



ARDUINO DASTURIDA DIGITAL INPUT VA ANALOG INPUT

*Ashirmatova Shaxrizoda Sharofiddin qizi**Nizomiy nomidagi Toshkent Davlat Pedagogika Universiteti talabasi*

Annaotatsiya: *Arduino dasturida dasturlash algoritmini tuzishda raqamli kirish. Arduino dasturida dasturlash algoritmini tuzishda analog kirish. Arduino dasturida dasturlash algoritmini tuzishda **raqamli** va **analog** kirishlarni ishlatish, turli sensorlar va qurilmalarni o'qish va ularni dasturda ishlatish uchun zarurdir. Arduino analog signallarni, shuningdek raqamli signallarni kiritishi va chiqarishi mumkin ekan. Analog signal-bu faqat ikkita qiymatga ega bo'lgan raqamli signaldan farqli o'laroq, har qanday qiymatni qabul qilishi mumkin: yuqori va past. Analog signallarning qiymatini o'lchash uchun Arduino o'rnatilgan analog-raqamli konvertorga (10-bitli ADC - Analog to Digital Converter orqali o'qiladi) ega. ADC analog kuchlanishni raqamli qiymatga aylantiradi. Analog signal qiymatini olish uchun foydalanadigan funksiya analogRead(pin). Ushbu funksiya analog kirish pinidagi kuchlanish qiymatini o'zgartiradi va mos yozuvlar qiymatiga nisbatan 0 dan 1023 gacha raqamli qiymatni qaytaradi. Standart mos yozuvlar kuchlanishi 5 V (5 V Arduino platalari uchun) yoki 3,3 V (3,3 V Arduino platalari uchun). U bitta parametrga ega, ya'ni pin raqami.*

Аннотация: *Цифровой ввод при создании алгоритма программирования в Arduino. Аналоговый ввод при построении алгоритма программирования в Arduino. При составлении алгоритма программирования в программе Arduino необходимо использовать цифровые и аналоговые входы, считывать различные сенсоры и устройства и использовать их в программе. Arduino может вводить и выводить аналоговые сигналы, а также цифровые сигналы. Аналоговый сигнал, в*



отличие от цифрового, который имеет только два значения, может принимать любое значение: высокое и низкое. Для измерения значения аналоговых сигналов Arduino имеет встроенный аналого-цифровой преобразователь (читается через 10-битный ADC - Analog to Digital Converter). ADC преобразует аналоговое напряжение в цифровое значение. Функция, используемая для получения значения аналогового сигнала `analogRead (pin)`. Данная функция изменяет значение напряжения на аналоговом входе и возвращает цифровое значение от 0 до 1023 относительно значения соответствующих записей.

Annotation: Digital introduction to the creation of a programming algorithm in the Arduino program. Analog input when creating a programming algorithm in the Arduino program. When creating a programming algorithm in the Arduino program, it is necessary to use digital and analog inputs, read various sensors and devices, and use them in the program. Arduino can input and output analog signals, as well as digital signals. An analog signal, unlike a digital signal, which has only two values, can take any value: upper and lower. For measuring the value of analog signals, Arduino has an integrated analog-to-digital converter (read through 10-bit ADC - Analog to Digital Converter). ADC converts analog voltage to a digital value. The function used to obtain the value of the analog signal is `analogRead (pin)`. This function changes the value of the voltage at the analog input pin and returns a numerical value from 0 to 1023 relative to the value of the corresponding records.

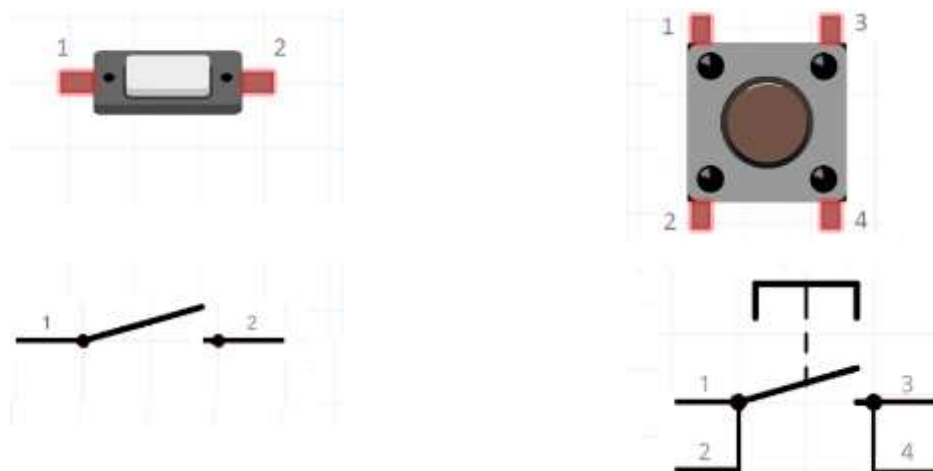
Tayanch so'zlar: **Raqamli kirish (Digital Input), Analog kirish (Analog Input),** Arduino, dasturlash, maket platasi, svetodiod.

Kirish. Raqamli kirish (Digital Input) - Raqamli kirish, faqat ikki holatda bo'lishi mumkin: **HIGH (yuqori) yoki LOW (past)**. Masalan, button (tugmacha) yoki **line sensor** kabi qurilmalar raqamli kirish sifatida ishlatiladi. Raqamli kirishda



0 (LOW) va **1** (HIGH) holatlari ishlatiladi. Sizlar bilan Arduino dasturida dasturlash algoritmini tuzishda raqamli kirish qurilmalarini ko‘rib chiqamiz.

Asosiy qism: Tugmacha (button) oddiy mexanik yoki elektr kaliti bo‘lib, uni bosish orqali faollashtirish uchun mo‘ljallangan. Bu elektr oqimini boshqarish, qurilma yoki tizimga signal berish yoki ma’lum bir harakatni boshlash uchun turli xil dasturlarda ishlatiladigan eng keng tarqalgan kalit turlaridan biridir. Bosish tugmalari kundalik narsalarda va sanoat tizimlarida ham qo‘llaniladi. Bosish tugmasi tasviri 1-rasmda ko‘rsatilgan.



1-rasm. Tugmacha (button)ning 2D va sxematik ko‘rinishi.

Tugmani bosish orqali elektr zanjirini ochish yoki yopish uchun ishlatiladigan oddiy, ammo ko‘p qirrali elektr kaliti. Ushbu kalitlar turli shakl va o‘lchamlarda bo‘ladi, lekin ularning barchasi bir xil asosiy prinsipga ega: tugmani bosish kalit holatini ochiqdan yopiqga yoki aksincha o‘zgartiradi. Ko‘pincha qoniqarli chertish bilan birga keladigan ushbu harakat elektr yo‘lini tugatadi yoki buzadi, qurilmaning funksiyasini yoqadi yoki o‘chiradi. Button qanday ishlashini tushunish uning ichki qismlarini diqqat bilan ko‘rib chiqishni talab qiladi.



Tugma Aktuatori: bu siz jismonan bosgan qism. U kalit ichidagi mexanizmga ulangan.

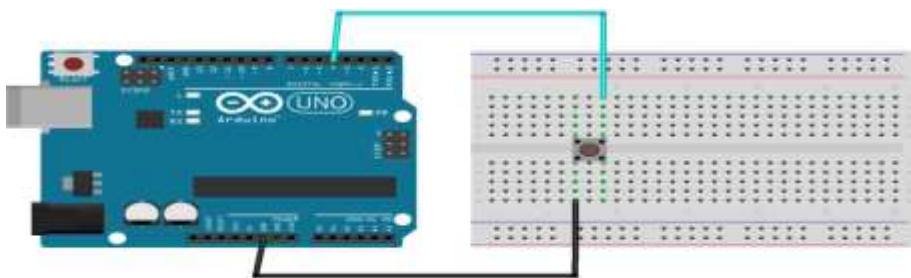
Kontaktlar: tugma kalitlari odatda metall kabi o'tkazuvchan materiallardan tayyorlangan bir yoki bir nechta kontaktlar to'plamini o'z ichiga oladi. Standart holatida bu kontaktlar ajratilgan.

Ichki mexanizm: tugmani bosganingizda, u elektr zanjirini to'ldirib, kontaktlarni bir-biriga itaradigan ichki mexanizmni ishga tushiradi. Qulflash kalitlarida ushbu mexanizm tugmachani yana bosmaguningizcha kontaktlarni joyida qulflaydi.

Qaytaruvchi: tugma bosilgandan so'ng asl holatiga qaytishini ta'minlash uchun ishlatiladi va ko'p hollarda bir lahzalik harakatni yaratadi.

Arduino bilan turli xil elektron konfiguratsiyalarga ega tugmachani qanday ishlashi bilan tanishib chiqamiz. Shuningdek, siz turli xil ilovalar uchun tugmachani qanday ishlatishni ko'rasiz va Arduining ba'zi imkoniyatlaridan foydalanasiz, masalan, uzilishlar. tugmachani Arduino dasturida dasturlash uchun Arduino Uno platasidan foydalanmoqchiman, ammo ushbu dasturlash algoritmi siz topishingiz mumkin bo'lgan boshqa Arduino platalari uchun ham ishlaydi.

Arduino platasidan foydalanib bosish tugmachani ulanish sxemasini ko'rib chiqamiz.



2-rasm. Tugmacha (button)ning Arduino platasiga ulanishining sxema ko'rinishi.



Sxemani qurish ketma-ketligi:

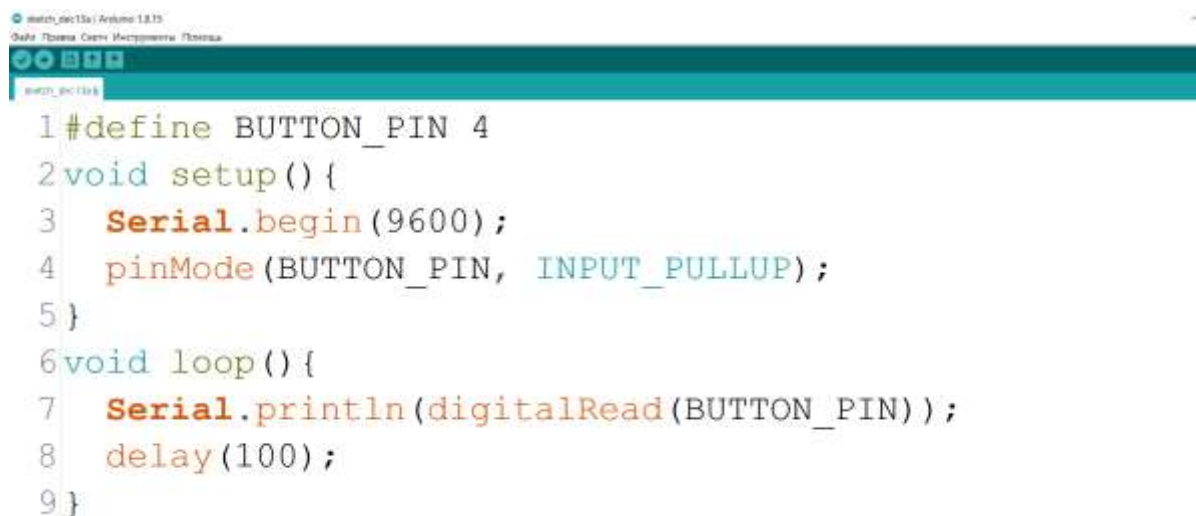
Arduinoni o‘chirilganligiga ishonch hosil qiling.

Rasmdagi kabi maket platasi o‘rtasiga bosish tugmasini ulang.

Bir tugmachaning oyoqida Arduino platasidagi GND piniga simni (iloji bo‘lsa qora) ulang.

Tugmaning yuqori chap va pastki chap oyoqlari bir-biriga, yuqori o‘ng va pastki o‘ng oyoqlari esa bir-biriga bog‘langan. Agar siz GND simini chap tomonga ulagan bo‘lsangiz, o‘ng tomonga boshqa simni ulang, shunda ular bir-biriga ulanmagan. Ushbu boshqa sim raqamli pinga o‘tadi, masalan 4. Hozircha sxemada rezistor mavjud emas.

Tugma holatini o‘qish uchun Arduino kodi: Biz bu erda qilmoqchi bo‘lgan narsa shunchaki tugmachaning holatni o‘qish va soniyasiga 10 marta monitor portda chop etishdir.



```
1#define BUTTON_PIN 4
2void setup() {
3  Serial.begin(9600);
4  pinMode(BUTTON_PIN, INPUT_PULLUP);
5}
6void loop() {
7  Serial.println(digitalRead(BUTTON_PIN));
8  delay(100);
9}
```

Keling! ushbu bosish tugmachasini sozlash uchun kodni satr bo‘yicha ajratamiz.

#define BUTTON_PIN 4



Avval biz tugma pin-kodi uchun `#define` yaratamiz, shuning uchun har safar ushbu pin-kodni kodimizda ishlatmoqchi bo'lganimizda "4" yozishimiz shart emas. Bu sizning dasturlash algoritmingizni osonlashtiradi, ayniqsa tugmani keyinroq boshqa pinga ulamoqchi bo'lsangiz. Shunday qilib, siz dasturning yuqori qismida joylashgan kodning bitta qatorini o'zgartirishingiz kerak va shuning uchun uni topish oson.

```
void setup(){
```

```
Serial.begin (9600); tekshirildi imloviy
```

Void setup () funksiyasi birinchi bo'lib, faqat bir marta chaqiriladi. Bu yerda biz ketma-ket aloqani boshlashimiz kerak, shunda biz tugmachadan olingan ma'lumotlarni chop etish uchun ketma-ket monitordan foydalanishimiz mumkin.

```
pinMode (BUTTON_PIN, INPUT_PULLUP);}
```

Kodingizdagi tugmachani shu tarzda ishga tushirasiz. `void setup ()` da siz `pinMode ()` funksiyasidan argument bilan foydalanasiz: avval tugma pin – bu erda `BUTTON_PIN` "4" bilan almashtiriladi – va keyin biz pin uchun kerakli rejimni tanlaymiz.

Biz tugmachadan ma'lumotlarni o'qiymiz "read" rejimini tanlaymiz – "write" yozish rejimini tanlamaymiz va keyin bizda hali ham 2 ta variant mavjud: kirish yoki `INPUT_PULLUP`. Bizning sxemamizda rezistor yo'qligi sababli, biz Arduino platasining ichki tortish rezistoridan foydalanish uchun `INPUT_PULLUP` dan foydalanamiz.

Tugma holatini o'qish uchun kod

```
void loop(){
```

```
Serial.println(digitalRead(BUTTON_PIN));
```

Endi tugma uchun pin-kod kirish sifatida ishga tushiriladi va biz cheksiz ko'p marta qayta-qayta chaqiriladigan `void loop ()` ga kiramiz.

Tugma holatini o'qish uchun biz `digitalRead()` funksiyasidan foydalanamiz, bitta argument bilan: tugma pin-kodi. Ushbu funksiyaning natijasi yuqori yoki past



bo'ladi. Bu erda biz tortib olinadigan konfiguratsiyadamiz (INPUT_PULLUP bilan), tugma bosilmasa, siz yuqori o'qiysiz. Tugma bosilganda siz past o'qiysiz.

Keyin, biz natijani qo'yamiz `digitalRead()` funksiyaga `Serial.println()`, shunchaki Serial Monitor ma'lumotlarni chop etadi.

```
delay(100); }
```

Tugma holatini o'qib, uni chop etgandan so'ng, biz 100 millisekundlik `delay()` qo'shamiz, ya'ni biz bu amalni sekundiga taxminan 10 marta bajaramiz. `void loop()` dan chiqqanimizdan so'ng, u yana chaqiriladi va hokazo va hokazo.

Endilikda kodni kompilyatsiya qilish va Arduinoga yuklash orqali sinab ko'rishingiz mumkin.

Serial monitorni oching va tugmachani bir necha marta bosib qo'yib yuboring. Siz shunga o'xshash narsani ko'rasiz:

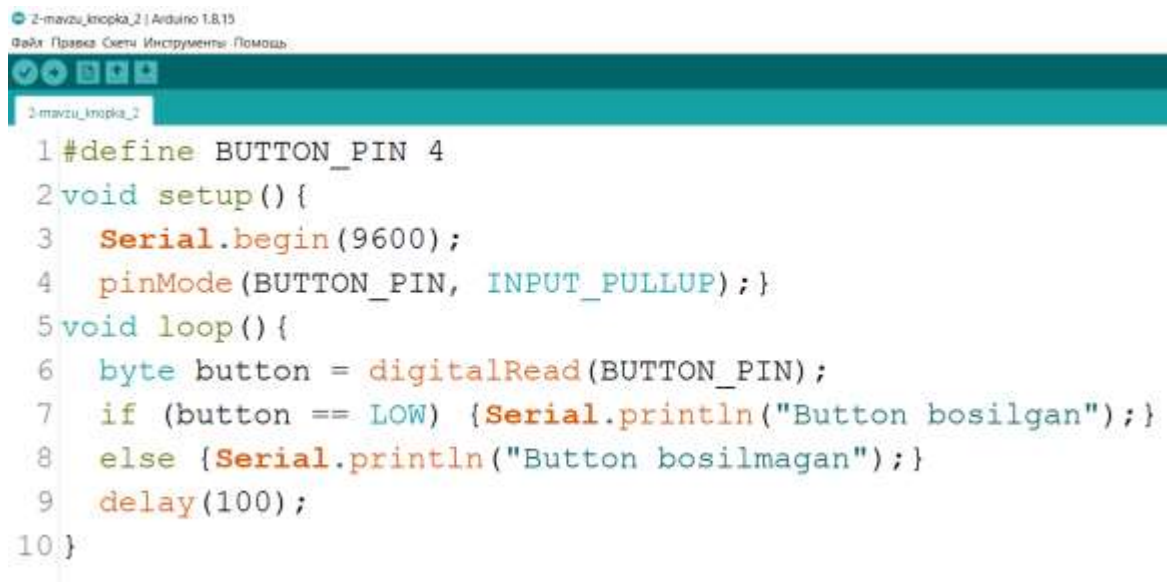
```
1
1
1
1
0
0
0
0
1
1
1
1
```

Har 100 millisekundda siz 0 yoki 1 bilan yangi qatorga ega bo'lasiz.

Tugma bosilmaganda siz "1" bosilganini ko'rasiz va tugma bosilganda siz "0"ni ko'rasiz. Ushbu birinchi kod tugmachadan keladigan ma'lumotlarni ko'rish uchun minimal misol sifatida juda yaxshi. Ammo haqiqiy dasturda siz tugma holatini chop etishdan ko'ra qiziqroq narsalarni qilishni xohlashingiz mumkin. Masalan, siz tugma holatini o'qishingiz, uni o'zgaruvchining ichida saqlashingiz



va keyin qiymatga qarab ma'lum bir amalni bajarishingiz mumkin. Quyidagi Arduino kodidan foydalanamiz.



```
1#define BUTTON_PIN 4
2void setup() {
3  Serial.begin(9600);
4  pinMode(BUTTON_PIN, INPUT_PULLUP);
5void loop() {
6  byte button = digitalRead(BUTTON_PIN);
7  if (button == LOW) {Serial.println("Button bosilgan");}
8  else {Serial.println("Button bosilmagan");}
9  delay(100);
10 }
```

O'rnatish bosqichi bir xil. `void loop()` da ko'rib turganingizdek, endi biz `digitalRead()` dan olingan qiymatni o'zgaruvchiga saqlaymiz, biz button deb nomlaymiz. Ushbu o'zgaruvchining ma'lumotlar turi `bayt`, bu sizga yuqori va past qiymatlar uchun kerak bo'lgan narsadir.

Keyin, oddiy if tuzilishi bilan biz holatning LOW ligini tekshiramiz, ya'ni ichki Arduino tortish qarshiligi bilan konfiguratsiyamizda tugma bosiladi. Buni aniqlaganimizda, biz xohlagan harakatni qilishga qaror qilishimiz mumkin va agar shart noto'g'ri bo'lsa, bu holat HIGH ekanligini anglatadi (chunki faqat 2 ta holat mavjud), biz tugma bosilmaganligini bilamiz.

Endi bu kodni ishlatishimiz mumkin va har bir bosilganda "Button bosilgan" ko'rish 100 millisekundlarda, va siz tugmachani bosmaganingizda, "Button bosilmagan".yozuvlar monitor portda hosil bo'ladi.

Tugmachani Arduino platasiga ulashda rezistorlaridan foydalanish



Biz 2-rasimda keltirilgan sxemamizda hech qanday rezistor ishlatmadik – aslida biz ichki Arduino rezistoridan foydalangan holda amalga oshirdik.

Asosan, tugmachani raqamli pin-kodga ulaganingizda, Arduino o'qiydigan qiymat 0V va 5V orasida bo'ladi. agar qiymat 0v ga yaqin bo'lsa, sizning kodingizda LOW bo'ladi va agar u 5V ga yaqin bo'lsa, HIGH qiymatga ega bo'ladi.

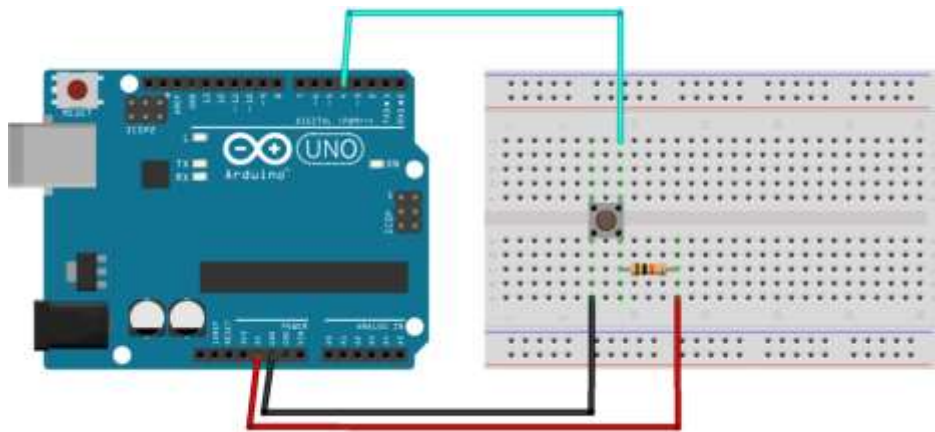
Agar siz hech qanday qarshilik ko'rsatmasangiz, qiymat 0V va 5V o'rtasida "o'zgaruvchi" bo'lishi mumkin, shuning uchun sizga tasodifiy va aniq bo'lmagan natijalar beradi. Rezistorni qo'shib, siz standart holatni LOW yoki HIGH bo'lishga majburlashingiz mumkin va tugmani bosganingizda, 1-holatga aksincha bo'ladi.

Rezistor yo'q, lekin ichki tortishish qarshiligini faollashtirish uchun INPUT_PULLUP rejimidan foydalanildi (standart o'qish kuchlanishini 5V qiymatgacha majbur qiladi). Biz 1-variantdan foydalanganmiz, endi 2 va 3-variantlar bilan qanday ishlashni ko'rib chiqamiz.

Nega 2 va 3-variantlarni ko'rib chiqishimiz kerak, agar biz sxemani hech qanday rezistorsiz qura olsak? Sabab, chunki ichki tortish qarshiligi biz sxemada qo'lda qo'shmoqchi bo'lgan rezistorga nisbatan ancha zaif. Agar sizda uzunroq simlar bo'lsa, bu ishonchsiz natijalarga olib kelishi mumkin shu sababli bizga qo'shimcha resistor kerak bo'ladi

Arduino platasi va tugmacha(button)ni tashqi qarshilik bilan ulanish
sxemasi

Ushbu sxema uchun bizga 10k rezistor va biz ilgari ishlatgan har qanday boshqa komponent kerak bo'ladi.



3-rasm. Arduino platasi va tugmacha(button)ni tashqi qarshilik bilan ulanish sxemasi

Bu sxemada quyidagi dasturlash algoritmi ishlaydi

2-mavzu_knopka_2 | Arduino 1.8.15
Файл Правка Скетч Инструменты Помощь

```
1 #define BUTTON_PIN 4
2 void setup() {
3   Serial.begin(9600);
4   pinMode(BUTTON_PIN, INPUT);
5 }
6 void loop() {
7   byte button = digitalRead(BUTTON_PIN);
8   if (button == HIGH) {Serial.println("Button bosildi");}
9   else {Serial.println("Button bosilmadi");}
10  delay(100); }
```

2. Analog kirish (Analog Input) - Arduino analog signallarni, shuningdek raqamli signallarni kiritishi va chiqarishi mumkin ekan. Analog signal-bu faqat ikkita qiymatga ega bo'lgan raqamli signaldan farqli o'laroq, har qanday qiymatni qabul qilishi mumkin: yuqori va past. Analog signallarning qiymatini o'lchash uchun Arduino o'rnatilgan analog-raqamli konvertorga (10-bitli ADC - Analog to



Digital Converter orqali o'qiladi) ega. ADC analog kuchlanishni raqamli qiymatga aylantiradi. Analog signal qiymatini olish uchun foydalanadigan funksiya analogRead(pin). Ushbu funksiya analog kirish pinidagi kuchlanish qiymatini o'zgartiradi va mos yozuvlar qiymatiga nisbatan 0 dan 1023 gacha raqamli qiymatni qaytaradi. Standart mos yozuvlar kuchlanishi 5 V (5 V Arduino platalari uchun) yoki 3,3 V (3,3 V Arduino platalari uchun). U bitta parametrga ega, ya'ni pin raqami.

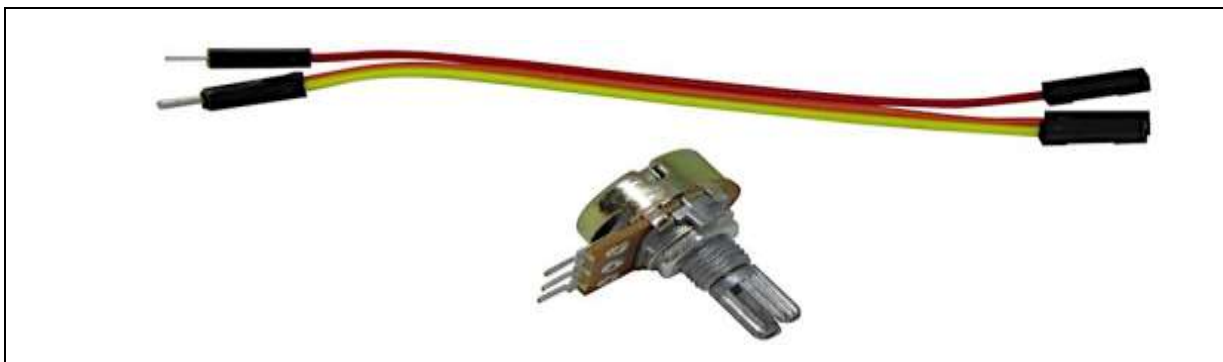
Arduino o'rnatilgan raqamli-analog konvertorga ega emas, lekin u analog chiqishning ba'zi funksiyalariga erishish uchun raqamli signalni impuls kengligida modulyatsiya qilishi mumkin. Signal signalini chiqarish uchun ishlatiladigan funksiya analog yozish (pin, qiymat). pin-bu PV chiqishi uchun ishlatiladigan pin raqami. qiymat-bu signalning ish tsikliga mutanosib raqam. Qachon qiymati = 0, signal har doim yopiq. Qachon qiymati = 255, signal har doim. Ko'pgina Arduino platalarida PVM funksiyasi pinlarda mavjud 3, 5, 6, 9, 10, va 11. Ko'pgina pimlardagi PV signalining chastotasi taxminan 490 Gts. Uno va shunga o'xshash platalarda 5 va 6 pinlar taxminan 980 Gts chastotaga ega. Leonardodagi 3 va 11 pinlar ham 980 Gts da ishlaydi. Bu jarayonni bosqichma-bosqich ko'rib chiqaylik. Raqamli kirishni o'qish uchun analogRead() funksiyasidan foydalanamiz

Arduino bilan AnalogRead (INPUT):

Arduino Uno va Arduino MEGA bilan analogRead usulidan foydalanib potentsiometr qurilmasini dasturlash algoritmini tuzib chiqamiz va arduino analog kirish pinidan analog qiymatni qanday oladi va o'qiydi kuzatamiz uchun potentsiometrni Arduino analog kirish piniga ulaymiz. Potansiyometr Arduino analog pinida 0v dan 5v gacha kuchlanish o'rnatadi. Shundan so'ng PIN ustida analog qiymatini o'qiydi va serial monitor oynada uni ko'rsatadi. Potansiyometr o'zgaruvchan rezistordir. Qarshiligini o'zgartirish uchun potentsiometrning milini yoki vintini burash kerak. Potentsiometrni Arduinoga ulashning eng oson usuli

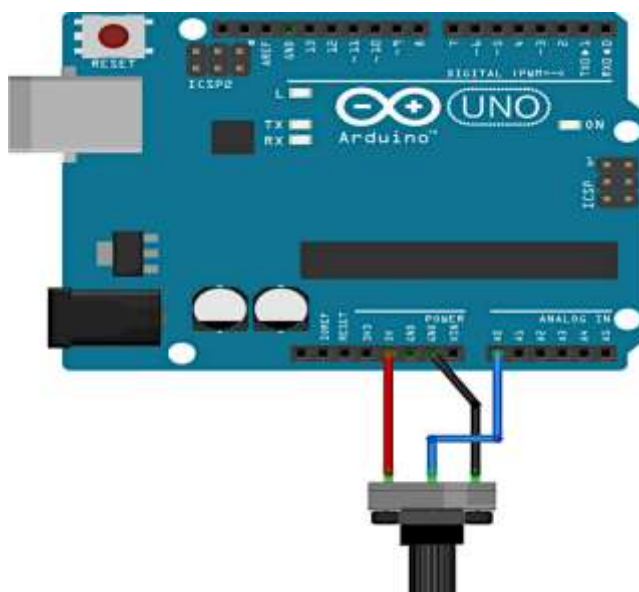


dupont simlaridan foydalanish. Asosan, dupont simlari potansiyometr pinlariga ulanadi. dupont simlarining boshqa uchi to'g'ridan-to'g'ri Arduino pinlariga yoki ulagichlariga ulanadi.



4-rasm. Potinsiyometr va dupont similar.

Birinchidan, potentsiyometrning tashqi pinlaridan biridan Arduinodagi 5V ga simni ulanadi. Ikkinchidan, qarama-qarshi tashqi pinni Arduino GND-ga ulanib. Nihoyat, potansiyometrning o'rta pinini A0 dan A5 gacha analog Arduino piniga ulanadi.



5-rasm. Arduino platasiga potensiyometrning ulanish sxemasi.

Serial monitor oynada ko'rsatadi 0 - qachonki 0v yoki GND A0 pinda mavjud. Boshqa tomondan, serial monitor oyna ko'rsatadi 1023 5V A0 pinda



mavjud bo'lsa. Potansiyometr milini aylantirib, A0 pinidagi kuchlanishni o'zgartirilgan natijada, ketma-ket monitor oynasi 0 va 1023 o'rtasidagi xom analog qiymatlarni ko'rsatadi. Arduino platasidagi asosiy mikrokontroller chipining ichida ADC mavjud. ADC raqamli Konverter uchun Analog uchun turadi. Ushbu qurilma analog kirish pinidagi analog kuchlanishni 10 dan 0 gacha bo'lgan 1023 bitli raqamga o'zgartiradi.

Natija: Arduino platformasi bilan ishlashda **raqamli** va **analog** kirishlar muhim o'rin tutadi, chunki ular mikrokontrollerga tashqi muhitdan ma'lumotlarni olish imkoniyatini yaratadi. Har ikkala turdagi kirishlar Arduino dasturlarida turli xil sensorlar, tugmalar, potentiometrlar va boshqa qurilmalar bilan ishlashda keng qo'llaniladi.

Raqamli kirishlar ikki holatda bo'lishi mumkin: yuqori (HIGH) yoki past (LOW). Arduino mikrokontrolleri bu holatlarni o'qib, ularga mos ravishda ishlov beradi. Masalan, **tugma yoki reed kontakt** kabi qurilmalar yordamida raqamli signal olinganda, Arduino tizimi bu signalni o'qiydi va unga javoban tegishli amallarni bajaradi. Raqamli kirishlar juda oddiy va oson ishlashni ta'minlaydi, ammo ular faqat ikki holatni (ON/OFF) aniqlay oladi. **Misol:** Tugma bosilganmi yoki yo'qmi, buni bilish uchun raqamli kirish ishlatiladi.

Analog kirishlar mikrokontrollerga uzatilgan uzluksiz o'zgaruvchi signalni o'qish imkoniyatini beradi. Arduino boardlari, masalan, **LDR (Light Dependent Resistor)**, **potentiometr**, **harorat sensorlari** kabi qurilmalardan analog signalni olishga imkon beradi. Analog kirishlar o'qilgan qiymatni 0 dan 1023 gacha bo'lgan sonlarga aylantiradi, bu esa o'zgarmas yoki uzluksiz o'zgarishlarni o'lchash imkoniyatini yaratadi. **Misol:** Potentiometr yordamida nur darajasi yoki haroratni o'lchashda analog kirish ishlatiladi.



FOYDALANILGAN ADABIYOTLAR

1. X.N.Nazarov —Robototexnika asoslaril. TDTU., Toshkent:, 2005 y.
2. “Робототехника для детей и родителей” С.А. Филиппов, Санкт-Петербург “Наука” 2010. - 195 с. 12
3. Бокселл Дж. Б78 Изучаем Arduino. 65 проектов своими руками. — СПб.: Питер, 2017. . — 400 с.
4. С. Монк Программируем Arduino. Профессиональная работа со скетчами . — СПб.: Питер, 2017.

Axborot manbaalari

1. <http://www.robotics.uc.edu>
2. <http://www.robotics.utexas.edu/rrg>
3. <http://www.lib.tpu.ru/fulltext2/m/2012/m85.pdf>