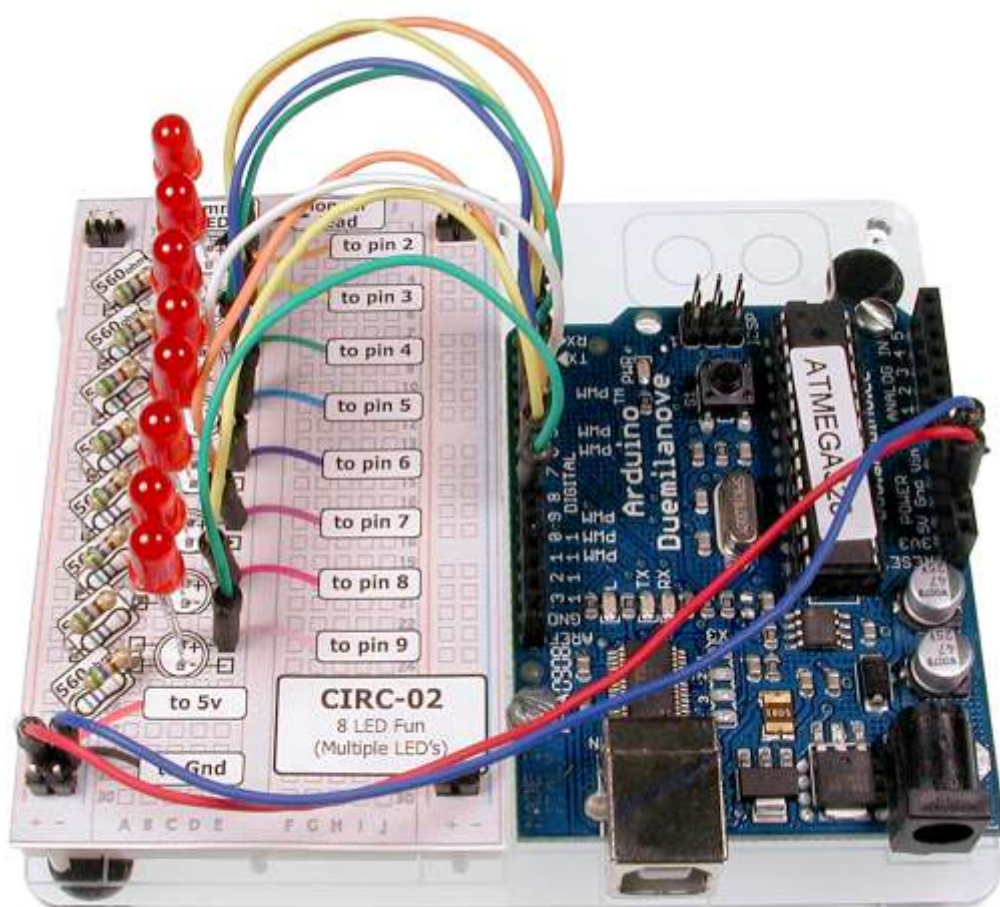


Arduinoni O`rganish Uchun Qo`llanma



ARDX



So`zboshi/proektlarga tayyorgarlik

{ASEM}	Qurilmani yig`ish	02
{INST}	Dasturiy ta`minotni o`rnatish	03
{PROG}	Dasturlash to`g`risida kichik ma`lumot	04
{ELEC}	Elektronika to`g`risida kichik ma`lumot	06

Qurilma

{CIRC01}	Ishni boshlaymiz (pirpirovchi svetodiod)	08
{CIRC02}	8-ta svetodiodli sxema	10
{CIRC03}	Motorlar bilan ishlash (tranzistor va motor)	12
{CIRC04}	Servomotorlar	14
{CIRC05}	8-ta svetodiodlar (74HC595 qadamli registrlar)	16
{CIRC06}	Musiqqa (pyezoelementlar)	18
{CIRC07}	Knopka bilan ishlash	20
{CIRC08}	O`zgaruvchan qarshilik (Potensiometrlar)	22
{CIRC09}	Yorug`lik (fotorezistorlar)	24
{CIRC10}	Temperatura (TMP36 temperatura datchiki)	26
{CIRC11}	Yuqori yuklamalar bilan ishlash (Relelar)	28
{CIRC12}	Ko`p rangli svetodiodlar (RGB svetodidolari)	30

01 ASEM

Komponentlarni
yig`ish

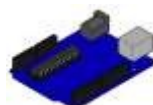
:: Qurilmani yig`ish ::



Arduino uchun
asos
x1



Bredbord
x1



Arduino
x1



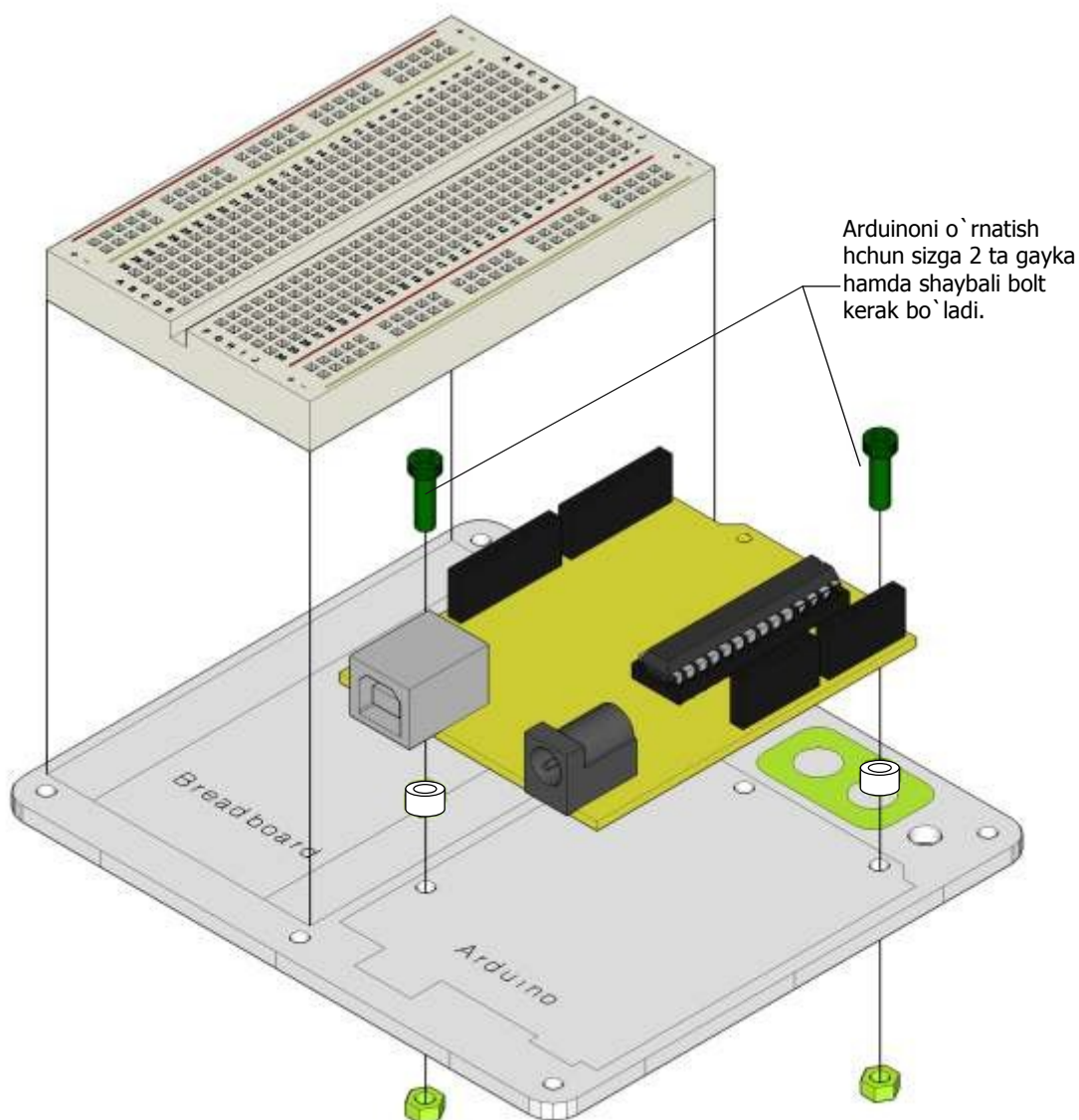
4-40 x 3/8" bolt
x3



4-40 gayka
x3



#4 x 1/16" shayba
x3



∴ Dasturiy taʼminotni oʻrnatish (IDE foydalanuvchilari interfeysi) ∴

02 INST

Installiyatsiya
(Dasturiy va Apparatli taʼminot)

Foydalanuvchi interfeysi Arduino dasturlanushi uchun moʻljallangan. Boshida hammasi qiyindek tuyuladi, lekin bu dasturni oʻrnatib foydalanishni boshlaganingizdan soʻng, uning sirlari sizga osondek boʻlib qoladi.

1 qadam: Dasturiy taʼminotni

Yuklab olish

Sahifani oching

<http://arduino.cc/en/Main/Software>

va oʻzingizning operatsion sistemangizni tanlang

Windows XP

2 qadam: Arxivni ochish

Dastur

[arduino-00XX-win.zip](#) (XX - versiya nomeri)

Eng qulayi shu papkaga joylash

[c:\Program Files\](#)

3 qadam: "Ярлык" yaratish

Oching

[c:\program files\arduino-00XX\](#) (XX - versiya nomeri)

Sichqonchani oʻng tomonini bosish

[Arduino.exe](#) (send to>Desktop ("ярлык" yaratish))

4 qadam: Ulash

Oʻzingizning Arduinoingizni USB kabel

Orqali boʻsh turgan USB portga ulang.

Ekkranda "диалоговое окно"

Punkti chiqishi kerak.

5 qadam: Yangi qurilmani qoʻshish

"поиск драйверов в интернете" opsiyasini oching. ("next" tugmasini bosing)

Keyingi

Bosing "Install from a list or specific location (Advanced)"

Duemilanove Board

[c:\program files\arduino-00XX\drivers\FTDI USB Drivers\](#)

Uno Board

[c:\program files\arduino-00XX\drivers\](#)

Vista, Seven

5 qadam: Yangi qurilmani qoʻshish

Device Manager oching

Start > Run > devmgmt.msc

Arduinoni tanlang

Other Devices > Arduino Uno (Uno)

Drayverni yangilang

"Update Driver" ni bosing

Drayverni tanlang

"Browse My Computer for Driver Software"ni bosing

[c:\program files\arduino-00XX\drivers\](#)

Mac OSX

2 qadam: .dmg ni oching

Oching (yuklang)

[arduino-00XX-mac.dmg](#) (XX - versiya nomeri)

3 qadam: Ilqovadan nusxa oling

Toping

"[Arduino](#)" (brauzerning "devices" qismida)

Joyini oʻzgartiring

"Arduino" ilovasini

"Applications" papkasiga joylang

4 qadam: Drayverni oʻrnating

∴ Faqat Duemilanove platasi uchun.

Toping

"[Arduino](#)" device

Sichqonchani 2-martta bosib, oʻrnating

[FTDI Drivers for Intel Macs \(XXXX\).pkg](#)

(FTDI Drivers for PPC Macs (XXXX).pkg)

Kompyuterni perzagruzka qiling

5 qadam: Arduino platasini ulang

Arduino platangizni kompyuteringizning

Boʻsh USB portiga ulang.

Arduino Programmalashtirish dasturi haqida

Arduino, C dasturlash tili bazasida dasturlanadi va C dastur tilini ixcham ko`rinishi hisoblanadi. C dasturlash tilida boshlang`ich tushunchalarga ega bo`lgan odam, Arduinoni tezda o`zlashtirib oladi. Agar, sizga bu qiyin tuyulayotgan bo`lsa, qo`rqmang tezda hammsini bilib olasiz. Chunki, Arduino dasturidagi beriladigan buyruqlar soddaligi bilan, C dastur tilidan farqlanadi. Quyida, dasturda ishlatiladigan ayrim tushunchalar bilan tanishib chiqamiz.

Tuzilishi

Arduinoning har qanday dasturi (ular odatda «sketch» deyiladi) ikkita asosiy funktsiyaga bo`linadi (podprogramma)

void setup() { }
Figurali qavs ichidagi barcha komandalar, dasturni birinchi ishga tushirganda, faqat bir marta takrorlanadi.

void loop() { }
Bu funktsiya setup() funktsiyasi yakunlangandan so`ng, to elektr manbaa uzilmaguncha, davriy ravishda takrorlanadi

Sintaksis (Arduino dasturida ishlatiluvchi belgilar)

C dasturlash tilidagi ayrim belgilar Arduinoda ham ishlatiladi (ammo C tilining strukturalari murakkabligi uni keng imkoniyatlarini ochadi). Siz uchun quyidagi belgilarni bilib olish yetarli.

// (bir qatorli kommentariyalar)

Odatda Arduinoda tuzilgan dasturni nimadan iboratligini sharqlash (Kommentariya) uchun qo`llaniladi. Tekst faqat shu belgidan keyin kiritiladi. Dastur kiritilgandan keyin kommentariyalar saqlanib qolmaydi (ya`ni, dastur plataga yozilganda yuklangan dasturda joy egalamaydi).

{ } (figurali qavs)
Komandalar blokining boshi va Oxirini bildiruvchi belgi hisoblanadi (odatda funktsiyaning boshlanishi va Yakunini bildiruvchi belgi).

/* */ (ko`p qatorli kommentariyalar). Siz bu belgini berilgan buyruqni batafsil yoritish maqsadida shu belgilarni orasiga yoziladi. Bu belgilar orqali berilgan kommentariyalar kompilyator (tekshirish) yordamida ignorirovat qilinadi.

; (nuqtali vergul)
Dasturda berilgan har bir kommandalardan keyin shu belgi qo`yiladi (bu belgini ikki nuqta nuqta bilan adashtirib bo`lmaydi, dastur uni hato deb tushunadi).

O`zgaruvchilar

Har qanday programma sonlar bilan murakkab amallarni bajaradi. o`zgaruvchilar esa bu amallarni oson qilishda yordamlashadi.

int (butun sonli)
Asosiy ishchi o`zgaruvchidir. Xotirada 2 bayt (16 bit) egallaydi. Butun sonlarni -32 768 ... 32 767 diapazonda o`ziga qabul qiladi.

long (uzun)
int o`zgaruvchisining hajmi Yetmaganda ishlatiladi. Xotirada 4 bayt (32 bit) joyni egallaydi va -2 147 483 648 ... 2 147 483 647 diapazondagi sonlarni qabul qiladi.

boolean (ikkilik)
True/False o`zgaruvchisini Oddiy ko`rinishidir. xotirada faqat 1 bit egallaydi.

float (suzuvchi nuqtali)
Ko`proq analog signallarni boshqarishda qo`llaniladi. Xotirada 4 bayt (32 bit) va -3,4028235E+38 diapazondagi ma`lumotlarni egallaydi.

char (simvol)
ASCII kodidagi bitta simvolni o`z ichiga oladi (misol, «A» =65). xotirada faqat 1 bayt (8 bit) joy egallaydi. Arduino uni massivli Simvol deb qabul qiladi.

Matematika operatorlar

Operatorlar sonlarni o'zgartirish uchun xizmat qiladi.

= (tenglik) nimanidir qaysingadir tengligini anglatadi. (masalan, $x=10*2$ x ni $10x2$ ga tengligini ifodalaydi).
% (bo'lishdagi qoldiq). Masalan, $12\%10$ 2 natijani beradi.
+ (qo'shish)
- (ayirish)
***** (ko'paytiruv)
/ (bo'luv)

Taqqoslash operatorlari

Bu operatorlar logik taqqoslash uchun qo'llaniladi.

== (teng) (masalan $12==10$ teng emas (FALSE), $5==5$ teng(TRUE).)
!= (teng emas) (masalan $12!=10$ to'g'ri (TRUE), $5!=5$ noto'g'ri (FALSE).)
< (kichik) (masalan $12<10$ noto'g'ri (FALSE), $12<12$ noto'g'ri (FALSE), $12<14$ to'g'ri (TRUE).)
> (katta) (masalan $12>10$ to'g'ri (TRUE), $12>12$ noto'g'ri (FALSE), $12>14$ noto'g'ri (FALSE).)

Boshqaruvchi funktsiyalar

Komandalar tartibini aniqlash uchun, boshqaruvchi funktsiya kerak bo'ladi. Quyida ayrim funktsiyalar keltirilgan. bu funktsiyalar haqida to'liq ma'lumotlarni arduino.cc saytidan olishingiz mumkin.

```
If (1 shart) {}  
else if (2 shart) {}  
else {}
```

Agar, 1 shart to'g'ri (TRUE) bo'lsa, birinchi figurali qavsdagi komandalar bajariladi. Agar, 1 shart noto'g'ri (FALSE) bo'lsa, 2 shart tekshiriladi. Agar, 2 shart to'g'ri bo'lsa, ikkinchi figurali qavsdagi komandalar bajariladi, aks holda uchinchi figurali qavsdagi komandalar bajariladi.

```
for (int i=0;  
i<qaytarilishlar soni;  
i++) {}
```

Bu funktsiya sikllarni aniqlash uchun ishlatiladi. nechchi son berilsa, shuncha marta takrorlanadi. i o'zgaruvchisi oshishi ham, kamayishi ham mumkin.

Raqamli signallar

`digitalWrite(pin, value);`
 Agar, port OUTPUT rejimiga ulangan bo'lsa, HIGH (logik bir, +5V) yoki LOW (logik nol, GND) ni yozish mumkin.

`pinMode(pin, mode);`
 Tanlangan portni ish rejimini aniqlash uchun qo'llaniladi. Siz, 0...19 (14 dan 19 gacha portlar analog portlardir) bo'lgan portlarni qo'llashingiz mumkin. Rejimlar, odatda INPUT(kirish) yoki OUTPUT(chiqish) bo'lishi mumkin.

`digitalRead(pin);`
 Agar, port INPUT rejimida bo'lsa, komanda chiqishdagi signalni HIGH yoki LOW ga o'zgartiradi.

Analogli signallar

Arduino — raqamli qurilma, lekin quyidagi ikki komanda orqali analogli signallarni boshqarish mumkin.

`analogWrite(pin, value);`
 Arduining ba'zi portlari (3,5,6,9,10,11) KIM (keng impulsli modulyatsiya) xususiyatiga ega. Bu rejimda port, logik bir va nol signalni juda katta tezlikda yuboradi. Shu asosda o'rta kuchlanish bir va nollar soni balansiga bog'liq bo'lib qoladi va bir va nol 0 (0V) dan 255 (+5V) gacha bo'lgan oralikda o'zgaradi.

`analogRead(pin);`
 Agar, analogli port INPUT rejimida sozlangan bo'lsa, undagi kuchlanishni o'lchash mumkin. 0 (0V) dan 1024 (+5V) gacha qiymatni qabul qilishi mumkin.

.: Elektronika to`g`risida kichik ma`lumot:.

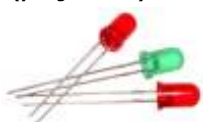
Elektronika asoslari

Arduinoni o`rganish davomida siz elektronika haqida tushunchaga ega bo`lishingiz kerak. Shuning uchun biz bu bo`limda ba`zi elektroni komponentlarning tuzilishi, xossalari, ulanish usullari haqida ma`lumot beramiz. Agar, elektron komponentlar to`g`risida batafsil bilmoqchi bo`lsangiz, help@oomlout.com pochta-siga savollaringizni yuboring.

Elektronik komponentlar

LED

(yorug`lik diodi)



Vazifasi:

Dioddan tok oqib o`tsa, undan yorug`lik chiqaradi. Yorug`lik diodidan tok faqat bir tomonga qarab o`tadi.

Ko`rinishi:

Kichkin lampochkaga o`xshaydi.

Oyoqchalar soni: 2

(yorug`lik diodining uzunroq oyoqchasi (anod) manbaaning musbat qutbiga ulanadi.)

Muhim:

Faqat to`g`ri ulangan holatda ishlaydi.

Tokning oqib o`tishini chegaralash

Maqsadida rezistor qo`llaniladi.

Qo`shimcha ma`lumot:

<http://ardx.org/LED>

Diod



Vazifasi:

Bir tomonli klappanning elektron ekvivalenti. Tok undan faqat bir tomonga qarab oqadi.

Ko`rinishi:

Odatda silindrik ko`rinishda bo`ladi.

oyoqchalari bir-biriga nisbatan qarama-qarshi joylashgan (silindrik korpusning bir tomoniga, chiziqliq tortilgan, bu diodni manfiy qutbini anglatadi).

Oyoqchalar soni: 2

Muhim:

Tokni faqat bir tomonga o`tkazadi.

Diodni chiziqliq tortilgan oyoqchasini manbaani

(-) qutbiga, qarama-qarshi tomondagisini esa, musbat qutbga ulanadi.

Qo`shimcha ma`lumot:

<http://ardx.org/DIOD>

Rezistor



Vazifasi:

Zanjirdan oqib o`tayotgan tok kuchini chegaralaydi.

Ko`rinishi:

Odatda, 2 oyoqchali silindrik ko`rinishda bo`ladi. Korpusga tortilgan rangli chiziqliqlar rezistorning nominalini korsa-tadi.

Oyoqchalar soni: 2

Muhim:

Rezistor nominalini adashtirish oson.

Qo`shimcha ma`lumot:

<http://ardx.org/RESI>

Tranzistor



Vazifasi:

Signalni yo`nalishi va kuchayishini o`zgartiruvchi element.

Ko`rinishi:

Xilma-xil korpuslarda ishlab chiqariladi. Markasi odatda, korpusning old qismiga yoziladi.

Oyoqchalar soni:

3 (Baza, Kollektor, Emitter)

Muhim:

Oyoqchalarni zanjirga ulayotganda, to`g`ri ulash lozim.

Qo`shimcha ma`lumot:

<http://ardx.org/TRAN>

Servomotorlar



Vazifasi:

Elektr impulslarini aylanish o`qining burchakka ogishini ta`minlaydi..

Ko`rinishi:

3 oyoqchali, kronshteynli aylanish o`qiga ega, plastik korpusda bo`ladi.

Oyoqchalar soni:

3

Muhim:

To`g`ri ulanganiga e`tibor bering!

Qo`shimcha ma`lumot:

<http://ardx.org/SERV>

Kollektorli Dvigatel



Vazifasi:

Aagar, motordan tok o`tsa, uning o`qi aylanadi.

Ko`rinishi:

Odatda, silindrik metall korpusda, markazida aylanish o`qi bo`lgan element.

Oyoqchalar soni:

2

Muhim:

Dvigatelni ulash uchun, quvvati mos keluvchi tranzistor yoki rele-dan foydalanish zarur.

Qo`shimcha ma`lumot:

<http://ardx.org/MOTO>

Elektronik komponentlar (Davomi)

Pyezoelement

**Vazifasi:**

Tok impulsi o'tganda, shivirlaydi. Impuls ketma-ketligi, elemetda musiqali ton hosil qiladi.

Ko'rinishi:

Kichik qora bochkaga, ba'zida tilla diskga o'xshab ketadi.

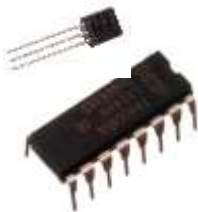
Oyoqchalar soni: 2**Muhim:**

To'g'ri ulash juda qiyin.

Qo'shimcha ma'lumot:

<http://ardx.org/PIEZ>

IC (Integral mikrosxema)

**Vazifasi:**

O'zida elektronaning xoxlagan qiyinchiligini saqlaydi.

Ko'rinishi:

Komponentning nomi va markasi korpusning ustiga yoziladi.

Oyoqchalar soni:

2...100 tagacha.

Важно:

Ulash jarayonida, mikrosxemaning oyoqchalarini to'g'ri ulash kerak.

Qo'shimcha ma'lumot:

<http://ardx.org/ICIC>

Tugma

**Vazifasi:**

Tugma bosilganda kontaktlarni tutashtiradi.

Ko'rinishi:

Tugmasi yuqorida va oyoqchalari pastda bo'lgan, kvadrat ko'rinishdagi element.

Oyoqchalar soni:

4

Muhim:

Zanjirga, ikki yon oyoqchalari ulanadi.

Qo'shimcha ma'lumotlar:

<http://ardx.org/BUTT>

Potensiometr

**Vazifasi:**

Nominali, aylanish o'qining burchak o'zgarishiga qarab o'zgaradi.

Ko'rinishi:

Turli xil korpuslarda ishlab chiqariladi.

Oyoqchalar soni:

3

Muhim:

Chiziqli yoki logarifmik shkalali bo'lishi mumkin.

Qo'shimcha ma'lumot:

<http://ardx.org/POTE>

Fotorezistor

**Vazifasi:**

Nominali, o'ziga tushgan yorug'lik intensivligiga qarab o'zgaradigan rezistor.

Ko'rinishi:

Odatda, shaffof qoplama tagida zig-zagli spiral ko'rinishiga ega kichkina disk.

Oyoqchalar soni:

2

Muhim:

Fotorezistorni kuchlanishni bo'luvchi sifatida ishlatish maqsadga muvofiq.

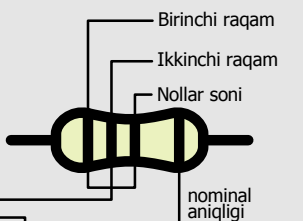
Qo'shimcha ma'lumot:

<http://ardx.org/PHOT>

Rezistorlarni raqamli kodirovkasi

Namunalar:

yashil-ko'k-jigarrang = 560 OM
qizil-qizil-qizil = 2 200 OM =
2.2 kOM
jigarrang-qora-to'q sariq = 10 000
OM = 10 kOM



0 — qora	5 — yashil	20% — chiziqli yo'q
1 — jigarrang	6 — ko'k	10% — kumushrang
2 — qizil	7 — binafsha	5% — tillarang
3 — to'q sariq	8 — kulrang	
4 — sariq	9 — oq	

Oyoqchalarni kaltalatish

Yuqorida keltirilgan ba'zi elektron komponentlar juda uzun oyoqchali bo'ladi. Ishlatish qulay bo'lishi uchun quyidagicha o'zgartirish qilsa ham bo'ladi:
yorug'lik diodlari:
Uzunroq oyoqchani 10 mm gacha, kaltarog'ini 7 mm gacha.

Rezistorlar:

Oyoqchalarini pastga 90°ga buking va 6 mm gacha kaltalating.

Qolgan komponentlar:

Boshqa komponentlarni oyoqchalarini siz, o'zingizning xoxishingizcha kaltalatishingiz mumkin.



Ishning ta`rifi:

Svetodiodlar turli qurilmalarda keng ishlatiladi, tuzilishi va ishlash prinsipi sodda bo`lganligi uchun o`rganishni shundan boshlaymiz. Bu mavzuda biz, svetodiodni cheksiz sikl bo`yicha o`chirib-yoqishni o`rganamiz. Pastdagi spiska bo`yicha detallarni yig`ing. Detallar yigilgandan so`ng, sketch ni yuklaymiz. Buning uchun Arduino platasini, bo`sh USB portga ulang. Keyin esa, programmadagi **Instrumenti>Port>** (odatda, Com 1..3 chiqadi) ni bosib portni o`rnatib. Sketchni yuklanishi, **Fayl>Vgruzit>** menyusi orqali amalga oshiriladi. Va shundan keyin kompilyatsiya boshlanadi. Agar, hamma narsani to`g`ri bajargan bo`lsangiz siz tuzgan birinchi kichik projekt ishlaydi.

Agar, siz yuklash jarayonida biror muammoga duch kelgan bo`lsangiz, quyidagi saytga murojaat qiling — <http://ardx.org/TRBL>

Qurilma:

Elementlar:



Topshiriq kartochkasi
CIRC-01
x1
560 omli rezistor
yashil-ko`k-jigarrang
x1



2` kontaktli
razyom
x4

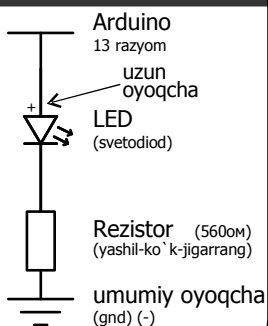


Svetodiod
10MM
x1



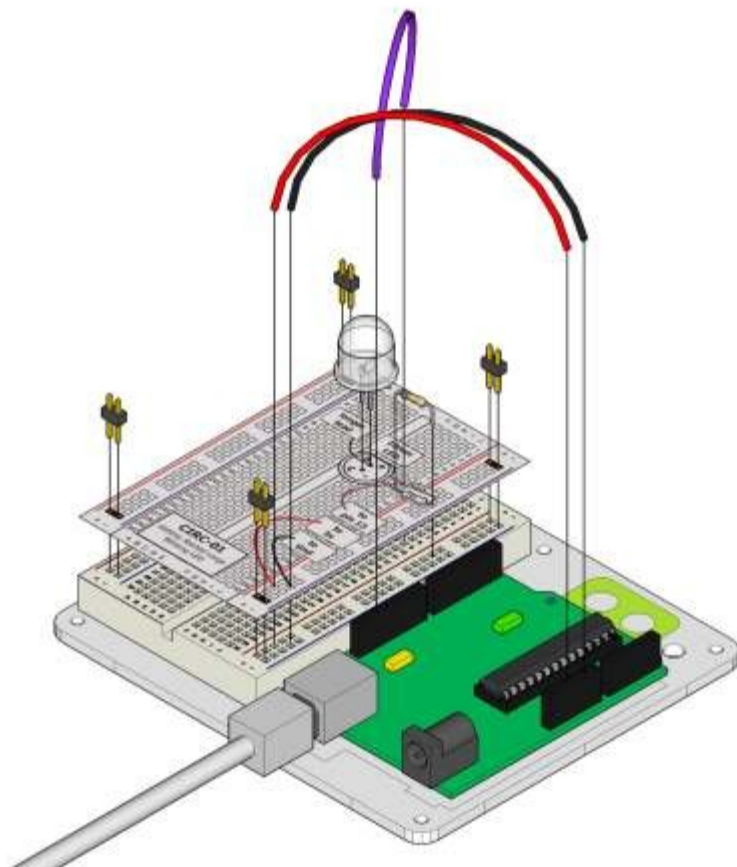
Simlar

Cxema



Internetda:

..:yuklash:.
Topshiriqlar kartochkasi
<http://ardx.org/BBL501>
..:tomosha qilish:.
videos
<http://ardx.org/VIDE01>



Fayl > Obrazci > 1.Basic > Blink

(bu namunalar arduino.cc saytidan, bu saytda yana ko`plab proektlar kodlari bor)

```

/* pirpirash (o`chib yonish)
 * svetodiodni 1 sekund yoqadi, 1 sekund o`chiradi,
 * funksiya takrorlanishiga buyruq beriladi
 * Created 1 June 2005 By David Cuartielles
 * http://arduino.cc/en/Tutorial/Blink
 * based on an original by H. Barragan for the wiring i/o board
 */

```

```
int ledPin = 13;    // svetodiod 13 oyoqcha chiqishiga ulangan
```

```

// boshlang`ich sozlamalar funksiyasi setup() sketch boshida 1 marta ishlaydi
void setup() {      // 13 kontaktini chiqish rejimiga sozlaymiz:
  pinMode(ledPin, OUTPUT);  }

```

```

// loop() funktsiyasi takroriy siklni amalga oshiradi
// hozircha Arduino manbaaga ulangan
void loop() {
  digitalWrite(ledPin, HIGH); // svetodiod yoqilishi
  delay(1000);                // 1 sekund to`xtashi
  digitalWrite(ledPin, LOW);  // svetodiod o`chishi
  delay(1000);                // 1 sekund to`xtashi
}

```

Ishlamayaptimi? (3 muammo va uning yechimi)**Svetodiod yonmayaptimi?**

Svetodiod, qutblari to`g`ri ulangandagina ishlaydi
Uni zanjirdan uzib, teskari ulang. Xavotirlanmang svetodiod teskari ulansa hech narsa qilmaydi.
Svetodiod aniq 13 portga ulanganiga ishonch xosil qiling.

Yozilgan sketch yuklanmadi?

Bu holat ba`zida yuz beradi.
Ko`proq bu holatning sababi noto`g`ri ko`rsatilgan port hisoblanadi.
Portni o`zgartirish menyusi:
Instrumenti>Port>

Shunda ham ishlamadimi?

Agar, sizda sababsiz nosozliklar kelib chiqsa, quyidagi manzilga pochta orqali savollaringizni yuboring

help@oomlout.com**Qurilmani chuqirroq o`rganamiz****Boshqaruvchi portni almashtiramiz:**

Svetodiod 13 portga ulangan, lekin biz Arduinoni xoxlagan portidan foydalanishimiz mumkin. Simni xoxlagan portlardan biriga ulang (0 dan 13 raqamli yoki 14 dan 19 analogli portlar). Keyin esa, komandani almashtiring:
int ledpin = 13; ni, int ledpin = port nomeri;
almashtiring. Sketchni arduinoga yuklang.

Pirpirash chastotasini o`zgartiramiz:

O`chib yonish vaqtini o`zgartirish uchun — delay(x) funksiyasining sonini o`zgartiring. To`xtash quyidagicha aniqlanadi: x = sekundlar soni * 1000.
Masalan: topshiriq uchun pirpirashni kichikroq chastotali sketchi ko`rinishi:

```

digitalWrite(ledpin,HIGH);
delay(2000);
digitalWrite(ledpin,LOW);
delay(2000);

```

Yonishning yorqinligini boshqarish:

Oddiy o`chib-yonishdan tashqari, svetodiodning yonish yorqinligini boshqarsa ham bo`ladi (kengroq ma`lumotni keyingi topshiriqlarda berib o`tamiz). Svetodiodni 9 portga ulang:

```

int ledpin = 13; o`zgartiring int ledpin = 9;
loop() protsedurasining figurali qavs ichidagi kodini quyidagicha o`zgartiring:
analogWrite(ledPin, new number);
newnumber — 0...255 diapazondagi xoxlagan son. 0 o`chirilgan svetodiodni anglatadi, 255 — maksimal yorqinlikni.

```

Yorqinlikni sekinlik bilan o`zgarish kodi:

Namunani oching:

Fayl > Obrazci > 3.Analog > Fading

Bu sketchni Arduinoga yuklab, hosil bo`lgan effektdan zavqlanang.

Yana savollar bormi?

Tafsilotlar, elektron detallar ma`lumoti, savollar bilan ishlash:

<http://ardx.org/CIRC01>



Ishning ta`rifi:

Biz svetodiodlarni pirpirashga majbur qildik, stavkani ko`tarish payti keldi! Keling birdaniga 8 ta svetodiodni ulaymiz. Bundan tashqari, ko`plab yorig`lik effektlari xosil qilishimiz mumkin bo`ladi. Bu sxema — Arduinoni o`rganishda, keyinchalik katta proektlar va eksperimentlar qilishda asos bo`la oladi. Svetodiodlarni Boshqarish bilan bir vaqtda, sketchni o`zgartirish, takomillashtirishni ko`rib chiqamiz.

for() loops - sketchni ma`lum bir qismini bir necha marta bajarish kerak bo`lganda ishltiladi.

arrays[] - o`zgaruvchilar bilan ishlashni osonlashtiradi (aslida o`zgaruvchilar guruhi bilan ishlashni).

Qurilma:

Elementlar:



Vazifa kartochkasi
CIRC-02
x1



2* kontaktli
razyom
x4



Svetodiod 5mm,
yashil
x8

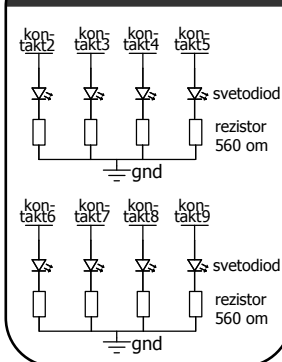


similar



Rezistor 560 Om
Yashil-ko`k-jigarrang
x8

Sxema



Internetda:

..yuklash:.

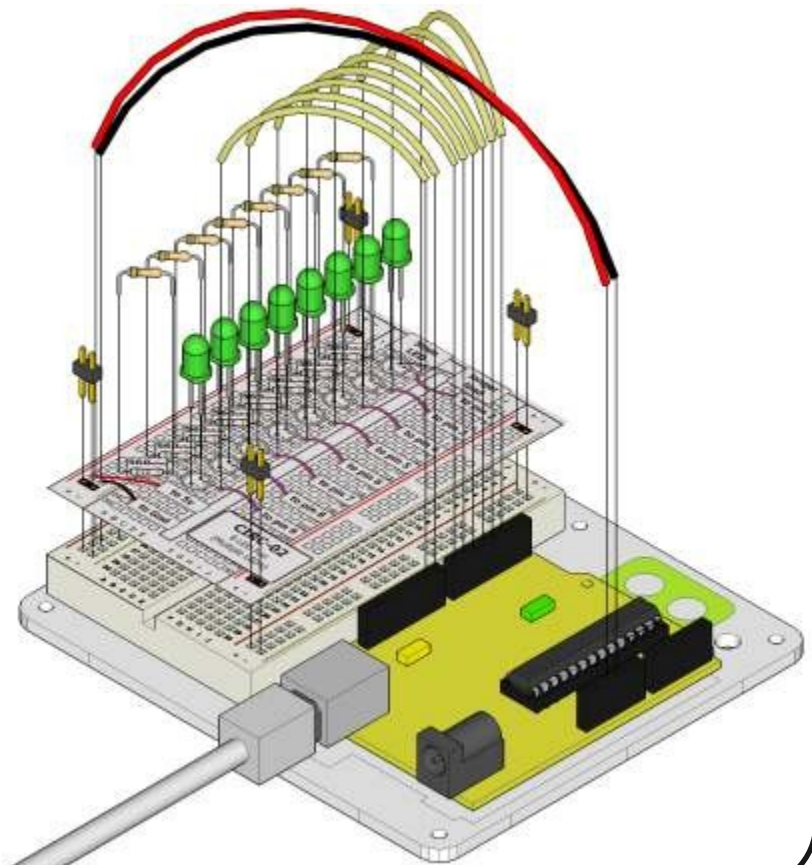
Vazifa kartochkasi

<http://ardx.org/BBL502>

..Tomosha qilish:.

video

<http://ardx.org/VIDE02>



Yuklab olish uchun manzil: <http://ardx.org/CODE02>

(tekstdan nusxa olib Arduino Sketch yangi oynasiga joylashtiring)

```
//svetodiodlar o'zgaruvchilari
int ledPins[] = {2,3,4,5,6,7,8,9};
//ko'plik, qaysi svetodiod
//qanday portga ulangani
//masalan, svetodiod №0 -
//2 portga

void setup()
{
  for(int i = 0; i < 8; i++){
    //8 marta qaytalanuvchi sikl
    pinMode(ledPins[i],OUTPUT);
    //chiqish rejimida bo'lgan, svetodiodlar
    //ulangan, port o'rnatiladi
  }
}

void loop() // tugallanmas sikl
{
  oneAfterAnotherNoLoop();
  //xar bir svetodiodni yoqadi va
  //xuddi shunday o'chiradi
  //oneAfterAnotherLoop();
  //oneAfterAnotherNoLoop qilgan vazifani bajaradi
  //lekin berilgan komanda kaltaroq
  //oneOnAtATime();
  //inAndOut();
}

/*
* oneAfterAnotherNoLoop() - bitta svetodiodni
```

```
Yoqadi, to'xtab turishi
delayTime da, keyingi
svetodiod yonishi,
o'chishi

void oneAfterAnotherNoLoop(){
  int delayTime = 100;
  //svetodiodni yoqishlar orasidagi
  //millisekunddagi to'xtash vaqti
  digitalWrite(ledPins[0], HIGH);
  //№0 raqamli svetodiodni yonishi
  //2 chi chiqishdagi

  delay(delayTime); //delayTime dagi to'xtash
  //(millisekundlarda)

  ...
  digitalWrite(ledPins[7], HIGH);
  //9 chiqishdagi, № 7 raqamli svetodiod yonishi

  delay(delayTime);
  //delayTime dagi to'xtash (mi;isekundlarda)
  //xar bir svetodiod o'chishi

  digitalWrite(ledPins[7], LOW); //№ 7 svetodiod
  o'chishi
  delay(delayTime); //delayTime dagi to'xtash
  //(millisekundlarda)

  ...
  -----to'liq dastur elektron versiyada-----
```

Ishlamayaptimi? (3 muammoning yechimi)

Ba'zi svetodiodlar yonmayapti

Adashtirib ulash juda oson, ishlamayotgan svetodiodni uzib qutblarini almashtirib ulang, qutblarni to'g'ri ulashga e'tibor bering.

Svetodiodlar noto'g'ri tartibda ishlayapti

При подключении восьми проводов легко можно перепутать пару. Убедитесь, что первый светодиод подключен к порту 2 и каждый последующий к следующему порту по порядку.

Boshidan boshlang

Ba'zida qurilmadagi nosozlikni qidirgandan ko'ra, uni yangidan yig'gan osonroq bo'ladi.

Qurilmani chuqurroq o'rganamiz

Sikllardan foydalanish:

loop() funksiyasida to'rtta komanda yozilgan. Oxirgi uchasi «//» simvol bilan boshlanadi. Bu, kompilyator qatorlarni (komentariy deb qaraydi) o'chirib yuborishini anglatadi. Sikllardan foydalanish uchun, programma Tekstini quyidagicha o'zgartiring:

```
//oneAfterAnotherNoLoop();
oneAfterAnotherLoop();
//oneOnAtATime();
//inAndOut();
```

Sketchni Arduino ga yuklang va e'tibor bering Programma oldingidek ishlayapti.

Ushbu ikki funksiyaga e'tibor qarating, ular bir xil bajarilayapti, lekin xar xil prinsipga asosan ishlayapti (ikkinch funksiya for strukturasini beradi)

Qo'shimcha effektlar:

Ushbu effektdan zerikdingizmi? Ikki namuna bilan ishlashga urinib ko'ring. Oxirgi ikki qatordagi komentariyadagi (//) belgilarni olib tashlab, sketchni arduinoga yuklang va ushbu effektdan baxra oling.

O'zingizni effektingizni yarating:

Sketchdagi biror tekstni o'zgartirishga harakat qiling. Svetodiodlarni yoqish uchun asosiy g'oya: komanda digitalWrite(pinNumber, HIGH); va o'chirish uchun komanda digitalWrite(pinNumber, LOW); Nima bo'lgan taqdirda ham qo'rqmang, chunki siz nimanidir o'zgartirsangiz ham hech narsa buzilmaydi.

Yana savollar bormi?

Tafsilotlar, elektron detallar ma'lumoti, savollar bilan ishlash:

<http://ardx.org/CIRC02>

..Motorlar bilan ishlash (tranzistor va motor):.



Ishning ta`rifi:

Arduino porti kam quvvatga ega elementlar uchun eng qulay hisoblanadi (masalan, svetodiod). Agar, katta quvvatga ega elementlar (masalan, motorlar) bilan ishlash uchun qo`shimcha element Tranzistorlar qo`llaniladi. Tranzistor foydali element hisoblanib, o`zi kichik tok sarflab, katta quvvatli yuklanmalarni ishga tushiradi.

Tranzistor uchta oyoqchadan iborat. NPN tipdagi tranzistorlarda yuklanmani kollektorga, emitterni esa, «yerga» (nol potensial, massa, manfiy qutb) ulanadi.

Bazadan emitterga kichik miqdorda tok oqib o`tsa (agar, bazaga Arduinodagi HIGH signali ulansa), Tranzistor «ochiladi», tok tranzistordan oqib o`tib motorni ishga tushiradi.

Tranzistorlarning juda ko`p xili mavjud bo`lib, ular hozirda elektronika, texnologiyalarning juda ko`p soxalarida keng miqyosda ishlatiladi. Biz uchun P2N2222AG markadagi tranzistor kerak bo`ladi.

Tranzistor yuqori kuchlanishdagi ishlashi 40 voltni tashkil qiladi. Maksimal tok esa, 600 mA.

Bu parametrlar, Arduino orqali motorni boshqarishda yetarli.

Ushbu tranzistor haqida ko`proq ma`lumotga ega bo`lmoqchi bo`lsangiz, ushbu saytga tashrif buyuring: <http://ardx.org/2222>.

Tokni o`rgaga oqishini saqlash uchun 1N4001 markali diod ishlatiladi. Ko`proq ma`lumot uchun: <http://ardx.org/4001>

Qurilma:

Elementlar:



**Vazifa kartochkasi
CIRC-03
x1**



**2` kontaktli
razyom
x4**



**Tranzistor
P2N2222AG (TO92)
x1**



Simlar



**Kollektorli
Dvigatel
x1**

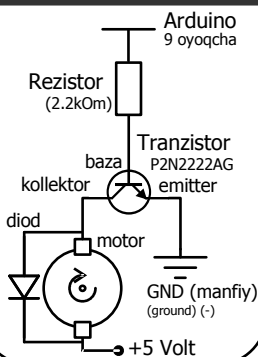


**Diod
(1N4001)
x1**



**Rezistor 2.2 kOm
qizil-qizil-qizil
x1**

Sxema



Internetda:

..yuklash..

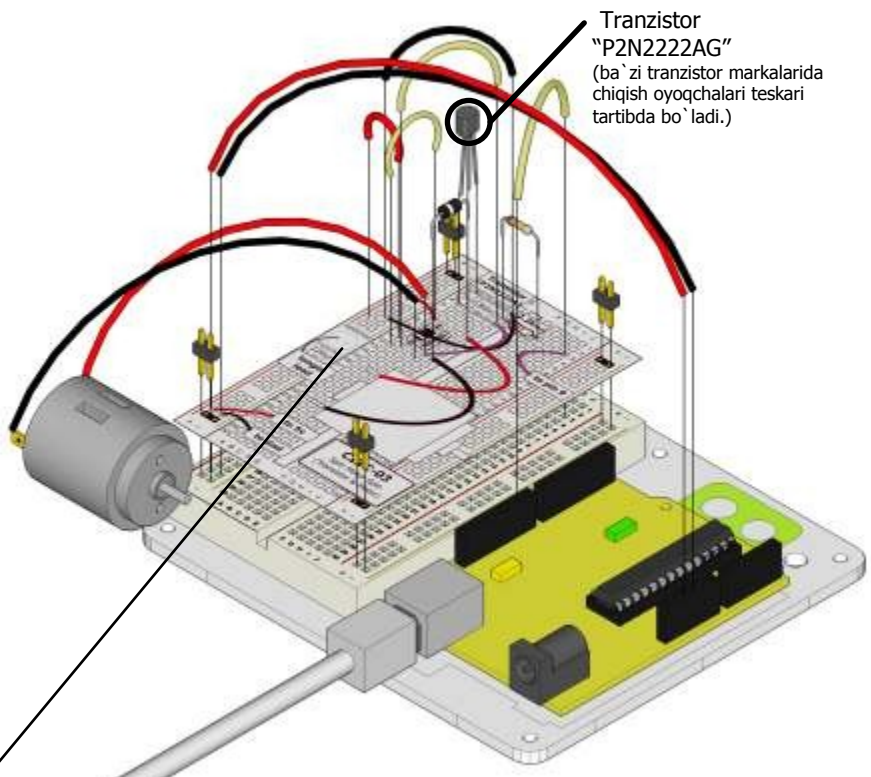
Vazifa kartochkasi

<http://ardx.org/BBLS03>

..Tomosha qil..

Video

<http://ardx.org/VIDE03>



..Eslatma: agar, motorni ishga tushirganda, Arduino perezagruska bo`lsa yoki qotib, ishlasa, motorni kondensator bilan qo`shib ulang:.

Kodni yuklash uchun: <http://ardx.org/CODE03>

(matnni Arduino Sketch yangi oynasiga ko`chirib o`ting)

```
int motorPin = 9; //motor 9 razyomga ulangan

void setup() //bir martta bajariladi
{
  pinMode(motorPin, OUTPUT);
}

void loop() //cheksiz bajariladi
{
  motorOnThenOff();
  //motorOnThenOffWithSpeed();
  //motorAcceleration();
}

/* motorOnThenOff() - motorni yoqadi keyin
Uni o`chiradi (e`tibor bering, kod svetodiod uchun
ishlatgan kodimizga o`xshaydi)
*/
void motorOnThenOff(){
  int onTime = 2500;
  //yoniq holatdagi vaqt
  int offTime = 1000;
  //o`xchiq holatdagi vaqt
  digitalWrite(motorPin, HIGH);
  // motorni yonishi
  delay(onTime);
  // onTime dagi kechikish (zaderjka) millisekunda
  digitalWrite(motorPin, LOW);
  // motoni o`chishi
  delay(offTime); //Off Timedagi zaderjka millisekunda
}
```

```
void motorOnThenOffWithSpeed(){
  int onSpeed = 200;
  // 0 (to xTash) va 255 (maksimal tezlik) orasidagi son
  int onTime = 2500;
  int offSpeed = 50;
  // 0 (to xTash) va 255 (maksimal tezlik) orasidagi son
  int offTime = 1000;
  analogWrite(motorPin, onSpeed);
  // motor yonishi
  delay(onTime); //onTimedagi zaderjka millisekunda
  analogWrite(motorPin, offSpeed);
  // motor o`chishi
  delay(offTime); //offTimedagi zaderjka millisekunda
}

void motorAcceleration(){
  int delayTime = 50; // tezlik orasidagi zaderjka
  for(int i = 0; i < 256; i++){
    //tezlikni 0 dan 255 gacha o`zgartiradi
    analogWrite(motorPin, i); //sets the new speed
    delay(delayTime); //delayTimedagi zaderjka millisekunda
  }
  for(int i = 255; i >= 0; i--){
    //tezlikni 255 dan 0 gacha o`zgartiradi
    analogWrite(motorPin, i); //yangi tezlik o`rnatish
    delay(delayTime); //delayTimedagi zaderjka millisekunda
  }
}
```

Ishlamayaptimi? (3 muammo va yechimi)

Motor aylanmayapti?

Tranzistorning oyoqchalari to`g`ri ulanganiga e`tibor bering.
Ko`p tranzistorlar oyoqchalar tartibi turlicha bo`ladi.

Shunda ham ishlamayapti?

Siz tanlangan motor 5 Volt dan kam kuchlanishga mo`ljallangan bo`lishi kerak.

Ishlaydimi o`zi?

Arduino ba`zida o`zi kompyuterdan uzilib, o`chib qoladi.
Arduino USB kabelini qaytadan uzib, ulab ko`ring.

Qurilmani chuqurroq o`rganamiz.

Tezlikni boshqarish:

Biz oldin Arduino yordamida svetodiod yorqinligini boshqarishni o`rganganmiz. Shu prinsipni motorni boshqarish uchun ham qo`llash mumkin. Arduino, KIM (keng impulsli modulyatsiya) deb nomlanuvchi signalni shakllantiradi. Chiqishda, yuqori chastotali logik nol va bir ketma - ketligda xosil bo`ladi.

Bir va nol qiymat sonining balansi, natijaviy kuchlanishni beradi. Masalan, 2.5 V kuchlanishni hosil qilish uchun birlik vaqt ichida nol qiymat soni bir qiymat soniga teng bo`lishi kerak.

Sketchdagi loop() funksiyasi matnini quyidagicha o`zgartiring:

```
// motorOnThenOff();
motorOnThenOffWithSpeed();
// motorAcceleration();
```

Arduinoga yuklang. Siz onSpeed va offSpeed o`zgaruvchilari orqali motorning aylanish tezligini boshqarishingiz mumkin.

Tezlanish va sekinlashish :

Nima uchun bu ikki parametrga e`tibor berishingiz kerak? Bu texnologiya orqali siz motorni tezlashtirib, sekinlashtirib mumkin. Tekshirish uchun sketchdagi loop() funksiya matnini quyidagicha o`zgartiring:

```
// motorOnThenOff();
// motorOnThenOffWithSpeed();
motorAcceleration();
```

Bu programmani yuklab, motorni maksimal tezlikka sekinlik bilan chiqib borishini va to`xtashini guvohi bo`lishingiz mumkin. Agar, siz tezlik o`zgarishini boshqarmoqchi bo`lsangiz delayTime o`zgaruvchisini (kattaroq qiymat kichikroq tezlanishga mos keladi) o`zgartiring.

Yana savollar bormi?

Tafsilotlar, elektron detallar ma`lumoti, savollar bilan ishlash:

<http://ardx.org/CIRC03>



Ishning ta`rifi:

Motorni aylantirish yetarli darajada qiziqarli, lekin biz harakat qismlardan iborat proekt qilayotganimizda, ko`proq funksiyaga ega element zarur bo`ladi. Servomotor xuddi shunday maqsadda ishlatiladi. Servomotorlar hozirda robotlardan tortib, o`yinchoqlarning harakat qismlariga o`rnatiladi. Servomotor ichiga kichik reduktor (quvvatni oshirish uchun) va bir nechta elektronik komponentlar (boshqarishga qulay) o`rnatilgan. Standart servomotor burilish burchagi 0 dan 180 gradusni tashkil qiladi. Burilish burchagini boshqarish uchun impuls beriladi, 1.25 ms (0 gradus) dan 1.75 (180 gradus), 1.5 ms (90 gradus). Bunday vaqtlar ishlab chiqariluvchi tomonidan har xil qilib o`rnatilishi mumkin. Agar, impulsni xar 25...50 ms da jo`natsa, servomotor tekis harakat xarakat qiladi. Arduining qulay tomoni shundaki, ikkita servomotorlarni bir vaqtda osongina boshqarish uchun, programma bibliotekasida podprogramma mavjudligidir.

Qurilma:

Komponentlar:



Vazifa kartochkasi
CIRC-04
x1



2x kontaktli
razyom
x4



3x kontaktli
razyom
x1

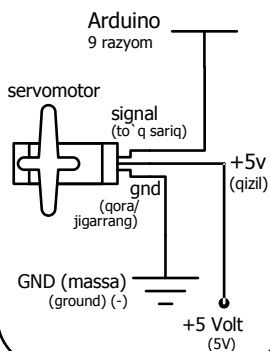


Simlar



Servomotor
x1

Sxema



Internetda:

..yuklash:.

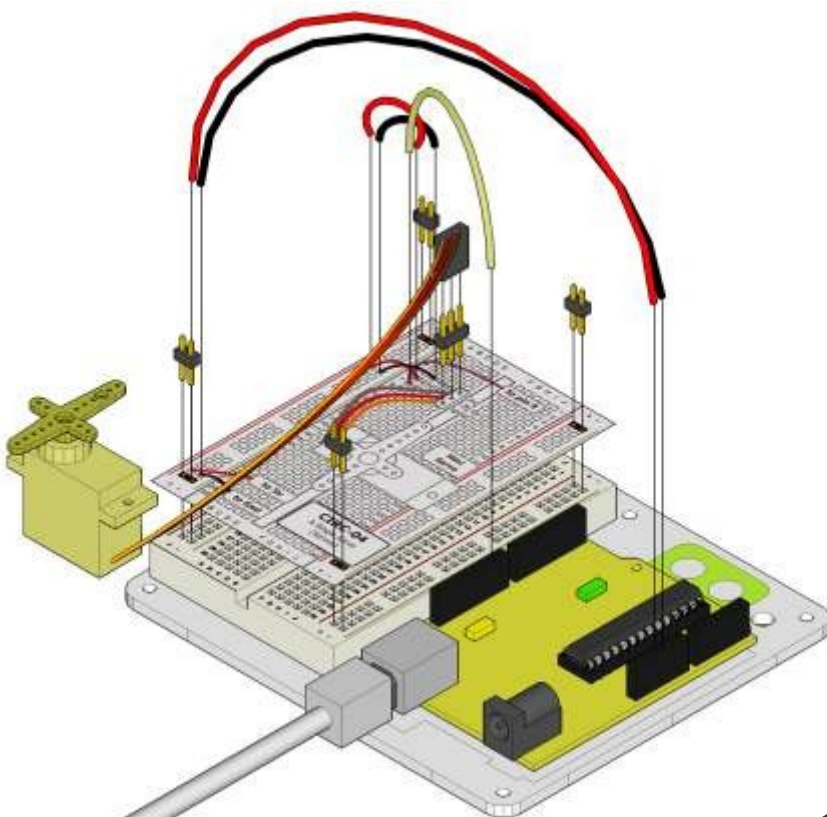
Vazifa kartochkasi

<http://ardx.org/BBL504>

..Tomosha qilish:.

Video

<http://ardx.org/VIDE04>



File > Examples > Servo > Sweep

(bu namunalar arduino.cc saytidan, bu saytda yana ko`plab proektlar kodlari bor)

//servomotor burlishi (by BARRAGAN <<http://barraganstudio.com>>)

```
#include <Servo.h>
Servo myservo; //servomotor uchun «sevo» obektini yaratish
int pos = 0; //servo pozitsiyasini saqlash uchun o`zgaruvchi

void setup() {
  myservo.attach(9); //9 oyoqchani servo obektni boshqarish uchun o`rnatiladi
}

void loop() {
  for(pos = 0; pos < 180; pos += 1) //0° dan 180° gacha 1° qadam bilan o`zgartiriladi
    myservo.write(pos); //servomotorga buyruq
    //pozitsiyaga 'pos' o`zgaruvchi bilan o`tish
    //berilgan pozitsiyani 15 ms zaderjkasi
    delay(15);
  position
}
for(pos = 180; pos>=1; pos-=1) //180° dan 0° ga o`zgartiriladi
{
  myservo.write(pos); //servomotorga buyruq
  //pozitsiyaga 'pos' o`zgaruvchi bilan o`tish
  //berilgan pozitsiyani 15 ms zaderjkasi
  delay(15);
}
}
```

Ishlamayaptimi? (3 muammo va yechimi)**Servomotor aylanmayaptimi?**

Xatto rangli simlar markirovkasi bilan ham oyoqchalarni oson adashtirish mumkin. Simlarni to`g`ri ulab ko`ring.

Shunda ham ishlamayaptimi?

Manbaani ulash kerak! Qizil va jigarrang simlar проводники должны быть подключены к +5В и к керек.

Agar servomotor siltanib harakatlanyaptimi?

Agar, servomotor harakat qilayotgan bo`lsa, Arduino svetodiody o`chib yonsa, manbaa quvvati yetmayotgan bo`ladi. USB orqali emas, alohida razyomga manbaani ulang.

Qurilmani chuqurroq o`rganamiz.**Potensiometr orqali boshqarish:**

Biz hali kirish signallarini boshqarishni o`rganmadik, agar sinamoqchi bo`lsangiz, namunani yuklang **File > Servo > Knob**. Bu namunada potensiometr (CIRC 08) dan foydalanadi. Chuqurroq tanishish uchun: <http://ardx.org/KNOB> manzilga kiring.

To`g`ridan-to`g`ri boshqarish:

Programmadagi biblioteka orqali servomotorni boshqarish osonroq. Ba`zida boshqarish uchun mustaqil programma tuzish ham kerak bo`ladi. Servomotorni bevosita boshqarish mukin, agar Arduinoning xoxlagan portiga kerakli davomiylikda impuls jo`natilsa. Sketch namunasi:

```
int servoPin = 9;
void setup() {
  pinMode(servoPin, OUTPUT);
}
void loop() {
```

```
int pulseTime = 2100;
//zaderjka kattaligi mikrosekundda
//900 mks 0 gradus uchun, 1500 mks 90 gradus uchun,
//2100 mks 180 gradus uchun
digitalWrite(servoPin, HIGH);
delayMicroseconds(pulseTime);
digitalWrite(servoPin, LOW);
delay(25);
}
```

Ajoyib g`oyalar:

Servomotor juda ko`p g`oyalarda keng qo`llanishi mumkin. Bir qancha proektlarni namunasi:

Yangi yil hisoblagichi
<http://ardx.org/XMAS>

Robot manipulyatori (Arduino va servomotor bilan yasaluvchi proekt)
<http://ardx.org/RARM>

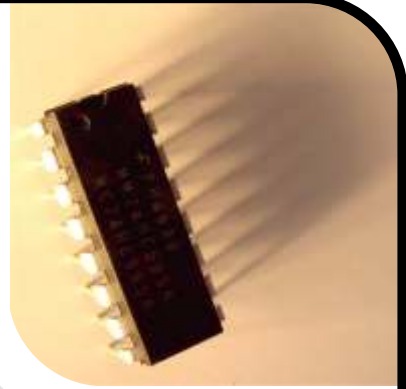
Servomotorli qadam tashlovchi robot
<http://ardx.org/SEWA>

Yana savollar bormi?

Tafsilotlar, elektron detallar ma`lumoti, savollar bilan ishlash:

<http://ardx.org/CIRC04>

8-ta svetodiodlar (74HC595 qadamli Registrlar):.



Ishning ta`rifi:

Mikrosxemalar bilan tanishish vaqti keldi. Mikrosxemaning tashqi ko`rinishi chalkash ko`rinadi. Masalan, Arduino platasidagi kontroller mikrosxemasi va qadamli registr mikrosxemasi juda o`xshash, lekin xarakteristikasi bo`yicha bir biridan juda farq qiladi. Atmega mikrokontroller narxi bir necha dollar turadi, 74HC595 qadamli registr mikrosxemasi narxi esa sentlarda baholanadi.

Qadamli registrni o`rganish boshlanishiga juda qulay hisoblanadi. Agar, siz ushbu mikrosxemani chuqur o`rgansangiz, siz uchun mikrosxema dunyosi qiyindek tuyilmaydi. Qadamli registr (ba`zida uni Ketma ket interfeysni parallelga o`zgartiruvchi ham deyishadi) Arduining 3 ta raqamli portidan foydalanib, sizga (svetodiodlar yoki shunga o`xshash narsalarni boshqarish uchun) 8ta qo`shimcha raqamli oyoqcha hosil qiladi. Yana ushbu registrni ketma- ket ulash mumkin, bunda Arduinoni 3ta oyoqchasidan foydalanib, ko`plab raqamli oyoqchalarni hosil qilish mumkin. Qadamli registrdan foydalanish uchun unga, ketma-ket interfeys orqali qiymat yozish kerak va yozilgan qiymatni parallel interfeys orqali chiqishga yuborish kerak. Ketma-ket interfeys o`z navbatida 2 ta chiqishni: ma`lumotlar chiqishi va taktlar chiqishi ni tashkil qiladi. Bunday interfeys orqali baytni uzatish uchun, qadamli registr chiqishiga tartibli ravishda sath (Level) o`rnatish kerak, bit va bayt mos keluvchi qiymatini impuls tariqasida taktili chiqishga yuborish kerak. Barcha baytlar qadamli registrdan o`tgandan keyin, parallel interfeysdan o`tish uchun buyruq beriladi. Ko`proq informatsiyalar bilan tanishish uchun quyidagi adressga kirishingizni so`raymiz: <http://ardx.org/SHIF>.

Qurilma:

Komponentlar:



Vazifa kartochkasi
CIRC-05
x1



2* kontaktli
razyom
x4



74HC595
qadamli
registr
mikrosxema
x1



Simlar

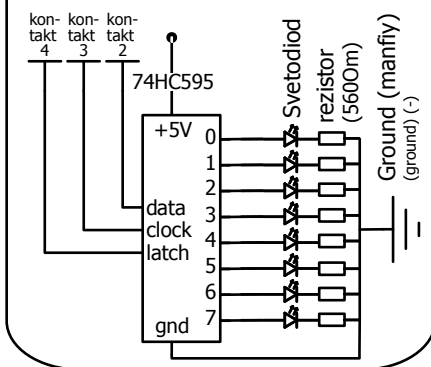


qizil
svetodiod
x8



Rezistor 560 Om
Yashil-Ko`k-Jigarrang
x8

Sxema



Internetda:

..yuklash..

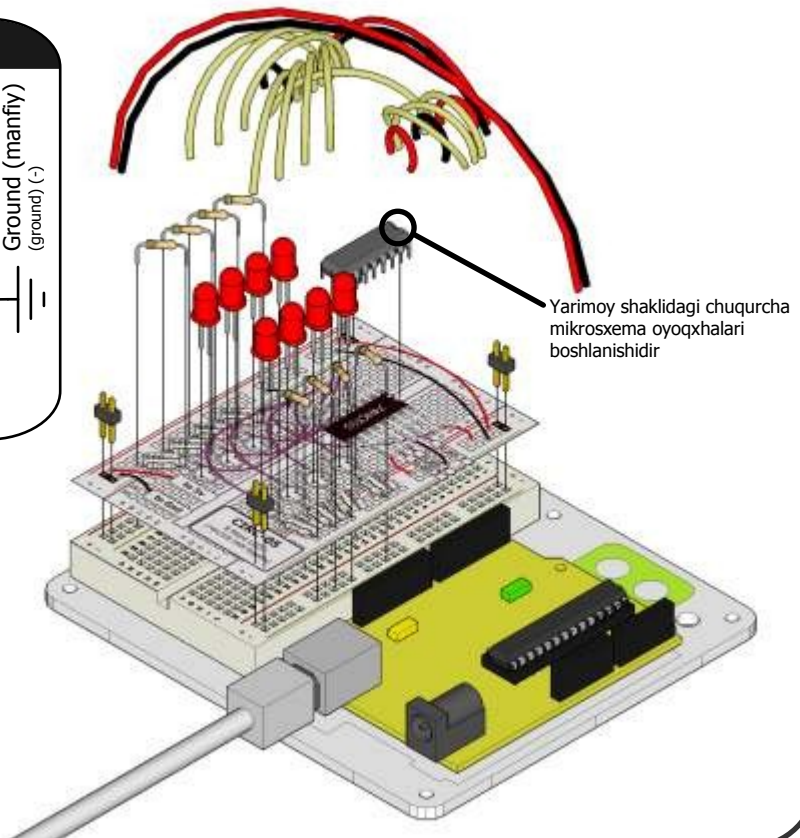
Vazifa kartochkasi

<http://ardx.org/BBLS05>

..Tomosha qil..

Video

<http://ardx.org/VIDE05>



Kodni yuklash uchun adres: <http://ardx.org/CODE05>

(matnni Arduino Sketch yangi oynasiga ko`chirib o`ting)

```
Razyomlarni aniqlash
//74HC595 SPI protokolidan foydalanildi
//3 razyomdagisi:
int data = 2;
int clock = 3;
int latch = 4;

void setup()
//funktsiya faqat bir marta ishlatiladi
{
  pinMode(data, OUTPUT);
  pinMode(clock, OUTPUT);
  pinMode(latch, OUTPUT); }

void loop()
//funktsiya cheksiz ishlatiladi
{
  int delayTime = 100;
  //svetodiodlar yangilanishidagi zaderjka
  for(int i = 0; i < 256; i++){
    updateLEDs(i);
    delay(delayTime); }
}
```

```
yuboriladi
*/

void updateLEDs(int value){

  digitalWrite(latch, LOW);
  //signal tirqishi LOW ga o`rnatilgan

  shiftOut(data, clock, MSBFIRST, value);
  //qadamli registrga 8 bit jo`natadi

  digitalWrite(latch, HIGH);
  //tirqish ochiladi – informatsiya kiritiladi
}

-----programma to`liq versiyasi elektron holatda-----
```

Ishlamayaptimi? (3 muammo va yechimi)

Arduinoning manbaa Indikatori yonmayapi

Qadamli registr mikroshemasini teskari ulangan. Agar, siz tezda manbaani uzib, mikroshemani qayta ulasangiz — xech narsa o`zgaraydi.

Noto`g`ri ishlayapti?

Simlar chalkashtirib ulangan bo`lsa ham, mikroshema ishlamasligi ham mumkin.

O`xshamadimi?

Sizni qiynayotgan muammolaringizni quyidagi manzilga jo`nating.
help@oomlout.com

Qurilmani chuqurroq o`rganamiz.

Topshiriqni qiyinlashtiramiz:

Arduino qiyin narsalarni nisbatan osonlashtirish mumkin. Ketma- ket interfeysda ma`lumot jo`natish — shunday topshiriqlardan biri. Buning boshqacha ko`rinishda ham qiyinroq qilib, tuzishingiz mumkin (o`zingizni xoxishingiz bo`yicha). Sketchdagi matnni quyidagicha o`zgartirib ko`ring:
updateLEDs(i); ni updateLEDsLong(i); ga o`zgartiring. Sketchni kontrollerga yuklab, siz hech narsa o`zgarmaganligiga ishonch hosil qilishingiz mumkin.
updateLEDsLong(i); funksiya matniga e`tibor bering, ma`lumotlar bir bitda yuklanayapti (ko`proq ma`lumot: <http://ardx.org/SPI>).

Svetodiodlarni mustaqil boshqaruvi:

CIRC02 vazifasiga qarab svetodiodlarni boshqaramiz. barcha 8 ta svetodiodlar bir bayt (8 bit) holatda saqlanadi.
при помощи одного байта (8 бит). Arduino может легко

Ikkilik operatsiyalar to`g`risida to`liq ma`lumotga ega bo`lish uchun quyidagi adressga kiring:

<http://ardx.org/BITW>.

Sketchdagi matnni quyidagicha o`zgartiring:

```
int delayTime = 100;
//sikllar orasidagi zaderjka vaqti mks da
for(int i = 0; i < 8; i++){
  changeLED(i,ON);
  delay(delayTime);
}for(int i = 0; i < 8; i++){
  changeLED(i,OFF);
  delay(delayTime);
}
```

Programmani Arduinoga yuklang. Svetodiodlar ketma-ket yonishi va o`chishi kerak.

Qo`shimcha effektlar:

Siz effektlar podprogrammasini CIRC02 dan ko`chirishingiz mumkini.
digitalwrite(led,state) ni changeLED(led,state) ga.

Yana savollar bormi?

Tafsilotlar, elektron detallar ma`lumoti, savollar bilan ishlash:

<http://ardx.org/CIRC05>



Ishning ta`rifi:

Biz shu paytgacha elektronlar, yorug`lik, harakatlarni boshqardi. Keling, ovozni boshqarib ko`ramiz. Musiq bu analogli hodisa hisoblanadi, как известно. Raqamli Arduino bilan musiqalarni boshqarish imkoni bormi? Mikrokontrollarning yuqori ishlash tezligi buning imkoni borligini ko`rsatadi. Pyezoelement, unga elektr impulslari kelganda shivirlaydi. Agar, ma`lum chastotada impulslar jo`natilsa (masalan, La notasini kuylash uchun 440 marta signal yuborilsa), shivirlashlar musiqali tonga aylanadi. Keling, Arduinoni qanaqadir melodiyanı kuylash uchun majbur qilamiz.

Qurilma:

Komponentlar:



Vazifa kartochkasi
CIRC-06
x1



2x kontaktli
razyom
x4



pyezoelement
x1



simlar

Sxema



Arduino
9 kontakt



Pyezo-
element



Massa
(ground) (-)

Internetda:

..yuklash..

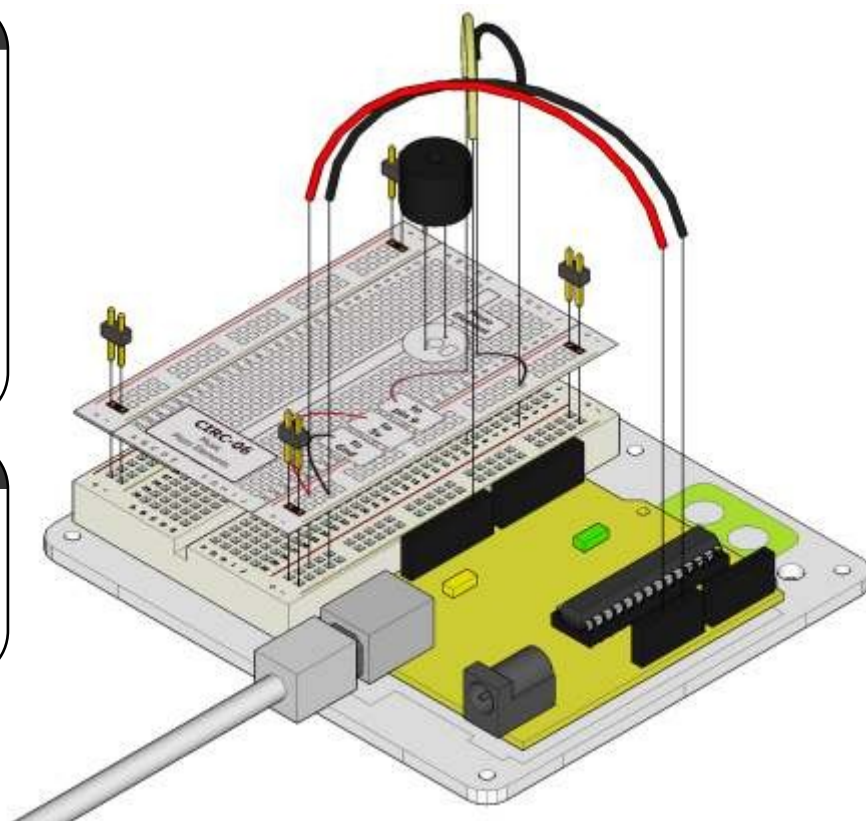
Vazifa kartochkasi

<http://ardx.org/BBL506>

..Tonosha qil ..

Video

<http://ardx.org/VIDE06>



kodni yuklash uchun: <http://ardx.org/CODE06>

(matnni Arduino Sketch yangi oynasiga ko`chirib o`ting)

```
/* Musiq
 * (cleft) 2005 D. Cuartielles for K3
 *
 * Bu namunada biz pyeoelement yordamida musiqa
 * yaratamiz. U chastotaga mos ravishda KIM (keng impulsli
 * modulyatsiya) signal yuboradi, natijada musiqali ton hosil
 * bo`ladi. Kerakli zaderjkani hisoblash uchun quyidagi
 * formulani yozamiz:
 *
 *      timeHigh = period / 2 = 1 / (2 * toneFrequency)
 *
 * zaderjka quyidagi jadval bo`yicha kiritiladi:
 *
 * nota      chastota (period)      timeHigh
 * c          261 Hz                 3830    1915
 * d          294 Hz                 3400    1700
 * e          329 Hz                 3038    1519
 * f          349 Hz                 2864    1432
 * g          392 Hz                 2550    1275
 * a          440 Hz                 2272    1136
 * b          493 Hz                 2028    1014
 * c          523 Hz                 1912    956
 *
 * http://www.arduino.cc/en/Tutorial/Melody
 */

int speakerPin = 9;
int length = 15; // nota soni
char notes[] = "ccggaagffeeddc "; // probel pauzani belgilaydi
int beats[] = { 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 4 };
int tempo = 300;

void playTone(int tone, int duration) {
  for (long i = 0; i < duration * 1000L; i += tone * 2) {
    digitalWrite(speakerPin, HIGH);
    delayMicroseconds(tone);
  }
}
```

```
digitalWrite(speakerPin,
LOW);
delayMicroseconds(tone);
}

void playNote(char note, int duration) {
  char names[] = { 'c', 'd', 'e', 'f', 'g', 'a', 'b', 'c' };
  int tones[] = { 1915, 1700, 1519, 1432, 1275, 1136, 1014, 956
};
  // nota nomiga mos ton kuylash
  for (int i = 0; i < 8; i++) {
    if (names[i] == note) {
      playTone(tones[i], duration);
    }
  }
}

void setup() {
  pinMode(speakerPin, OUTPUT);
}

void loop() {
  for (int i = 0; i < length; i++) {
    if (notes[i] == ' ') {
      delay(beats[i] * tempo); // pauza
    } else {
      playNote(notes[i], beats[i] * tempo);
    }
    // notalar orasidagi pauza
    delay(tempo / 2);
  }
}
```

Ishlamayaptimi? (3 muammo va yechimi)

Ovoz chiqmayapti

Pyeoelement o`lchamlari va tuzilishi tufayli uning oyoqchalarini oson adashtirish mumkin. Pyeoelement to`g`ri ulanganiga e`tibor bering.

Musiqa chalinmayapti

Pyeoelementni bredborddan uzib, to`g`ri ulab ko`ring. To`g`ri ulanganda ishlashi kerak.

Ushbu musiqadan charchadingizmi?

Programma tayyor, siz faqat programmadagi ba`zi bir notani o`zgartiring.

Qurilmani chuqurroq o`rganamiz

Tezlik o`zgarishi:

Kuylash tezligini oshirish uchun, quyidagi qatorni o`zgartirish kerak

```
int tempo = 300; --->
int tempo = (yangi qiymat)
```

Tempni kamaytirish uchun qiymatni oshiramiz, tempni oshirish uchun qiymatni kamaytiramiz.

Notani sozlash uchun:

Agar, siz biror notani nosoz chiqishini sezsangiz, uni to`g`irlashingiz mumkin. Buning uchun tones[] massividagi o`zgaruvchini o`zgartirsangiz bas.

```
char names[] = { 'c', 'd', 'e', 'f', 'g', 'a', 'b', 'c' };
int tones[] = { 1915, 1700, 1519, 1432, 1275, 1136, 1014, 956 };
```

Shaxsiy melodiyni yozish:

Programmada 'Twinkle Twinkle Little Star' kuyi yozilgan, lekin kuyni o`zgartirish mumkin. Xar qanday kuy bitta o`zgaruvchi (nota soni -int length) va ikki massiv bilan o`zgaradi. Bitta massiv, notes[] notalar ketma-ketligini beradi. boshqasi, beats[] ijro etish davomiyligini. Masalan:

Twinkle Twinkle Little Star

```
int length = 15;
char notes[] = {"ccggaagffeeddc "};
int beats[] = { 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 4 };
```

Happy Birthday (first line)

```
int length = 13;
char notes[] = {"ccdcfcgccgcf "};
int beats[] = {1,1,1,1,1,2,1,1,1,1,1,2,4};
```

Yana savollar bormi?

Tafsilotlar, elektron detallar ma`lumoti, savollar bilan ishlash:

<http://ardx.org/CIRC06>



Ishning ta`rifi:

Biz shu paytgacha boshqarishning turli variantlari bilan ishladik. Endi esa, Arduinoni eshitishga, ko`rishga, xis qilishga o`rgatish vaqti keldi. Keling oddiy knopkadan boshlaymiz ishimizni.

Knopkani ulash juda ham oson. Faqatgina bitta komponent, rezistor kerak. Gap shundaki, Arduino informatsiyani biz tushungan darajada tushuna olmaydi. U knopka bosilganmi yo`qmi bilmaydi, ustiga ustak u, faqatgina o`zining kirish qismidagi kuchlanish signallarini boshqara oladi. Knopkani bosilishi, kirishdagi kuchlanish uzilishini (LOW) bildiradi, knopka qo`yib yuborilganda, kuchlanish maksimal qiymatga erishadi (HIGH). Rezistor, knopka qo`yib yuborilganda, Arduino chiqishidagi logik qiymatni chegaralab turish uchun xizmat qiladi.

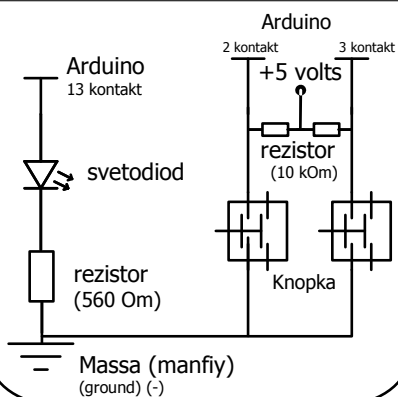
E`tibor bering: birinchi namunada faqat bitta knopkadan foydalanilyapti.

Qurilma:

Komponentlar:

 Vazifa kartochkasi CIRC-07 x1	 Rezistor 10 kOm jigarrang-qora-to`q sariq x2	 Rezistor 560 Om yashil-ko`k-jigarrang x1	 Провод
 Knopka x2	 Qizil svetodiod x1	 2` kontaktli razyom x4	

Sxema



Internetda:

..yuklash..

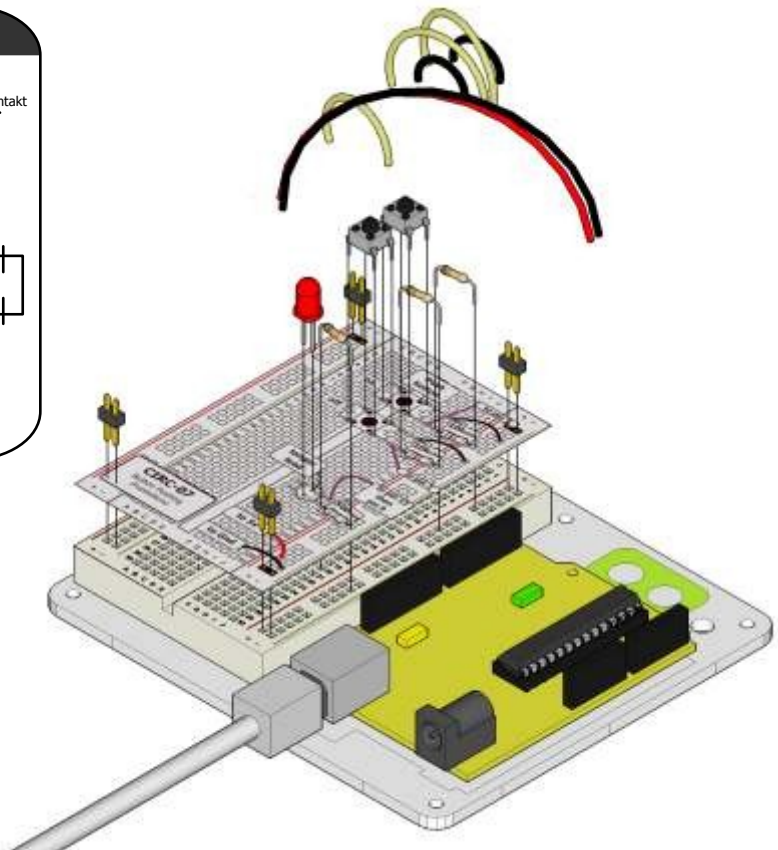
Vazifa kartochkasi

<http://ardx.org/BBL507>

..Tomosha qil..

video

<http://ardx.org/VIDE07>



File > Examples > 2.Digital > Button

(bu namunalar arduino.cc saytidan, bu saytda yana ko`plab proektlar kodlari bor)

```
/*
 * Knopka
 * by DojoDave <http://www.0j0.org>
 *
 * Knopka bosilishi orali svetodiodni yoqib-o`chiramiz
 * http://www.arduino.cc/en/Tutorial/Button
 */
int ledPin = 13;           // svetodiod chiqish nomerini yozamiz
int inputPin = 2;          // knopkaga ulangan chiqish nomerini yozamiz
int val = 0;               // knopka holatini saqlash uchun o`zgaruvchi

void setup() {
  pinMode(ledPin, OUTPUT);
  // svetodiodga ulangan, chiqish vazifasidagi, kontakt hosil qilamiz
  pinMode(inputPin, INPUT);
  // knopkaga ulangan, kirish vazifasidagi, kontakt hosil qilamiz
}

void loop(){
  val = digitalRead(inputPin); // knopka chiqishidagi qiymatni hisoblaymiz
  if (val == HIGH) {           // knopka bosilganligini tekshiramiz
    digitalWrite(ledPin, LOW); // svetodiod o`chishi
  } else {
    digitalWrite(ledPin, HIGH); // svetodiod yonishi
  }
}
```

Ishlamayaptimi? (3 muammo va yechimi)

Svetodiod

yonmayapti

Knopka kvadrat shaklida, uni 90 gradusga burib, qayta ulash mumkin. Knopkani to`g`ri ulanganiga e`tibor bering.

Svetodiod bir

tekisda yonmayapti (o`chadi)

Bu vazifa uchun svetodiodni 13 portdan, 9 portga olib ulang.

O`xshamadimi?

Agar, har ikkala holda ham ushbu qurilma yonmasa yoki ishlamasa, svetodiodning o`zi ishdan chiqqan bo`ladi. Yangi svetodiod olib qurilmani qayta yig`ing.

Qurilmani chuqurroq o`rganamiz.

O`chiruvchi knopka, yoquvchi knopka:

Birinci namuna ozgina hayron qoldirishi mumkin (masalan, «menga bu ishni qilish uchun Arduino shart emas»), keling, uni ozgina Takomillashtiramiz. Birinci knopka svetodiodni yoqadi, ikkinchisi o`chiradi. Sketchni quyidagicha tarzda o`gartiring:

```
int ledPin = 13;
//svetodiod uchun razyom tanlang
int inputPin1 = 3; // 1 knopka
int inputPin2 = 2; // 2 knopka

void setup() {
  pinMode(ledPin, OUTPUT);
  // svetodiod chiqishga yoziladi
  pinMode(inputPin1, INPUT); // 1 knopka - chiqish
  pinMode(inputPin2, INPUT); // 2 knopka - chiqish
}

void loop(){
  if (digitalRead(inputPin1) == LOW) {
    digitalWrite(ledPin, LOW); // svetodiodni
    o`chirish
  } else if (digitalRead(inputPin2) == LOW) {
    digitalWrite(ledPin, HIGH); // svetodiodni
    yoqish
  }
}
```

Sketchni yuklang va ishlashini tekshiring

Svetodiodni bir tekisda o`chib/yonishi:

Keling knopkani analogli signallarni boshqarish uchun ishlatib ko`ramiz. Buning uchun svetodiodni 13 portdan, 9 portga ko`chiramiz va sketchni o`zgartiramiz:

```
int ledPin = 13; ----> int ledPin = 9;
Endi, loop() funksiyasini quyidagicha o`zgartiring:

int value = 0;
void loop(){
  if (digitalRead(inputPin1) == LOW) { value--; }
  else if (digitalRead(inputPin2) == LOW) { value++; }
  value = constrain(value, 0, 255);
  analogWrite(ledPin, value);
  delay(10);
}
```

Svetodiodning yonib/o`chish tezligini o`zgartirish:

Agar, siz svetodiodni tezroq o`chib/yonishini xoxlasangiz, sketchdagi bitta komandani o`zgartirsangiz kifoya:

```
delay(10); ----> delay(yangi qiymat);
```

Effekt tezlashishi uchun ushbu qiymatni kamaytiring. Yonib/o`chishni pasaytirish uchun ushbu qiymatni oshiring.

Yana savollar bormi?

Tafsilotlar, elektron detallar ma`lumoti, savollar bilan ishlash:

<http://ardx.org/CIRC07>

Ishning ta`rifi:

Arduino raqamli portlaridan tashqari, 6 ta analogli portlar bilan jixozlangan. Bunday port bilan, raqamli kodni 0...5V kuchlanish diapazoni bilan, 0...1023 diapazonda (10 bit kattalikka mos keladi) o`zgartirish mumkin bo`ladi. Potensiometr (o`zgaruvchan qarshilik) bu juda kerakli element bo`lib, bunday portlar bilan ishlashda qo`l keladi. Potensiometr chekka oyoqchalari «manfiy» va +5V qutbga ulansa, o`rtadagi oyoqchada 0...5V diapazonda kuchlanish hosil bo`ladi. Aylanuvchi qismini buraganda, potensiometr kerakli kuchlanishni hosil qiladi (burchak kattaligi 0...5V diapazonda yotadi). Aylanuvchi o`qning o`rta holatdagi paytida, o`rta oyoqchadagi kuchlanish 2.5V ni tashkil qiladi. O`rta oyoqchadagi o`lchangan kuchlanish qiymatini programmada o`zgaruvchi sifatida qabul qilish mumkin.



Qurilma:

Komponentlar:



Vazifa kartochkasi
CIRC-08
x1



Yashil svetodiod
x1



2` kontaktli razyom
x4



Rezistor 560 Om
Yashil-Ko`k-jigarrang
x1

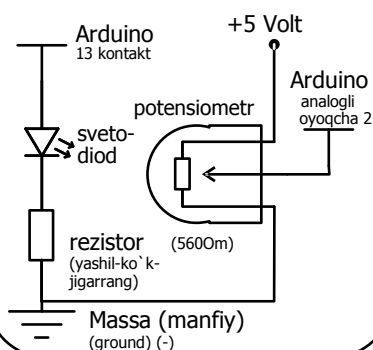


Potensiometr
10 kOm
x1



Simlar

Sxema



Internetda:

..yuklash..

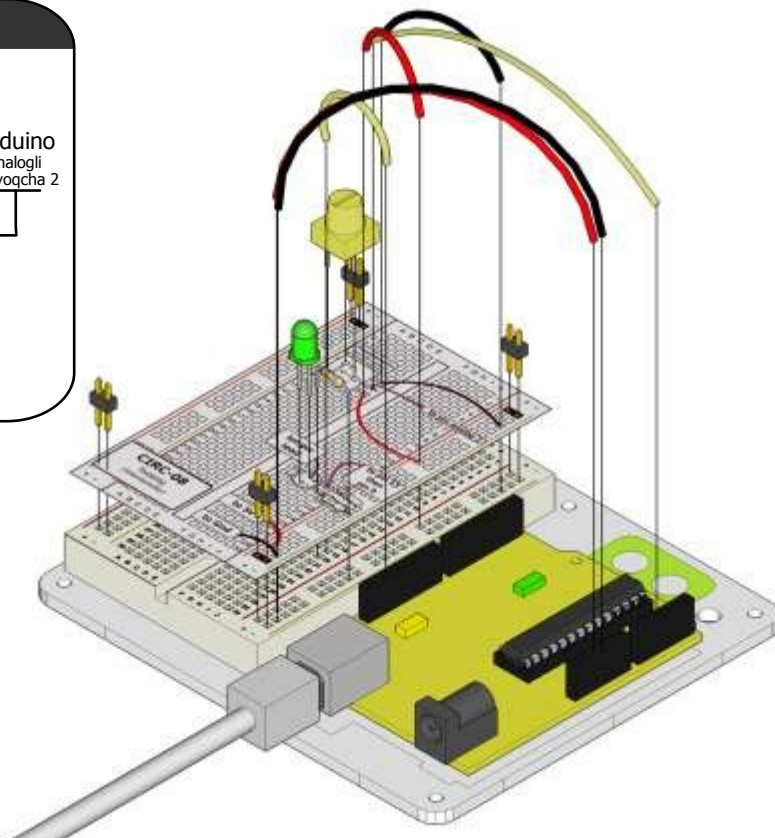
Vazifa kartochkasi

<http://ardx.org/BBS08>

..Tomosha qil..

video

<http://ardx.org/VIDE08>



File > Examples > 3.Analog > AnalogInput

(bu namunalar arduino.cc saytidan, bu saytda yana ko`plab proektlar kodlari bor)

```

/*
 * Svetodiodni o`chirib/yoqish jarayonida, 0 chiqishdagi analog sensorni qiymatini hisobga olib,
 * analog chiqishni hosil qiladi. Svetodiodni o`chib/yonish vaqti, analogRead() ga
 * bog`liq bo`ladi.
 * Created by David Cuartielles
 * Modified 16 Jun 2009
 * By Tom Igoe
 * http://arduino.cc/en/Tutorial/AnalogInput
 */

int sensorPin = 0;    // potensiometr uchun kirish nomerini beramiz
int ledPin = 13;      // svetodiod chiqish nomerini beramiz
int sensorValue = 0;  // sensordan kelayotgan qiymatlarni saqlash uchun o`zgaruvchi

void setup() {
  pinMode(ledPin, OUTPUT); //svetodiodlarga ulangan kontaktlarni chiqishdek hosil qilamiz
}

void loop() {
  sensorValue = analogRead(sensorPin); // sensor qiymatini hisoblaymiz
  digitalWrite(ledPin, HIGH);          // svetodiodni yoqadi
  delay(sensorValue);                  // <sensorValue> zaderjkasi millisekundlarda
  digitalWrite(ledPin, LOW);           // svetodiodni o`chiradi
  delay(sensorValue);                  // <sensorValue> zaderjkasi millisekundlarda
}

```

Ishlamayaptimi? (3 muammo va yechimi)**Potensiometr
davriy ravishda
ishlamayaptimi**

Bu potensiometrning yomon kontakt tegishi bilan aniqlanadi. Potensiometrni maketli plataga qattiqroq bosing.

Ishlamayapti

Potensiometrning

o`rta oyoqchasini 2 analogli oyoqchaga emas, 2 raqamli oyoqchaga tiqilsa ham, ishlaydi (analog oyoqchalar manbaa teshigi bilan bir qatorda joylashgan).

Shunda ham ishlamayapti

Potensiometr qutbini

teskari ulab ko`ring. Ba`zida bu yordam berishi mumkin.

Qurilmani chuqurroq o`rganamiz.**Tasodifiy uzib ulagich:**

Ba`zida biror o`lchangan kattalikdagi jarayonni kerakli qiymatga o`zgartirish kerak bo`ladi (masalan, kerakli bosqichga yetgandan keyin, suv bosimini kamaytirish Kerak bo`ladi). Bu ishni potensiometr orqali bajarish uchun programmani quyidagicha o`zgartirish kerak bo`ladi:

```

void loop() {
  int threshold = 512;
  if(analogRead(sensorPin) > threshold){
    digitalWrite(ledPin, HIGH);}
  else{ digitalWrite(ledPin, LOW);}
}

```

Svetodiod kerakli qiymatga erishganda yonib/o`chadi. Siz, threshold parametrini o`zgartirishingiz bilan Sezgirlikni oshirishingiz mumkin.

Yorqinlik nazorati:

Svetodiod yorqinligini bevosita potensiometr yordamida boshqarish mumkin. Buning uchun svetodiodni 13 portdan 9 portga o`zgartirishingiz kerak va programma matnini

o`zgartiring:

```

int ledPin = 13; ----> int ledPin = 9;

loop funksiyasini o`zgartiring:
void loop() {
  int value = analogRead(potPin) / 4;
  analogWrite(ledPin, value);
}

```

Programmani mikrokontrollerga yuklang va potensiometr holati o`zgarishiga bog`liq holda svetodiod yorqinligi o`zgarishiga e`tibor bering. Biz o`lchangan qiymatni 4 ga bo`ldik, nimagaki yorqinlik Faqatgina 0...255 (8 bit) oraliqda o`zgarishi mumkin, analogRead() funksiyasi esa 0...1024 (10 bit) oraliqda o`zgaradi.

Servomotor nazorati:

Bu ikkita qurilmani bittada jamlovchi ajoyib namuna. Servomotorni CIRC-04 vazifasiga o`xshab ulang, quyidagi namunani oching **File>Examples>Servo>Knob**, qatorni o`zgartiring:

```
int potpin = 0; ----> int potpin = 2;
```

Programmani mikrokontrollerga yuklang, potensiometr holati servomotor holatini ifodalaydi.

Yana savollar bormi?

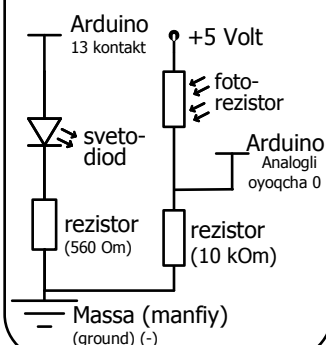
Tafsilotlar, elektron detallar ma`lumoti, savollar bilan ishlash:

<http://ardx.org/CIRC08>

**Ishning ta`rifi:**

Biz oldingi mashg`ulotlarda, inson qo`li bilan informatsiyani potensiometr orqali kiritishni o`rgangan edik. Atrof muhitdan kerakli informatsiyani olish uchun turli xil sensorlar mavjud.

Masalan, fotorezistorni yoritilganlik datchiki sifatida ishlatish mumkin. Bunda Arduino ishlash printsipi o`zgarmas saqlanadi. Arduino bevosita qarshilikni o`lchay olmaydi (kuchlanishni o`lchash mumkin), shuning uchun fotorezistor kuchlanishni bo`luvchi sifatida qo`llaniladi (<http://ardx.org/VODI>). Analogli chiqishda kuchlanish qiymatini aniq o`lchash mumkin, lekin bizning vazifamizda bu kerak emas. Biz nisbiy yoritilganlik bosqichini o`lchaymiz. Fotorezistorning kichik qiymati yorqin yorug`likka mos keladi, katta qiymati qorong`ulikka mos keladi.

Qurilma:**komponentlar:****Sxema****Internetda:****.:yuklash:.**

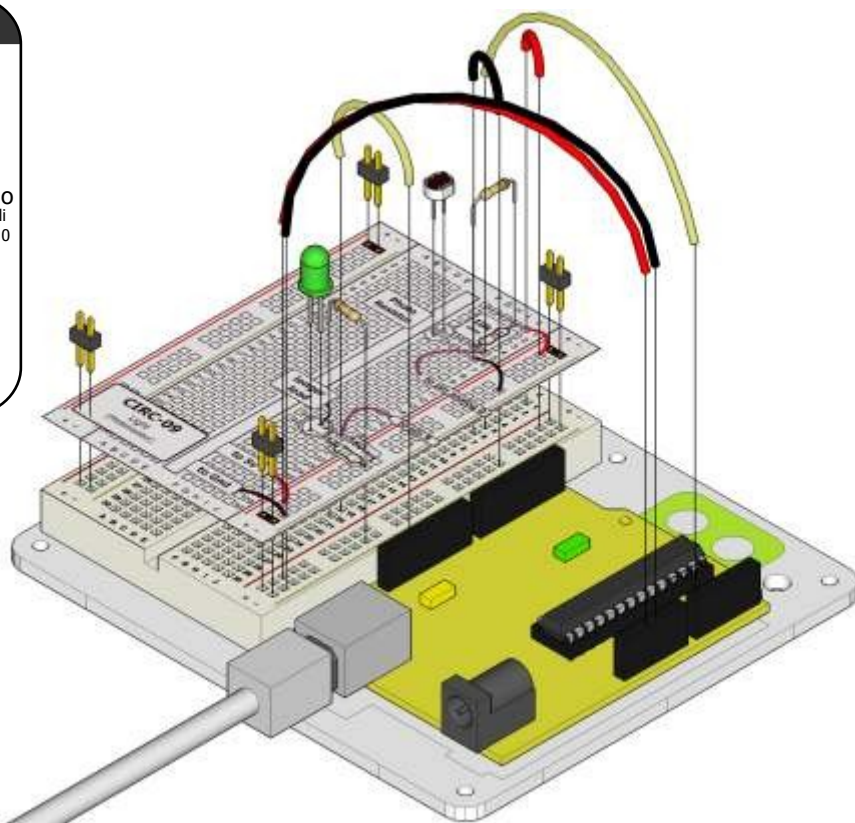
Vazifa kartochkasi

<http://ardx.org/BBL09>

.:Tomosha qil:.

video

<http://ardx.org/VIDE09>



kodni yuklash uchun: <http://ardx.org/CODE09>

(matnni Arduino Sketch yangi oynasiga ko`chirib o`ting)

```
/*
 * fotorezistorga yorug`lik tushishi natijasida
 * svetodiod yonish intensivligini
 * o`zgartiruvchi oddiy programma
 */

//PhotoResistor Pin
int lightPin = 0; //fotorezistor ulangan
// analogli chiqish porti.
// fotorezistor biror o`lchob birligiga
// to`g`rilanmagan, faqatgina yoritilganlik
// bosqichiga qarab, kuchlanish beryapti

value (relative light)
//LED Pin
int ledPin = 9;
//svetodiod ulangan kontakt.
//biz yorug`likni boshqaramiz, shuning uchun
Analogli chiqsh KIM (keng impulsli modulyatsiya)
Dan foydalanamiz.

void setup()
{
```

```
pinMode(ledPin, OUTPUT);
//Svetodiodga ulangan kontakti
//chiqish kabi to`g`rilaymiz
}
/*
 *loop() funksiyasi funksiya tugaganidan keyin boshlanadi
 * setup takrorlanadi
 */
void loop()
{
  int lightLevel = analogRead(lightPin);
  //yoritilganlik bosqichini hisoblaymiz

  lightLevel = map(lightLevel, 0, 900, 0, 255);
  //0-900 ora;iqda o`zgaruvchi qiymati beriladi
  lightLevel = constrain(lightLevel, 0, 255);
  //0-255 oraliqda o`zgaruvchi qiymati bilan chegaralaymiz
  analogWrite(ledPin, lightLevel);
  //o`zgaruvchi qiymatini kiritamiz
}
```

Ishlamayaptimi? (3 muammo va yechimi)

Svetodiod qoraligicha qolyapti

Juda ko`p hollarda svetodiod noto`g`ri ulanadi. Uni manbaadan uzib, qaytadan to`g`ri ulab ko`ring.

Qurilma yorqinlikka reaksiya bermayapti

Fotorezistor to`g`ri ulanganiga e`tibor bering.

Yana ishlamayapti?

Siz eksperiment qilayotgan xona kerakli darajada yorug`lik bo`lmayotgani uchun ishlamaydi. Agar, qo`shimcha yorug`lik manbaalarini qo`llasangiz bu muammoni yechimini topishingiz mumkin.

Qurilmani chuqurroq o`rganamiz

Chiqishni inverslash:

Svetodiod inverslangan rejimda ishlashi uchun Arduino sketchini quyidagicha o`zgartiring:

```
analogWrite(ledPin, lightLevel); ---->
analogWrite(ledPin, 255 - lightLevel);
```

Tun chiroq:

Svetodiod yorqinligini o`zgartirish bilan bir qatorda uni Tun chiroq sifatida o`chirib yoqsa ham bo`ladi. Buning uchun siz Sketchdagi ba`zi matnlarni quyidagicha o`zgartirishingiz kerak:

```
void loop(){
  int threshold = 300;
  if(analogRead(lightPin) > threshold){
    digitalWrite(ledPin, HIGH);
  }else{
    digitalWrite(ledPin, LOW);
  }
}
```

Yorug`lik yordamida boshqariluvchi servomotor

Keling, ushbu sensori servomotorni boshqarish uchun qo`llaymiz. Servomotorni 9 portga ulang (CIRC-04 dagi kabi). Kontrollerga Knob File > Examples > Servo > Knob (CIRC-08 dagi namuna) programmani yuklang. Servomotor qanday ishlashini kuzating. Siz, servomotorni hamma diapazonda ishlamayotganini sezishingiz mumkin. Bu holatda, 0..5V dan kam diapazonda ishlovchi kuchlanish bo`luvchisi orqali foydalanayotganimizni bildiradi. Buni to`g`irlash uchun biz, val = map(val, 0.1023, 0, 179); komandasi orqali to`g`rilashimiz mumkin. Bu funksiya haqida batafsil ma`lumot olish uchun quyidagi adressga murojaat qilishingizni so`raymiz: <http://arduino.cc/en/Reference/Map>.

Yana savollar bormi?

Tafsilotlar, elektron detallar ma`lumoti, savollar bilan ishlash:

<http://ardx.org/CIRC09>

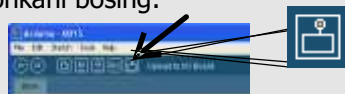
CIRC-10

.:Temperatura (TMP36 temperatura datchiki):.

Ishning ta`rifi:

Arduino yordamida yana nimani o`rgansa bo`ladi. Temperaturani! Temperaturani o`rganish uchun yetarli darjada qiyin bo`lgan mikrosxemani, tranzistorga o`xshash, korpusga ega, P2N222AG datchik kerak bo`ladi. Bu mikrosxema uchta oyoqchadan tashkil topgan: «manfiy», signal va +5v manbaa. U 10mV/gradus Selsiy aniqlikda (manfiy temperaturalarni o`lchash uchun, 500mV kuchlanishni surish orqali amalga oshiriladi. Masalan, 25°C = 750 mV, 0°C = 500 mV). Kuchlanishlar qiymatini temperatura birliklariga aylantirish uchun, Arduining matematik qobiliyatini ishga solamiz. Temperaturani hosil qilish uchun, Arduining sozlovchi oyna interfeysidan foydalanamiz. Temperatura qiymatini jo`natish uchun, kompyuter ekranida akslantiruvchi ketma-ket portdan foydalanamiz.

Bu qurilma ketm-ket portli monitorda ishlaydi. Uni ishga tushirish uchun, programmani Arduinoga yuklab, tepa qismdagi panellardan to`rtburchak shakldagi (yuqori qismida antennaga o`xshash tasvir bor) ikonkani bosib:



TMP36 datchik haqida: <http://ardx.org/TMP36>



Qurilma:

Komponentlar:



Vazifa kartochkasi
CIRC-10
x1



2* kontaktli
razyom
x4

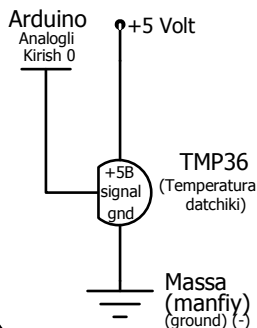


Temperatura
datchiki TMP36
x1



Simlar

Sxema



Internetda:

..yuklash..

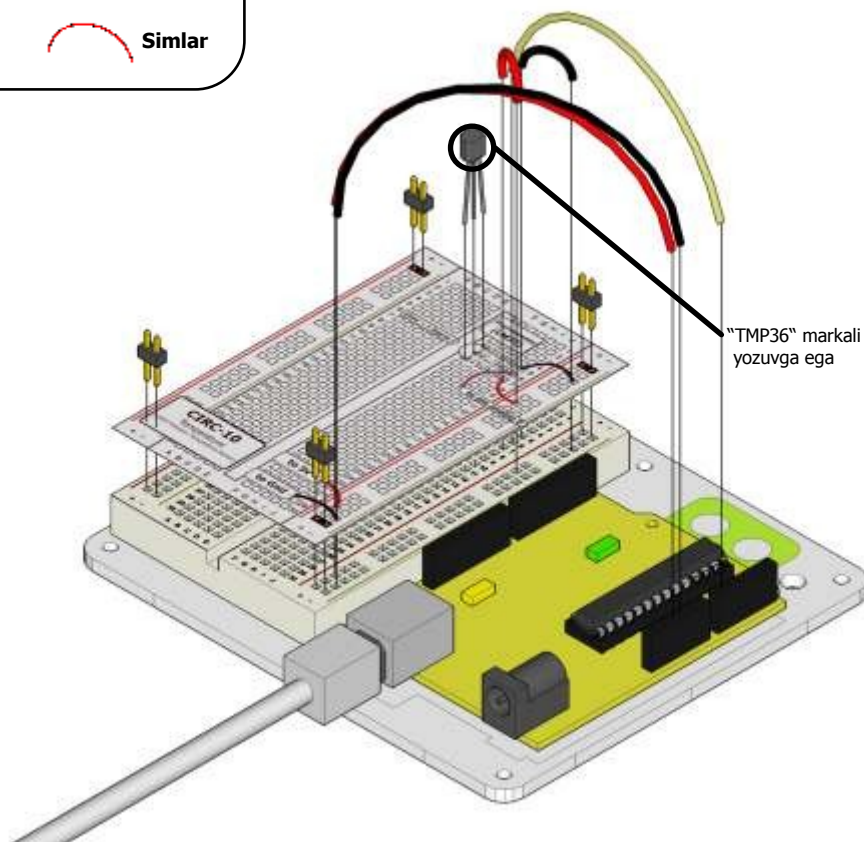
Vazifa kartochkasi

<http://ardx.org/BBL10>

..Tomosha qil.

video

<http://ardx.org/VIDE10>



Kodni yuklash uchun: <http://ardx.org/CODE10>

(matnini Arduino Sketch yangi oynasiga ko`chirib o`ting)

```

/*
 * | Arduino eksperiment uchun to`plami |
 * | CIRC-10 vazifasi uchun sketch namunasi |
 * | .. Temperatura datchiki :. |
 *
 * Ketma-ket portli monitorda joriy temperaturani
 * ko`rsatish uchun oddiy programma

//TMP36 Pin variables
int temperaturePin = 0;
// TMP36 datchiki chiqishi ulangan, analog portni
kiritamiz. Chegaraviy qiymat - 10mV/gradus selsiy
(500mV qo`shilishi past temperaturalar
uchun)

void setup()
{
  Serial.begin(9600);
  //открываем последовательное соединение. Для
  отображения результата необходимо запустить монитор
  последовательного порта (последняя кнопка под главным
  меню, выглядит как прямоугольник с антенной)
}

void loop() //cheksiz sikl
{
  float temperature = getVoltage(temperaturePin);
  /*temperatura datchikidan kuchlanish qiymatini
  qabul qiladi*/

```

```

  temperature = (temperature - .5)
  * 100;
  //qabul qilingan qiymatni 10mV/gradus dan (kuchlanish-500mV)x100
  ga o`zgartiradi

  Serial.println(temperature); //выводим результат
  delay(1000); //задержка 1 сек
}

/*
 * getVoltage() - kuchlanish qiymatini analog chiqishiga
  o`zgartiradi
 */
float getVoltage(int pin){
  return (analogRead(pin) * .004882814);
  //qiymatni 0...1024 diapazonda 0...5 mV diapazonga
  o`zgartiradi (xar bir ko`rsatkich ~ 5 millivolt)
}

```

Ishlamayaptimi? (3 muammo va yechimi)**Xech narsa sodir bo`lmayapti**

Bu programa indikatsiyani
ishlatmaydi. Rezultatni ko`rish
uchun, siz kema-ket
monitorli portni
ishga tushirishingi kerak.

**Noma`lum ma`lumotlar
paydo bo`lyapti**

Bu holat, sizning ketma-ket portli
monitoringiz kerakli tezlikda
ma`lumotlar qabul qilayotganidan
dalolat beradi.
Monitoring ishlash tezligini
«9600 baud» ga o`zgartiring.

**Temperatura qiymati
o`zgarmayapti**

Datchikni qizdirish uchun
uni barmoqlaringiz bilan ushlang.
Yoki muz parchasini olib,
Uni sovitishga harakat
qiling.

Qurilmani chuqurroq o`rganamiz.**Kuchlanishni tasvirlash:**

Buni bajarish juda ham oson, chunki datchik kuchlanish
uzatayapti. Faqatgina qatorni o`chirish kifoya
delete the line `temperature = (temperature - .5) * 100;`

Temperaturani farangeytda tasvirlash:

Bu ham qiyin emas, nimagaki faqat hisoblash qismiga
tegilyapti. Gradus selsiydan, gradus farengeytga o`tish
uchun, quyidagi formuladan foydalanish
maqsadga muvofiq:

$$(F = C * 1.8) + 32)$$

Qator qo`shing
`temperature =`
`((temperature - .5) * 100) * 1.8 + 32;`
qatordan oldin `Serial.println(temperature);`

Ko`proq informatsiyali ma`lumotlar porti:

Keling ma`lumotga xabar qo`shamiz, keyin ketma-ket
port orqali jo`natamiz. Programmani oldingi holatiga
qaytib, quyidagilarni o`zgartiring:
`Serial.println(temperature);`

```

----->
Serial.print(temperature);
Serial.println(" degrees centigrade");
Birinchi qatordagi o`zgarishlar shuni anglatadiki, keyingi
ma`lumotlar chiqishi displeydagi qatorda amalga oshiriladi.
Keyingi qator yordamchi informatsiyani chiqaradi va yangi
qatorga o`tishni ta`minlaydi.

```

Jo`natish tezligini o`zgartirish:

Agar, siz ketma-ket port orqali katta hajmli informatsiyani
jo`natmoqchi bo`lsangiz, unda imkoni bor darjada tez jo`natish
fikri paydo bo`lishi aniq. Namunamizda biz, ma`lumotlarni
9600 bod tezlikda jo`natamiz, lekin yana katta tezliklarni
qo`llash imkoni bor.

Qatorni o`zgartiring:
`Serial.begin(9600);` -----> `Serial.begin(115200);`

Shunday qilib, jo`natish tezligi 12 marta ortadi.
Faqatgina ketma-ket monitor porti tezligini o`zgartirishni
unutmang.

Yana savollar bormi?

Tafsilotlar, elektron detallar ma`lumoti, savollar bilan ishlash:

<http://ardx.org/CIRC10>

Yuqori yuklamalar bilan ishlash (relelar):



Ishning ta`rifi:

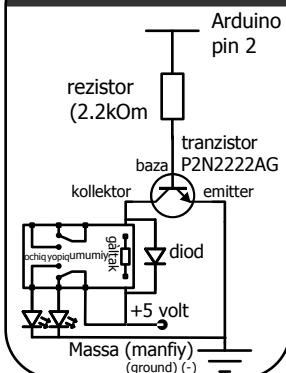
Последние задания всегда в какой-то степени являются испытанием. Мы используем наши знания о транзисторах (CIRC03) чтобы управлять реле. Реле - это управляемый электронно-механический переключатель. Внутри пластиковой коробочки содержится электромагнит, при подаче питания на который происходит переключение механического контакта (обычно с заметным щелчком). Различные реле очень сильно отличаются по размерам и параметрам (от совсем крошечных, до реле размером с холодильник). Обычно, чем больше размер реле — тем больший ток оно способно коммутировать. При помощи реле и Arduino Вы сможете управлять чем-то значительным.

Qurilma:

Komponentlar:

 Vazifa kartochkasi CIRC-11 x1	 Diod (1N4001) x1	 Tranzistor P2N2222AG (TO92) x1	 Rele (DPDT) x1
 Rezistor 2,2 kOm qizil-qizil-qizil x1	 Rezistor 560 oM Yashil-ko`k-jigarrang x2	 Yashil svetodiod x1	 qizil svetodiod x1
 2* kontaktli razyom x4			

Sxema



Internetda:

..yuklash..

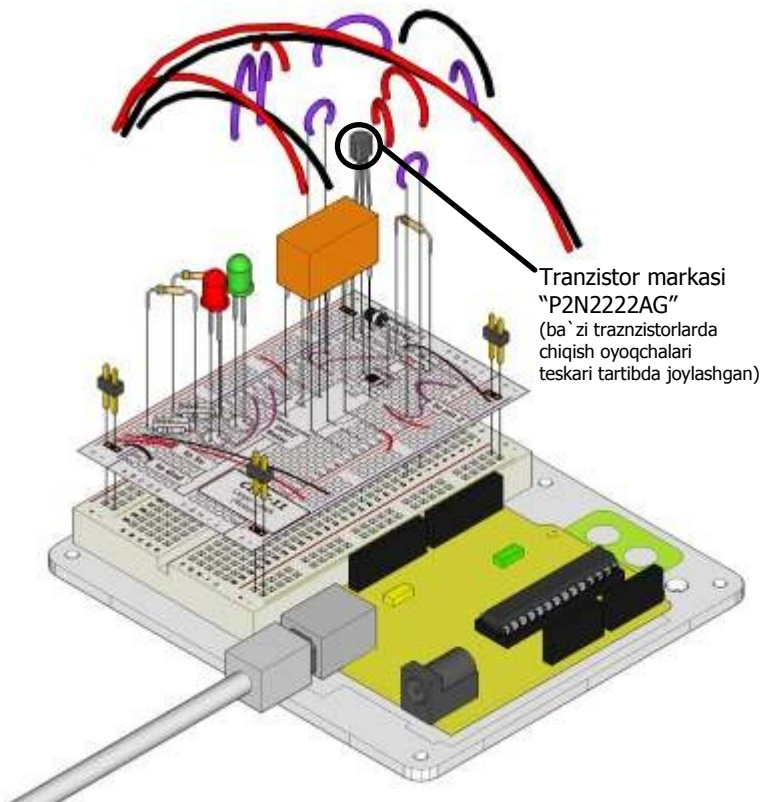
Vazifa kartochkasi

<http://ardx.org/BBLS11>

..Tomosha qil..

video

<http://ardx.org/VIDE11>



File > Sketchbook > 1.Basic > Blink

(bu namunalar arduino.cc saytidan, bu saytda yana ko`plab proektlar kodlari bor)
 /* pirpirash

* sodda namuna: programma svetodiodni 1 sek yoqadi, keyin 1 sek o`chiradi, va hokazo,. Biz 13 oyoqchani ishlatamiz, bu oyoqcha sizning Arduinoingiz turiga bog`liq, agar, sizning Arduino platangizda svetodiod bo`lsa, rezistor va svetodiod kerak bo`lmaydi.

* <http://www.arduino.cc/en/Tutorial/Blink>
 */

```
int ledPin = 2;                // *****pin 2 ga o`zgartiring*****

void setup()                  //bir martta bajariladi, keyin sketch ishga tushadi
{
  pinMode(ledPin, OUTPUT);    //raqamli kontaktni chiqishga o`zgartiramiz
}

void loop()                   //funksiya cheksiz qaytariladi
{
  digitalWrite(ledPin, HIGH); //svetodiodni yoqadi
  Delay(1000);                 //1 sek zaderjka
  digitalWrite(ledPin, LOW);   //svetodiodni o`chiradi
  Delay(1000);                 //1 sek zaderjka
}
```

Ishlamayapti? (3 muammo va yechimi)**Hech narsa sodir bo`lmayapti**

Programma 13 portdan foydalanadi

Rele 2 portga ulangan.

Faqatgina programmani
o`zgartirishni
unutmang.

**Ovoz chiqmayapti
(rele ulangandagi ovoz)**

Tranzistor ishlamayotgan
bo`lishi mumkin. Diqqat
bilan qurilmani tekshiring.

Umuman ishlamayapti

Qurilmadagi rele payvandlashga
mos bo`lishi kerak. Agar, ozgina
kuch bilan releni maketli plataga
bosing. Kontakt yaxshi
bo`lmaganda ham, rele
ishlamaydi.

Qurilmani chuqurroq o`rganamiz.**Teskari tok impulsini vizualizatsiyasi**

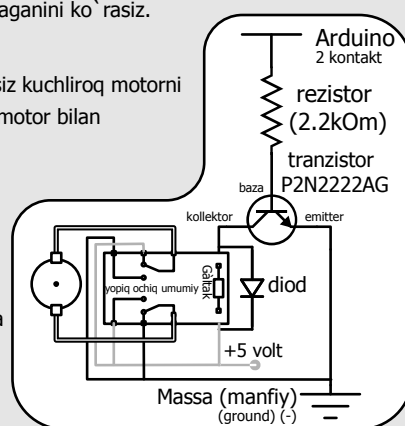
Diodni svetodiod bilan almashtiring. Siz releni o`chirganingizda svetodiodni pirpiraganini ko`rasiz.

Motor bilan boshqarish

CIRC-03 vazifasida motorni tranzistor bilan boshqarishni o`rganagan edik. Agar, siz kuchliroq motorni Boshqarmoqchi bo`lsangiz, reledan foydalaning. Buning uchun, qizil svetodiodni motor bilan almashtiring (560 oM li rezistorni almashtiring).

Motorni aylanish yo`nalishini o`zgartirish

Qurilmaga ozgina o`zgartirish kiritamiz. Doimiy motorni aylanish yo`nalishini o`zgartirish uchun, tok oqish yo`nalishini o`zgartirish kifoya. Agar, buni qo`lda bajarmoqchi bo`lsangiz — simlar o`rnini alishtirish kifoya. Elektron kommutatsiya uchun, H-ko`priikka o`xshash elementdan foydalanish kerak. Bu jarayonni DPDT rele yordamida bajarish mumkin. Quyidagi sxemani yig`ing:

**Yana savollar bormi?**

Tafsilotlar, elektron detallar ma`lumoti, savollar bilan ishlash:

<http://ardx.org/CIRC11>

CIRC-12

..Ko`p rangli svetodiodlar (RGB svetodiodlar):.



Ishning ta`rifi:

Siz, svetodiod pipirashi va KIM orqali motorni boshqarishni bilamiz. Keling, shu olgan bilimlar asosida xar xil svetodiodlar yordamida xoxlagan intensivlikda (RGB svetodiod yordamida) yoqamiz. RGB svetodiod 3 xil svetodioddan tashkil topgan.

R — qizil, G — yashil, B — ko`k.

Bu 3 xil rang yordamida xar xil kombinatsiyalarni yaratish, sintez qilish mumkin.

RGB svetodiodlarni xar bir kanalida oddiy svetodiod ishlatiladi, shunday qilib, ranglarni qo`shilib yonishi yordamida animatsiyalar yaratish imkoni paydo bo`ladi.

Qurilma:

Komponentlar:



Vazifa kartochkasi
CIRC-RGB
x1



rezistor 560 Om
yashil-ko`k-jigarrang
x6



2` kontaktli
razyom
x4



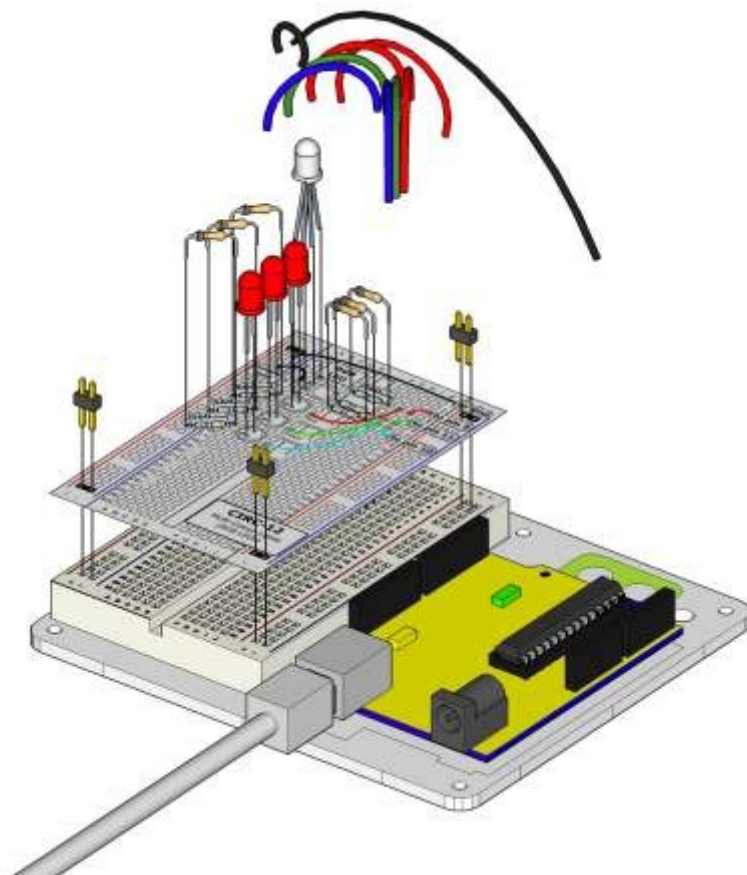
RGB sveto-
Diod 5 mm
x1



simlar



Qizil svetodiod
x1



Internetda:

..yuklash..

Vazifa kartochkasi

<http://ardx.org/BBS12R>

Kodni yuklash uchun: <http://ardx.org/CODE12R>

(matnni Arduino Sketch yangi oynasiga ko`chirib o`ting)

```
//RGB LED pins
int ledDigitalOne[] = {9, 10, 11};
//svetodiodlarning 3 ta raqamli oyoqchalari
//9 = redPin, 10 = greenPin, 11 = bluePin

const boolean ON = LOW;
//Low ga o`xshash yoniq holatini beramiz
// (nimagaki RGB svetodiod umumiy anodga
// ulangan), umumiy oyoqcha +5V ga ulangan
const boolean OFF = HIGH;
//HIGH ga o`xshash o`chiq holatini beramiz

//berilgan rang
const boolean RED[] = {ON, OFF, OFF};
const boolean GREEN[] = {OFF, ON, OFF};
const boolean BLUE[] = {OFF, OFF, ON};
const boolean YELLOW[] = {ON, ON, OFF};
const boolean CYAN[] = {OFF, ON, ON};
const boolean MAGENTA[] = {ON, OFF, ON};
const boolean WHITE[] = {ON, ON, ON};
const boolean BLACK[] = {OFF, OFF, OFF};

//berilgan rangni saqlovchi massiv
const boolean* COLORS[] =
{RED, GREEN, BLUE, YELLOW, CYAN, MAGENTA,
WHITE, BLACK};

void setup(){
```

```
for(int i = 0; i < 3; i++){
  pinMode(ledDigitalOne[i], OUTPUT);
}
//3 ta svetodiod oyoqchalarini chiqishga o`zgartiramiz
}

void loop(){
  setColor(ledDigitalOne, CYAN);
  //svetodiod angini kiritamiz

  //randomColor()
}

void randomColor(){
  int rand = random(0, sizeof(COLORS) / 2);
  //ranglar diapazonidagi tasodifiy raqamni qaytaradi
  setColor(ledDigitalOne, COLORS[rand]);
  //svetodiod rangini tasodifiy beradi
  delay(1000);
}

void setColor(int* led, boolean* color){
  for(int i = 0; i < 3; i++){
    digitalWrite(led[i], color[i]);
  }
}
```

Ishlamayapti? (3 muammo va yechimi)

Svetodiod yonmayapti, yoki noto`g`ri rang yonayapti

RGB svetodiodni diqqat to`g`ri ulanganini tekshiring. umumiy oyoqchani «massa» ga ulashni unutmang.

Hammasi o`ta qizil

RGB svetodioddagi qizil svetodiod boshqa ranglarga qaraganda yorqinroq yonyapti Bu holatni R kanalidagi Rezistor nominalini kuchaytirish orqali to`g`rilanadi.

Juda ko`p svetodiodlarmi?

Bir xil rangli svetodiodlar RGB kanallarini to`g`rilash uchun ishlatiladi. Siz, bu svetodiodlar ishlashini bilib olganingizdan so`ng uni olib tashlashingiz mumkin.

Qurilmani chuqurroq o`rganamiz

Ko`proq ranglar

Menimcha, svetodiodni ko`k yonishi sizni juda hayron qoldirgan bo`lsa kerak. Uni boshqa rangga o`zgartirish uchun programmani o`zgartiring: setColor(ledDigitalOne, CYAN); ----> setColor(ledDigitalOne, **NEW COLOR**);

Tasodifiy ranglarni tasvirlash

Albatta biz, bitta rangni yondirishdan ko`ra har xil rangni yondirishni yaxshi ko`ramiz. Pastdagi programmani quyidagicha o`zgartiring:

```
void loop(){
  //setColor(ledDigitalOne, CYAN);
  randomColor()
}
```

Ranglarni analogli boshqaruvi

Bir rangdan boshqasiga o`tqazish yetarli darajada qiziqarli. Lekin analogli boshqaruvni amalga oshirsa, yetarli darajada ranglar uyg`unligini olsa bo`ladi.

Namunani quyidagi adressdan yuklang:

<http://ardx.org/MABE12R>

Yana savollar bormi?

Tafsilotlar, elektron detallar ma`lumoti, savollar bilan ishlash:

<http://www.solarbotics.com>

This image shows a single page from a notebook or ledger. The page is white with thin, black horizontal lines spaced evenly apart. The top-left corner is rounded, and the bottom-left corner features a small, dark grey tab-like shape containing the number "32". The rest of the page is blank, with no handwriting or other markings.

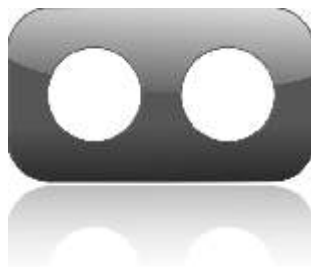
.:Qaydlar uchun joy:.

Qo`shimcha

This image shows a single page from a notebook or ledger. The page is white with thin, black horizontal lines spaced evenly apart. The top-left and bottom-right corners are rounded. In the bottom right corner, there is a small black semi-circular tab containing the number "33" in white.



www.oomlout.com



Эти материалы выпущены под лицензией Creative Commons Attribution-Share Alike 3.0 Unported License. Чтобы ознакомиться с лицензией, посетите: <http://creativecommons.org/licenses/by-sa/3.0/> или напишите в Creative Commons по адресу: 171 Second Street, Suite 300, San Francisco, California 94105, USA.

