

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"
ІНСТИТУТ ПІДПРИЄМНИЦТВА ТА ПЕРСПЕКТИВНИХ ТЕХНОЛОГІЙ

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторної роботи № 3

“ЗНАЙОМСТВО З ARDUINO.

ПРИНЦИПИ РОБОТИ З ВХІДНИМИ ДАВАЧАМИ”

з дисципліни ”Архітектура комп’ютерів”

для студентів спеціальності 6.050101 «Комп’ютерні науки»

Львів 2017

Методичні вказівки для виконання лабораторної роботи № 2 “ЗНАЙОМСТВО З ARDUINO. ПРИНЦИПИ З ВХІДНИМИ ДАВАЧАМИ” з дисципліни ”Архітектура комп’ютерів” для студентів спеціальності 6.050101 «Комп’ютерні науки» / Укл.: В.М. Теслюк, О.Борейко, В.Я.Коваль.

Укладачі:

зав. кафедри ІСТ, д.т.н., професор Теслюк Василь Миколайович;
викладач кафедри ІСТ, Борейко Олег ,
викладач кафедри ІСТ, Коваль Володимир Ярославович.

Відповідальний за випуск В.М. Теслюк

Методичні вказівки до виконання магістерських кваліфікаційних робіт обговорено та схвалено на засіданні кафедри інформаційних систем і технологій.
Протокол № від « » _____ 2017 р.

Зав. кафедри ІСТ _____ проф., д.т.н., В.М. Теслюк

Методичні вказівки до виконання магістерських кваліфікаційних робіт обговорено та схвалено на засіданні Науково-методичної ради Навчально-наукового інституту підприємництва та перспективних технологій Національного університету «Львівська Політехніка».

Протокол № _____ від « » _____ 2017 р.

Голова НМР _____ доц., к.ф.-м.н. Ю.М. Слюсарчук

ЗМІСТ

1	Мета роботи:	4
2	Теоретичні відомості	4
3	Порядок виконання роботи	13
3.1.1	Підключити кнопку та світлодіоди до плати Arduino згідно зі схемою варіанту.	13
3.1.2	Запустити на комп'ютері середовище розробки Arduino IDE	13
3.1.3	Розробити алгоритм роботи згідно завдання та написати код програми керування.	13
3.1.4	Підключити плату Arduino через USB до ПК	13
3.1.5	Натисніть кнопку Verify и переконайтесь, що у нижній частині вікна з'явився надпис Done Compiling. Це значить, що унаписаній програмі не знайдено помилок.....	13
3.1.6	Виберіть у Tools->Board ваш тип плати. Перевірте, чи правильно обрано USB-порт в Tools->Serial port. Після натисніть на кнопку Upload.....	13
3.1.7	Якщо внизу з'явився надпис "Done uploading" – процес завантаження програми в контролер пройшов успішно.....	13
3.1.8	Перевірити виконання завдання на контролері.....	13
3.1.9	Дати відповідь на контрольні запитання.	13
3.1.10	Скласти звіт з виконання лабораторної роботи та захистити його до початку виконання наступної лабораторної роботи.....	13
4	Контрольні запитання.....	13
5	Зміст звіту	14
6	Список використаної літератури.....	14
7	Варіанти індивідуальних завдань.....	15

Лабораторна робота №3

ЗНАЙОМСТВО З ARDUINO. ПРИНЦИПИ РОБОТИ ЗІ ВХІДНИМИ ДАВАЧАМИ

1 Мета роботи:

Ознайомитись з платою Arduino. Оволодіти, основними принципами роботи та підключення вхідних давачів до плати Arduino.

Здобуття практичних навичок роботи з мікроконтролером типу Arduino

2 Теоретичні відомості

Кнопка – найпростіший електромеханічний пристрій, для передачі електричного сигналу різним пристроям шляхом замикання або розмикання двох і більше контактів. В залежності від стану, в якому залишається кнопка після натискування на неї, вона буває двох видів:

- кнопки (перемикачі) з фіксованим положенням;
- тактові кнопки або нефіксовані (які повертаються у вихідне положення).

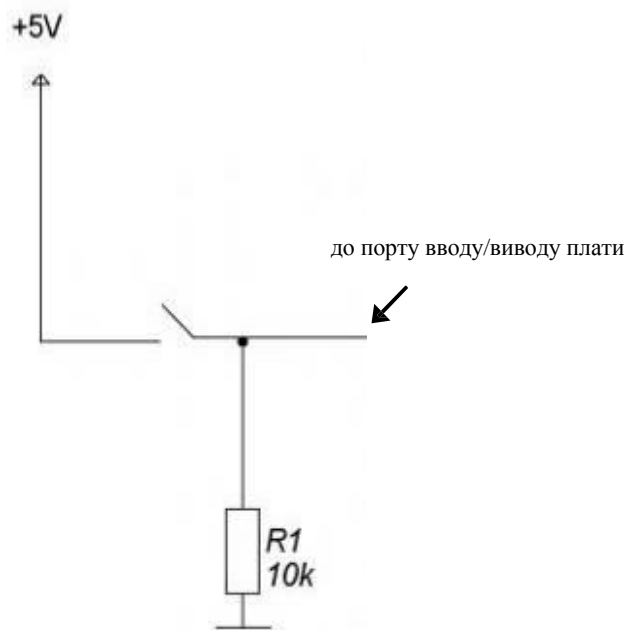


Рисунок 2-1Схема правильного підключення кнопки до піна плати.

Пін плати на вході повинен мати стан 0 або 1. Коли стан «підвішений», тобто не визначений, на вході будуть збиратися різні зовнішні наводки (статичні, електричні, електромагнітні), що буде призводити до хибного спрацювання кнопки. Щоб підвести пін до 0 або 1 використовують підтягуючі резистори. Вони бувають двох видів, верхньої або нижньої підтяжки. Верхні підключаються до плюса живлення, а нижні до мінуса.

Після підключення схеми переходимо до написання програми. Необхідно пін, до котрого буде підключатися кнопка, ініціалізувати на вхід за допомогою команди:

```
pinMode(buttonPin, INPUT); //де buttonPin – це номер  
піна до якого підключена кнопка
```

Також необхідно об'явити змінну в яку буде записуватися статус кнопки.

2.1 Простий приклад вмикання та вимикання світлодіода за допомогою кнопки

Розглянемо простий приклад вмикання та вимикання світлодіода за допомогою кнопки, підключеної до 7 піна. Схема підключення зображена на рисунку 2.2.

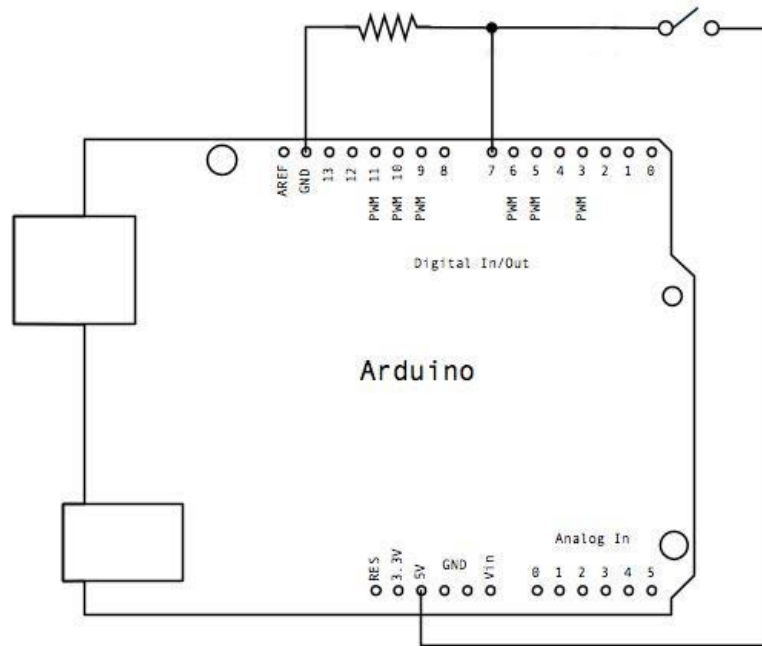


Рисунок 2-2 Схема підключення кнопки до Arduino Uno

Текст програми, у якій за допомогою натискання кнопки відбувається включення вбудованого світлодіода на виводі 13 виглядає так:

```
const int buttonPin = 7; // номер піна, до якого
                           підключена кнопка
int buttonState = 0; // змінна для
                     запису стану кнопки

void setup ()
{
  pinMode(buttonPin, INPUT); // ініціалізація піна на
                             введення
}

void loop ()
{
  buttonState = digitalRead(buttonPin);
  if (buttonState == HIGH) // якщо кнопка натиснута
  {
    digitalWrite(13, HIGH); // вбудований світлодіод
                             вмикається
  }
  else {
    digitalWrite(13, LOW); // вбудований світлодіод
                             вимикається
  }
}
```

2.2 Потенціометр, як аналоговий ввідний пристрій

Потенціометр – це змінний резистор з регульованим опором. Потенціометри використовуються в робототехніці як регулятори різних параметрів – гучність звуку, потужності, напруги тощо.

Потенціометр має три контакти (рис. 2.3). Середній контакт йде на аналоговий вихід. Зовнішні контакти йдуть до піна із живленням та до землі.



Рисунок 2-3 Зображення потенціометра

Для детального розуміння роботи потенціометра розглянемо приклад його роботи.

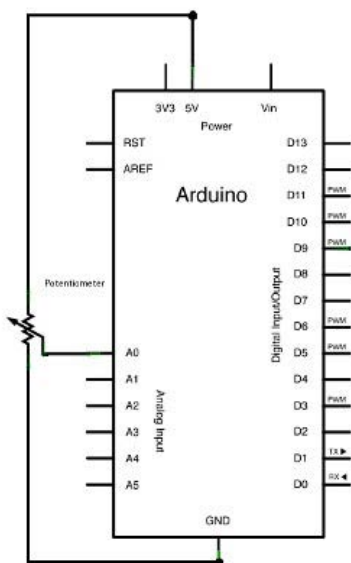


Рисунок 2-4 Схема підключення потенціометра до плати Arduino

На рисунку 2.4 зображена схема підключення потенціометра до плати. У цьому випадку потенціометр підключений між землею і +5 V потенціалами (оскільки плата Arduino живиться +5 V), а движок сполучений з каналом аналого-цифрового перетворювача мікроконтролера. У такому разі можна регулювати вихідну напругу потенціометра в межах від 0 до 5 V.

Приклад програми для роботи контролера з потенціометром:

```
void setup() { Serial.begin(9600);  
}  
void loop() {
```

```

    int sensorValue = analogRead(A0); // отримуємо поточне
    значення
    Serial.println(sensorValue, DEC); // виводимо результат
    на монітор
}

```

При ініціалізації встановлюємо потрібну швидкість зв'язку: `Serial.begin(9600);`. Далі в циклі ми постійно зчитуємо дані, що поступають з потенціометра за допомогою методу `analogRead()`. Оскільки значення знаходиться в діапазоні від 0 до 1023, можемо використовувати тип `int` для змінної `sensorValue`.

Отриманий результат будемо виводиться у вікно послідовного монітора в десятковому форматі.

2.3 Джойстик – маніпулятор, пристрій введення координат або напрямку руху.

Джойстик - це вхідний пристрій, що складається зі стержня, який обертається на базі, і повідомляє про кут або напрямок пристрою, який він контролює. Джойстики часто використовуються для керування відеоіграми та роботами.

Джойстик

PS2

використовується

тут.

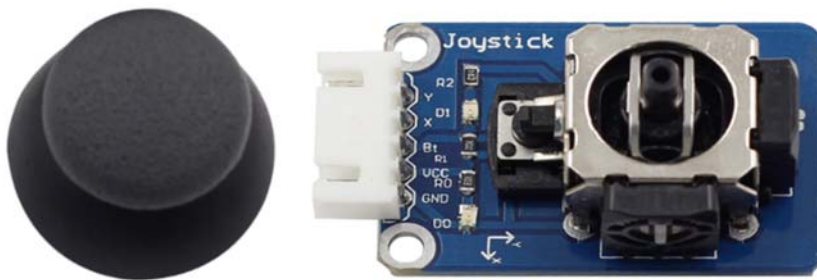


Рисунок 2-5 Зовнішній вигляд джойстика

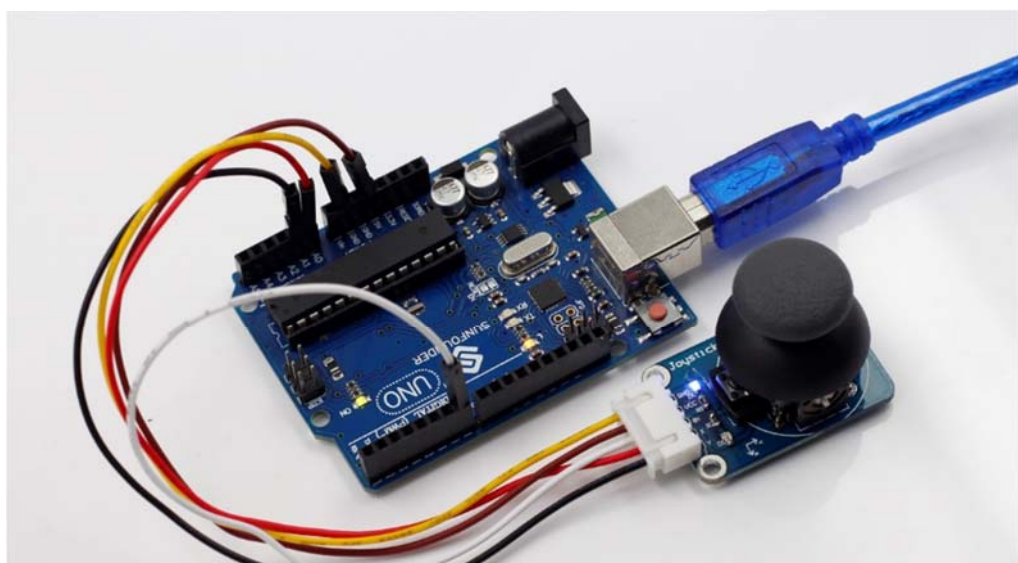


Рисунок 2-6 Приклад підключення джойстика до плати Ардуіно

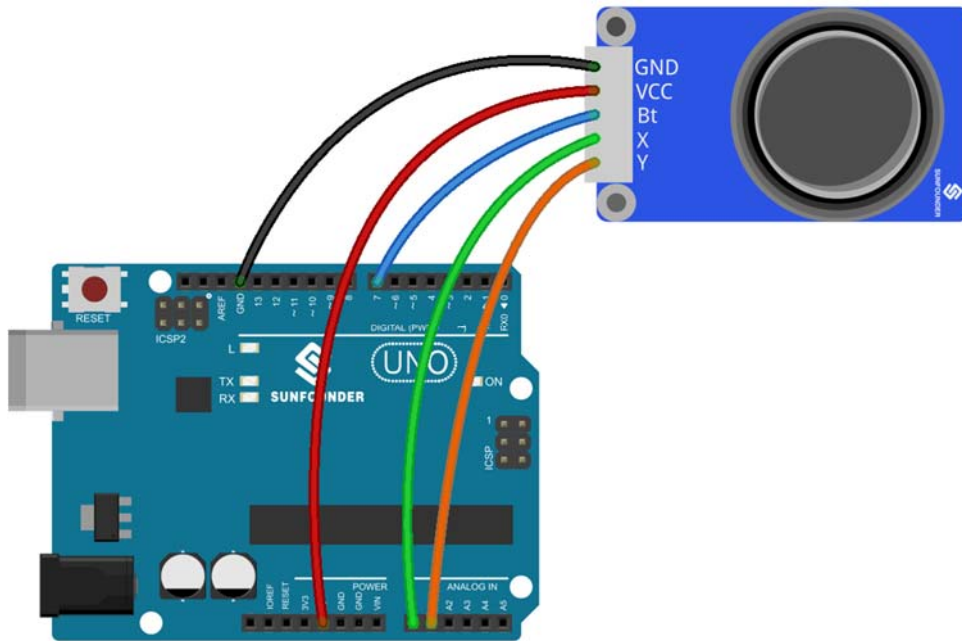


Рисунок 2-7 Схема підключення джойстика до алати Ардуіно

Програма (приклад) використання джойстика в якості ввідного пристрою:

```

/*****
* function:push the joystick and the coordinates of X
and Y axes displayed on Serial Monitor will change
accordingly;
* press down the joystick, and the coordinate of Z=0
will also be displayed.
connection:
Joystick Uno R3
GND    GND
VCC     5V
Bt      7
x       A0
y       A1
*****/
const int xPin = A0; //X attach to A0
const int yPin = A1; //Y attach to A1
const int btPin = 7; //Bt attach to digital 7
void setup()
{
  pinMode(btPin,INPUT); //set btpin as INPUT
  digitalWrite(btPin, HIGH); //and HIGH
  Serial.begin(9600); //initialize serial
}
void loop()
{
  Serial.print("X: "); //print "X: "
  Serial.print(analogRead(xPin),DEC); //read the value
of A0 and print it in decimal

```



```

Serial.print("\tY: "); //print "Y: "
Serial.print(analogRead(yPin),DEC); //read the value
of A1 and print it in decimal
Serial.print("\tZ: "); //print "Z: "
Serial.println(digitalRead(btPin));    ////read the
value of pin7 and print it
delay(100); //delay 100ms
}

```

2.4 Фоторезистор та Arduino

Фоторезистори дають вам можливість визначати інтенсивність освітлення. Вони маленькі, недорогі, вимагають мало енергії, легкі у використанні, практично не схильні до зносу. Саме через це вони часто використовуються в іграшках, гаджетах і пристроях.. Фоторезистори за своєю суттю є резисторами, які змінюють свій опір (вимірюється в Ом) в залежності від того, яка кількість світла потрапляє на їх чутливі елементи. Як вже говорилося вище, вони дуже дешеві, мають різні розміри і технічні характеристики, але в більшості своїй не дуже точні. Кожен фоторезистор поводить себе трохи інакше в порівнянні з іншим, навіть якщо вони з однієї партії від виробника. Відмінності в показаннях можуть досягати 50% і навіть більше! Так що розраховувати на прецизійні вимірювання не варто. В основному їх використовують для визначення загального рівня освітленості в конкретних, "локальних", а не "абсолютних" умовах.

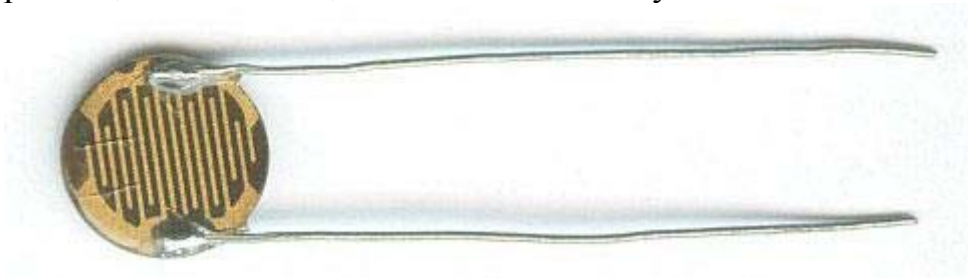


Рисунок 2-8 Загальний вигляд фоторезистора

Фоторезистори є відмінним вибором для вирішення завдань на зразок "навколо темно або світло", "чи є щось перед датчиком (що обмежує надходження світла)", "яка з ділянок має максимальний рівень освітленості".

Так як фоторезистори по суті є опором, вони не мають полярності. Це означає, що ви можете їх підключати їх ноги 'як завгодно' а вони будуть працювати!

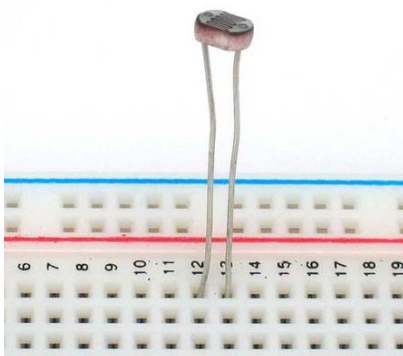


Рисунок 2-9 Приклад включення фоторезистора

Найпростіший варіант використання: підключити одну ногу до джерела

живлення, другу - до землі через понижуючий резистор. Після цього точка між резистором з постійним номіналом і змінним резистором - фоторезистором - підключається до аналогового входу мікроконтролера. На малюнку нижче показана схема підключення до Arduino.

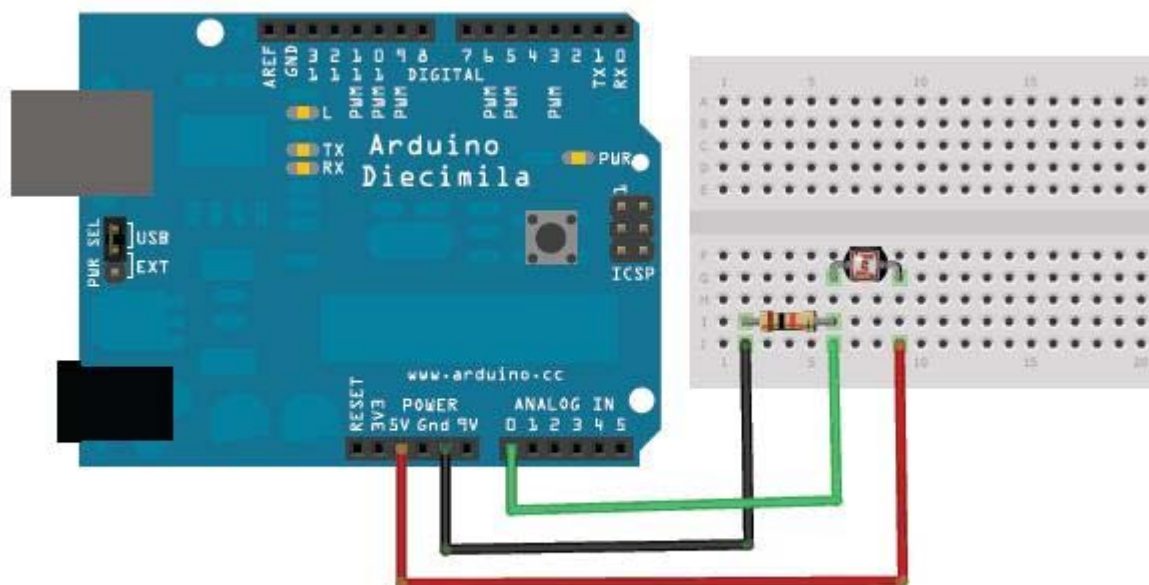


Рисунок 2-10 Підключення фоторезистора до Ардуїно

Простий приклад використання фоторезистора в проекті Ардуїно

У цьому скетчі зчитуються аналогові значення для визначення яскравості світлодіода. Чим темніше буде, тим яскравіше буде світити світлодіод! Не забудьте, що світлодіод повинен бути підключений до ШІМ контакту для роботи даного прикладу. В даному випадку використовується контакт 11.

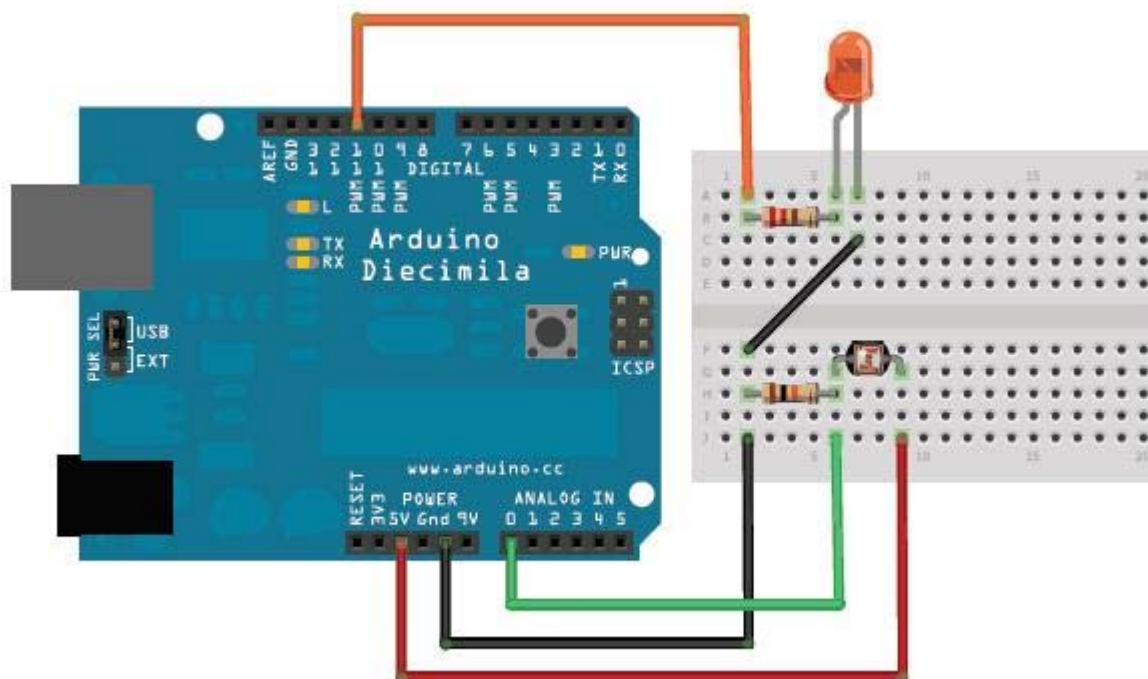


Рисунок 2-11 Схема підключення до Ардуїно

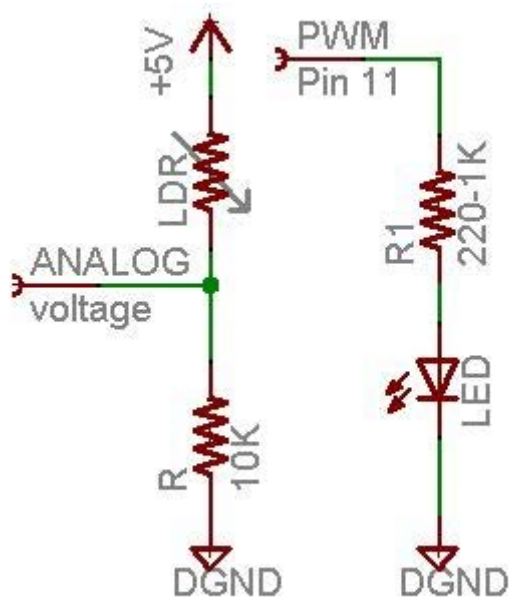


Рисунок 2-12 Електрична схема прикладу використання фоторезистора в проекті Ардуїно

Програма

```

/* простий скетч для фоторезистора. Підключіть одну
ногу фоторезистора до 5 В, другу до аналогового піну
Analog 0. Після цього підключіть резистор на 10 кОм
між Analog 0 і землею. Через резистор підключіть
світлодіод між 11 піном і землею.*/
int photocellPin = 0; // сенсор і понижаючий резистор 10
кОм підключені до a0
int photocellReading; // зчитуємо аналогові значення з
сенсора
int LEDpin = 11; // світлодіод на пін 11 (ШІМ пин)
int LEDbrightness; //
void setup(void) {
// інформацію для відлагоджування направляємо на системний
монітор
Serial.begin(9600);
}
void loop(void) {
photocellReading = analogRead(photocellPin);
Serial.print("Analog reading = ");
Serial.println(photocellReading); // аналогові
значення з сенсора
// світлодіод горить яскравіше, якщо рівень освітленості на
датчику зменшується
// це значить, що ми повинні інвертировать зчитуються значення
від 0-1023 до 1023-0
photocellReading = 1 023 - photocellReading;

```

```
// тепер ми повинні перетворити діапазон 0-1023 в 0-255, так
// як саме такий діапазон використовує
analogWriteLEDBrightness = map(photocellReading,
0, 1023, 0, 255);
analogWrite(LEDpin, LEDbrightness);
delay(100);
}
```

2.5 Керування яркістю світіння світлодіода за допомогою двох кнопок (використання ШІМ каналів)

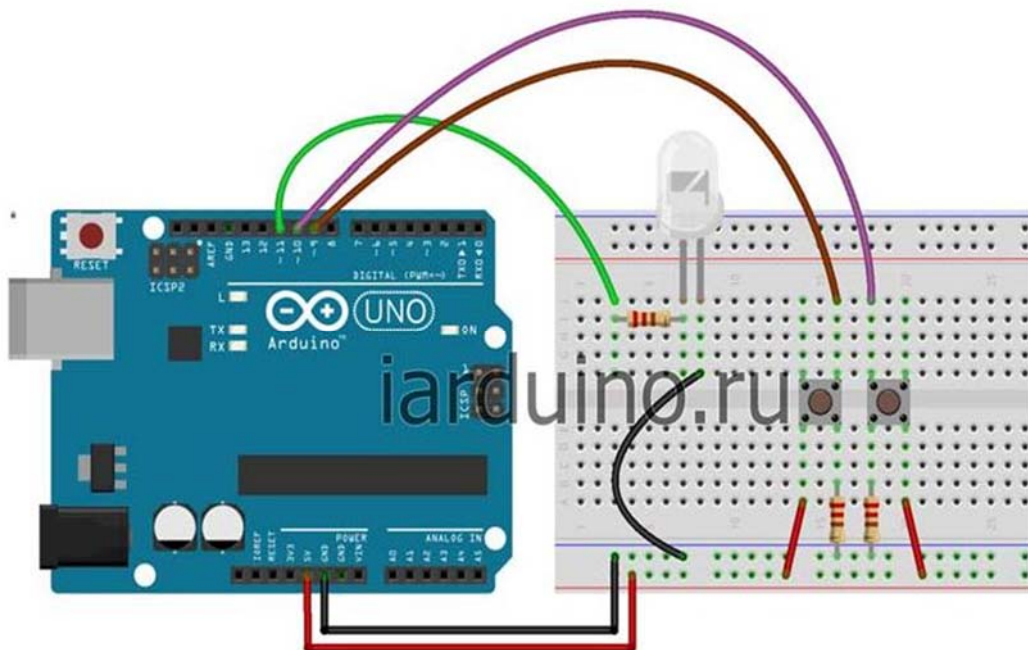


Рисунок 2-13 Схема підключень на макетній платі

Інтенсивність світіння буде регулюватися від 0 до 254 одиниць, де 0 – світлодіод виключений, а 254 - горить максимально (Наприклад при 127 яскравість буде на 50%).

Для підключення світлодіода необхідно використовувати резистор. Номінал резистора підбирається для кожного світлодіода індивідуально, залежно від його робочої напруги і струму. Зазвичай резистор в 1 кОм. Можна використовувати як універсальний варіант. Управління реалізуємо за допомогою двох тактових кнопок. Підключати тактові кнопки необхідно через підтягуючі резистори (До землі, GND), для захисту від статичних наведень (Помилкові спрацьовування кнопок від одного лише дотику і тд.). Номінал може бути різний. Рекомендуємо вам використовувати резистори 10 кОм - 100 кОм.

Приклад програми:

```
int led = 11; // Номер Pin до якого підключено діод
int brightness =0; // Змінна яскравості(від 0 до 254)
int buttonPlus=9; // Номер Pin до якого підключена кнопка 1
int buttonMinus=10; // Номер Pin до якого підключена кнопка 2
void setup() {
pinMode(led, OUTPUT); // Порт 11 (led) як вихід
```

```

}
void loop() {
  if (digitalRead(buttonPlus) == HIGH) {
    brightness += 5;
  } // Перевіряємо, якщо вхід під номером 9 (buttonPlus) має стан 5
  В збільшуємо значення змінної яркості на 5 одиниць
  if (digitalRead(buttonMinus) == HIGH) {
    brightness -= 5;
  }
  brightness = constrain(brightness, 0, 254);
  // Ця функція контролює, щоб змінна brightness не стала більше 254
  і менше 0, якщо значення виходить за ці межі то функція 0 або 254
  analogWrite(led, brightness); // встановлюємо стан
  яскравості діода
  delay(50); // Пауза 50 мілісекунд.
}

```

3 Порядок виконання роботи

- 3.1.1 Підключити кнопку та світлодіоди до плати Arduino згідно зі схемою варіанту.
- 3.1.2 Запустити на комп'ютері середовище розробки Arduino IDE
- 3.1.3 Розробити алгоритм роботи згідно завдання та написати код програми керування.
- 3.1.4 Підключити плату Arduino через USB до ПК
- 3.1.5 Натисніть кнопку Verify и переконайтесь, що у нижній частині вікна з'явився надпис Done Compiling. Це значить, що унаписаній програмі не знайдено помилок.
- 3.1.6 Виберіть у Tools->Board ваш тип плати. Перевірте, чи правильно обрано USB-порт в Tools->Serial port. Після натисніть на кнопку Upload.
- 3.1.7 Якщо внизу з'явився надпис “Done uploading” – процес завантаження програми в контролер пройшов успішно.
- 3.1.8 Перевірити виконання завдання на контролері
- 3.1.9 Дати відповідь на контрольні запитання.
- 3.1.10 Скласти звіт з виконання лабораторної роботи та захистити його до початку виконання наступної лабораторної роботи.

4 Контрольні запитання

1. В чому полягає особливість підключення вхідних датчиків до платформи

Arduino, особливості?

2. Як здійснюється підключення світлодіодів до джерела живлення?
3. Як встановлюються режими роботи вихідних портів контролера Arduino?

5 Зміст звіту

1. Тема і мета роботи.
2. Короткі теоретичні відомості.
3. Хід роботи.
4. Висновки.

6 Список використаної літератури

7 Варіанти індивідуальних завдань

Підключення та керування світлодіодним рядком за допомогою вхідного давача

Варіант	Кількість світлодіодів, та порядок вмикання
1.	Натискання кнопки 7, поступовий рух одиночного вогника вперед
2.	Фоторезистор 11, поступовий рух двох вогників
3.	Натискання кнопки 11, поступовий рух 7 вогників
4.	Фоторезистор 14, поступове запалювання всіх світлодіодів та одночасне погашення
5.	Натискання кнопки 14, поступове запалювання всіх світлодіодів та поступове погашення
6.	Фоторезистор 14, поступове запалювання всіх світлодіодів та зворотнє погашення
7.	Натискання кнопки 7, поступовий рух одиночного вогника вперед і назад
8.	Джойстик 14, поступовий рух вогника непарних світлодіодів вперед і парних назад
9.	Фоторезистор 7, поступовий рух одиночного вогника вперед
10.	Натискання кнопки 11, поступовий рух двох вогників
11.	Фоторезистор 11, поступовий рух 7 вогників
12.	Натискання кнопки 14, поступове запалювання всіх світлодіодів та одночасне погашення
13.	Фоторезистор 14, поступове запалювання всіх світлодіодів та поступове погашення
14.	Натискання кнопки 14, поступове запалювання всіх світлодіодів та зворотнє погашення
15.	Джойстик 7, поступовий рух одиночного вогника вперед і назад
16.	ВАРІАНТ НА ВИБІР (Комбінація влсаних режимів)