

ARDUINO TARAQQIYOTI

Ma'ruza 3

Mavzu: Puls kengligi modulyatsiyasi

Reja:

1. Puls kengligi modulyatsiyasi o'zi nima?
2. Arduinoda puls kengligi modulyatsiyasining ishlatilishi
3. For sikli
4. Puls kengligi modulyatsiyasi asosida kichik loyiha amalga oshirish

Shunday qilib, o'tgan darslar davomida Arduino platformasining pinlarini raqamli tarzda boshqarishni o'rganib oldik. Bu esa yorug'lik diodlarini yoqib o'chirish, relarlarni boshqarish, o'zgarmas tok motorlarini bir xil tezlikda ishga tushirishda juda qo'l keladi. Ammo, Arduino platformasidan 0 yoki 5Voltdan boshqa qiymatdagi kuchlanish chiqarish talab etilsa nima qilamiz?

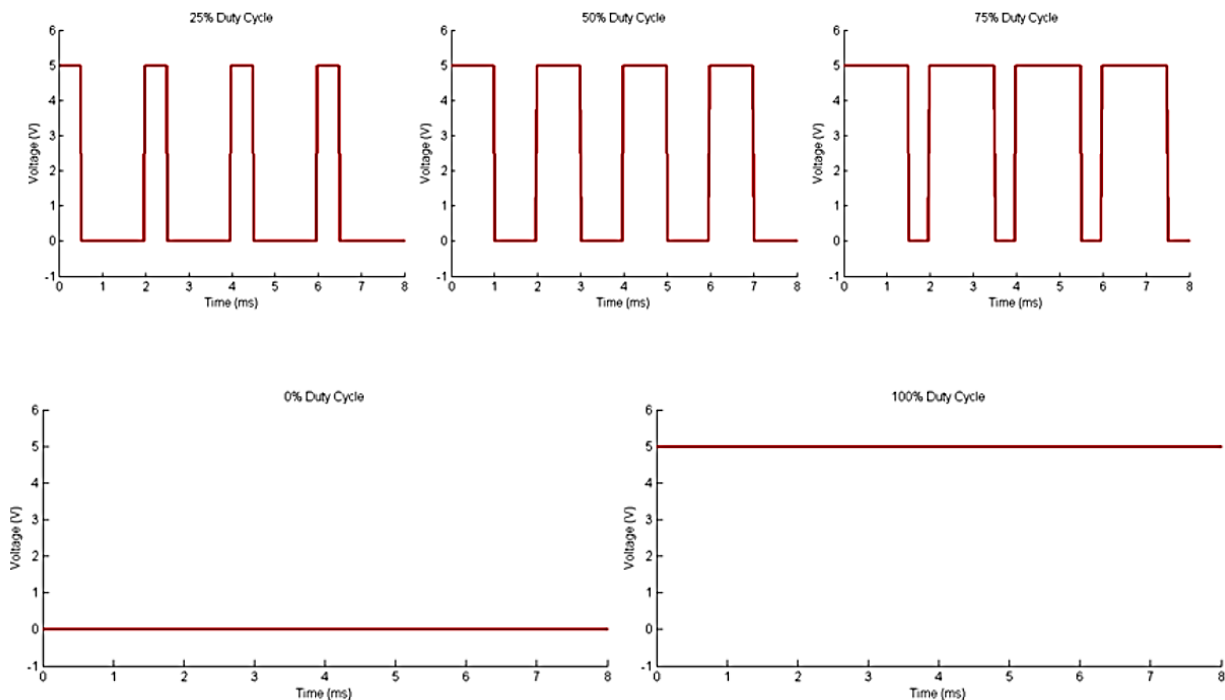
Xo'sh, ushbu holatda biz analog signal chiqarishning ajoyib bir usulini ya'ni puls kengligi modulyatsiyasini ishga tushiramiz. Arduino platformasida ~ belgisi bilan belgilangan bir nechta pinlar mavjud. Ushbu pinlardan analog signal chiqarish mumkin. Masalan, Arduino Uno platformasida bular 3, 5, 6, 9, 10 va 11 pinlardir.

Puls kengligi modulyatsiyasi (PWM) chiqishlari 8 bit qiymatda bo'ladi. Boshqacha qilib aytganda, siz unda 0 dan $2^8 - 1$ yoki 0 dan 255 gacha qiymat yozishingiz mumkin.

PWM boshqaruvi juda ko'p analog signal chiqarilishi talab etilgan holatlarda ishlatilishi mumkin. Lekin, u har doim ham analog signal kirishida ishlatila olinmaydi. Masalan, PWM o'zgarmas tok motorlarini turli xil tezliklarda boshqarish uchun juda ham qulay xisoblanadi. Biroq, u ovoz kuchaytirgichlarni boshqarishda yaxshi ish bermaydi.

Quyida PWM qanday ishlashini tushunib olishimiz mumkin.

Quyidagi grafikka qarang.



PWM signallari turli xil ijro sikllarida

Puls kengligi modulyatsiyasi kvadrat to'liq signallarining ijro sikllarini modulyatsiya qilish orqali ishlaydi. Ijro sikli bu bitta to'liq kvadrat to'liqidagi necha foiz mantiqiy "high" (signal mavjud) qismi mavjud bo'lsa, shuncha miqdorga teng o'lcham. Balki siz 50% ijro sikliga ega kvadrat to'liqlar bilan tanishdirsiz. Unda bir to'liq kvadrat to'liq davrining teng yarmida signal mavjud qolgan yarmida mavjud emas bo'ladi.

analogWrite() komandasi chiquvchi kvadrat to'liqining fozidagi ijro siklini o'rnatadi.

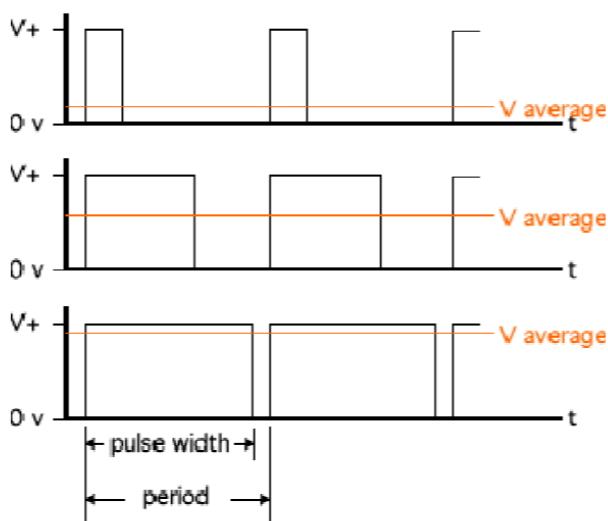
- analogWrite() funksiyasi 0 qiymat bilan berilgan bo'lsa, ushbu holatda kvadrat to'liq faqat nol qiymatga ega bo'ladi.
- analogWrite() funksiyasi 255 qiymat bilan berilgan bo'lsa, ushbu holatda kvadrat to'liq faqat bir qiymatga ega bo'ladi.
- analogWrite() funksiyasi 127 qiymat bilan berilgan bo'lsa, ushbu holatda kvadrat to'liqning 50 foizi bir qiymatga va qolgan 50 foizi nol qiymatga ega bo'ladi.

Yuqorida keltirilgan rasm 25%, 50%, 75%, 0% va 100% ijro sikliga ega signallarni ko'rsatadi. Ushbu kvadrat to'liqning chastotasi Arduinoda 490Hz ni tashkil etadi. Ya'ni, signal 5Volt bilan 0Votning o'rtasida 1 sekundda 490 marta o'zgaradi.

Shunday qilib, agar siz yorug'lik diodiga yuborilayotgan tokni o'zgartirmayotgan bo'lsangiz, unda nega yorug'lik diodi xira yonmoqda? Bunga sabab shuki, inson ko'zining

sezgirlik darajasidir. Masalan, yorug'lik diodi har 1 millisekundda yonib o'chayotgan bo'lsa, huddi u yarim yorug'lik chiqarayotganga o'xshaydi. Chunki, inson ko'zi seza olmaydigan darajada tez yonib o'chmoqda.

Puls kengligi modulyatsiyasini bajarishni tushunib olish uchun avvalo u nima qilishini tushunish lozim. Puls kengligi modulyatsiyasi shunday uslubki, u o'zgaruvchan signal bilan ta'minlaydi. Puls kengligi modulyatsiyasi buni puls kengligini o'zgartirish orqali amalga oshiradi, o'z navbatida bog'langan komponentaga qancha quvvat berilishiga ko'ra kvadrat to'lqin kengligining ijro siklini o'zgartiradi. U o'zagaradi chunki signal ijro siklini oladi va chiquvchi signal quvvatini o'rtacha qiymatga keltiradi. Bu quyidagi rasmda ifodalangan.



Ijro siklining o'zgarishi

Shunday qilib, kattaroq ijro sikli foizi kattaroq chiqish signali deganidir. Yuqoridagi rasmda va Arduino platformasida o'zgaruvchi parameter bu tok kuchlanishidir.

Arduino platformasida Puls kengligi modulyatsiyasi signalini yaratish

Arduino platformasida 6 ta pinlar puls kengligi modulyatsiyasi uchun mo'ljallangan. Quyidagi dastur Arduino platformasining 9-pinida puls kengligi modulyatsiyasi signalini hosil qilish uchun berilmoqda.

```
File Edit Sketch Tools Help
sketch_sep30a $
int PWM_pin = 9;

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

void loop() {

  analogWrite(PWM_pin, 51);
  delay(3000);

  analogWrite(PWM_pin, 127);
  delay(3000);

  analogWrite(PWM_pin, 165);
  delay(3000);

}
```

Ushbu dastur quyidagi seksiyalarga bo'lingan.

Birinchi seksiya

Ushbu seksiyada o'zgaruvchi e'lon qilingan. Puls kengligi modulyatsiyasi signalining chiqish pini ya'ni 9-pin PWM_pin o'zgaruvchisi bilan e'lon qilindi. Bu dasturni tushunarli bo'lishiga yordam beradi.

Ikkinchi seksiya

Ushbu seksiya dasturning void setup qismi bo'lib, bu yerda dastur davomida ishlatiluvchi pinlar chiqish yoki kirish pini sifatida sozlanadi. Yuqoridagi dasturda PWM_pin ya'ni

Arduino platformasining 9-pini chiqish pini sifatida sozlandi. Odatda, bu pinMode funksiyasi yordamida amalga oshiriladi.

Uchinchi seksiya

Ushbu seksiya dasturning void loop qismi bo'lib, dasturning asosiy kodlari shu yerga kiritiladi. Bu yerda biz uch xil turdagi puls kengligi modulyatsiyasi signallarini hosil qilish komandalarini beradik. Har bir puls kengligi modulyatsiyasi signali 3 sekundda boshqasiga o'zgarib turadi.

AnalogWrite funksiyasi funksiya ichida o'zgaruvchan kvadrat to'lqini hosil qiladi. Funksiya ishlashi uchun ikkita parameter ya'ni signal chiquvchi pin nomeri va ijro siklining qiymati kiritilishi zarur. Ijro sikli sifatida kiritilayotgan o'zgaruvchi 0 va 255 sonlari o'rtasida bo'lishi shart. Ijro sikli 100% bo'ladi qachonki, unga 255 raqami kiritilsa. Va u 0 % bo'ladi qachonki uning qiymati sifatida 0 kiritilsa. AnalogWrite funksiyasi uchun maxsus qiymat belgilash uchun quyidagi formula ishlatiladi:

$$Ijro_sikli * 255 = analogWrite_qiymati$$

Yuqoridagi dasturda avvalo 20% ijro sikli va 3 sekunndan keyin 50% va yana 3 sekunndan keyin 65% ijro sikli amalga oshishi uchun maxsus komandalar kiritilgan. Natijada quyidagi rasmlarda ko'rsatilgandek puls kengligi modulyatsiyasi signallari hosil bo'ladi.



20% Ijro sikli



50% Ijro sikli



65% Ijro sikli

Endi esa puls kengligi modulyatsiya bo'yicha kichik bir loyiha amalga oshiramiz. Bunda avval esa for siklini o'rganib olishimiz darkor, chunki aynan ushbu for sikli bajariladigan loyihamizda ishlatiladi.

For sikli

For sikli jingalak qavslar ichida yozilgan dasturni qayta-qayta takrorlash uchun ishlatiladi. Yuqoriga sanovchi odatda yuqoriga sanash va siklni tugatish uchu ishlatiladi.

For siklining 3 ta muhim qismlari mavjud:

```
for (boshlash; holat; ko'tarilish){  
// komandalar;  
}
```

Qavslar

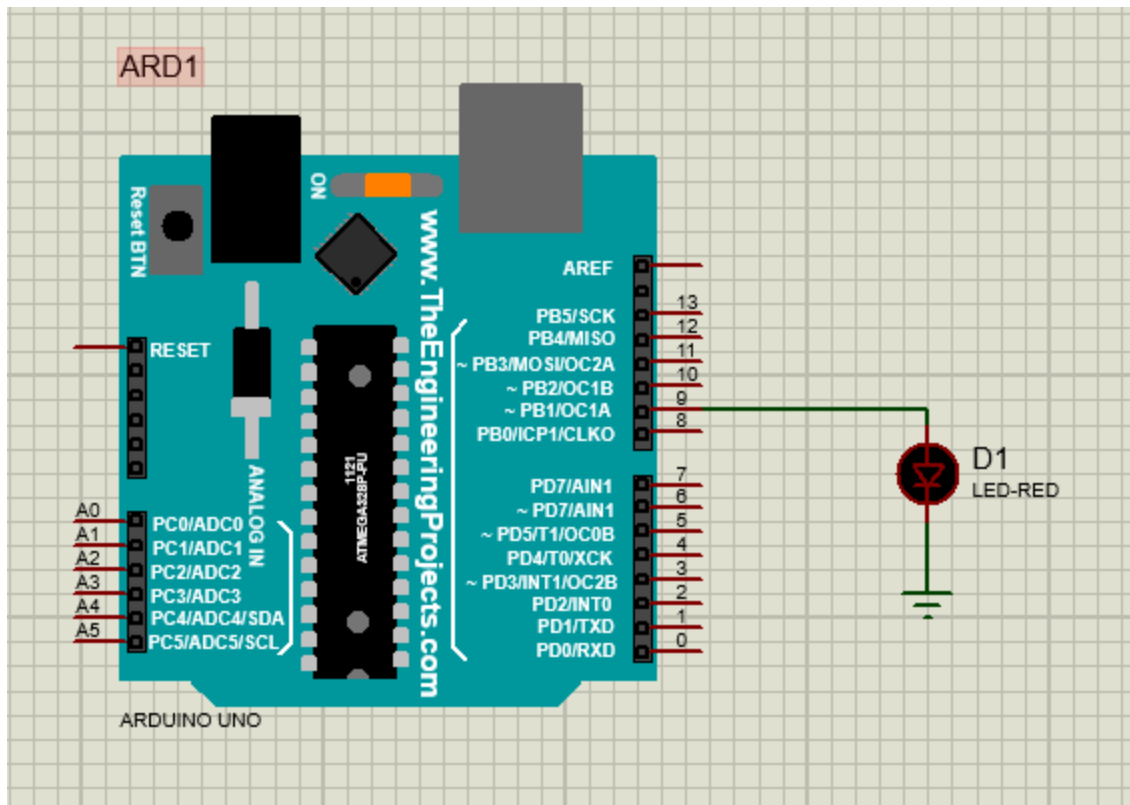
```
for(int x = 0; x < 100; x++){  
    println(x); // prints 0 to 99  
}
```

Boshlash birinchi bo'lib va faqat bir marta sodir bo'ladi. Har safar sikl aylanganda holat tekshiriladi. Tekshirilganda agar holat to'g'ri bo'lsa, oshirish bajariladi va aksincha agar holat noto'g'ri bo'lsa for sikli tugallanadi.

Puls kengligi modulyatsiya bo'yicha kichik loyiha

Ushbu loyihamizda Arduino platformasiga yorug'lik diodini ulab, ushbu yoqug'lik diodini puls kengligi modulyatsiyasi signali yordamida turli xil yorqinlikda yoqamiz.

Buning uchun avvalo loyiha uchun elektorn sxema tuzib olamiz. Electron sxema quyidagi rasmda keltirilgan.



Loyiha uchun dasturni esa quyidagicha yozamiz.

```
analogWrite
```

```
int PWM_pin = 9;

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

void loop() {

  for(int i = 0; i<=255; i++){
    analogWrite(PWM_pin, i);
    delay(10);
  }

  for(int i = 255; i>=0; i--){
    analogWrite(PWM_pin, i);
    delay(10);
  }
}
```

```
Done compiling.
```

Foydalanilgan adabiyotlar

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