkoniyatlarini qo‘srimcha kutubxonalar orqali kengaytirishingiz mumkin.

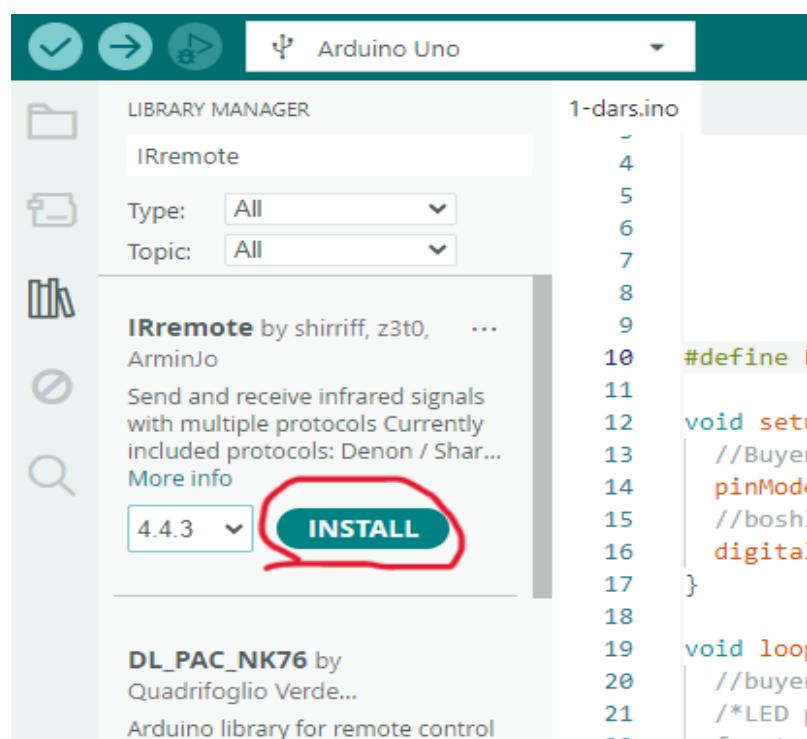
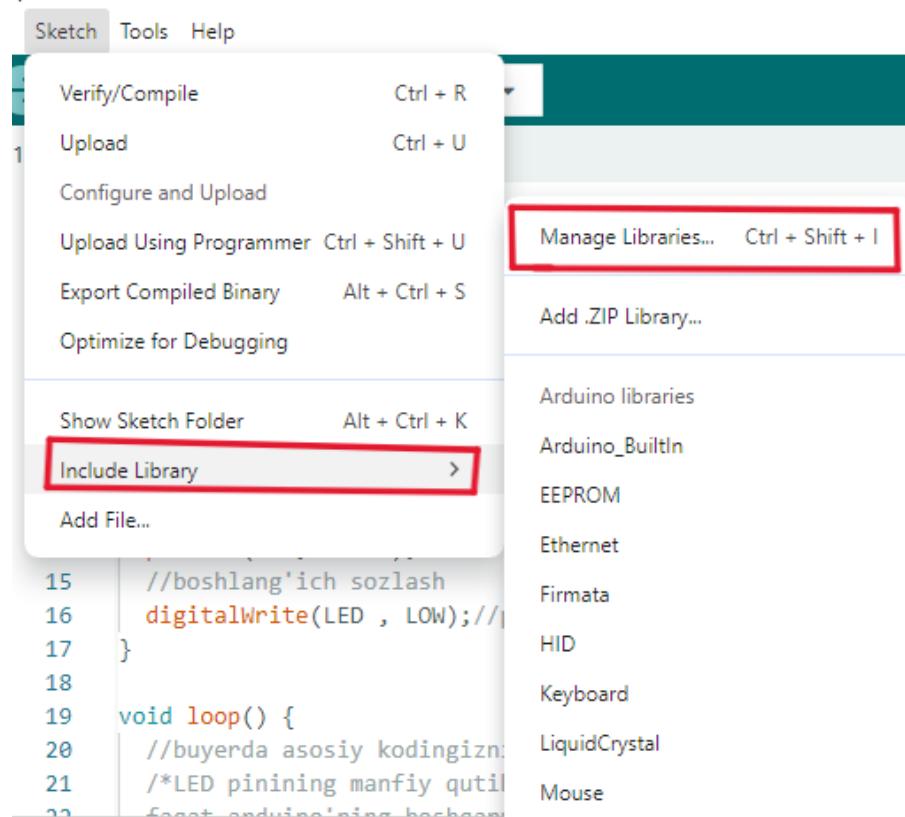
1.3. Kutubxonalarni o‘rnatish

Kutubxonalar bu sizga sensorlar, displeylar va modular bilan ulanishni osonlashtiradigan kodlar to‘plami. Misol uchun Liquid Crystal kutubxonasi sizga LCD Displayga belgilarni oson chiqarishga yordam beradi. Internetda yuklab olish uchun yuzlab qo‘srimcha kutubxonalar mavjud. Qo‘srimcha kutubxonalardan foydalanish uchun avval ularni o‘rnatishingiz kerak.

Kutubxona menejeridan foydalanish. Arduino IDE dasturida kutubxona menejeridan foydalanishingiz mumkun. Arduino IDE dasturini oching va "Sketch" menyusiga bosing va keyin "Include Library" bo‘limini tanlang keyin "Manage Libraries." Bo‘limidan kutubxonalarni qo‘sishingiz mumkun. "Add .ZIP Library..." bo‘limidan .ZIP farmatdagi kutubxonalarni qo‘sishingiz mumkun. Misol: .ZIP farmatdagi *IRremote* kutubxonasini qo‘sish. Arduino IDE dasturini oching va quyidagi menyularga o‘ting "Sketch" > "Include Library" > "Add .ZIP Library...".



Ikkinci usulda esa ‘Add .ZIP Library...’. Menyusi o‘rniga ‘Manage Libraries.’ Menyusiga o‘ting va qidiruv bo‘limidan IRromote kutubxonasini qidiring keyin ‘install’ tugmasini bosing.



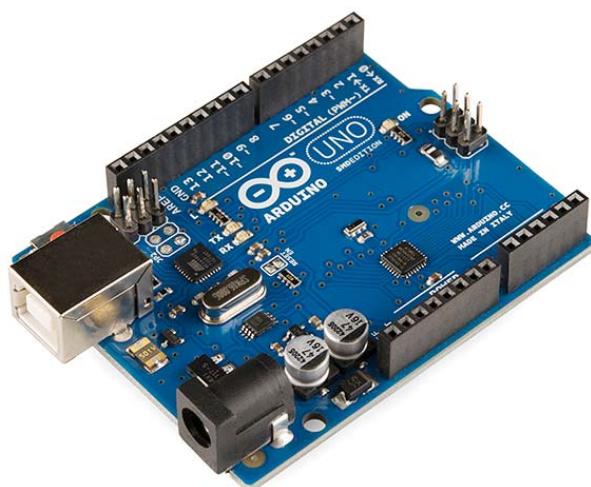
1.4. Arduino

Arduino - bu foydalanish uchun qulay bo‘lgan apparat va dasturiy ta’minotga asoslangan ochiq manbali elektron platforma hisoblanadi. Interaktiv loyihalarda ishlayotgan har qanday kishiga mos keladi. Odatda, Arduino loyihasi sxemalar va kodlardan iborat bo‘ladi. Arduino platasi mikroprotsessor, kirish - chiqish interfeyslari kabi komponentlar yig‘indisidan iborat bo‘lgan sxema platasi hisoblanadi. Arduino platasi sensorlar yordamida atrof-muhit haqida ma’lumot oladi va foydalanuvchi amallarini qabul qilib ekranga malumot chiqarish, LED yoritgichlarni boshqarish va boshqa shu kabi funksiyalarni bajarishi mumkin. Biz faqat sxemani yig‘ishimiz va kod yozishimish kifoya. Hozirda Arduino platasining bir nechta modellari mavjud bo‘lib, ularning o‘rtasidagi kodlar umumiyligi bo‘ladi (biroq texnik farqlari tufayli ayrim platalar to‘liq mos kelmasligi mumkin). Ommabop boshqaruv platasi modellari quyidagilarni o‘z ichiga oladi.

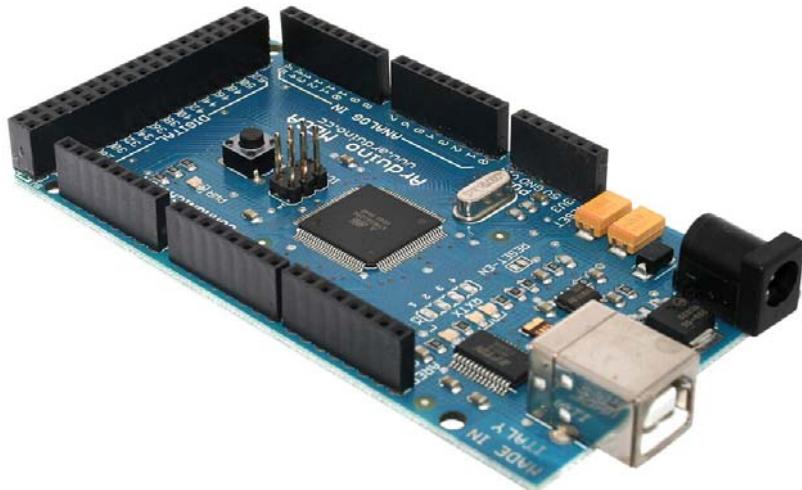
Arduino nano



Arduino uno



Arduino mega



1.5. Arduino Uno haqida ma'lumot

Arduino Uno - bu ATmega328P mikrokontrolleriga asoslangan ochiq manbalni mikrosxemali platforma bo'lib, elektron loyihalar uchun juda mashhur. U 16 MHz chastotada ishlaydi va USB orqali dasturlanadi.

Quvvat bilan ta'minlash usullari

Arduino Unoni quyidagi usullar bilan quvvatlantirish mumkin:

Quvvat manbai	Tavsif
USB port	Kompyuter orqali 5V, 500mA gacha quvvat beradi.
Barrel Jack (DC)	7-12V tavsija etiladi. 6V dan past yoki 12V dan yuqori bo'lsa, muammo bo'lishi mumkin.
VIN pin	Tashqi manbadan 7-12V quvvat berish uchun ishlatiladi.
5V va 3.3V pinlari	Tashqi komponentlarga quvvat berish uchun. 5V - asosiy, 3.3V - past kuchlanishli qurilmalar uchun.
GND pinlari	Yer (ground) ulanishi. Barcha sxemalar umumiylashtirilgan GNDga ega bo'lishi kerak.

Raqamli pinlar (Digital Pins)

Arduino uno da 14 ta raqamli pin mavjud: D0 - D13. D0 (RX) va D1 (TX): Serial aloqa uchun. D3, D5, D6, D9, D10, D11: PWM (Pulse Width Modulation) funksiyasiga ega. Har bir pin input/output sifatida ishlatilishi mumkin.

DigitalWrite(), digitalWrite(), pinMode() funksiyalari orqali boshqariladi. Har bir pin maksimal 40 mA tok bera oladi, lekin tavsiya etilgan - 20 mA.

Analog pinlar (Analog Pins)

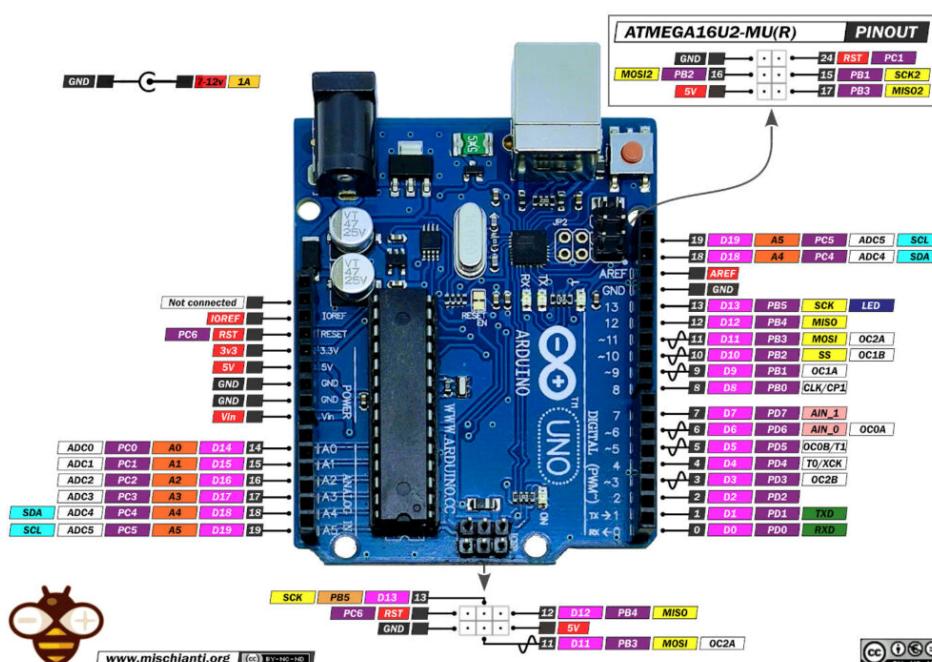
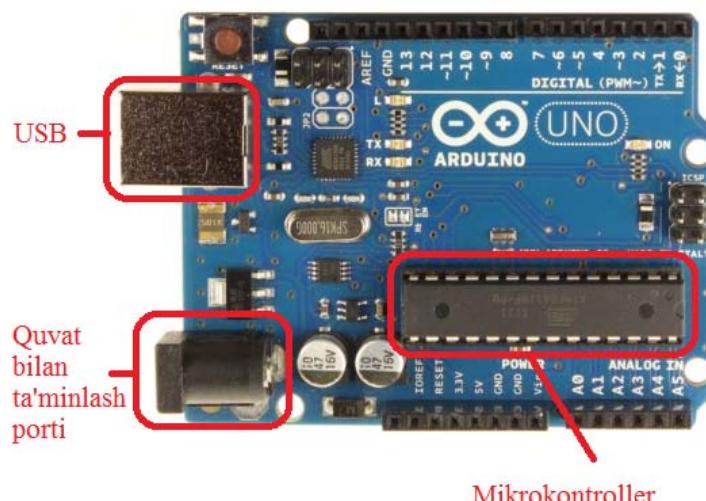
Arduino uno da 6 ta analog pin mavjud: A0 - A5. Bu pinlar 0 - 5V oralig'idaqan analog signalni o'qiydi. ADC (Analog to Digital Converter) orqali 10-bit raqamli qiyamatga aylantiriladi (0 - 1023). Analog pinlar digital pin sifatida ham ishlatalishi mumkin.

Maxsus pinlar

AREF: Analog kirishlar uchun tashqi voltaj ma'lumotini belgilash.

RESET: Mikrokontrollerni qayta ishga tushirish.

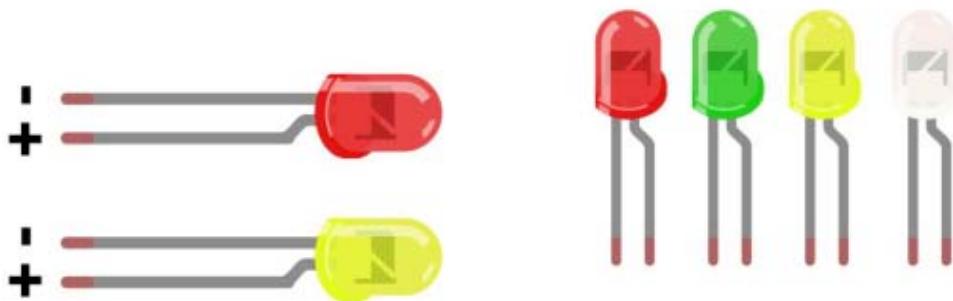
IOREF: Mikrokontroller ishlayotgan voltajni bildiradi.



2-DARS LED YORITGICH

2.1. Umumiy ma'lumot

Ushbu loyiha orqali siz 10 mm LED modulini yoqish uchun Arduino unodan qanday foydalanishni o'rGANISHINGIZ mumkin. Dastur yuklab olinganidan so'ng va simlar ulangandan keyin, LED yoritgich muvaffaqiyatli yonayotganini ko'rasiz.



Agar LED yoritgich yonmasa, simlar to'g'ri ulanganmi yoki yo'qmi - shuni tekshiring. Shuningdek, ulanayotgan Arduino uno pin raqamlari mos kelish - kelmasligini ham tekshiring.

2.2. Ishlash prinsipi

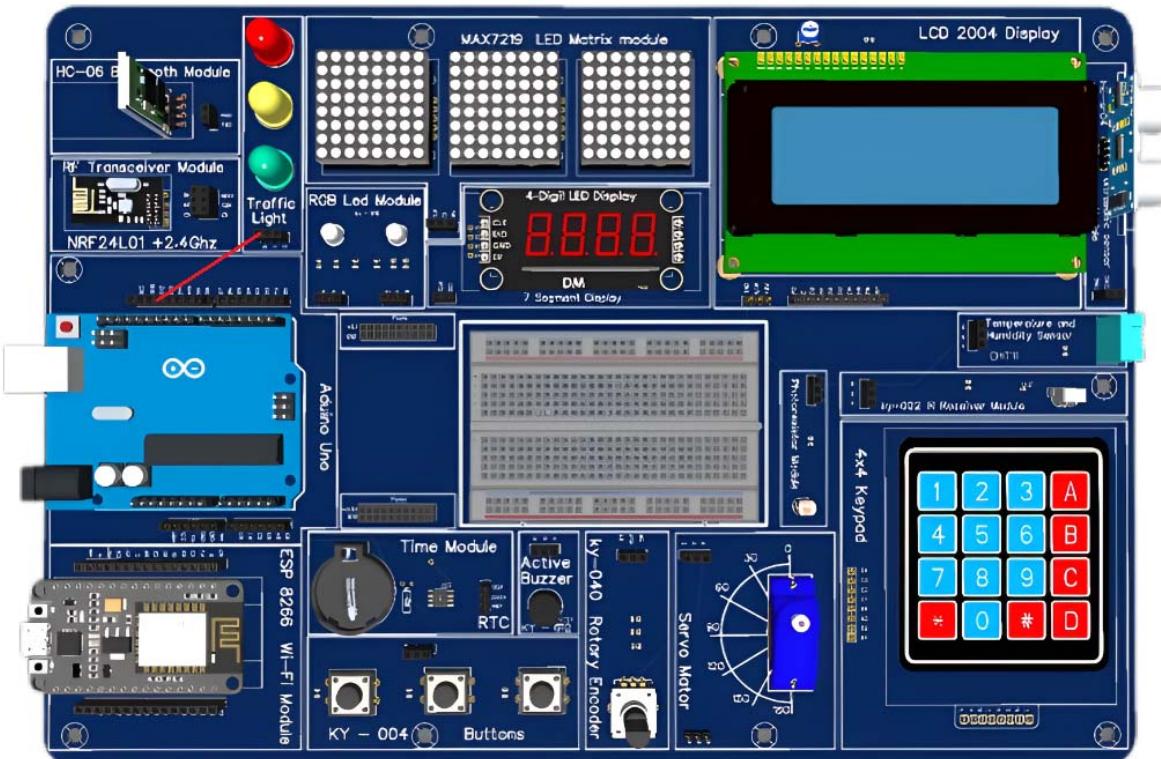
LED elektr energiyasini yorug'lik energiyasiga aylantiradi, va u bir yo'nalishli o'tkazuvchanlikka ega. Shuningdek, taxminan 5V bo'lgan teskari buzilish kuchlanishiga ega. Uning volt-amper xarakteristikasi oldiniga juda keskin bo'ladi, shuning uchun tokni cheklovchi rezistor ketma-ket ulanishi kerak. 5 voltli zanjirda, odatda 400 om atrofidagi rezistor ishlataladi. LEDning ikki pinidan uzunrog'i musbat qutb hisoblanadi. Ulashning ikki usuli mavjud agar LEDning musbat qutbi cheklovchi rezistor orqali Arduinoga, ya'ni I/O portga ulanib, boshqa uchi yerga (GND) ulangan bo'lsa, Arduino chiqishi yuqori holatda (HIGH) bo'lganda LED yonadi, past (LOW) holatda bo'lganda esa o'chadi. LEDning manfiy qutbi Arduinoning I/O portiga ulangan bo'lsa, boshqa uchi tokni cheklovchi rezistor orqali 5V kuchlanishiga ulanadi. Bunday holatda, Arduino chiqishi past (LOW) bo'lsa - LED yonadi, chiqishi yuqori bo'lsa - LED o'chadi.

Agar LED bilan birga rezistor ishlatalmasa, ortiqcha tok oqimi sababli LED darhol ishdan chiqishi mumkin. Bu tok LEDni qizdirib yuboradi va yorug'lik hosil qilinadigan 'birikma'ni buzadi. LEDning qaysi pini musbat, qaysi pini manfiy ekanini aniqlashning ikki usuli mavjud Birinchi usul, musbat qutbdagi pin(oyoqcha) uzunroq bo'ladi. Ikkinci usul, LED ko'rpusida manfiy qutbdagi pin kiradigan joyda

korpusning cheti tekis bo‘ladi. Agar sizda uzun pin yonida tekis chetli LED bo‘lsa, bu pin musbat qutb deb qabul qilishingiz mumkin.

2.3. Ulanish tavsifi

Simlardan foydalanib, mikrokontrollerning D13 pinini Traffic Lights modulining istalgan piniga ulang. Ulanish sxemasi quyidagi rasmda ko‘rsatilgan.



2.4. Dasturiy taminot qismi

LED yoritgich uchun faollashtiruvchi pinni belgilash.

```
#define LED 13 //chiqish pinini belgilash
```

Boshlash Funktsiyasi, 13-pinnni chiqish pini sifatida belgilash.

```
void setup() {
    // buyerga sozlash kodini yozing , u bir marta ishlaydi:
    pinMode(LED, OUTPUT);
    //boshlang'ich sozlash,
    digitalWrite(LED , LOW); // pin qiymati low (pas) bo'ladi.
}
```

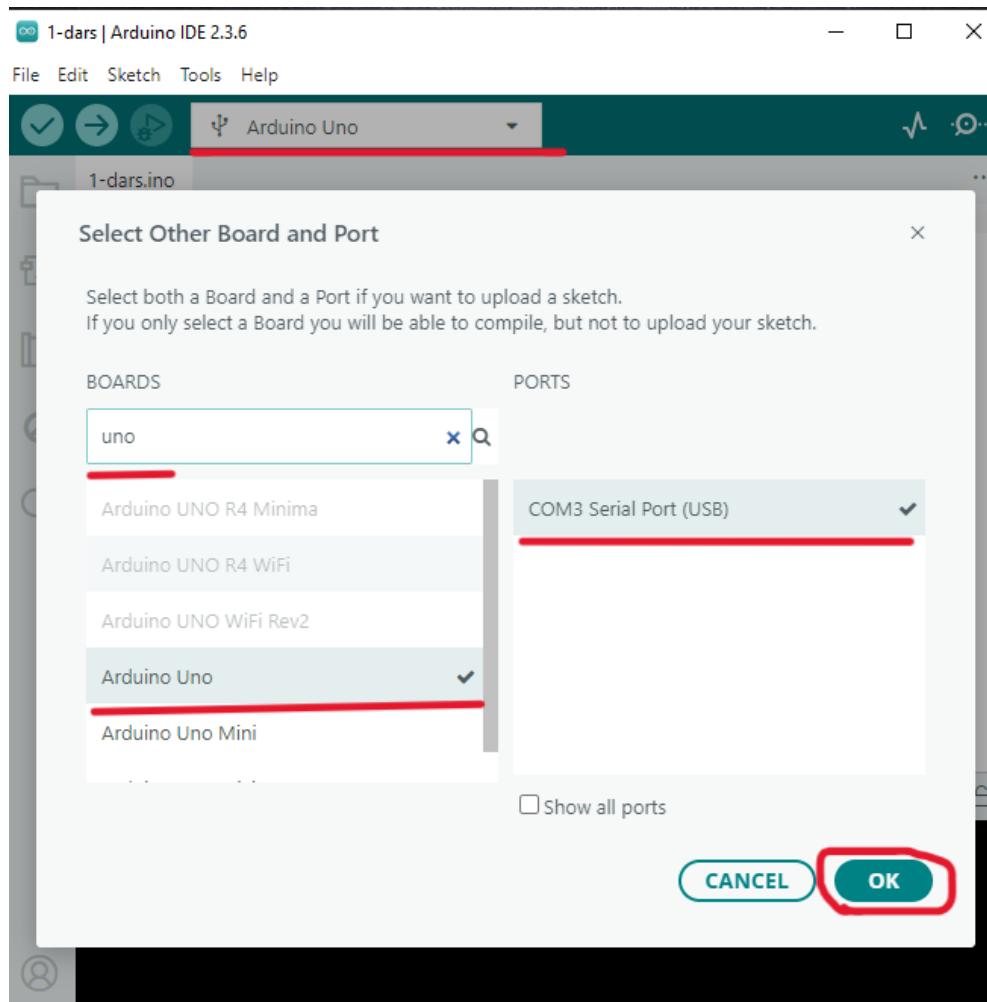
Asosiy funksiyada, 13-pin yuqori (HIGH) darajada chiqish holatiga o‘tkazish.

```
void loop() {
    //Bu yerga asosiy kodingizni yozing, takroran bajariladi:
    /*LED pinning manfiy qutbi oldindan ulab qo'yilganligi sababli,
    faqat Arduino'ning boshqaruv pinidan yuqori darajadagi
    (ya'ni musbat qutb) signal berish kifoya — shunda LED yonadi.*/

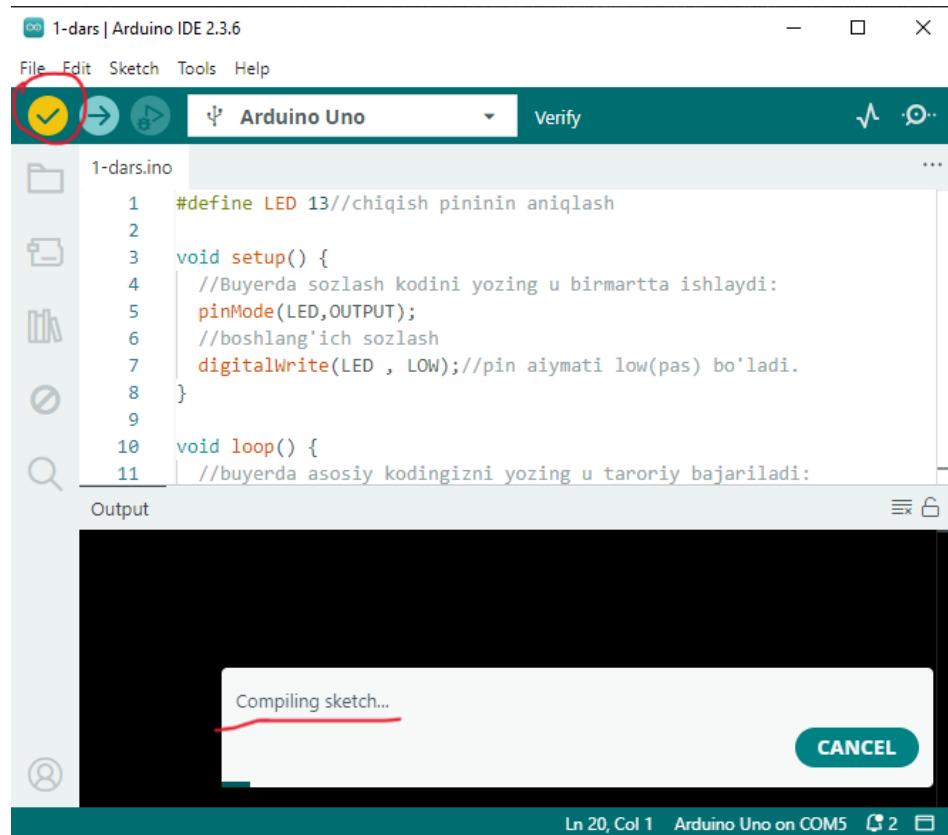
    digitalWrite(LED ,HIGH);
}
```

2.5. Kodni yuklash

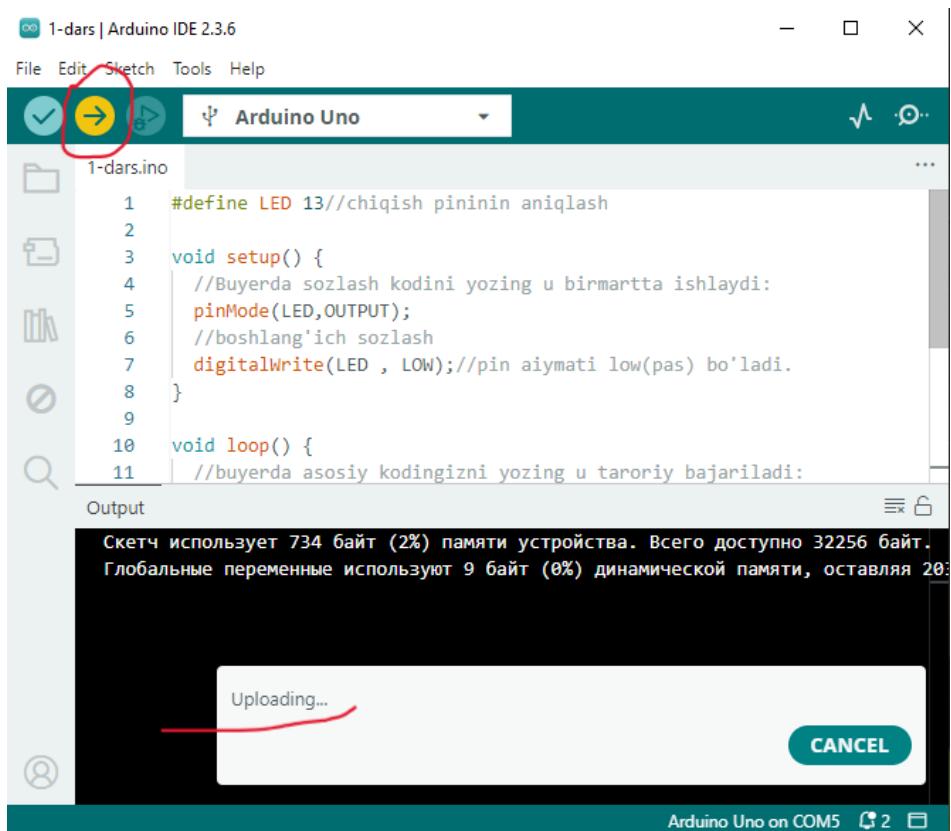
Arduino IDE dasturining yuqori qismidagi usb belgisi ustiga bosing Arduino unoni tanlang. Portni tanlang va ‘OK’ tugmasini bosing.



Kompilatsiya qilishni bosing.



Yuklashni bosing.



3-DARS TUGMA YORDAMIDA LEDNI BOSHQARISH

3.1. Umumiy ma'lumot

Ushbu loyiha orqali quyidagilarni o'rganishingiz mumkin. Buttons modulidagi tugmalardan biri orqali LED yoritgichni yoqishni o'rganasiz. Dastur yuklab olinganidan va simlar to'g'ri ulanganidan so'ng, tugmani bosishingiz kerak. Tugma bosilgach, LED yoritgich yonadi. Uni yana bosganingizda, LED yoritgich o'chadi.

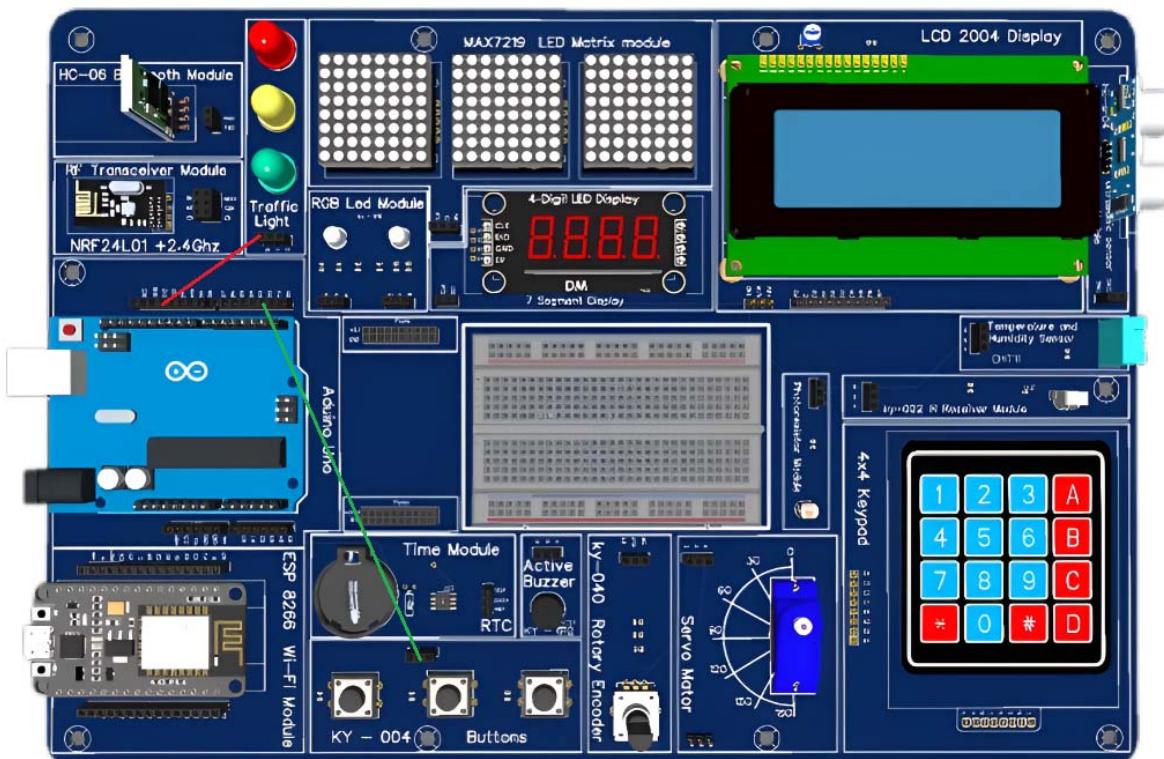
Agar LED yonmasa yoki o'chmasa, quyidagilarni tekshirib chiqing:

Elektr sxemasi to'g'ri ulanganmi?

Arduino unoga ulangan pin raqami to'g'ri tanlanganmi?

3.2. Ulanish tavsifi

Simlardan foydalanib, Arduino unoning D13 pinini Traffic Lights modulining istalgan piniga ulang. Shundan so'ng Arduino unoning D2 pinini buttons modulining istalgan piniga ulang. Bu bilan tugma uchun kerakli ulanish ham bajariladi. Quyidagi rasmida ulanishlar ko'rsatilgan.



3.3. Dasturiy taminot qismi

LED yoritgich uchun faollashtiruvchi pinni belgilash.

```
#define LED 13
#define button 2
```

Global o‘zgaruvchilarni tayinlang, ular tugma va LED holatlarini belgilovchi holat ko‘rsatgichi sifatida ishlataladi. Boshlash funksiyasida zarur bo‘lgan boshlang‘ich sozlamalar kiradi. LED uchun 13-pin chiqish (OUTPUT) ish holatida belgilanadi. Tugma holatini o‘qish uchun 2-pin kirish (INPUT) ish holatida belgilanadi.

```
//Loyiha uchun zarur bo‘lgan data(ma'lumot) o‘zgaruvchilarini belgilang
int key_ok=0;//holat ko‘rsatgichi o‘zgaruvchilarini
int LED_en=0;
void setup() {
    // buyerga sozlash kodini yozing u birmartta ishlaydi:
    pinMode(button, INPUT);//pinlarning ishchi holatini belgilang
    pinMode(LED, OUTPUT);
}
```

Asosiy funksiya quyidagilarni bajaradi. Tugma bosilganligini aniqlash. Tugma bosilgandan so‘ng, holat ko‘rsatkichi orqali u haqiqatan bosilganligini tekshiradi. So‘ng, ushbu holat ko‘rsatkichi asosida LED yonganmi yoki o‘chganmi shuni aniqlaydi.

```
void loop() {
    // Buyerga asosiy kodingizni yozing, takroran bajariladi:
    /*tugma bosilganligini o‘qish
     * tugma holat ko‘rsatgichini o‘qish
     */
    if(digitalRead(button)) {
        if(key_ok){//tugma bosilganligini aniqlang
            key_ok = 0;
            if(LED_en)//oxirgi holat ko‘rsatgichini o‘zgartiring
                LED_en=0;
            else LED_en = 1;
        }
    } else {
        delay(20);//kutish
        if(!digitalRead(button)) key_ok = 1;
    }
    //tugma bosilganda LED pini o‘zgartiriladi
    if(LED_en) digitalWrite(LED,HIGH);
    else digitalWrite (LED,LOW);
}
```

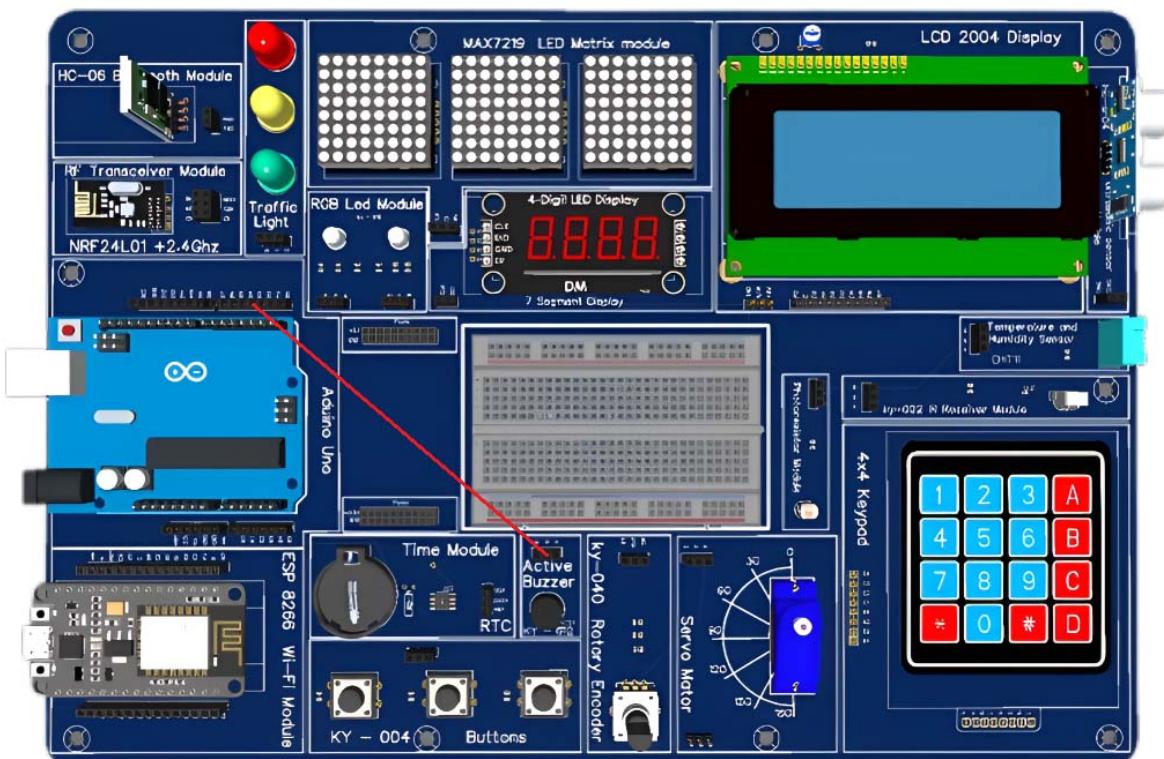
4-DARS: AKTIV BUZZER

4.1. Umumiy ma'lumot

Ushbu loyiha orqali siz Arduino unoni ishlatib aktiv buzzer orqali signal hosilqilishni o'rghanasiz. Bu loyihada buzzer yarim soniya signal chiqarib turadi keyin yana yarim soniya signal chiqarmay turadi va bu jarayan takroriy ravishda bajariladi. Dastur yuklab, barcha simlar to'g'ri ulangach, buzzerdan signal tovushi eshitiladi. Agar buzzer tovush chiqarmasa, avvalo elektr zanjiri to'g'ri ulanganligiga ishonch hosil qiling, shuningdek, Arduino unoga ulangan pin raqami to'g'ri ekanligini tekshiring.

4.2. Ulanish tavsifi

Arduino unoning D3 pinini Active Buzzer modulining 'S' piniga ulang. Ulanish sxemasi quyidagicha bo'ladi.



4.3. Dasturiy taminot qismi

Arduino unoning D3 pinini Aktiv buzzer uchun signal pini sifatida belgilang.

```
#define Buzzer 3
```

Boshlash funksiyasi ishga tushganda 3 pin chiqish pini sifatida belgilanadi.

```
void setup() {
// buyerga sozlash kodini yozing u birmartta ishlaydi:
pinMode(Buzzer,OUTPUT);
}
```

Asosiy funksiya Arduino unoga 500 millisekund yuqori (HIGH) darajadagi chiqish qiymatini beradi keyin esa 500 millisekund past (LOW) darajadagi qiymat beradi. Bu jarayon takrorlanadi. Shuning uchun Aktiv buzzer ovoz chiqaradi.

```
void loop() {
//Buyerga asosiy kodingizni yozing, takroran bajariladi:
digitalWrite(Buzzer,HIGH);
delay(500);
digitalWrite(Buzzer,LOW);
delay(500);
}
```

5-DARS. SVETAFOR YASASH

5.1. Umumiy ma'lumot

Avvalgi darslarning birida biz bir dona LED yoritgichni boshqarishni o'rgangan edik. Endi esa biroz murakkabroq bo'lgan svetofor loyihasini amalga oshiramiz. Aslida bu loyiha oddiy LED yoritgich loyihasining uch xil rangli LED yoritgichli ko'rinishi. Bu luyiha orqali biz svetoforni simulyatsiya qilishga erishamiz.

5.2. Ishlash prinsipi

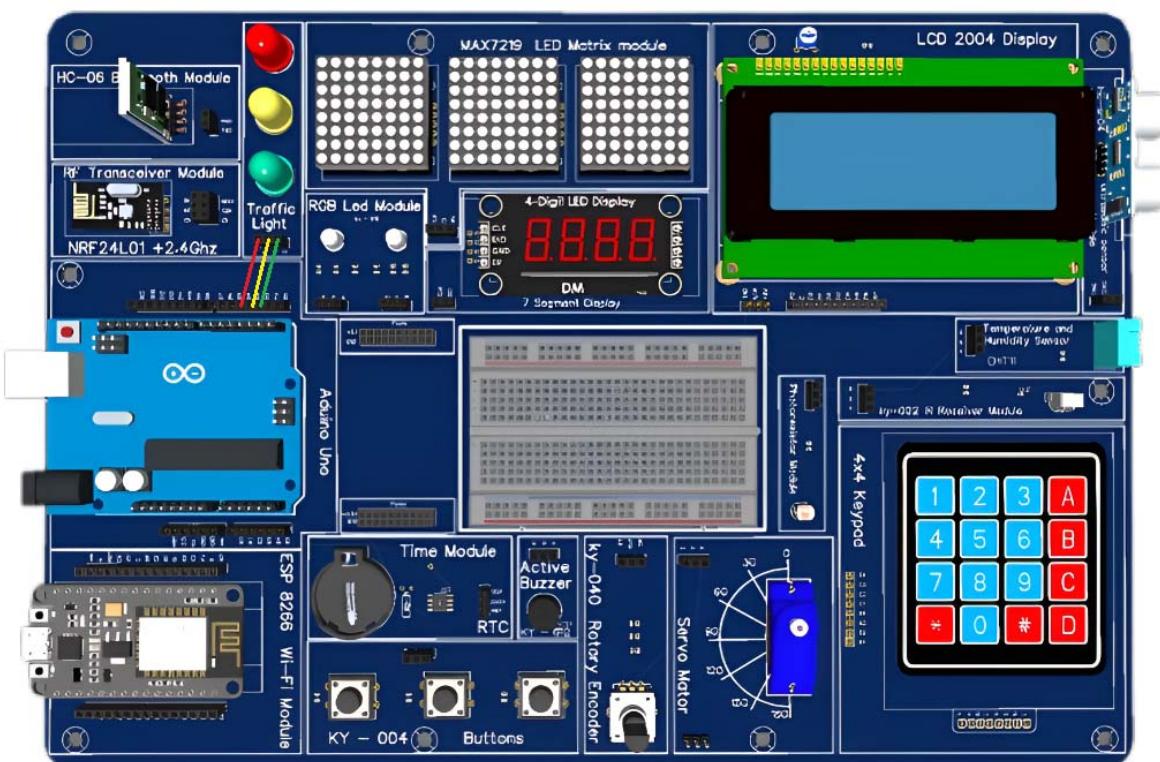
Svetafor chiroqlari yo'l harakati belgilarining muhim qismi bo'lib ular yo'l harakatining asosiy tili hisoblanadi. Yo'l harakati svetofori qizil chiroq (harakat taqiqlangan), yashil chiroq (harakatga ruxsat etilgan) va sariq chiroqdan (ogohlantirish) iborat bo'ladi. U quyidagilarga bo'linadi:

- Motorli transport vositalari uchun svetofor,
- Nomotorli transportlar uchun svetofor,
- Piyodalar o'tish joyi svetofori,

- Harakat yo‘nalishi svetofori,
- Yo‘nalishni ko‘rsatuvchi signal chiroqlari,
- Miltillovchi ogohlantiruvchi signal chiroqlari,
- Avtomobil va temir yo‘l kesishmasi uchun svetoforlar.

Yo‘l harakati svetoforlari - yo‘l xavfsizligini ta’minlash vositalarining bir turi bo‘lib, yo‘l harakati boshqaruvini kuchaytirish, yo‘l transport hodisalarining oldini olish, yo‘llardan harakatlanish samaradorligini oshirish hamda harakat sharoitlarini yaxshilashda muhim vosita hisoblanadi. Bu svetoforlar chorrahalar, T shaklidagi kesishmalar va boshqa yo‘l kesishmalarida qo‘llaniladi. Ular maxsus yo‘l harakati svetofor boshqaruv qurilmasi orqali nazorat qilinadi va transport hamda piyodalarining xavfsiz va tartibli harakatlanishini yo‘naltiradi.

5.3. Ulanish tavsifi



5.4. Dasturiy taminot qismi

Arduino unoning D2, D3, D4 pinlari Traffic light modulining musbal pinlariga ulanganligi sababli bu pinlar ishchi rejimi OUTPUT (chiqish) rejimiga sozlanadi.

```
#define Green 2
#define Yellow 3
#define Red 4

void setup() {
//Buyerga sozlash kodini yozing u birmartta ishlaydi:
pinMode(Green,OUTPUT); //Yashil rang
pinMode(Yellow,OUTPUT); //Sariq rang
pinMode(Red,OUTPUT); //Qizil rang
}
```

Yashil chiroq 5 soniya yonib turishi uchun 2 pinni HIGH (yuqori) holatga qolgan pinlarni esa LOW (pas) holatga sozlaymiz va 5 soniya kutib turamiz.

```
void loop() {
//Buyerga asosiy kodingizni yozing, takroran bajariladi:
digitalWrite(Green,HIGH); // Yashil chiroq yonadi
digitalWrite(Yellow,LOW); // Sariq chiroq o'chadi
digitalWrite(Red,LOW); //Qizil chiroq o'chadi
//Keyingi bosqichga o'tishdan oldin 5 sekund kutamiz
delay(5000);
```

Keyin esa 3 marotaba yashil chiroqni 500 milli sekund davomida yonib 500 milli sekund davomida o'chish kodini yozamiz.

```
//Yashil chiroq har 500 milli sekunda yonadi.
digitalWrite(Green,HIGH);
delay(500);
digitalWrite(Green,LOW);
delay(500);
digitalWrite(Green,HIGH);
delay(500);
digitalWrite(Green,LOW);
delay(500);
digitalWrite(Green,HIGH);
delay(500);
```

Keyin esa 1 sekund davomida sariq chiroq yonadi, qizil va yashil chiroqlar o‘chadi. Keyin esa qizil chiroq 5 sekund yonadi.

```
//Bir sekund sariq chiroq yonadi
digitalWrite(Green,LOW);
digitalWrite(Yellow,HIGH);
digitalWrite(Red,LOW);
delay(1200);
//Qizil chiroq 5 sekund yonadi.
digitalWrite(Green,LOW);
digitalWrite(Yellow,LOW);
digitalWrite(Red,HIGH);
delay(5000);
```

Keyin esa 3 marotaba qizil chiroqni 500 milli sekund davomida yonib 500 milli sekund davomida o‘chish kodini yozamiz.

```
//Qizil chiroq har 500 milli sekunda yonadi
digitalWrite(Red,HIGH);
delay(500);
digitalWrite(Red,LOW);
delay(500);
digitalWrite(Red,HIGH);
delay(500);
digitalWrite(Red,LOW);
delay(500);
digitalWrite(Red,HIGH);
delay(500);
```

6-DARS: FOTOREZISTOR

6.1. Umumiylumot

Fotorezistor bu yarim o‘tkazgichli fotoelektrik effekt asosida ishlovchi komponent bo‘lib, qarshilik qiymati tushayotgan yorug‘likning intensivligiga qarab o‘zgaradi. Yorug‘lik kuchli bo‘lsa qarshilik kamayadi, Yorug‘lik kam bo‘lsa qarshilik oshadi.

Fotorezistorlar odatda quyidagi sohalarda qo‘llaniladi:

- Yorug‘lik darajasini o‘lchashda
- Yorug‘likni avtomatik boshqarishda

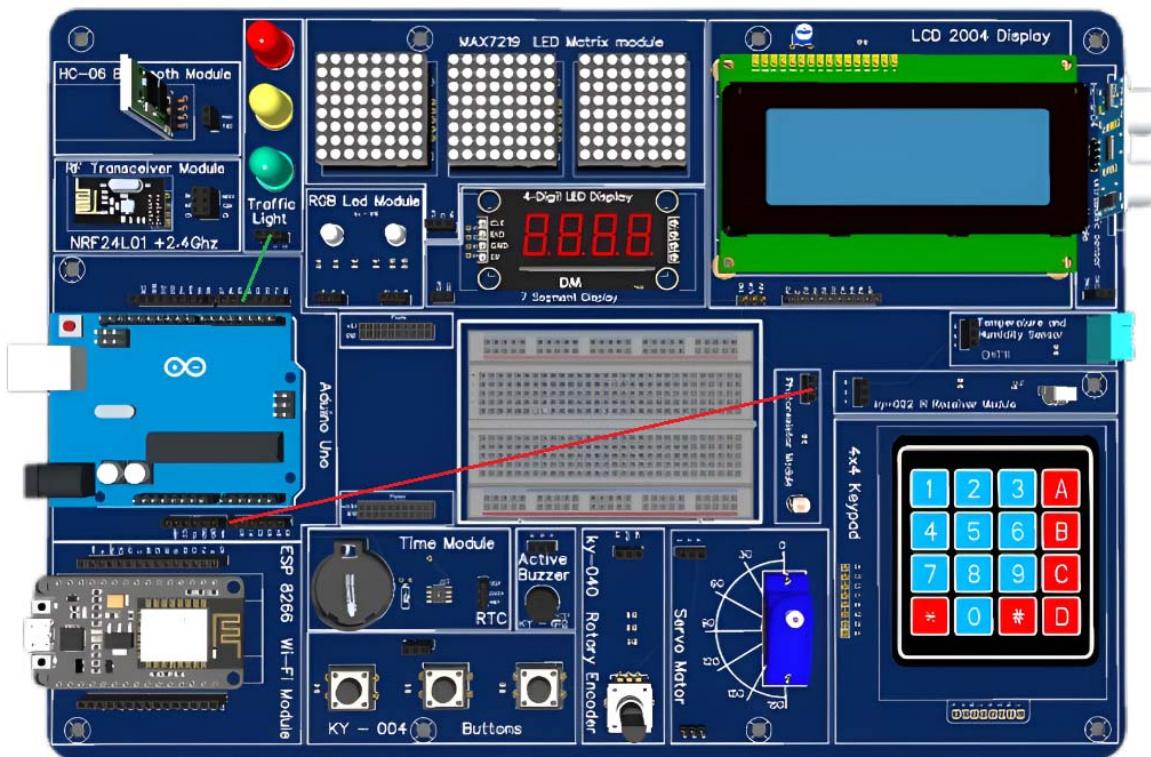
- Fotoelektrik konversiyada (yorug‘likdagi o‘zgarishlarni elektr o‘zgarishlarga aylantirish)

6.2. Ishlash prinsipi

Bu darsimizda fotorezistordan foydalanib yorug‘likni avtomatik boshqarish tizimini loyihalaymiz. Bu tizimda fotorezistoriga yorug‘lik tushsa LED yoritgich o‘chadi agar yorug‘lik tushmasa LED yoritgich yonadi. Bu loyihani aqilli yoritgich dep atashimiz ham mumkin chunki tunda ko‘chalarni yoritib kunduzi Avtomat tarzda o‘chadigan aqilli yoritgichar bilan bir xil ishlaydi. Fotorezistoriga kuchli yorug‘lik tushsa uning qarshiligi kamayadi agar yorug‘lik tushmasa qarshilik kattalashadi. Shuning uchun fotorezistorni Arduino unonig analog piniga ulaymiz. LED yoritgichni esa raqamli pinga ulaymiz.

6.3. Ulanish tavsifi

Arduino unoning A0 pini fotoresistor modulining ‘S’ piniga ulanadi. Arduino unoning D4 pini Traffic light modulining istalgan piniga ulang.



6.4. Dasturiy taminot qismi

Fotorezistor uchun A0 pin LED uchun D4 pinini belgilaymiz.

```
//portlarni belgilash
#define RES_Pin A0 //A0 fotorezistoriga ulanadi
#define LED_Pin 4 //D4 LEDga ulandi.
```

Fotorezistor kam yorug'lik sezganda LED ni yoqadi shuning uchun D4 pinning ishchi holati chiqish (OUTPUT) dep belgilaymiz.

```
void setup() {
    Serial.begin(9600);
    pinMode(LED_Pin, OUTPUT);
}
```

Fotorezistordan ma'lumotni o'qish uchun analog o'qish (analogRead) funksiyasidan foydalaning.

```
void loop(){ //Fotorezistor dan malumot o'qish kodi
    inputValue = analogRead(RES_Pin);
```

So'ngra bu ma'lumotni kunduz yoki tun holatini aniqlash mezoni sifatida qabul qiling. Agar o'qilgan qiymat 500 dan kichik bo'lsa, demak, atrof-muhit qorong'ilashgan. Shunda LED chiroqqa ulangan pinni yuqori signal (HIGH) holatiga o'tkazing va LED chiroqni yoqing.

```
Serial.print("Value=");
Serial.println(inputValue);

//Yorug'lik kamayganda LEDni yonadi.
if (inputValue < 500){
    digitalWrite(LED_Pin, HIGH);
//Yorug'lik ko'payganda LED o'chadi.
} else {
    digitalWrite(LED_Pin, LOW);
}
}
```

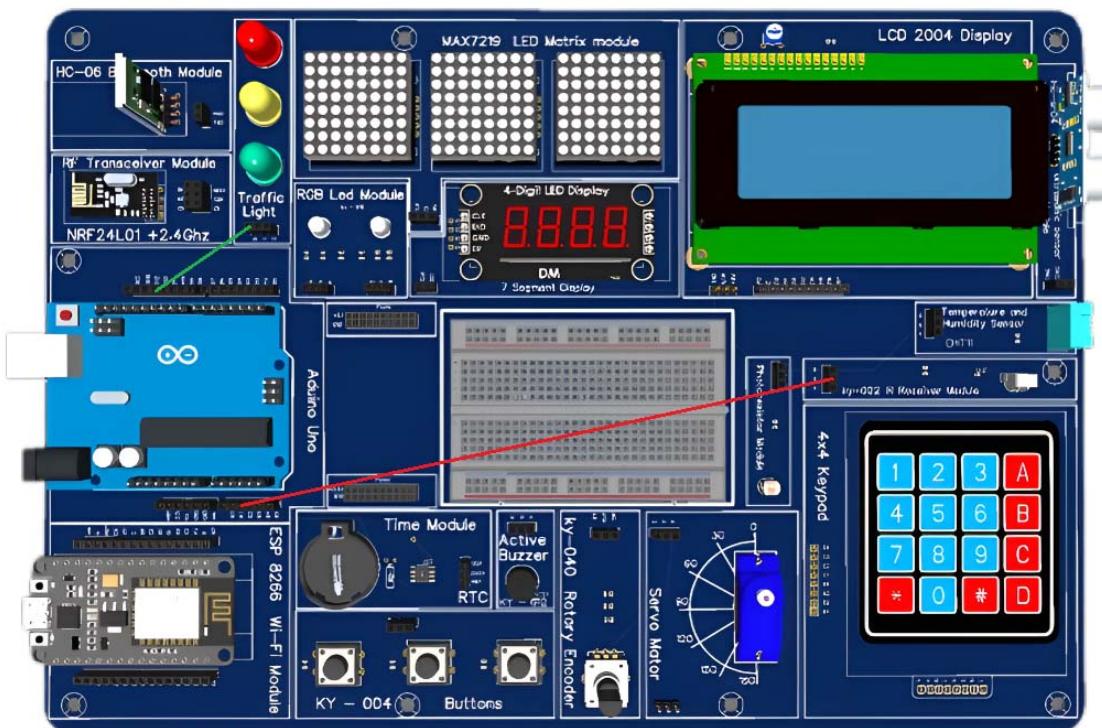
7-DARS. IR SENSOR ORQALI LED YORITGICHNI BOSHQARISH

7.1. Umumiy ma'lumot

Ushbu loyiha orqali siz infraqizil pult yordamida LED yoritgichni masofadan yoqish va o'chirishni o'rghanasiz. Dastur yuklab olingandan so'ng, simlarni ulagningizdan keyin pultdagi 'play' tugmasini bosish orqali LED lampani yoqish va o'chirish mumkin bo'ladi.

Ulanish tavsifi Arduino unoning D2 pinini IR reciver modulining 'S' pinig simlar bilan ulang, D13 pinini esa Traffic light modulining istalgan piniga ulang.

7.2. Ulanish tavsifi



7.3. Dasturiy taminot qismi

Boshlash funktsiyaning va pin ulanishlar avvalgi darslardan bizga ma'lum bo'lganligi sababli, bu yerda takrorlamaymiz. Asosan infraqizil signalni qayta ishlovchi funktsiyaga e'tibor beramiz. Infracqizil signallarni qayta ishslash Infracqizil signal qabul qilish funktsiyasi pulit tugmalarinig (kod) qiymatini qabul qilganda, ushbu qiymatni saqlaydi. Bu qiymatni serial port orqali chop etish (print qilish) va ko'rish mumkin. LED holatini boshqarish Belgilangan kodiga mos tugma kodi aniqlangach, LED holati teskari qilib o'zgartiriladi ya'ni agar LED yoqilgan bo'lsa, o'chiriladi. Agar o'chirilgan bo'lsa, yoqiladi. Shu tarzda LEDni yoqib o'chirish amalga oshiriladi. Loyihada IRremote by shirriff 4.4.3v kutubxonasi ishlatilgan.

```

#include "IRremote.h"
#define HW 2
#define LED 13
int BIT_LED = 0;
IRrecv irrecv(HW);
decode_results results;

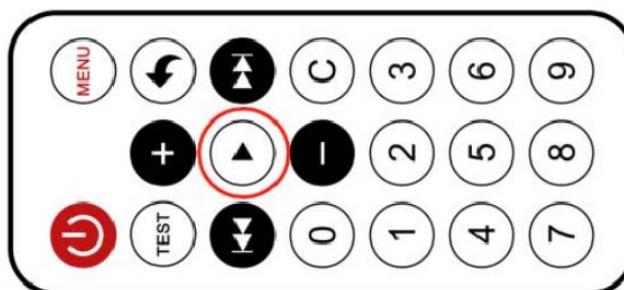
void setup() {
    irrecv.enableIRIn();
    Serial.begin(9600);
    pinMode(LED, OUTPUT);
}

void loop() {
    if (irrecv.decode(&results)) { //Bizda qabul qilingan IR signal bormi
        translateIR();
        Serial.println(results.value, HEX);
        irrecv.resume(); //Keyingi signalni qabul qilish
    }
}

void translateIR(){
//Qabul qilingan IR kod (signal) asosida LED ni yoqamiz
    if (results.value == 0xFFA857){
        if (BIT_LED)BIT_LED = 0;
        else BIT_LED = 1;
//LED holatini o'zgartiramiz
        if (BIT_LED)digitalWrite(LED, HIGH);
        else digitalWrite(LED, LOW);
    }
}

```

Quyidagi suratda qizil aylana bilan belgilanga pulit tugmasi dasturda LED yoritgichni yoqish uchun belgilangan.

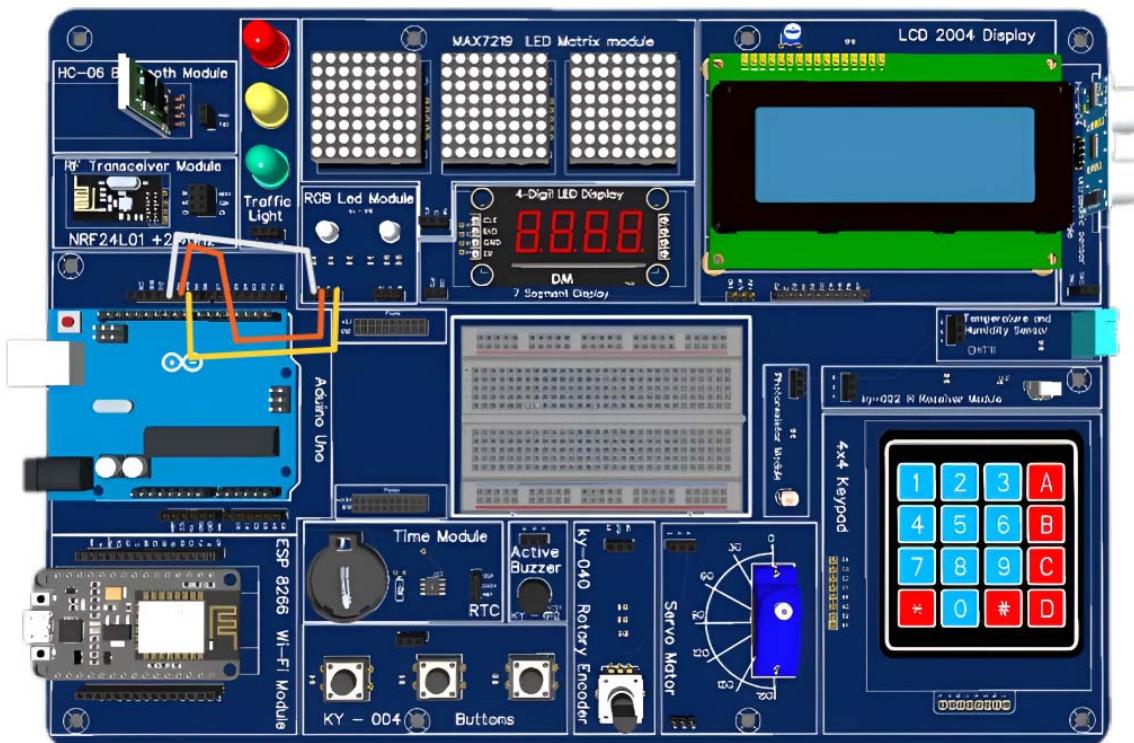


8-DARS: RGB LED MODULINI BOSHQARISH

8.1. Umumiylumot

Bu darsimizda RGB LED modulidan qanday foydalanishni o'rganasiz. RGB LED oddiy LED yoritgich bilan dearli bir xil. RGB LED 3 ta qizil, ko'k, yashil LED yoritgichlarning birlashtirilgan ko'rinishi. RGB LEDning 2 ta turi bor umumiylumot (+) pinli va umumiylumot (-) pinli. RGB LEDda 4 ta pin bo'ladi umumiylumot (-) pinli RGB LEDda 3 ta pin qizil, yashil, ko'k yoritgichlarning musbat pini bo'ladi. 1 ta pini yoritgichlarning manfiy pini hisoblanadi. Umumiylumot pinli RGB LEDda 3 ta manfiy pin va bitta musbat pin bo'ladi. Bu loyihada biz RGB LEDni ketma - ket likda 3 ta rangini ham yoqishni o'rganamiz.

8.2. Ulanish tavsifi



8.3. Dasturiy taminot qismi

Chiqish pinlarini belgiyaymiz va yorqinlik darajasini saqlash uchun val nomli o'zgaruvchi eloq qilamiz.

```
int redPin = 9;
int greenPin = 10;
int bluePin = 11;
int val; // Yorqinlik darajasini saqlaydi
```

Setup funksiyasida pinlarni chiqish pini deb belgilaymiz.

```
void setup() {
    pinMode(redPin, OUTPUT);
    pinMode(greenPin, OUTPUT);
    pinMode(bluePin, OUTPUT);
    Serial.begin(9600);
}
```

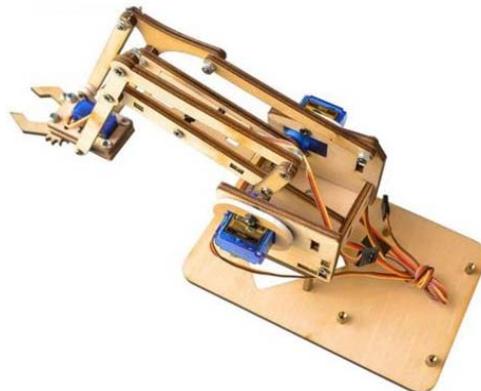
Yorqinlik darajasini o‘zgartirish uchun for siklidan foydalanamiz. Birinchi siklda val qiymatini 255 dan 0 gacha kamaytiramiz ikkinchi siklda esa val qiymatini 0 dan 255 gacha oshiramiz.

```
void loop() {
    for (val = 255; val > 0; val--) {
        analogWrite(redPin, val);
        analogWrite(greenPin, 255 - val);
        //128-val < 0 bo'lsa funksiya qiymatni 0 dep hisoblaydi
        analogWrite(bluePin, 128 - val);
        delay(5);
    }
    for (val = 0; val < 255; val++) {
        analogWrite(redPin, val);
        analogWrite(greenPin, 255 - val);
        //128-val < 0 bo'lsa funksiya qiymatni 0 dep hisoblaydi
        analogWrite(bluePin, 128 - val);
        delay(5);
    }
}
```

9-DARS. SERVO MATOR BOSHQARISH

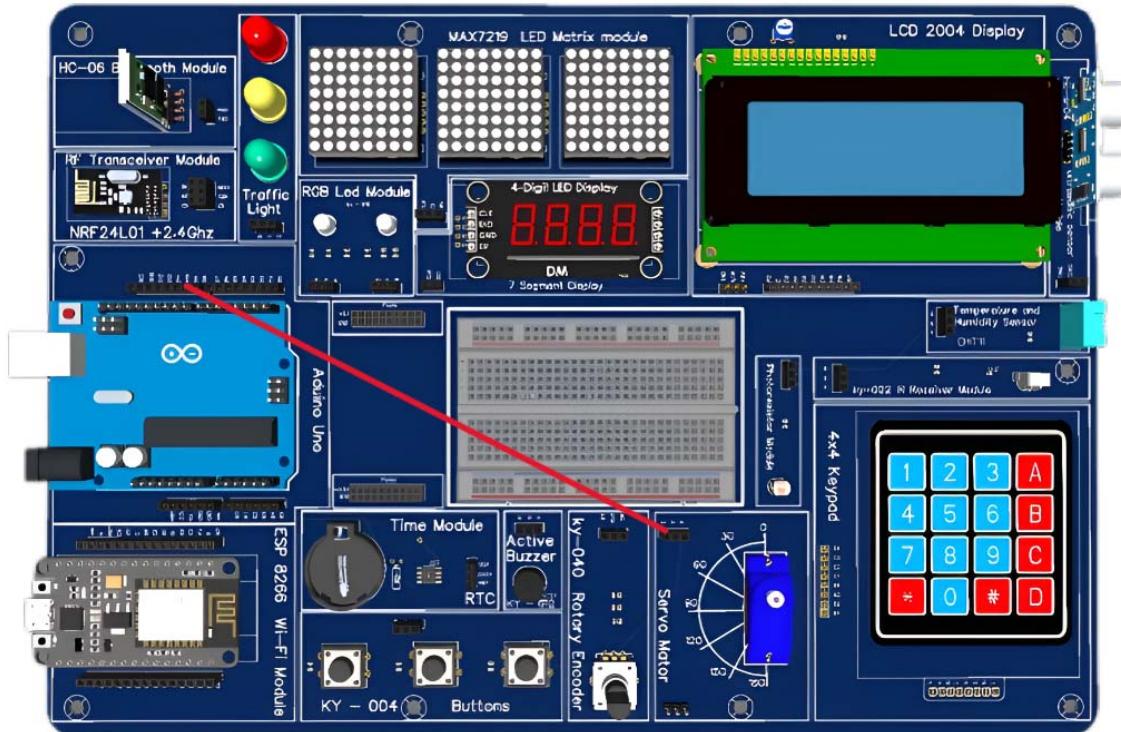
9.1. Umumiylumot

Servo mator burchakni, tezlikni aniq nazorat qilish uchun mo'ljallangan maxsus elektr motoridir. Yopiq zanjirli teskari aloqa tizimidan foydalangan holda ishlaydi, ya'ni u doimiy ravishda sensorlardan olingan real vaqt ma'lumotlari asosida harakatini kerakli maqsadga moslashtiradi. Servo mator robot qo'llar yasashda juda keng qo'llaniladi.



Bu loyihada biz servo matorni 0 dan 180 gradusgacha 30 gradusli qadam bilan buramiz keyin esa ortga 180 gradusga buramiz.

9.2. Ulanish tavsifi



9.3. Dasturiy taminot qismi

Bu loyihamizda biz <Servo.h> kutubxonasidan foydalanamiz. Servo mator ulash uchun Arduino unoning D9 pinidan foydalanamiz. Bizga pinMode() funksiyasi kerak emas o‘rniga <Servo.h> kutubxonasi taqdim qilgan attach() funksiyasidan foydalanishning o‘zi yetarli bo‘ladi. Quyida kutubxona ma’lumotlari qoldirilgan Servo by Michael Margolis 1.2.2v.

```
#include <Servo.h>

Servo myServo;

void setup() {
| myServo.attach(9); // Ulanish pini
}
```

Servo matorni boshqarishda for siklidan foydalamaniz.

```
void loop() {
| for (int pos = 0; pos <= 180; pos+=30) {
| | myServo.write(pos);
| | delay(1000);
| }
| myServo.write(0);
}
```

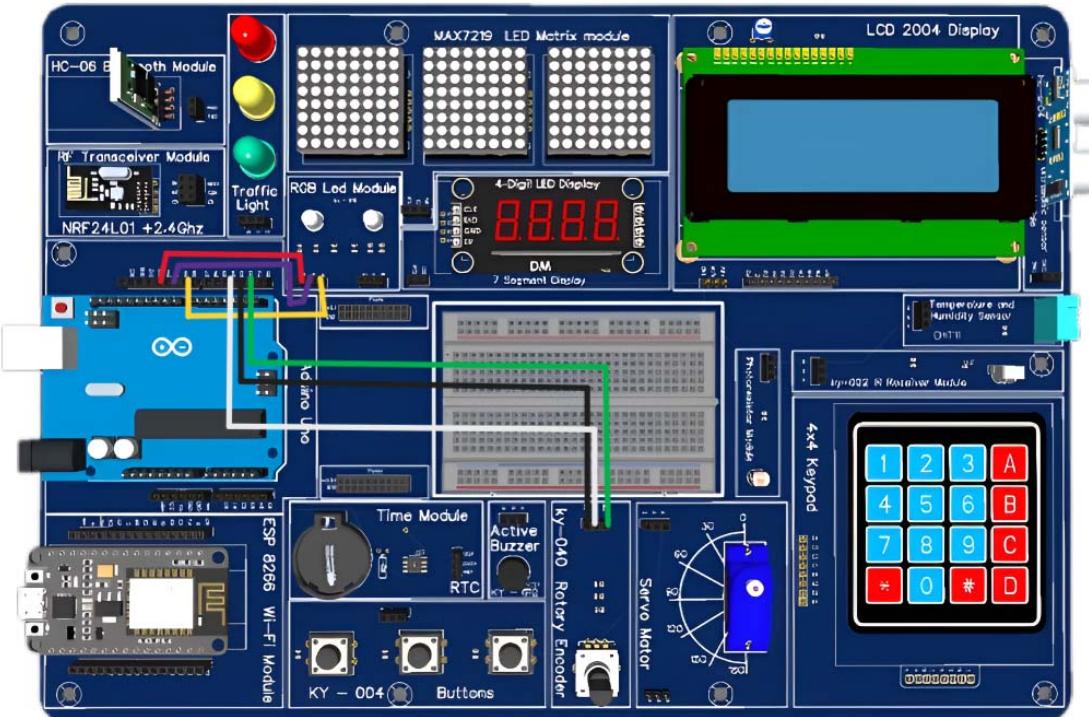
10-DARS: AYLANADIGAN ENKODER

10.1. Umumiylar

Aylanadigan kodlovchi aylanishni elektor signalga aylantirib beradi. U potinsiometrga o‘xshaydi ammao cheksiz aylanishi mumkunligi bilan ajralib turadi. Bu loyiha orqali siz RGB LEDni boshqarish uchun aylanadigan enkoderdan foydalanishni o‘rganishingiz mumkin. Ulanish tavsifi: enkoder va Arduino uno aylanishi quyidagicha CLK - D4, DT - D3, SW - D2. Arduino uno va RGB LED aylanishi quyida keltirilgan. Arduino unoning D9 - R, D10 - G, D11 - B pinlarga aylanadi.

10.2. Ulanish tavsifi

Yuqorida keltirilganlar asosida similar yordamida pinlarni ulab olamiz.



10.3. Dasturiy taminot qismi

Birinchi bo‘lib pinlarni belgilab olamiz.

```
//Enkoder pinlari
int CLK = 4;
int DT = 3;
int SW = 2;
//RGB LED pinlari
int redPin = 9;
int greenPin = 10;
int bluePin = 11;
```

Keyin esa dastur davomida bizga kerak bo‘ladigan o‘zgaruvchilarni belgilaymiz.

```
//yorqinlik darajasi
int red = 0;
int green = 0;
int blue = 0;
//Enkoder aylanishini aniqlash uchun
int counter = 0;
int currentStateCLK;
int lastStateCLK;
//Qaysi rangni boshqarishni aniqlash uchun
int state = 0;
//Tugma bosilganini aniqlash uchun
unsigned long lastButtonPress = 0;
```

Setup funksiyasida pinlarning ishchi holatini belgilaymiz.

```
void setup() {
    pinMode(CLK, INPUT);
    pinMode(DT, INPUT);
    pinMode(SW, INPUT_PULLUP);

    pinMode(redPin, OUTPUT);
    pinMode(greenPin, OUTPUT);
    pinMode(bluePin, OUTPUT);
    Serial.begin(9600);
    lastStateCLK = digitalRead(CLK);
}
```

Aylanadigan kodlovchining aylanish yo‘nalishini aniqlaymiz va qiymatni o‘zgartiramiz. Aylanish yo‘nalishiga qarab counter o‘zgaruvchisining qiymatini o‘zgardiramiz. Qiymat 0 va 255 orasida bo‘lishi uchun cheklov o‘rnatamiz. Keyin esa state o‘zgaruvchisi asosida counter o‘zgaruvchisining qiymatini rang qiymatlaridan biriga beramiz. State o‘zgaruvchisi hozirda qaysi rangni o‘zgartirish kerakligini saqlaydi.

```
void loop() {
    currentStateCLK = digitalRead(CLK);
    if (currentStateCLK != lastStateCLK && currentStateCLK == 1) {
        if (digitalRead(DT) != currentStateCLK) {
            if (counter > 0)
                counter--; //Soat miliga teskari aylanish
        } else {
            if (counter < 255)
                counter++; //Soat miliga teskari aylanish
        }
        if (state == 0)
            red = counter;
        else if (state == 1)
            green = counter;
        else blue = counter;
    }
    lastStateCLK = currentStateCLK;
```

Kodning keyingi qismida enkoder tugmasini bosilganini aniqlaymiz va satate o‘zgaruvchisining qiymatini o‘zgartiramiz. Natijada tugma bosilganda boshqa ranglarning qiymatini o‘zgartirishimiz mumkun bo‘ladi.

```
int btnState = digitalRead(SW);
if (btnState == LOW) { //tugma bosilganini aniqlash
    if (millis() - lastButtonPress > 50) {
        state++;
        state = state % 3;
        if (state == 0)
            counter = red;
        else if (state == 1)
            counter = green;
        else
            counter = blue;
    }
    lastButtonPress = millis();
}
```

Kodning yakuniy qismida olingan natijalar asosida RGB LEDga kuchlanish beramiz.

```
delay(1); //RGB LEDga signal berish
analogWrite(redPin, red);
analogWrite(greenPin, green);
analogWrite(bluePin, blue);
}
```

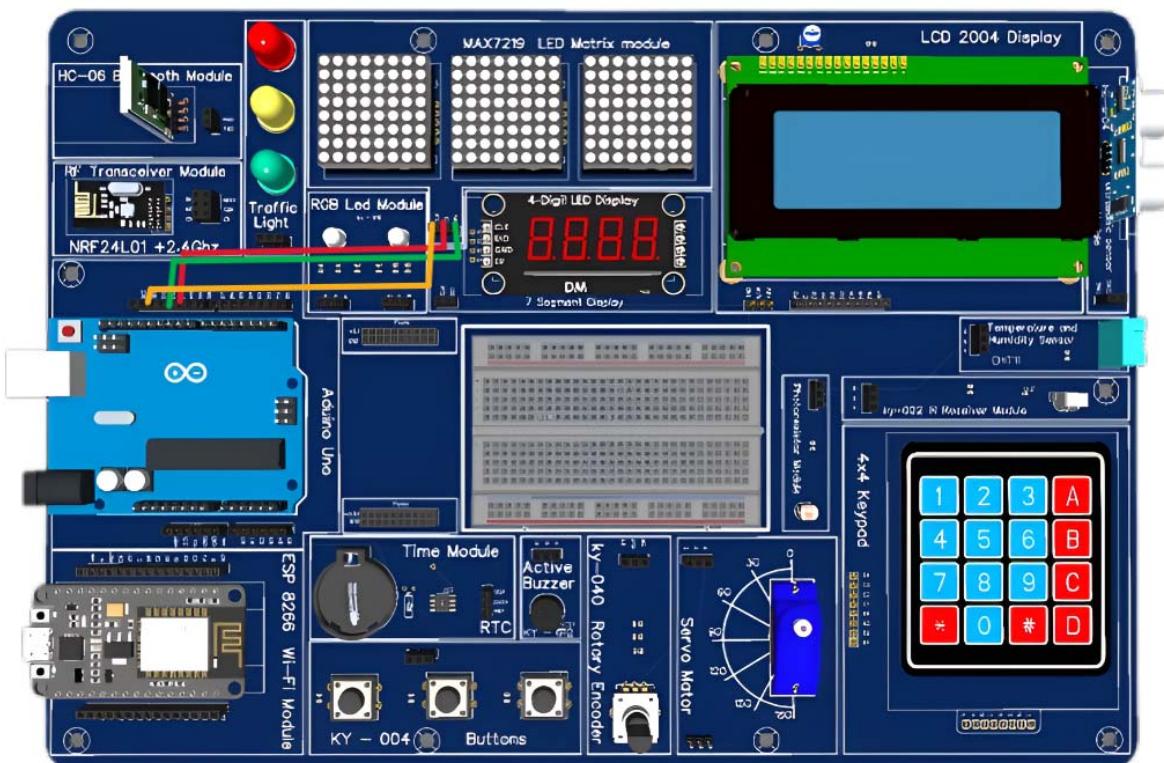
11-DARS. LED MATRITSAGA MATN YOZISH

11.1 Umumiy ma'lumot

LED matritsa moduli 8 qator va 8 ustunda joylashgan 64 ta LEDdan tashkil topgan displaydir. Odatda Matnni harakatlantirish Belgilar va animatsiyalar Sensor ma'lumotlari yoki holat ko'rsatkichlarini ko'rsatish uchun ishlataladi MAX7219 moduli bilan bog'langanda, atigi uchta Arduino pinlari yordamida boshqarish mumkun bo‘ladi.

11.2. Ulanish tavsifi

Biz MAX7219 modulidan foydalanamis shu sababli 3 ta sim ulanishini amalga oshirsak bo‘ldi. Arduino unoning qaysi pinlari LED matritsaning qaysi pinlariga ulanishi kerakligini quyida ko‘rishingiz mukun.



11.3 Dasturiy taminot qismi

Bu loyihada birnechta tashqi kutubxonalaridan foydalananamiz. Bu kutubxonalar Ichida SPI.h kutubxonasini o‘rnatish shart emas chunku u Arduino IDEga o‘rnatilgina bo‘ladi. Qolgan kutubxonalar ma’lumotlari quyida keltirilgan MD_MAX7xx by majicDesigns 3.5.1v, MD_Parola by majicDesigns 3.7.3v

Din pini D11 ga Cs pini D10 ga Clk pini D13 ga ulanadi. Kodda esa faqatgina D10 pinni belgilash kifoya qolgan pinlar Avtomat tarzda kutubxona tomonidan belgilangan.

```
#include <MD_Parola.h>
#include <MD_MAX7xx.h>
#include <SPI.h>

#define HARDWARE_TYPE MD_MAX7XX::FC16_HW
#define MAX_DEVICES 3
#define CS_PIN 10

MD_Parola display = MD_Parola(HARDWARE_TYPE, CS_PIN, MAX_DEVICES);
```

MAX_DEVICES nechta moduldan foydalanishimizni bildiradi. Biz uchta max7219 modulidan foydalanganimiz uchun MAX_DEVICES 3 ga teng bo‘ladi.

Quyidagi kodda setIntensity funksiyasi yorqinlik darajasini boshqaradi. displayScroll funksiyasi esa harakantanuvchi yozuv shakillantirish uchun ishlataladi. Buyerda ‘Arduino uno’ harakatlanuvchi yozuv, PA_LEFT matinni chapga tartiblaashni bildiradi. PA_SCROLL_LEFT esa matinni chapga harakatlantirish vazifasini bajaradi. Undan keyingi 100 harakatlanish tezligini belgilaydi.

```
void setup() {
    display.begin();
    display.setIntensity(5); //Yorqinlik (0-15)
    display.displayClear();
    display.displayScroll("Arduino uno ", PA_LEFT, PA_SCROLL_LEFT, 100);
}
```

Dasturning asosiy qismida yani loop funksiyasida animatsiya tugasa qayta boshlash buyrug‘i yozilgan.

```
void loop() {
    if (display.displayAnimate()) {
        display.displayReset();
    }
}
```

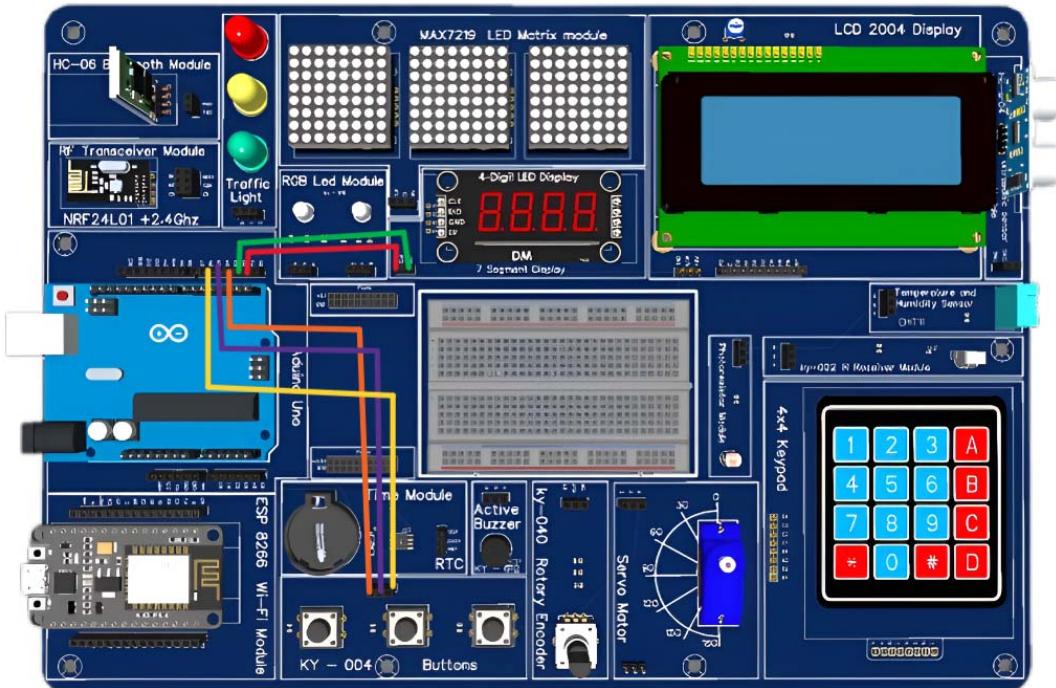
12-DARS: 7 SEGMENTLI DISPLAYDAN SANAGICH YASASH

12.1. Umumiy ma’lumot

Yetti segmentli display moduli ikkita kirish orqali ma’lumot qabul qiladigan va yorqinlikni boshqara oladigan TM1637 mikrosxemasi bilan jihozlangan. Bu modul raqamli soat, taymer, sanagich, harorat ko‘rsatgichi kabi qurilmalar yasash uchun juda qulay hisoblanadi. Bunday displaylar foydalanish uchun qulay va tejamkor hisoblanadi. Bu loyihada sanagich yasaymiz birinchi tugma bosilganda qiymat ko‘payadi. Agar ikkinchi tugma bosilsa qiymat kamayadi. Uchinchi tugma bosilganda qiymat 0 ga tenglashadi.

12.2. Ulanish tavsifi

Quyidagi rasimda 7 segmentli display, tugmalar va Arduino unoning ulanish sxemasi ko‘rsatilgan. Arduino unoning D2 pini displayning CLK pimiga D3 pini DIO piniga D4 pini (+) tugmaga D5 pini (-) tugmaga D6 reset tugmasiga ulanadi.



12.3 Dasturiy taminot qismi

Ushbu dasturda TM1637 Display kutubxonasidan foydalanamiz. Kutubxona ma'lumotlari quyida keltirilgan TM1637 by Avishay Orpaz 1.2.0v. Pinlarni belgilaymiz. Bu yerda BTN_INC qiymatni oshirish tugmasi pini BTN_DEC qiymatni kamaytirish tugmasi pini BTN_RST esa reset yani qiymatni nolga tenglashtirish tugmasi pini.

```
#include <TM1637Display.h>

#define CLK 2
#define DIO 3
#define BTN_INC 4
#define BTN_DEC 5
#define BTN_RST 6

TM1637Display display(CLK, DIO);
int counter = 0;
```

Pinlarning ishchi holatini INPUT_PULLUP qilib belgilaymiz. CLK va DIO pinlarini ishchi holatini belgilash shart emas chunki pinlarning ishchi holati TM1637 kutubxonasi tomonidan avtomat belgilanadi.

```
void setup() {
    //Yorqinlik darajasini belgilash
    display.setBrightness(7);
    display.clear();

    pinMode(BTN_INC, INPUT_PULLUP);
    pinMode(BTN_DEC, INPUT_PULLUP);
    pinMode(BTN_RST, INPUT_PULLUP);
}
```

Dasturning asosiy qismida har bir tugma bosilganida nima vazifani bajarishini yozamiz. Dastur so‘ngida ma’lumotlarni displayga chiqaramiz.

```
void loop() {
    if (digitalRead(BTN_INC) == LOW) {
        counter++;
        delay(200); // Kutish
    }

    if (digitalRead(BTN_DEC) == LOW) {
        counter--;
        delay(200);
    }

    if (digitalRead(BTN_RST) == LOW) {
        counter = 0;
        delay(200);
    }
    //Displayga yozu chiqarish
    display.showNumberDec(counter, true);
}
```

13-DARS. HARORAT VA NAMLIKNI ANIQLASH

13.1. Umumiy ma'lumot

Ushbu darsda Siz harorat va namlik sensori tomonidan o'lchangan ma'lumotlarni ko'rsatish uchun displeyni qanday ulash va undan qanday foydalanishni o'rghanasiz.

Display xususiyatlari

Display LED yoritgichiga ega.

Har bir qatorda 20 ta belgigacha bo'lgan to'rtta qatorda matnni ko'rsata oladi.

Har bir belgi to'g'ri burchaklar shaklida namoyon bo'ladi va ularni tashkil etuvchi piksel tuzilishini ekranda ko'rish mumkin. Ushbu darsda biz harorat va namlikni o'lchash uchun LCD Display va DH11 harorat va namlik sensoridan foydalanishni o'rGANAMIZ.

LCD Display

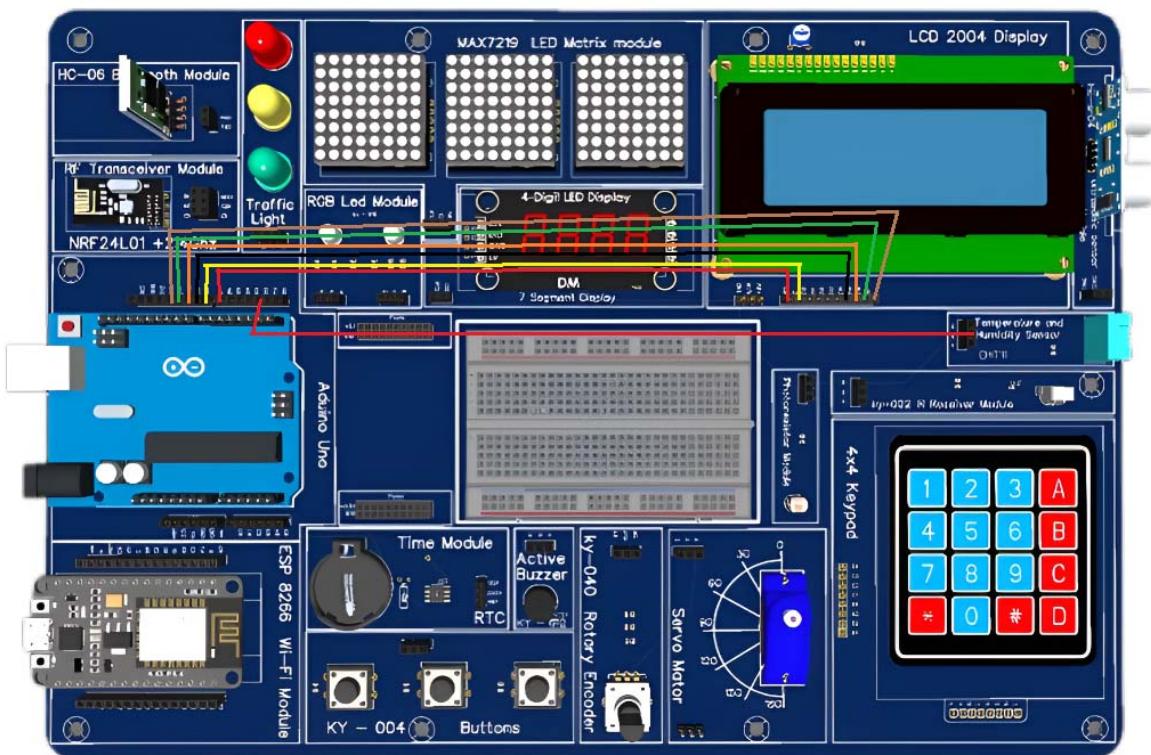


LCD2004 pinlari:

- VSS: Yerga ulanadigan pin.
- VDD: +5V quvvat manbaiga ulanadigan pin.
- VO: LCD1602 displayining kontrastini sozlovchi pin.
- RS: Registr tanlash pini. Bu pin orqali LCD xotirasining qaysi qismiga ma'lumot yozilishini boshqarasiz - ekranga chiqadigan ma'lumot saqlanuvchi data registri yoki buyrug'lar saqlanadigan instruksiya registrini tanlashingiz mumkin.
- R/W: O'qish/yozish holatini belgilovchi pin.
- E (Enable): Past darajali energiya berilganda LCD moduli tegishli buyrug'larni bajarishini ta'minlovchi pin.
- D0-D7: Ma'lumotni o'qish va yozishga xizmat qiluvchi pinlar.
- A va K: LCD displayning orqa yoritgichini boshqaruvchi pinlar (LED backlight).

13.2. Ulanish tavsifi

Quyida LCD Display, harorat va namlik dachigi va Arduino unoning ulanish diagrammasi keltirilgan. LCD Displayning yuqori qismida joylashgan potensiometr yoedamida Displayning yorqinligini sozlashingiz mumkun. Displayning pastki qismida esa R/W pini ham sozlashingiz mumkun bunda R/W pini GNDga yoki +5 volt kuchlanishga ulashingiz mumkun.



13.3 Dasturiy taminot qismi

Harorat va namlik sensorining (DHT11) hamda LCD displeyning kutubxonasi dasturga qo'shish kerak. Arduino unonig D2 pini temperature and humidity sensor modulining 'S' piniga ulanadi. LCD display qaysi pinlari Arduino unoning qaysi pinlariga ulanishi quyida keltirilgan.

Arduino uno	D7	►	RS	LCD Displey
	D8	►	E	
	D9	►	D4	
	D10	►	D5	
	D11	►	D6	
	D12	►	D7	

Bu loyihada ishlataligan kutubxonalar ma'lumotlari quyida keltirilgan DHT sensor library by Adafruit 1.4.6v Wire.h va LiquidCrystal.h kutubxonalarini o'rnatish shart emas chunku ular oldindan o'rnatib qo'yilgan.

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

//Pinlarni belgilaymiz
#define DHTPIN 2
//Sensor turini belgiyaymiz
#define DHTTYPE DHT11
//Boshlang'ich sozlamalar
DHT dht(DHTPIN, DHTTYPE);
LiquidCrystal lcd(7, 8, 9, 10, 11, 12);
```

DHT11 sozlandi va ishga tushurildi LCD display sozlandi va ishga tushurildi.

```
void setup() {
    Serial.begin(9600);
    dht.begin(); //DHT ishga tushuramiz
    Wire.begin();
    lcd.begin(20, 4);
    lcd.clear();
}
```

Sensordan olingan malumotlar asosida harorat va namlik serial portiga (tekshirish uchun) va LCD Displayga chiqariladi.

```
void loop(){
// 0.5 sekud kutamiz
//bu sensor haroratni o'lchab ilishi uchun
delay(500);
//Harorat yoki namlikni o'qish uchun
// 250 millisekund kerak bo'ladi
float h = dht.readHumidity(); //Namlikni o'qish
float t = dht.readTemperature(); //Haroratni o'qish
Serial.print("Namlik: "); //Namlik
Serial.println(h);
Serial.print("Harorat: "); //Harorat
Serial.print(t);
Serial.println(" °C ");
```

Ma'lumotlarni LCD Displayga chiqarish kodlari quyida keltirilgan.

```
//Display cursorini 1 satr 0 belgiga sozlaymiz
lcd.setCursor(0, 0);
lcd.print("Harorat:");
lcd.setCursor(5, 0);
lcd.print(t);
//Display cursorini 2 satr 0 belgiga sozlaymiz
lcd.setCursor(0, 1);
lcd.print("Namlik:");
lcd.setCursor(5, 1);
lcd.print(h);
}
```

14-DARS. ULTIRASONIK ORQALI MASOFANI O'LCHASH

14.1. Umumiyligi ma'lumot

Ushbu darsda Siz ultira tovushli masofani aniqlash sensori tomonidan o'lchanigan masofani ko'rsatish uchun displeyni qanday ulash va undan qanday foydalanishni o'rjasiz.

Display xususiyatlar

Display LED yoritgichiga ega.

Har bir qatorda 20 ta belgigacha bo'lgan to'rtta qatorda matnni ko'rsata oladi.

Har bir belgi to'g'ri burchaklar shaklida namoyon bo'ladi va ularni tashkil etuvchi piksel tuzilishini ekranda ko'rish mumkin.

HC-SR04 ultratovush datchigi moduli 2 sm dan 400 sm gacha bo'lgan masofani kontaktsiz o'lchanish funksiyasini taqdim etadi, o'lchanish aniqligi esa 3 mm gacha yetishi mumkin. Modul quyidagilarni o'z ichiga oladi: ultratovush yuboruvchi, qabul qiluvchi va nazorat qilish sxemasi. Ishlashning asosiy prinsipi:

Kiritish chiqarish trigger orqali kamida 10 mikrosoniya davomida yuqori darajali signal yuborilganda, modul avtomatik ravishda 8 marta 40 kHz chastotali ultratovush signallarini jo'natadi va qaytuvchi impuls signali bor-yo'qligini aniqlaydi.

Agar signal qaytsa, yuqori darajadagi chiqish signali davomiyligi - bu ultratovush yuborilganidan to qaytgunicha bo'lgan vaqtini bildiradi.

Sinov masofasi = (yuborishga ketgan vaqt × tovush tezligi (340 m/s)) / 2.

Masofa o'lchanishni boshlash uchun siz faqat trigger kirishiga 10 mikrosoniyalik qisqa impuls yuborishingiz kerak. Shundan so'ng modul 40 kHz

chastotada 8 siklli ultratovush to‘lqinini yuboradi va qaytgan aks sado signalini hosil qiladi.

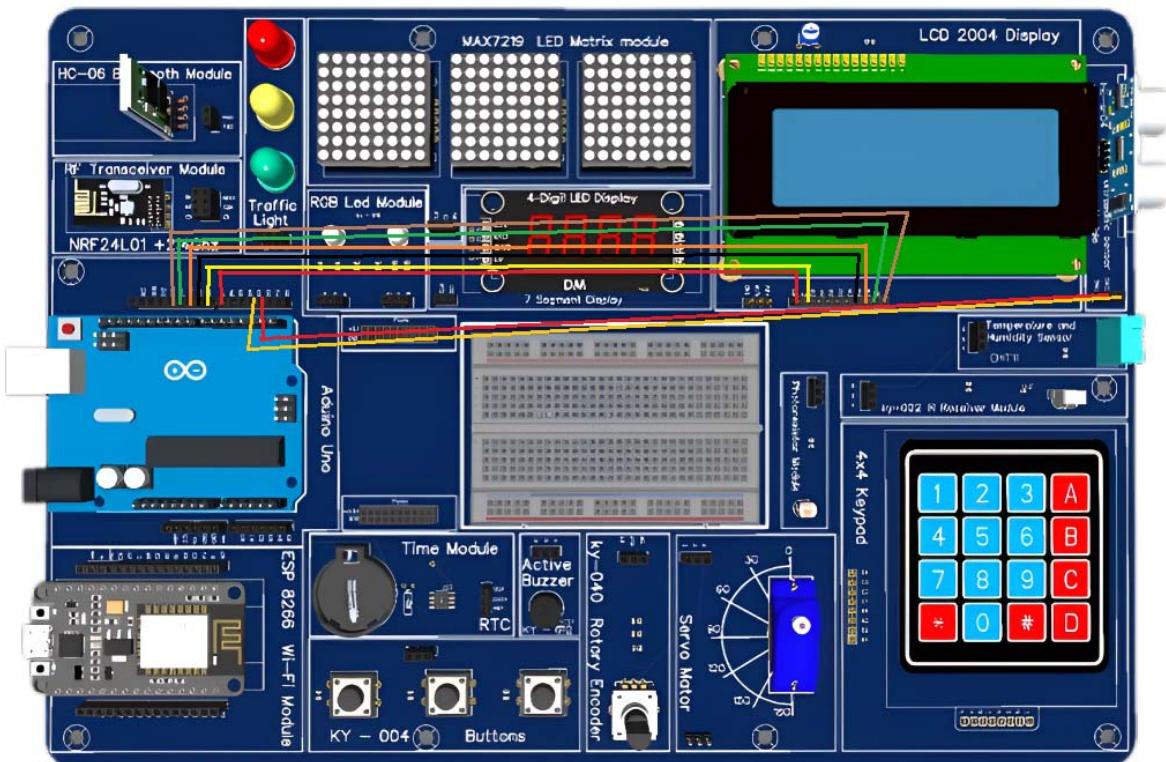
Echo - bu masofa obyektidir, u impulsning kengligi bilan bog‘liq bo‘lib, masofa bilan proporsionaldir. Siz masofani trigger signali yuborilgan va echo signali qabul qilingan oradagi vaqt intervali orqali hisoblappingiz mumkin.

Formula:

$$\text{masofa} = (\text{yuborishga ketgan vaqt} \times \text{tezlik (340 m/s)}) / 2$$

Trigger signalining aks sado signaliga ta’sir qilmasligi uchun, o‘lchov siklini 60 ms dan ortiq qilish tavsiya etiladi.

14.2. Ulanish tavsifi



HC-SR04 ultratovush datchigi moduli tomonidan o‘lchangan ma’lumotlar Arduino uno platasida qayta ishlanadi va LCD ekranda ko‘rsatiladi.

14.3. Dasturiy taminot qismi

LCD display va ultira tovush sensor uchun kutubxonalar qo‘shamiz. Bu loyihada bizga Wire.h va LiquidCrystal.h kutibxonalari kerak bo‘ladi avvalgi darsdan ma’lumki bu kutubxonalarni o‘rnatish shart emas ular Arduino IDE dasturida oldindan o‘rnatilgan.

Quyida dasturning pinlarni sozlash qismi keltirilgan.

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

//ehco:D3 trig:D2
#define Trig 2
#define Echo 3

LiquidCrystal lcd(7, 8, 9, 10, 11, 12); //LCD pini
```

Serial port va ekranni sozlaymiz hamda HC-SR04 ultratovush datchigi modulini sozlaymiz.

```
void setup() {
    Serial.begin(9600);
    Wire.begin();
    lcd.begin(20,4); // LCD 2004 in sozlash
    lcd.clear(); // LCD ekranni tozalash
    pinMode(Trig,OUTPUT); //Tring pinni chiqish qilib sozlimiz
    pinMode(Echo,INPUT); //Echo pinni kirish qilib sozlaymiz
}
```

HC-SR04 ultratovush datchigi moduli tomonidan o‘lchangan ma’lumotlarni olamiz va ularni serial porti oynasida hamda LCD Displayda ham ko‘rsating.

```
void loop(){
    int dis = GetDistance(); //Ultratovush ma’lumotlarini DIS ga saqlash.
    Serial.print(dis);
    Serial.print("cm");
    Serial.println();
    //Display kursorini 0 qator 0 belgiga sozlash
    lcd.setCursor(0, 0);
    lcd.print(dis); //Masofa malumotini ko'rsatish
    if(dis<10){
        lcd.setCursor(1, 0);
        lcd.print(" ");
    }else if( (dis>9) && (dis<100) ){
        lcd.setCursor(2, 0);
        lcd.print(" ");
    }
    lcd.setCursor(5, 0);
    lcd.print("cm:");
    delay(200);
}
```

Quyida getDistance funksiyasi kodi keltirilgan. By funksiyada masofani o'qish algoritmi yozilgan.

```
float GetDistance(){ //Ultiratovush ma'lumotini o'qish
    float distance;
    digitalWrite(Trig, LOW);
    delayMicroseconds(2);
    digitalWrite(Trig, HIGH);
    delayMicroseconds(10);
    digitalWrite(Trig, LOW);
    distance = pulseIn(Echo, HIGH);
    return distance;
}
```

15-DARS: VAQT MODULI ORQALI SOAT YASASH

15.1. Umumiylumot

Ko'plab elektron qurilmalarda amallarni vaqtga asoslanib bajarish zarur bo'ladi. Asosiy tizim ishlamay qolgan hollarda ham, kompyuterlar, mobil telefonlar kabi qurilmalar uchun vaqt va sanani hisoblashni to'xtatmaslik kerak. Shu sababli vaqt moduli (RTC Real Time Clock) qo'llaniladi.

Ushbu bo'limda siz RTC DS1307 moduli va Arduino unodan foydalanib, LCD ekran orqali vaqt ko'rsatuvchi qurilma yasashni o'rganasiz.

Vaqt moduli DS1307 haqida umumiylumot DS1307 - bu past quvvat sarf qiluvchi, 56 bayt hajmga ega bo'lgan doimiy xotira (non-volatile RAM) bilan jihozlangan, to'liq BCD (Binary-Coded Decimal) formatidagi soat-kalendar funksiyasiga ega real vaqt chipi hisoblanadi. Manzil va ma'lumot uzatish ikki simli ikki yo'nalishli seriiali shina orqali amalga oshiriladi. Ushbu chip soniyalar, minutlar, soatlar va boshqa vaqt birligi haqidagi ma'lumotlarni taqdim etadi. Oy kunlari avtomatik tarzda moslashtiriladi va kabisa yili (leap year) kcompensatsiyasi funksiyasi ham mavjud. Real Time Clock (RTC) tizimi - bu joriy vaqtini kuzatib boruvchi tizim bo'lib, aniq vaqtini saqlab turishi zarur bo'lgan har qanday qurilma bilan ishlatilishi mumkin. Siz RTC tizimisiz ham aniq vaqtini kuzatishingiz mumkin, ammo RTC quyidagi muhim afzallikkarga ega:

RTC tizimining afzalliklari:

Tizim vaqtini hisoblash: Bu juda muhim xususiyat, chunki ko'plab holatlarda CPU sensor ma'lumotlarini qabul qilish kabi nozik vazifalarni bajaradi. Agar RTC

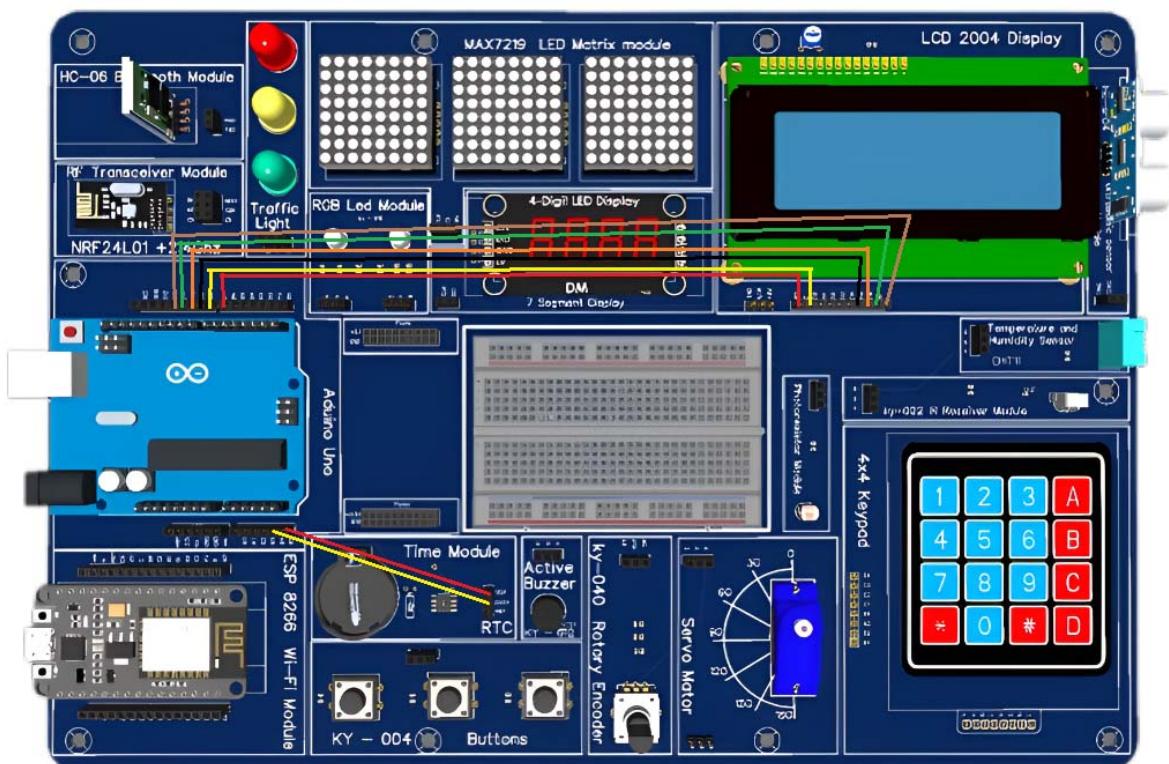
ishlatilmasa, CPU vaqtni ham kuzatib borishi kerak bo‘ladi, bu esa uning asosiy vazifasini buzishi mumkin.

Zaxira quvvat manbasi mavjud: RTC odatda CR2025 turidagi batareyalar yordamida zaxira quvvatga ega bo‘ladi, shuning uchun asosiy quvvat o‘chganida ham ma’lum vaqt davomida ishlashda davom etadi.

Kristall tebranishi - 32.768 kHz: Nega aynan 32,768? Chunki bu son 2^{15} ga teng bo‘lib, 1 soniyani oson hosil qilish imkonini beradi. Bunday kristall kichik bo‘ladi, o‘rtacha kenglikka ega, past quvvat talab qiladi va 32.768 kHz chastota talablarni to‘liq qondiradi.

Tezlik vs kuchlilik: Chastota qanchalik yuqori bo‘lsa, kristall shunchalik zaif bo‘ladi; chastota past bo‘lsa esa, quvvat sarfi ortadi.

15.2. Ulanish tavsifi



15.3. Dasturiy taminot qismi

LCD display va DS1307 uchun kutubxonalarni o‘rnatish shart emas chunki ular oldindan o‘rnatilgan. LCD display va Arduino uno ulanadigan pinlarini belgilang bu pinlar raqamli portlar 7, 8, 9, 10, 11, 12 hisoblanadi.

(Eslatma): Kodni yuklashdan oldin mavjud ‘DS1307_Write.ino’ faylidagi kodni yuklang shunda vaqt moduli aniq vaqtga sozlanadi. Kod muvaffaqiyatlidir.

yuklangach, "15-dars.ino" faylini ham yuklang. Yuklash tugallangach, mahalliy joriy vaqt ekranda to‘g‘ri ko‘rsatiladi.

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

// Kutubxonalar va pinlarni elon qiling
LiquidCrystal lcd(7, 8, 9, 10, 11, 12);
#define ADDRESS_DS1307 0x68
```

Dastur ishga tushishidan oldin, ba’zi boshlang‘ich sozlamalarni belgilash kerak: serial portning tezligini va displeyning sozlamalarini sozlang.

```
void setup() {
    Wire.begin();
    Serial.begin(9600);
    lcd.begin(20, 4);
    lcd.clear();
}
```

DS1307 funksiyasini chaqiring: Siz avval sanani DS1307 ichiga yozishingiz, so‘ng esa ma’lumotni o‘qishingiz kerak bo‘ladi.

```
void DS1307() {
    //Vaqtni o'qish
    Wire.beginTransmission(ADDRESS_DS1307);
    Wire.write(0x00);
    Wire.endTransmission();
    Wire.requestFrom(ADDRESS_DS1307, 7);
    if (Wire.available() >= 7) {
        for (int i = 0; i < 7; i++) {
            timeBcd[6-i] = Wire.read();
        }
    }
}
```

Quyida Sana va vaqtni serial portiga yozish kodlari keltirilgan.

```
//Serial portga yozish
Serial.print("20");
Serial.print(timeBcd[0], HEX);
Serial.print("/");
Serial.print(timeBcd[1], HEX);
Serial.print("/");
Serial.print(timeBcd[2], HEX);
Serial.print(" ");
Serial.print(BcdToDay(timeBcd[3]));
Serial.print(" ");
Serial.print(timeBcd[4], HEX);
Serial.print(":");
Serial.print(timeBcd[5], HEX);
Serial.print(":");
Serial.print(timeBcd[6], HEX);
Serial.println();
```

Quyida vaqtini LCD Displayga chiqarish kodi keltirilgan.

```
//LCD Displayga yozish
lcd.setCursor(0, 0);
lcd.print("20");
lcd.setCursor(2, 0);
lcd.print(timeBcd[0], HEX);
lcd.setCursor(4, 0);
lcd.print("/");
lcd.setCursor(5, 0);
lcd.print(timeBcd[1], HEX);
lcd.setCursor(7, 0);
lcd.print("/");
lcd.setCursor(8, 0);
lcd.print(timeBcd[2], HEX);
lcd.setCursor(10, 0);
lcd.print("Hafta:");
lcd.setCursor(15, 0);
lcd.print(timeBcd[3], HEX);
lcd.setCursor(0, 1);
lcd.print(timeBcd[4], HEX);
lcd.setCursor(2, 1);
lcd.print(":");
```

```

lcd.setCursor(3, 1);
lcd.print(timeBcd[5], HEX);
lcd.setCursor(5, 1);
lcd.print(":");
lcd.setCursor(6, 1);
lcd.print(timeBcd[6], HEX);
}

```

Hafta kunini raqamdan hafta kuniga aylantirish algoritmi.

```

// Ikkilik kodni hafta kuniga o'girish
String BcdToDay(byte val) {
    String res;
    switch(val) {
        case 1: res = "Yakshanba"; break;
        case 2: res = "Dushanba"; break;
        case 3: res = "Seshanba"; break;
        case 4: res = "Chorshanba"; break;
        case 5: res = "Payshanba"; break;
        case 6: res = "Juma"; break;
        case 7: res = "Shanba"; break;
        default: res = "Xato!";
    }
    return res;
}

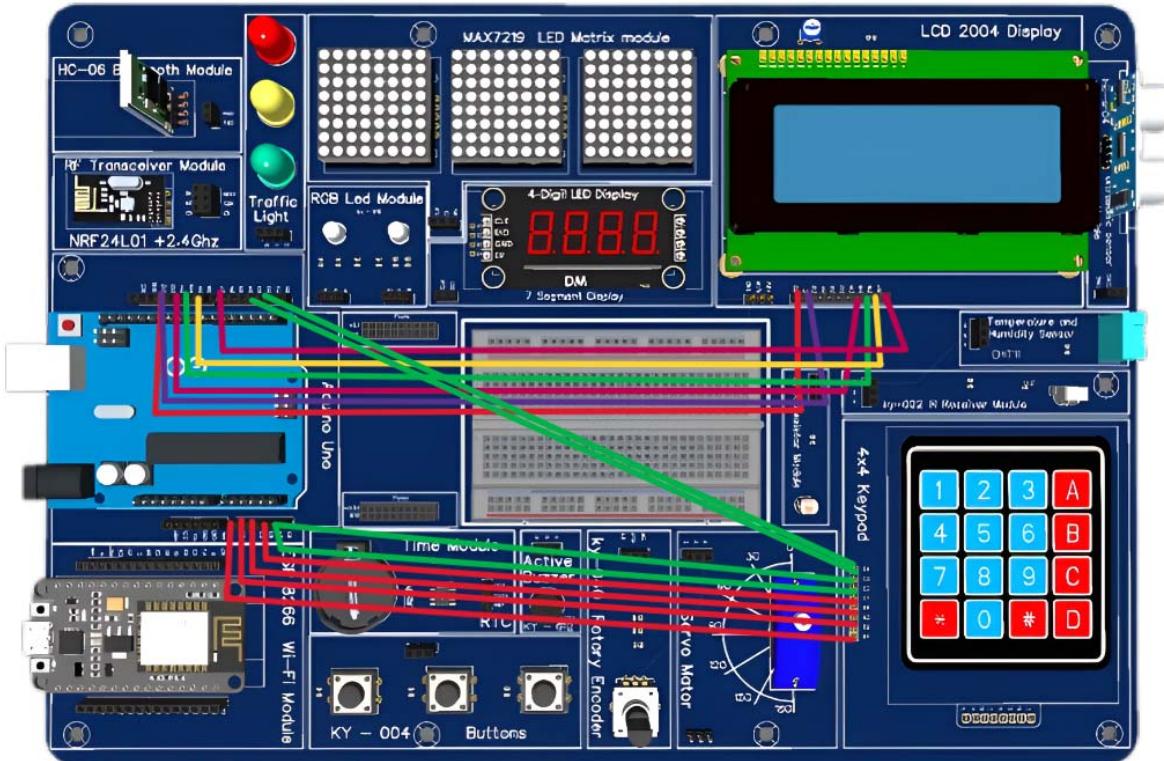
```

16-DARS: KEYPAD ORQALI EKRANGA YOZISH

16.1. Umumiylumot

Arduinoning 4x4 keypadi foydalanuvchilarga raqamlar, belgilar yoki buyruqlarni kiritishga yordam berish uchun Arduino bilan ko‘p ishlataladigan matritsa uslubidagi kiritish qurilmasidir. U 4 ta qator va 4 ta ustunda joylashgan 16 ta tugmadan iborat bo‘lib, bu uni kalkulyatorlar, parollarni kiritish va menuy navigatsiyasi kabi loyihalar uchun ixcham va samarali qiladi. Modulda jami 8 ta pin bo‘lib bular qator pinlari R1, R2,R3, R4 ustun pinlari C1, C2, C3, C4 lardir. Ushbu loyihada 4x4 keypad orqali kiritilgan raqamlar va belgilarni o‘qib LCD Displayga chiqaradi.

16.2. Ulanish tavsifi



16.3. Dasturiy taminot qismi

Ushbu loyihada LiquidCrystal va Keypad kutubhonalaridan foydalanamiz. Bu loyihada keypad yordamida kiritilgan belgilari LCD Displayga chiqariladi. Quyidagi rasimda Display pinlarni belgilash va keypad pinlarini belgilash qismi ko‘rsatilgan va keypadni sozlash uchun zarur bo‘lgan o‘zgaruvchilar elon qilamiz. Foydalanilgan kutibxonalar quyida keltirilgan keypad by Mark Stanley 3.1.1v, LiquidCrystal kutibxonasi oldindan o‘rnatilgan. Quyida pinlarni sozlash kodlari keltirilgan ROWS qatorlar soni COLS ustunlar soni.

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

// LCD pinlar: ( RS, E, D4, D5, D6, D7 )
LiquidCrystal lcd(12, 11, 10, 9, 8, 7);

// Keypad sozlash
const byte ROWS = 4;
const byte COLS = 4;
```

Quyidagi kodda belgilardan iborat massiv o‘zgaruvchiai ochilgan bo‘lib bu massiv keypaddan kiritilgan malunotlarni aniqlashda bizga yordam berdi. Bu belgilarni o‘zgartirishimizham mumkun.

```
char keys[ROWS][COLS] = {
    {'1','2','3','A'},
    {'4','5','6','B'},
    {'7','8','9','C'},
    {'*','0','#','D'}
};
// Qarot pinlari
byte rowPins[ROWS] = {A0, A1, A2, A3};
// Ustunlar
byte colPins[COLS] = {A4, A5, 2, 3};
// Keypadni sozlash
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);
```

Quyida LCD Displayning dastlabki sozlash kodlari keltirilgan.

```
//Displayn sozlaymiz
void setup() {
    lcd.begin(20, 4);
    lcd.print("Kiritilgan belgilar");
}
```

Quyida dasturning asosiy qismi keltirilgan va unda kiritilgan ma’lumotlarni displaygan chiqarish algoritmi keltirilgan.

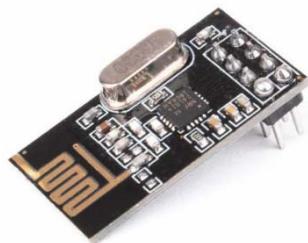
```
void loop() {
    char key = keypad.getKey();
    if (key) {
        lcd.setCursor(0, 1);
        lcd.print(key);
    }
}
```

17-DARS. NRF24L01 MODULDAN FOYDALANISH

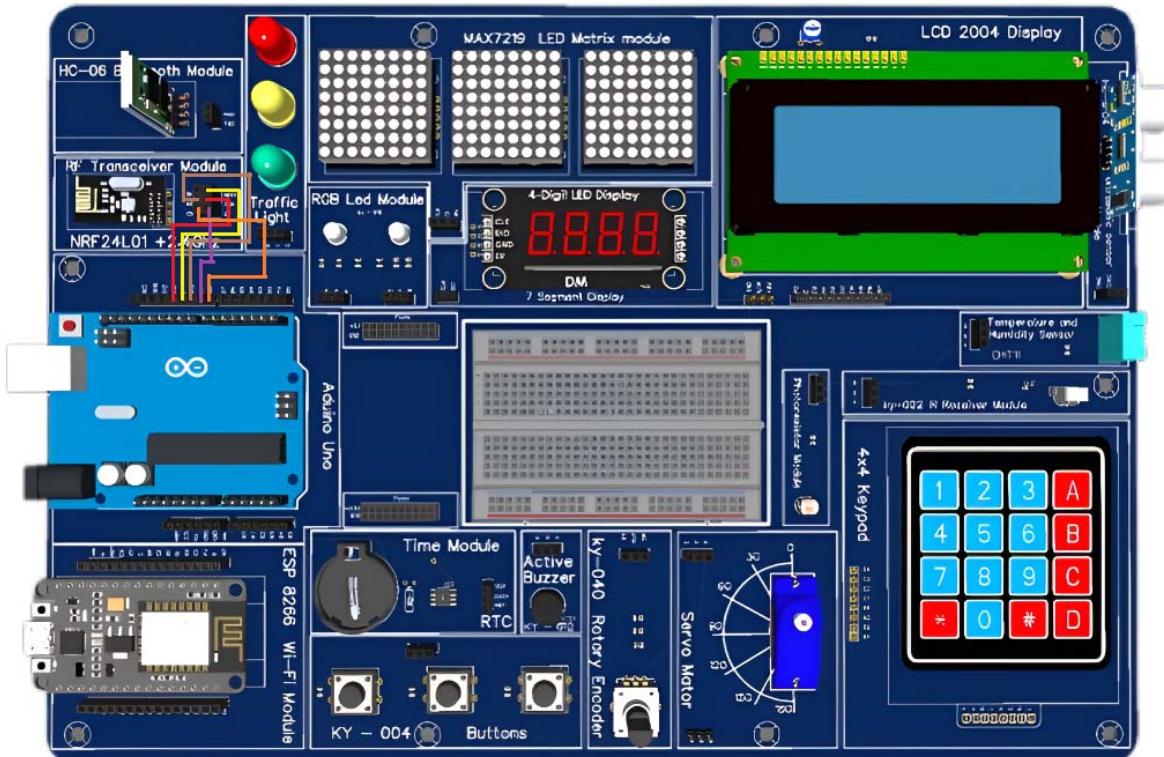
17.1 Umumiy ma'lumot

Ushbu loyiha orqali siz NRF24L01 modulidan foydalanib ma'lumot yuborishni va IDE kompilatorining serial porti orqali ko'rsatishni o'rGANASIZ.

Ulanishni tavsifi: Arduino unoning D9, D10, D11, D12 va D13 pinlarini kabellar yordamida RF Transceiver Modulining "CSN", "CE", "MOSI", "MISO" va "SCK" pinlariga ulang. Shuningdek, NRF24L01 modulini platada belgilangan joyga joylashtiring.



17.2. Ulanish tavsifi



17.3 Dasturiy taminot qismi

Bu loyihada RF24 by TMRh20 1.5.0 kutibxonasidan foydalanamiz. nRF24L01 kutubxonasi RF24 kutubxonasi tarkibida mavjud. Shu sababli RF24 kutubxonasini o'rnatishning o'zi yetarli.

```
#include <SPI.h>
#include "nRF24L01.h"
#include "RF24.h"
```

Kodda CE va CSN pinlarini belgilash kifoya qolgan pinlar kutubxona tomonidan Avtomat belgilanadi.

```
// RF24(cePin, csnPi)
RF24 radio(9,10);

//Bu biz hozir o'rnatmoqchi bo'lgan uzatish kanalining kodi
///!Boshqa modul bilan mos bo'lishi zarur
const uint64_t pipe = 0xE8E8F0F0E1LL;

//Uzatilishi kerak bo'lgan ma'lumotlar
int data = 0;
```

Ish uzatish tezligi "57600" ekanligini aniqlash uchun funktsiyani ishga tushiring. Bu Serial portning uzatish tezligi bilan bir xil bo'lishi kerak, aks holda buzilgan belgilar hosil bo'ladi.

```
void setup(void) {
    Serial.begin(57600);
    radio.begin();
    //Yozish kanalini ochish
    radio.openWritingPipe(pipe);
}
```

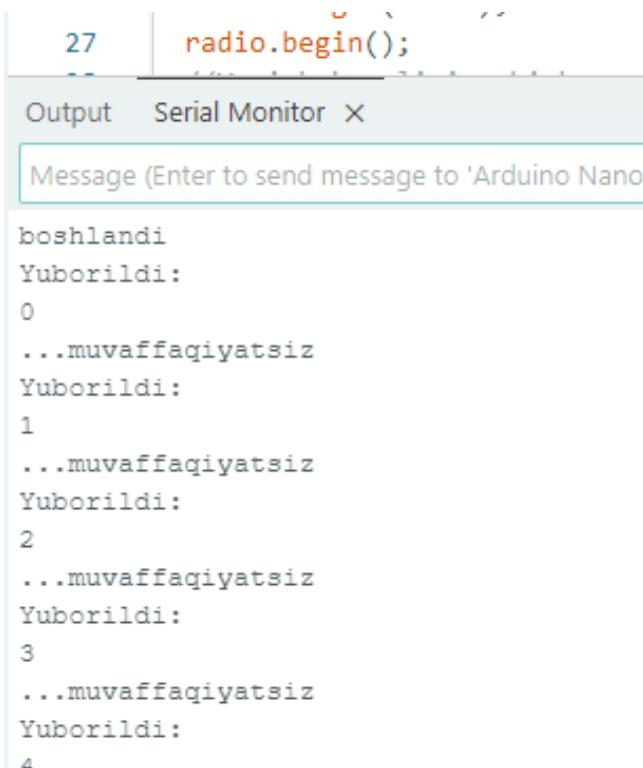
IDE kompilatori orqali dasturni yuklab olish uchun serial port uzatilgan 'data' ma'lumotlarini doimiy tarzda chop etadi kabellarni ulaganingizdan so'ng, IDE'ning nosozlikni tuzatish (debug) oynasini ochib, ma'lumotlarni ko'rishingiz mumkin.

Baud tezligi sozlamalariga alohida e'tibor bering: standart baud tezligi 9600 ga teng, lekin uni 57600 ga o'zgartirish kerak bo'ladi.

```
void loop(void) {
    Serial.print("Yuborildi:");
    Serial.print(data);

    bool ok = radio.write(&data, sizeof(int));
    if(ok)
        Serial.println("...Muvaffaqiyatli");
    else
        Serial.println("...muvaffaqiyatsiz");
    data++; //Har 200ms da 1 qo'shiladi
    delay(200);
}
```

Serial portida qanday natijalar chiqishi kerakligini quyidagi rasmda ko'rishingiz mumkun.



The screenshot shows the Arduino IDE Serial Monitor window. At the top, there are tabs for 'Output' and 'Serial Monitor'. The 'Serial Monitor' tab is active, indicated by a blue border. Below the tabs, the text 'Message (Enter to send message to 'Arduino Nano')' is displayed. The main area of the window shows the following text output:

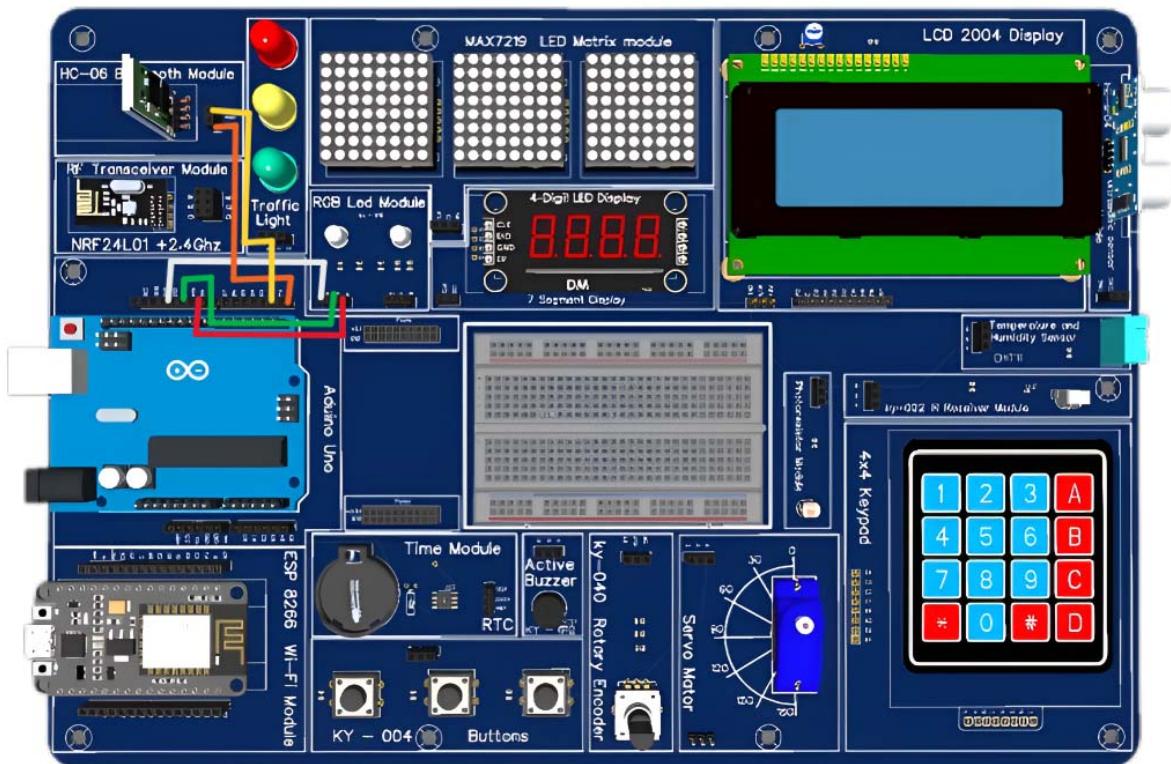
```
boshlandi
Yuborildi:
0
...muvaffaqiyatsiz
Yuborildi:
1
...muvaffaqiyatsiz
Yuborildi:
2
...muvaffaqiyatsiz
Yuborildi:
3
...muvaffaqiyatsiz
Yuborildi:
4
```

18-DARS: BLUETOOTH ORQALI RGB LEDNI BOSHQARISH

18.1 Umumiy ma'lumot

Ushbu loyiha orqali siz Bluetooth moduldan foydalanish uchun mobil ilovadan dan foydalanishni o'rganasiz. Bu ilovaning nomi 'Arduino Bluetooth controller' bo'lib 'Giristudio' tomonidan taqdim etilgan. Ilova versiyasi v2.7 Masofadan boshqarishni amalga oshirish uchun hc-06 Bluetooth modulini bordga joylashtiring. Dastur kodini Arduino unoga yuklang. So'ng, kabellarni ulang. Bluetooth-ga ulaning, Bluetooth-ga ulanish uchun mobil ilovadan foydalaning, 'RGB Picker' bo'limiga o'ting va rangni o'zgartiring.

18.2. Ulanish tavsifi



18.3 Dasturiy taminot qismi

Mobil ilova biz tanlagan rangni Bluetooth modulga matinli habar ko'rinishida yuboradi. Misol uchun biz tanlagan rang sariq bo'lsa ilova 255255000 yuboradi chunki sariq rangning RGB kodi qizil:255 yashil:255 havorang:0. Dastur ko'dida yuborilgan raqamlarni uchga bo'lish kodlari yoziladi. RGB LED pinlari R-D9, G-D10, B-D11 pinlaridan foydalanamiz. HC-06 ning RX pini D1 pinga, TX pini esa D0 pinga ulanadi. Chunki bu pinlar seriyali aloqa pinlari hisoblanadi.

```
#define redPin 9
#define greenPin 10
#define bluePin 11

void setup() {
    // Led pinlarini OUTPUT deb belgilaymiz
    pinMode(redPin, OUTPUT);
    pinMode(greenPin, OUTPUT);
    pinMode(bluePin, OUTPUT);
    // Bluetooth bilan bog'lanish standart tezligi
    Serial.begin(9600);
    setColor(037, 166, 154); // Boshlang'ich rang
}
```

Kodning asosiy qismida qabul qilingan matnni 3 ga bo'lamiz va matindan raqamga o'giramiz. Olingan ma'lumotlar asosida RGB LEDga quvvat beramiz.

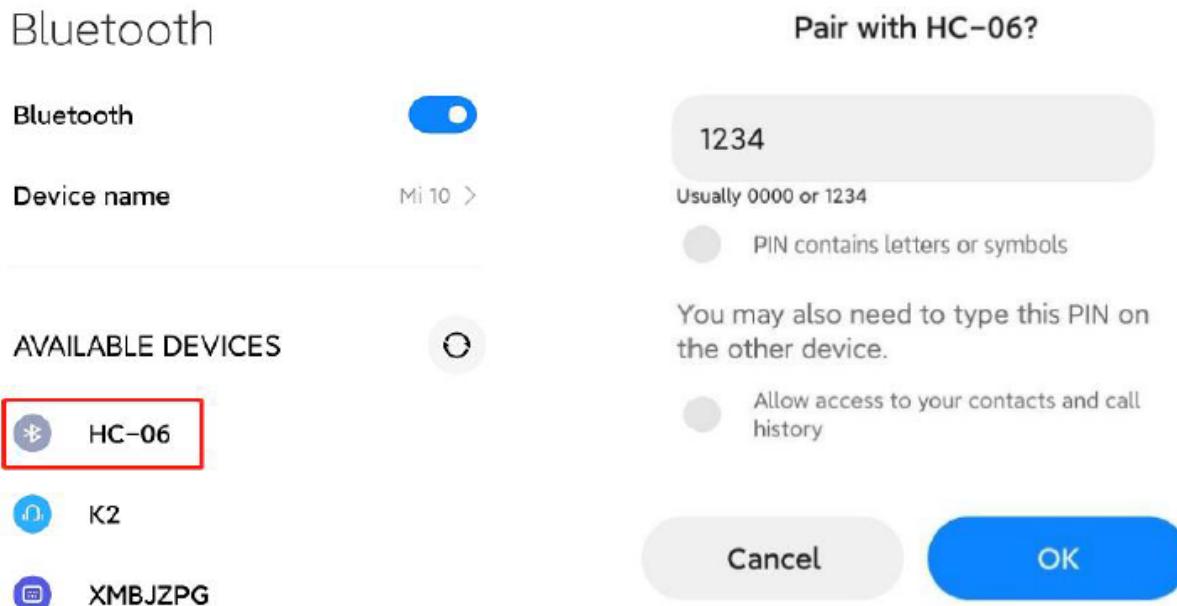
```
void loop() {
    if (Serial.available()) {
        String command = Serial.readStringUntil('\n');
        command.trim();
        // Qabul qilingan buyruqni uzunligi 9 ga tengligini tekshiramiz
        if (command.length() == 9) {
            // Buyruqdan RGB rang kodini ajratib olish
            int redValue = command.substring(0, 3).toInt();
            int greenValue = command.substring(3, 6).toInt();
            int blueValue = command.substring(6).toInt();
            // RGB LED rangini sozlash
            setColor(redValue, greenValue, blueValue);
        }
    }
}
```

Bu funksiada berilgan am'lumotlarni analog signal ko'rinishida chiqarib bermiz.

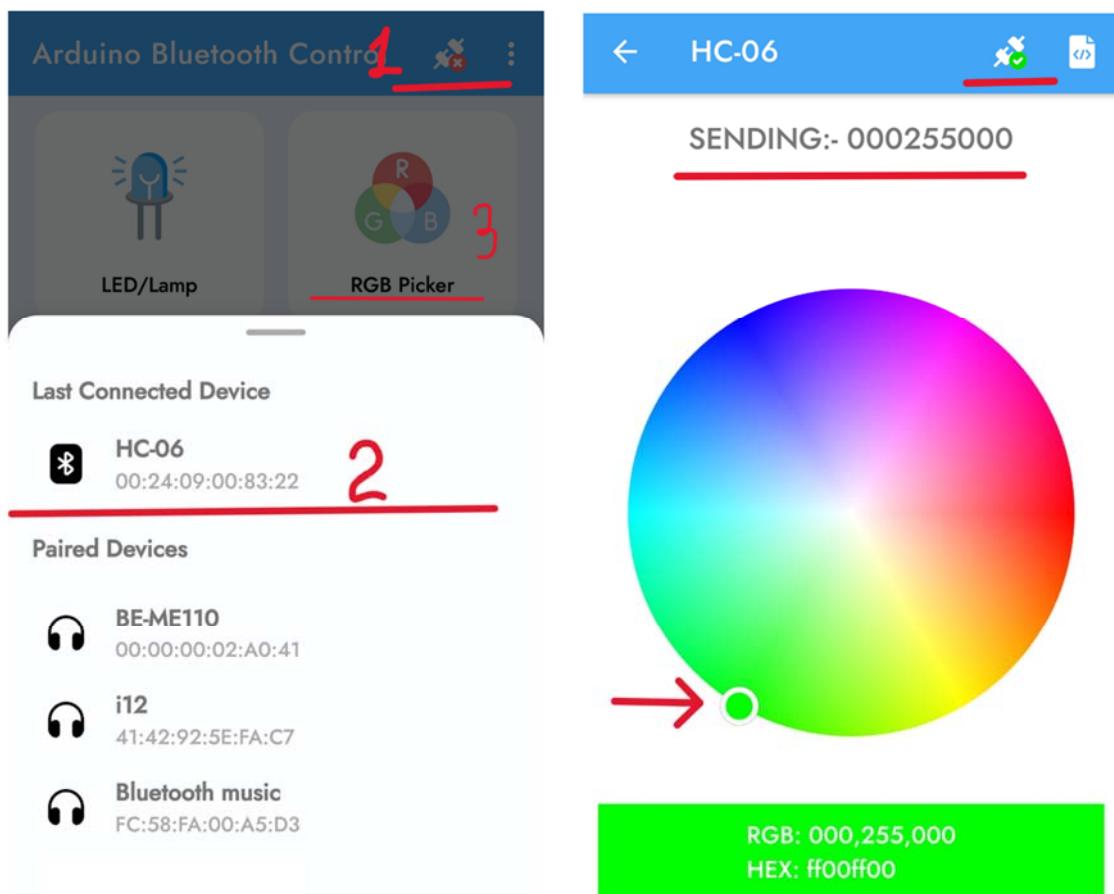
```
void setColor(int red, int green, int blue) {
    analogWrite(redPin, red);
    analogWrite(greenPin, green);
    analogWrite(bluePin, blue);
}
```

18.4. Illovani sozlash

Bluetooth-ni yoqing va hC-06 qurilmasini topish uchun Bluetooth qurilmalarini qidiring. Standart ulanish paroli 1234.



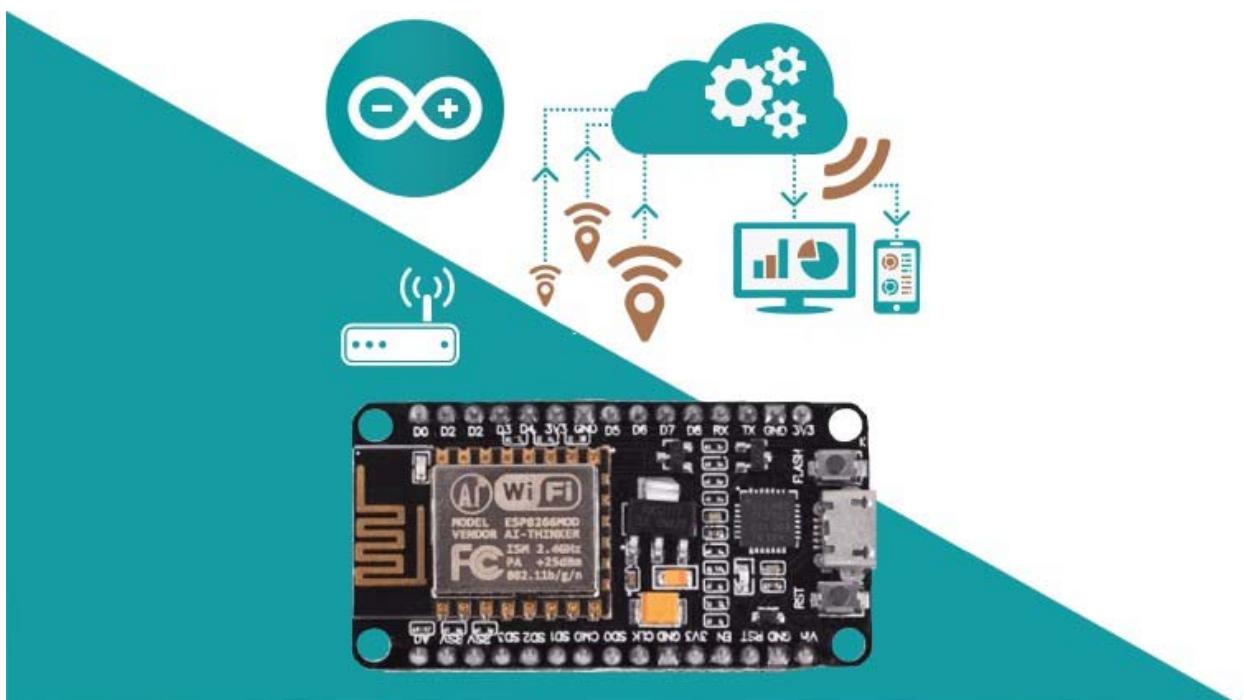
Ulanish bo‘limoga o‘ting HC-06 ni tanlang ‘RGB Picker’ bo‘limiga o‘ting rangni o‘zgartiring. Yuborilgan habar SENDING bo‘limida ko‘rsatiladi.



19-DARS. ESP8266 WI-FI MODULI

19.1 Umumiy ma'lumot

ESP8266 o'rnatilgan Wi-Fi imkoniyatlariga ega ixcham, arzon mikrokontroller bo'lib, uni IoT (Internet of Thinks) va uyni avtomatlashtirish loyihalari uchun qulay qiladi. Espressif Systems tomonidan ishlab chiqilgan bo'lib, u qurilmalarga internetga ularish yoki o'zlarining simsiz tarmoqlarini yaratish imkonini beradi.



19.2. ESP8266 texnik xususiyatlari

Asosiy xususiyatlari

Protsessor: Tensilica Xtensa® L106 32-bitli RISC protsessori

Soat tezligi: 80 MGts (standart), 160 MGts gacha

Operativ xotira: 128 KB (64 KB ko'rsatma + 64 KB ma'lumot)

Flash xotira: Odatda 512 KB dan 4 MB gacha (16 MB gacha kengaytirilishi mumkin)

Ishlash kuchlanishi: 3,0 V dan 3,6 V gacha (odatda 3,3 V)

Quvvat iste'moli: chuqur uyqu rejimida ultra past quvvat (~20 mKA)

Wi-Fi imkoniyatlari

Standartlar: IEEE 802.11 b/g/n

Xavfsizlik: WEP, WPA/WPA2

Tartiblar: Station, SoftAP yoki ikkalasi

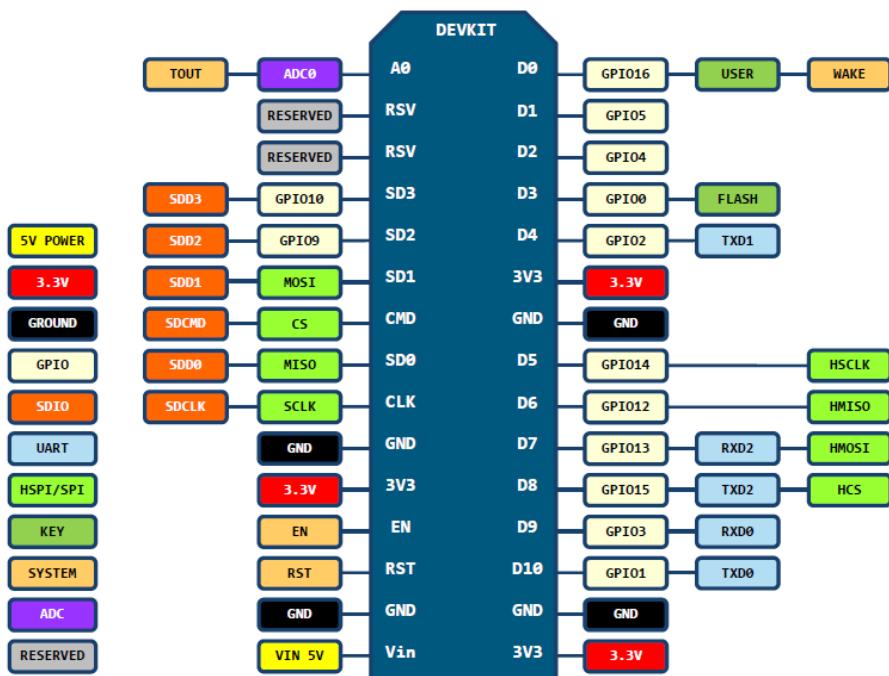
TCP/IP to'plami: to'liq integratsiyalangan

19.3 ESP8266 versiyasi

ESP8266 bir nechta versiyalarda mavjud (quyidagi rasmga qarang). Bizning fikrimizcha, ESP-12E yoki ko‘proq tanilgan nomi bilan ESP-12E NodeMCU to‘plami hozirda mavjud bo‘lgan eng keng qo‘llaniladigan versiyadir.



ESP8266 modulining sxematik diagrammasi:



19.4. NODEMCU ESP8266 pinlar

GPIO raqamlari pin diagrammasidagi yorliqlar bilan to‘g‘ri kelmaydi. Masalan, D1 - GPIO5, D2 - GPIO4 bilan mos keladi.

Quyidagi pinlar kirish/chiqish (input/output) sifatida muammosiz ishlatalishi mumkin:

GPIO5 (D1) - Odatda I2C uchun SCL sifatida ishlataladi

GPIO4 (D2) - Odatda I2C uchun SDA sifatida ishlataladi

GPIO0 (D3) - FLASH tugmasiga ulanagan; agar past (LOW) qiyomat belgilansa, ishga tushish muvaffaqiyatsiz bo‘ladi.

GPIO2 (D4) - Onboard LED bilan ulangan; past(low) yoki startda yuqori (HIGH) holatda bo‘lsa, ishga tushishda muammolar bo‘ladi

GPIO14 (D5) - SPI SCLK

GPIO12 (D6) - SPI MISO

GPIO13 (D7) - SPI MOSI

ADC kirish:

A0 pin AO deb belgilanadi - bu analog kirish (0 - 3.3V oralig‘ida o‘lchash)

Ushbu pinlar ishlatalishi mumkin, ammo ulardan foydalanishda ehtiyyot bo‘lish zarur ayniqsa startup (ishga tushish) paytida kutilmagan holatlar yuzaga kelishi mumkin.

GPIO16 (D0) - Ishga tushishda HIGH holatga o‘tadi va bu chuqur uyqu (deep sleep) rejimidan uyg‘onish uchun ishlataladi

GPIO15 (D8) - Yerga (GND) tortilishi kerak; agar HIGH holatga tortilsa, ishga tushishda xato yuz beradi

GPIO3 - RX sifatida belgilanadi va ishga tushishda HIGH holatda bo‘ladi

GPIO1 (TX) - Ishga tushish paytida debug chiqishi sifatida ishlataladi; agar LOW holatga tortilsa, ishga tushishda muvaffaqiyatsizlik yuzaga keladi.

TX pinlari emas, balki chiqish sifatida RX va ADO etiketli pinlardan foydalanish tavsiya etiladi

kirishlar sifatida.

GPIO6 orqali GPIO11 deb nomlangan pinlar ESP8266 flesh chipiga ulanadi.

Shuning uchun, kirish/chiqish funktsiyalari uchun ushbu pinlardan foydalanish tavsiya etilmaydi.

Agar siz o‘rni ishlatmoqchi bo‘lsangiz, GPIO4 va GPIO5 foydalanish uchun eng xavfsiz GPIO pinlari hisoblanadi.

20-DARS ARDUINO IDE DASTURI YORDAMIDA ESP8266 MODULIGA DASTUR YUKLASH

ESP8266 hamjamiyati Arduino IDE uchun maxsus qo'shimcha yaratgan bo'lib, uning yordamida siz ESP8266 modulini Arduino IDE va uning dasturlash tili orqali dasturlashingiz mumkin.

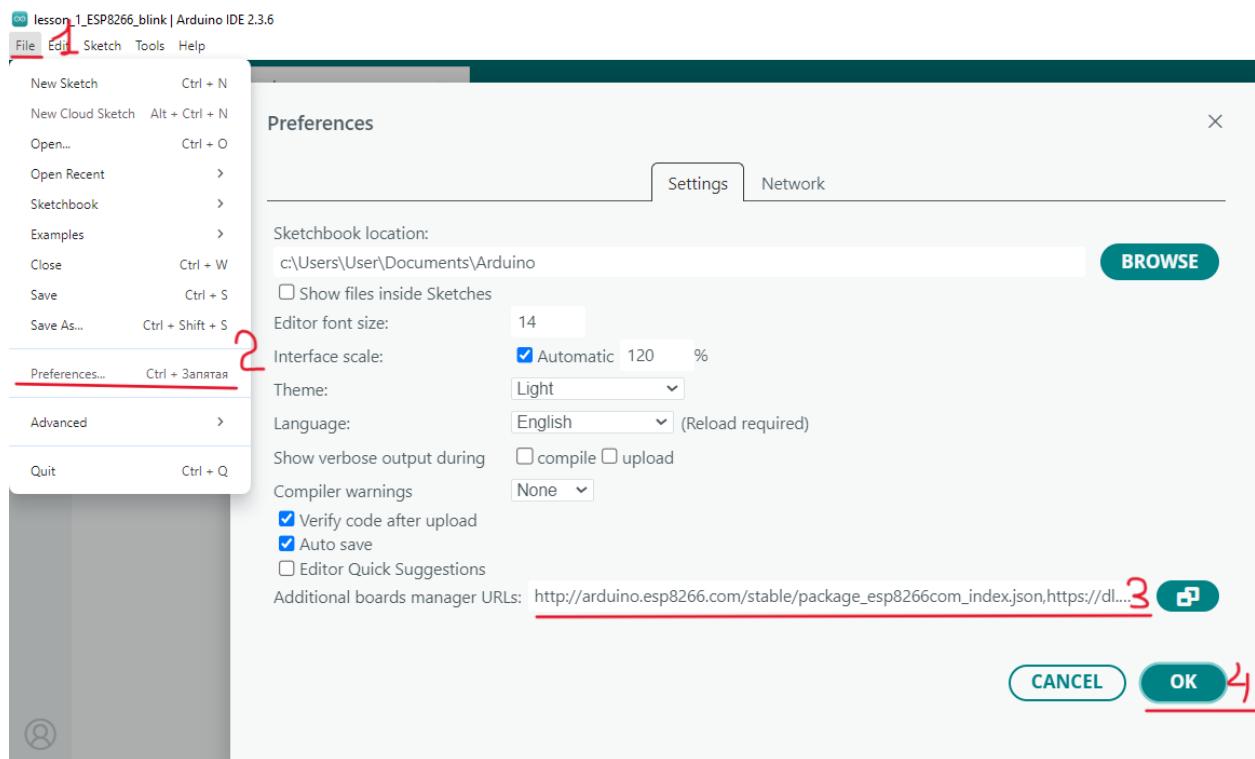
Ushbu qo'llanma ESP8266 pluginini Arduino IDEga qanday o'rnatishni ko'rsatadi - siz Windows, Mac OS X yoki Linux tizimidan foydalananayotgan bo'lsangiz ham farqi yo'q.

O'rnatishni boshlashdan oldin, kompyuteringizda Arduino IDEning eng so'nggi versiyasi o'rnatilganiga ishonch hosil qiling. Agar u eskirgan bo'lsa, avval uni o'chirib, qayta o'rnatish. Aks holda, to'g'ri ishlamasligi mumkin.

Arduino ide dasturining so'ngi versiyasini o'rnatish uchun quyidagi linkga bosing:
<https://www.arduino.cc/en/software/>

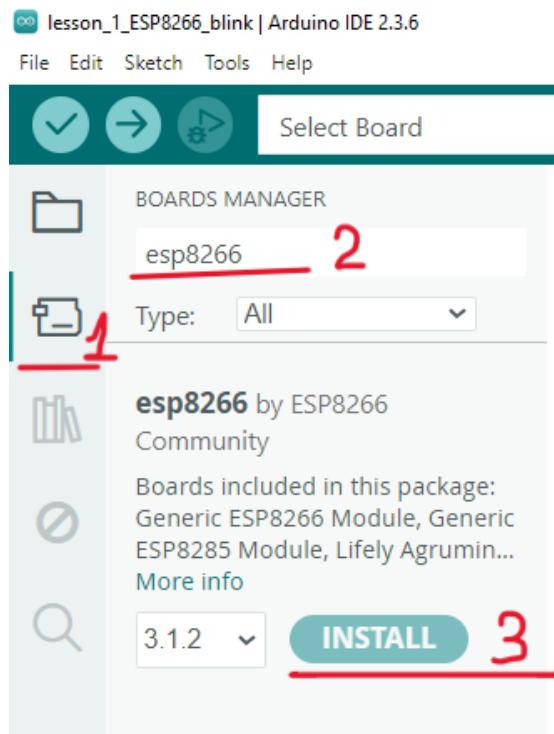
20.1 Arduino IDEga ESP8266 pluginini o'rnatish

ESP8266 pluginini Arduino IDE-ga o'rnatish uchun quyidagi ko'rsatmalarga amal qiling: Arduino IDE-da 'file' bo'limiga o'ting uyerdan Preferences bo'limiga oting.



Additional boards manager URLs: bo'limoga quyidagi havolalarni vergul bilan ajratib yozing va 'ok' tugmasini bosing.

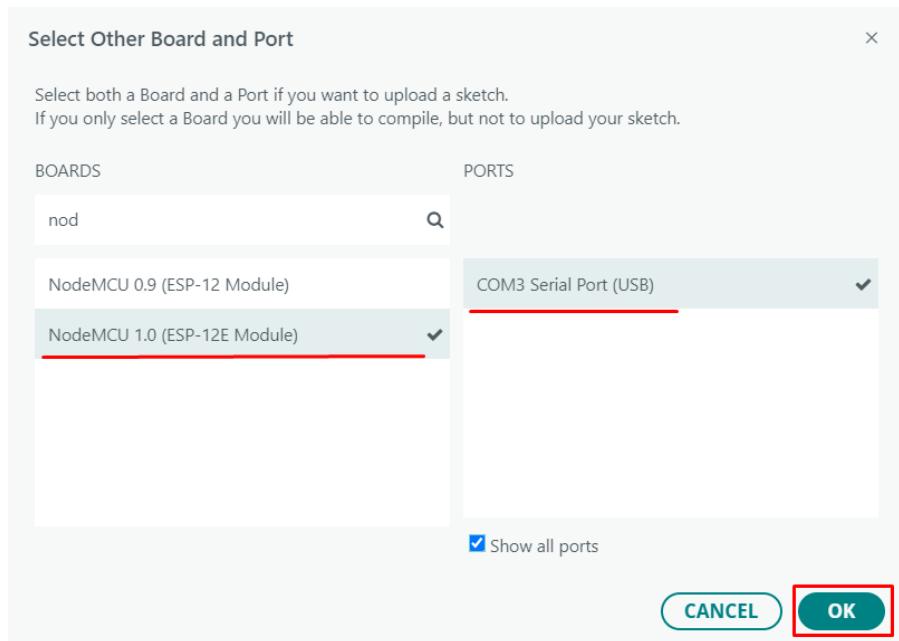
https://dl.espressif.com/dl/package_esp32_index.json
http://arduino.esp8266.com/stable/package_esp8266com_index.json
 Keyin esa boards manager bo‘limiga o‘ting va qidiruv bo‘limidan ‘ESP8266’ ni qidirng va INSTALL tugmasini bosing.



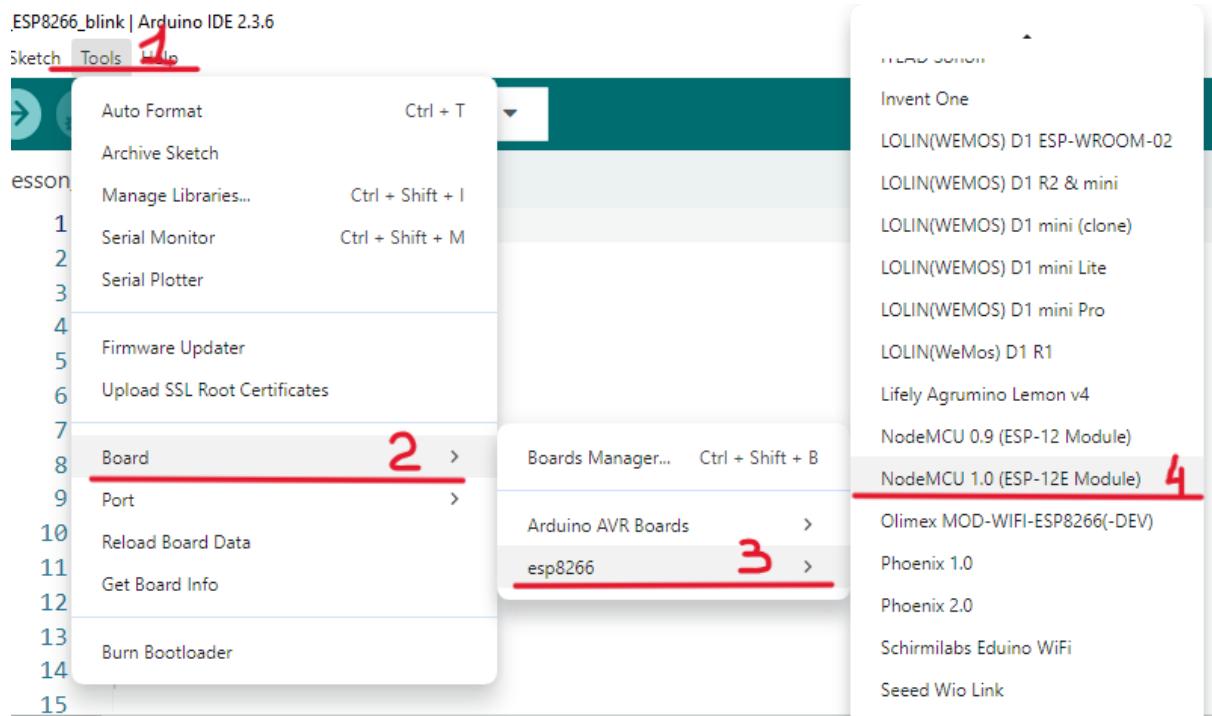
O‘rnatish biroz vaqt olishi mumkun.

20.2. Sinov uchun sodda loyiha

Bu loyihada LED yoritgich 0.5 soniys yons va 0.5 soniya o‘chiq holatda turadi. Bu soda loyiha orqali ESP8266 moduliga dastur yuklashni o‘rganamiz. Avval Arduino IDE dasturidan ESP8266 modulini tanlab olaiz.



Dasturni yuklash dan oldin portni tanlang va board ni to‘g‘ri tanlanganligini tekshiring.

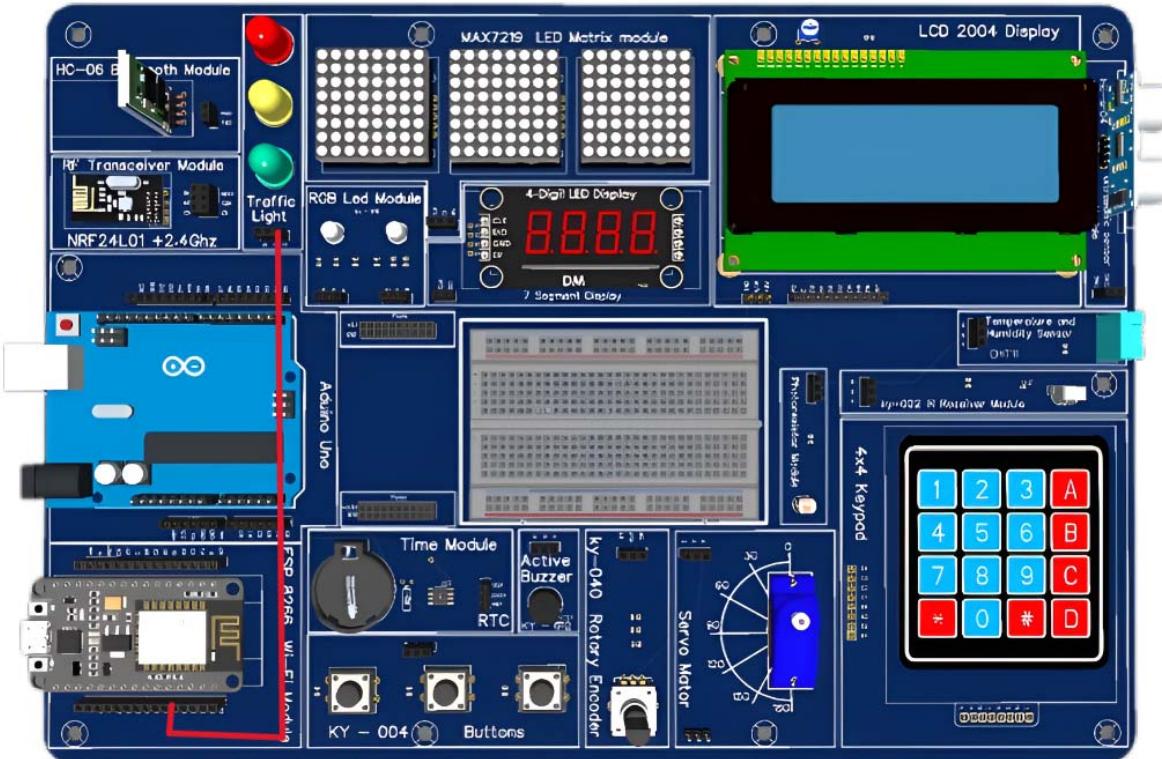


Dastur kodi juda soda va kod avvalgi darslarimizdan bizga tanish bo‘lgan elementlardan iborat.

```
//ESP8266
int pin = 2;
void setup() {
    pinMode(pin, OUTPUT);
}

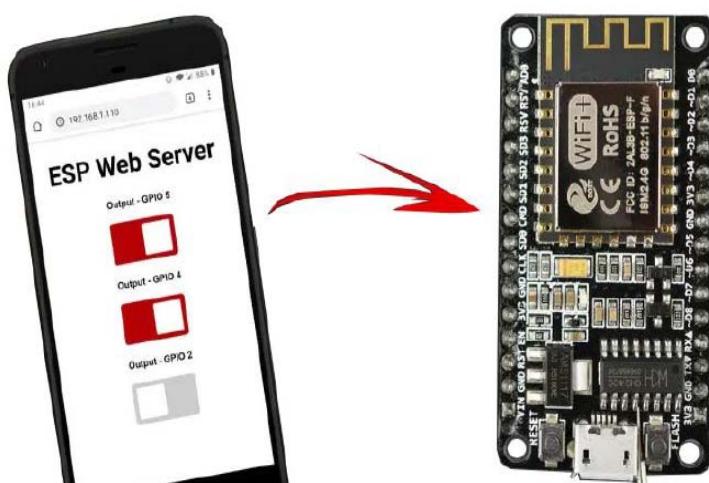
void loop() {
    digitalWrite(pin, HIGH);
    delay(500);
    digitalWrite(pin, LOW);
    delay(500);
}
```

Bu loyihada ESP8266 ning D4 pini Traffic lights modulining istalgan piniga ulanadi. Chunki yuqorida aytib o’tilganigek ESP8266 modulining pinlari dasturdagidan farqqiladi va GPIO2 pini D4 pinga to‘g‘ri keladi. Ulanish tasnifi quyidagi rasimda keltirilgan.



21-DARS: ESP8266DAN FOYDALANIB WI-FI ORQALI TRAFFIC LIGHT MODULINI BOSHQARISH

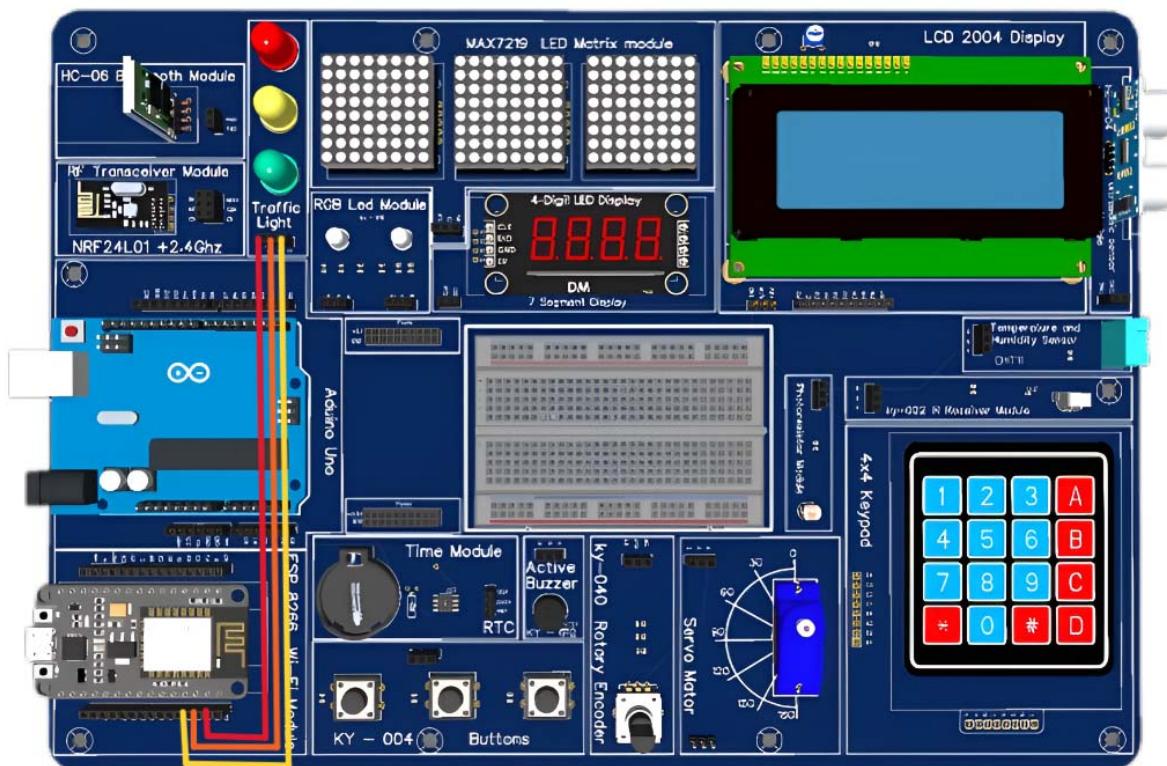
Bu darsda siz ESP8266 modulining chiqish pinlarini boshqarish uchun Asinxron veb server qurishni o‘rganasiz. Kod yozish va yuklash uchun Arduino IDE dan foydalanamiz va bizga ESPAsyncWebServer kutubxonasi kerak bo‘ladi.



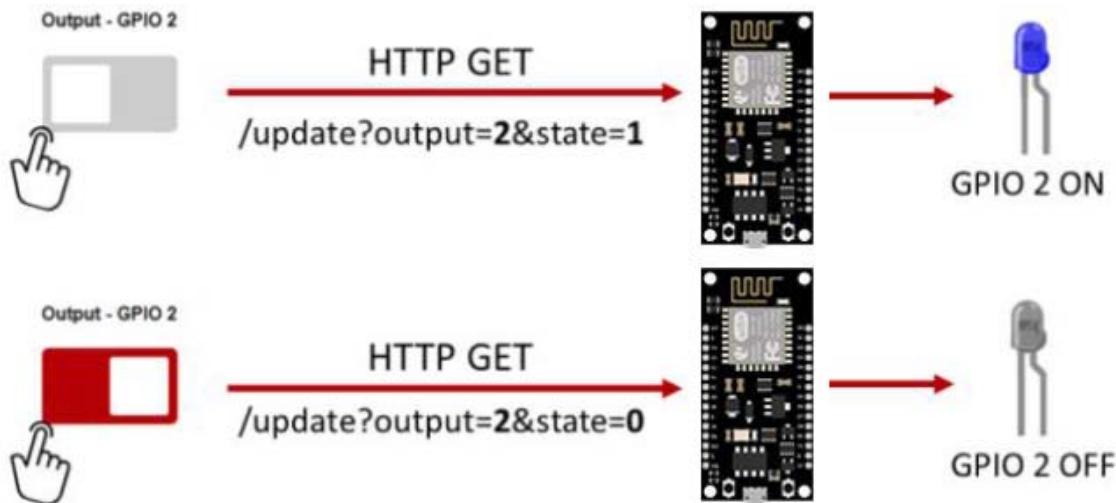
21.1. Asinxron tarmoq server

ESP-ga asoslangan mikrokontroller (masalan, ESP8266 yoki ESP32) yordamida mustahkam va samarali veb-server yaratish uchun biz ESPAsyncWebServer kutubxonasining kuchli imkoniyatlariiga tayanamiz. Ushbu kutubxona asinxron veb-serverni yaratish jarayonini soddalashtiradi, bu an'anaviy sinxron usullarga nisbatan yumshoqroq ishlash va tezroq javob berish vaqtlarini ta'minlaydi. Asinxron aloqadan foydalangan holda, server boshqa vazifalarni bloklamasdan bir vaqtning o'zida bir nechta mijoz ulanishlarini boshqarishi mumkin, bu esa uni IoT (Internet of Thinks) loyihalari uchun mos qiladi.

21.2. Ulanish tavsif



Bu loyihada veb sahifa yordamida LED yoritgichlarni boshqarishni o'rGANAMIZ. Bunda ESP8266 moduli wi-fi tarmog'iGA ulanadi va u bilan bitta wi-fi tatmog'iGA ulangan qurilmadan (telefon, kompyuter...) internet brouzer orqali (chrome, MS Edge...) kerakli internet manzilga o'tiladi. Sahifada uchta tugma bo'ladi shu tugmalar orqali ESP8266 ning chiqish pinlari boshqariladi. Bu qanday ishlaydi biz tugmani bosganimizda wi-fi tarmog'i orqali HTTP pratakoli asosida biz qurgan serverga so'ro'v yuboriladi va shu so'ro'v asosida chiqish pinlari boshqariladi. Biz bu chiqish pinlariga LED yoritgichlar ulaymiz va veb - sahifa yordamida LED yoritgichlarni boshqarish imkoniyatiga ega bo'lamiz.



21.3. Dasturiy taminot qismi

Biz loyihada foydalangan ESPAsyncWebServer kutubxonasinig me-no-dev tomonidan taqdim etilga orginal versiyasini Arduino IDE dasturidan qidirib topa olmaysiz. Shuning uchun ESPAsyncWebServer kutubxonasini .zip fayilida qo'shishingiz kerak. Biz qo'llanma bilan barcha kutubxonalarining .zip farmatini taqdim etamiz va quyida qoldirilgan havola oqali github.com platformasidan ham yuklab olishingiz mumkun. Buning uchun havolaga o'ting va 'code' tugmasini bosing va 'Download ZIP' tugmasini bosing va yuklab olingan kutubxonani Arduino IDE dasturiga qo'shing.

Havola : <https://github.com/me-no-dev/ESPAsyncWebServer>

Quyida kutubxonalarni elon qilish va Wi-fi tarmoq nomi va parolini sozlash kodi keltirilgan. Bu yerda PARAM_INPUT_1 va PARAM_INPUT_2

```
// Kutubxonalar
#include <ESP8266WiFi.h>
#include <ESPAsyncTCP.h>
#include <ESPAsyncWebServer.h>

// Tarmoq ma'lumotlatingiz bilan almash tiring.
const char* ssid = "Arduino board"; // Wi-Fi tarmoq nomi
const char* password = "12345678"; // Wi-fi paroli

const char* PARAM_INPUT_1 = "output";
const char* PARAM_INPUT_2 = "state";
```

o‘zgaruvchilari tugma holati va qaysi pinga tegishliligini aniqlovchi url parametrlaridir

Quyida server obyektini yaratish kodi keltirilgan

```
// 80-portda AsyncWebServer obyektini yaratting
AsyncWebServer server(80);
```

Veb sahifa kodlari

Biz HTML, CSS va JavaScript kodlari orqali veb sahifa yasaymiz va bu kodlarni index_html nomli o‘zgaruvchiga yuklaymiz. Qiyida keltirilgan kod sarlavha hisoblanadi va brouzeringizning yuqori qismida ko‘rinadi.

```
<title>ESP Veb Server</title>
```



Quyidagi <meta> tegi veb sayitingizni istalgan brouzerga moslashuvchan qiladi.

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

Quyidagi qator veb sayt logotipini bloklaydi. Agar bu qator bo‘lmassa brouzer

```
<link rel="icon" href="data:, ">
```

ESP8266 dan logotip so‘raydi.

Veb saytlarga dizayn berish uchun css dan foydalaniladi. Biz dizayn kodlarimizni <style> </style> teglari orasiga yozamiz.

<body> </body> teglari orasiga veb sahifaning asosiy kodlari yoziladi.

Quyida veb sahifaga katta shiriftda yozuv chiqarish uchun <h2> tegidan foydalanish ko‘rsatilgan.

```
<h2> ESP Veb Server Sahifasi </h2>
```

Veb sahifaning tugmalat turadiga qismiga %BUTTONPLACEHOLDER% ko‘rinishidagi matn yozilgan bu matn o‘rniga veb sahifa ishga tushganda vaziyatga mos tugma kodlari yoziladi.

Yuqorida aytilganidek veb sahifada java Scriptdan ham foydalanilgan. Java Sctipt kodlari <script></script> teglati orasiga yozilgan. Java sctiptda yozilgan funksiya tugma bosilganda serverga qaysi tugma bosilganini va uning holati qandayligini yuboradi serverda esa shu ma’lumotlar asosida qaysi chiqish piniga qanday signal yani yuqori (HIGH) yoki pas (LOW) yuborish kerakligini aniqlash kodi yozilgan.

```

<script>
    function toggleCheckbox(element) {
        var xhr = new XMLHttpRequest();
        if(element.checked){
            xhr.open("GET", "/update?output="+element.id+"&state=1", true);
        }else{
            xhr.open("GET", "/update?output="+element.id+"&state=0", true);
        }
        xhr.send();
    }
</script>

```

Quyida processor funksiyasi keltirilgan bu funksiya yuqorida aytilgan %BUTTONPLACEHOLDER% matni o‘rniga kerakli html kodlarini joylab beradi.

```

// Shifaga tugma qo'shish kodi
String processor(const String& var) {
    if (var == "BUTTONPLACEHOLDER") {
        String buttons = "";
        buttons += "<h4>Chiqish - GPIO 5</h4>
<label class=\"switch\">
<input type=\"checkbox\" onchange=\"toggleCheckbox(this)\" id=\"5\" " + outputState(5) + "
<span class=\"slider\">
</span>
</label>";
        buttons += "<h4>Chiqish - GPIO 4</h4>
<label class=\"switch\">
<input type=\"checkbox\" onchange=\"toggleCheckbox(this)\" id=\"4\" " + outputState(4) + "
<span class=\"slider\">
</span>
</label>";
        buttons += "<h4>Chiqish - GPIO 2</h4>
<label class=\"switch\">
<input type=\"checkbox\" onchange=\"toggleCheckbox(this)\" id=\"2\" " + outputState(2) + "
<span class=\"slider\">
</span>
</label>";
        return buttons;
    }
    return String();
}

```

Fubksiyaga BUTTONPLACEHOLDER matni berilsa u tugmalarning html kodlarini qaytaradi.

Quyidagi `outputState()` funksiyasini berilgan raqamdagini GPIO pini bo'sh bo'lsa "checked" aks holda bo'sh string qaytaradi.

```
String outputState(int output) {
    if (digitalRead(output)) {
        return "checked";
    } else {
        return "";
    }
}
```

Serial portini sozlaymiz va chiqish pinlarini ishchi holatini belgilaymiz.

```
void setup() {
    // Serial portini 115200 ga sozlaymiz
    Serial.begin(115200);
    // Pinlarni belgilash
    pinMode(5, OUTPUT);
    digitalWrite(5, LOW);
    pinMode(4, OUTPUT);
    digitalWrite(4, LOW);
    pinMode(2, OUTPUT);
    digitalWrite(2, LOW);
```

Dastlab ularga pas (LOW) qiymat beramiz.

Wi-fi ga ulanamiz ulanish muvofaqiyatlari bo'lmasa serial porti orqalini 'wi-fi ga ulanmoqda' degan yozuv chiqaramiz. Ulanish muvofaqiyatlari bo'lsa serial porti orqali IP manzilni chiqaramiz. Serverga ulanish uchun shu IP ga murojat qilishimiz kerak bo'ladi

```
// Wi-Fi tarmog'iga ulanish
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("WiFi ga ulanmoqda...");
}

// Lakal tarmoq manzilini chop etish
Serial.println(WiFi.localIP());
```

Quyidagi kod orqali biz veb sahifamiz html kodi yordamida server yaratamiz va IP manzilgiga murojat qilinganda biz yozgan veb sahifa oynasi ochiladigan qilamiz.

```
// Veb sahifa kodini yuklash
server.on("/", HTTP_GET, [](AsyncWebServerRequest* request) {
    request->send_P(200, "text/html", index_html, processor);
});
```

Quyidagi koda veb sahifa uchun veb manzil belgilash ko'satilgan. Kodning keyingi qismida veb so'rov orqali yuborilgan parametrlarni saqlash uchun o'zgaruvchilar belgilangan. Keyin esa veb so'rovda parameter borligi tekshirilmoqda. Agar parameter bor bo'lsa parametrlar digitalWrite Funksiyasi orqali chiqish pinlariga signal yuboradi. Avval aytganimizdek ikki o'zgaruvchilardan biri pin raqamini ikkinchisi tugma holatini bildiradi. Agar veb so'rovda parameter bo'lmasa o'zgaruvchilarga 'Habar yuborilmagan' matni saqlanadi.

```
// GET HTTP so'rovini yuborish <ESP_IP>/update?output=<inputMessage1>&state=<inputMessage2>
server.on("/update", HTTP_GET, [](AsyncWebServerRequest* request) {
    String inputMessage1;
    String inputMessage2;
    // Birinchi qiymatni olish <ESP_IP>/update?output=<inputMessage1>&state=<inputMessage2>
    if (request->hasParam(PARAM_INPUT_1) && request->hasParam(PARAM_INPUT_2)) {
        inputMessage1 = request->getParam(PARAM_INPUT_1)->value();
        inputMessage2 = request->getParam(PARAM_INPUT_2)->value();
        digitalWrite(inputMessage1.toInt(), inputMessage2.toInt());
    } else {
        inputMessage1 = "Habar yuborilmagan";
        inputMessage2 = "Habar yuborilmagan";
    }
});
```

Quyidagi koda veb so'rovdan olingan parametrlar tekshirish uchun serial porti orqali ekranga chop etiladi. Keyin esa brouzerga 'ok' Habari va status kodi yuboriladi. Server.begin(); esa serverni ishga tushuradi. Asosiy loop() funksiyasida hechqanday kod yozilmaydi. Server asinxron ishlashi uchun hamma kod setup() funksiyasida yoziladi.

```
Serial.print("GPIO: ");
Serial.print(inputMessage1);
Serial.print(" - yuborildi : ");
Serial.println(inputMessage2);
request->send(200, "text/plain", "OK");
});

// Start server
server.begin();
}

void loop() {
```

Dasturni yuklaganingizdan keyin seria portidagi IP manzilni brouzeringizga kiriting va quyidagicha oyna ochiladi.



Serial portida qaysi tugma bosilgani ko‘rsatiladi. Agar serial portida IP manzil chiqmasa ESP8266ning reset tugmasini bosing.

Output Serial Monitor X

Message (Enter to send message to 'NodeMCU 1.0 (ESP-12E Module)' on 'COM3')

```
rlid|ld|l||||$c|?????{?c?c??nn?doo?????c|x|||:  
WiFi ga ulanmoqda...  
192.168.43.124  
GPIO: 5 - yuborildi : 1  
GPIO: 5 - yuborildi : 0  
GPIO: Habar yuborilmagan - yuborildi : Habar yuborilmagan  
GPIO: 2 - yuborildi : 1  
GPIO: 4 - yuborildi : 1  
GPIO: 5 - yuborildi : 1
```