

Міністерство освіти і науки України
Національний університет водного господарства та
природокористування
Кафедра будівельних, дорожніх, меліоративних,
сільськогосподарських машин і обладнання

02-01-542М

МЕТОДИЧНІ ВКАЗІВКИ

до практичних робіт
з освітнього компоненту

«Інновації в АПК»

для здобувачів вищої освіти другого (магістерського) рівня
за освітньо-професійною програмою «Агроінженерія»

спеціальності

208 «Агроінженерія»

денної форми навчання

Рекомендовано науково-
методичною радою з якості
ННМІ
Протокол № 4 від 27.12.2022 р.

Рівне – 2023

Методичні вказівки до практичних робіт з освітнього компоненту «Інновації в АПК» для здобувачів вищої освіти другого (магістерського) рівня за освітньо-професійною програмою «Агроінженерія» спеціальності 208 «Агроінженерія» денної форми навчання / Бундза О.З. – Рівне : НУВГП, 2023. – 39 с.

Укладач: Бундза О. З., к.т.н., доцент, доцент кафедри будівельних, дорожніх, меліоративних, сільськогосподарських машин і обладнання.

Відповідальний за випуск: Налобіна О. О., д.т.н., професор, в.о. завідувача кафедри будівельних, дорожніх, меліоративних, сільськогосподарських машин і обладнання.

Керівник групи забезпечення спеціальності 208 «Агроінженерія»: Налобіна Олена Олександрівна, д.т.н., професор, професор кафедри будівельних, дорожніх, меліоративних, сільськогосподарських машин і обладнання

© О. З. Бундза, 2023

© НУВГП, 2023

Зміст

Вступ	4
Практична робота 1. Керування світлодіодними індикаторами за допомогою платформи на базі контролера AtMega328P	4
Практична робота 2. Керування силовим навантаженням за допомогою платформи на базі контролера AtMega328P	13
Практична робота 3. Звукова індикація подій зумером за допомогою платформи на базі контролера AtMega328P	16
Практична робота 4. Програмне керування яскравістю світлодіодних індикаторів з застосуванням платформи на базі контролера AtMega328P	20
Практична робота 5. Контроль освітленості за допомогою за допомогою платформи на базі контролера AtMega328P	25
Практична робота 6. Вимірювання відстані за допомогою платформи на базі контролера AtMega328P	28
Практична робота 7. Керування електродвигуном за допомогою ШИМ та платформи на базі контролера AtMega328P	33
Практична робота 8. Керування пристроями за допомогою IR-приймача та платформи на базі контролера AtMega328P	36
Список рекомендованої літератури	39

Вступ

Практичні роботи з освітньої компоненти «Інновації в АПК» дають змогу на практиці ознайомитись з принципами роботи окремих компонентів систем автоматизованого керування окремими системами сільськогосподарської техніки, ознайомитись з технічними засобами віддаленого керування виконавчими механізмами у сільськогосподарській техніці та обладнанні в АПК.

У практичних роботах розглянуто:

- основні принципи керування виконавчими механізмами;
- способи віддаленого керування виконавчими механізмами у сільськогосподарській техніці та обладнанні в АПК;
- способи і засоби виведення інформації про параметри механізмів у сільськогосподарській техніці та обладнанні в АПК;
- графічне виведення інформації про параметри механізмів у сільськогосподарській техніці та обладнанні в АПК.

Практична робота 1. Керування світлодіодними індикаторами за допомогою платформи на базі контролера AtMega328P

Мета роботи: ознайомитися з апаратною та програмною частинами платформи Arduino на базі контролера AtMega328P. Розглянути основи програмного керування світлодіодними індикаторами.

Теоретичні відомості

Сучасні мехатронні системи неможливі без мікропроцесорної складової, в кожній наявний один або декілька мікроконтролерів або мікропроцесорів. Наймасовіше використовують саме мікроконтролери, які містять в одній мікросхемі й обчислювальне ядро, й пам'ять і периферійні модулі вводу-виводу. Вони виконують роль центрального цифрового керуючого елемента, збираючи дані від датчиків, обмінюючись даними з іншими вузлами бортової мережі, формуючи сигнали керування приводами.

Створення компонентів техніки з мікроконтролерами вимагає як розробки апаратної частини (друкованої плати з розпаяним мікроконтролером й іншими компонентами, до якої підключаються датчики, драйвери приводів, інтерфейсні кабелі, засоби індикації тощо), так і програмного забезпечення («прошивки» мікроконтролера). Пришвидшити процес розробки прототипу дозволяє використання налагоджувальних плат, що включають мікроконтролер і потрібні для його роботи компоненти, а також роз'єми для підключення всього іншого.

Arduino – це відкрита електронна платформа для швидкого прототипування (створення прототипів) електроніки, що базується на простому у використанні апаратному та програмному забезпеченні.

Апаратна частина включає мікроконтролерну налагоджувальну плату, а програмна – середовище Arduino IDE. Платформа Arduino є відкритою, тому існує ряд аналогів Arduino-сумісних мікроконтролерних плат різного виконання.

Arduino Uno

Arduino Uno – це плата на основі мікроконтролера ATmega328P. Великий робочий діапазон температур (-40...+125 °C) дозволяє використовувати цей мікроконтролер у рухомій техніці. Плата Arduino Uno має 14 цифрових входів/виходів (з них 6 можуть використовуватися в якості ШІМ-виходів), 6 аналогових входів, роз'єм USB, роз'єм живлення, роз'єм для внутрішньосхемного програмування (ICSP) і кнопка скидання.

Плата може живитись від джерела постійного струму напругою 7...17 В через пін V_{in} , через роз'єм External Power Supply (постійна напруга 7...17 В) або за допомогою USB-кабелю. У останньому випадку максимальний струм на цифрових виходах буде дещо меншим.

Входи і виходи

Кожен з 14 цифрових виводів може працювати в якості входу або виходу. Рівень напруги на виводах обмежений 5В. Максимально допустимий (критичний) струм, який може віддавати або споживати один вивід, становить 40 мА, проте рекомендується у тривалому режимі не перевищувати 20-25 мА. Усі виводи з'єднані з внутрішніми підтягуючими резисторами (за

замовчуванням відключеними) опором 20-50 кОм. Крім цього, деякі виводи Arduino Uno можуть виконувати додаткові функції:

Послідовний інтерфейс: виводи 0 (RX) і 1 (TX) (рис. 1.1). Використовуються для отримання (RX) і передачі (TX) даних по послідовному інтерфейсу. Ці виводи з'єднані з відповідними виводами мікроконтролера ATmega8U2 або ATmega16U2, що виконує роль перетворювача USB/UART.

У платах інших виробників можна зустріти замість нього USB/UART-перетворювачі CH340 або CP2102.

Зовнішні переривання: виводи 2 і 3. Можуть служити джерелами переривань, що виникають при фронті, спаді або при низькому рівні сигналу на цих виводах.

ШИМ: виводи 3, 5, 6, 9, 10 і 11. За допомогою функції analogWrite() можуть виводити 8-бітові псевдоаналогові значення у вигляді ШИМ-сигналу. Це дозволяє, наприклад, реалізувати керування сервоприводами, двигунами постійного струму та світлодіодами.

Інтерфейс SPI: виводи 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Із застосуванням бібліотеки SPI дані виводи можуть здійснювати зв'язок по інтерфейсу SPI.

Світлодіод: 13. Вбудований світлодіод на платі, приєднаний до виводу 13.

Аналогові входи A0... A5 (таблиця 1.1) за допомогою мультиплектора приєднані до 10-розрядного аналого-цифрового перетворювача. На них можна подати аналоговий сигнал напругою 0...5 В та отримати аналого-цифрове перетворення вхідного сигналу у вигляді 10-бітного числа (1024 різних значення). Саме завдяки наявності вбудованого аналого-цифрового перетворювача плата може без додаткових елементів працювати з багатьма різними датчиками.

Таблиця 1.1
Основні характеристики Arduino Uno

Мікроконтролер	АТmega328P
Робоча напруга	5V
Вхідна напруга (рекомендована)	7-12V
Дискретні (цифрові) входи/виходи	14 (6 із них можуть використовуватися як ШІМ-виходи)
Аналогові входи	6
Максимальний струм одного вивода	40 mA
Максимальний струм вивода 3,3В	50 mA
Флеш-пам'ять	32 KB (АТmega328P) з яких 0.5 KB використовуються завантажувачем
SRAM	2 KB (АТmega328P)
EEPROM	1 KB (АТmega328P)
Тактова частота	16 MHz

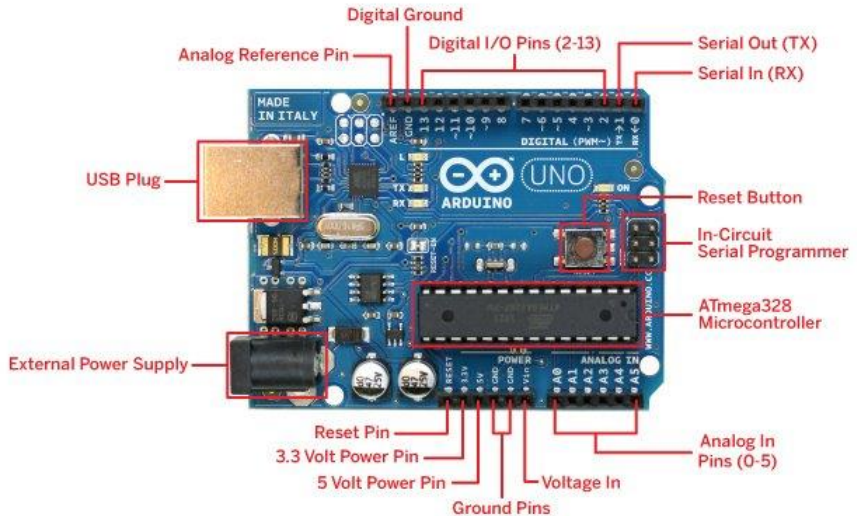


Рис. 1.1 Розпіновка плати Arduino Uno

Програмування Arduino

Програмування мікроконтролерних плат Arduino здійснюється у вільно розповсюджваному середовищі Arduino IDE.

Мікроконтролери в платах Arduino випускаються з прошитим завантажувачем (bootloader), що дозволяє завантажувати в мікроконтролер нові програми без необхідності використання зовнішнього програматора. Взаємодія з ним здійснюється за оригінальним протоколом STK500.

Також мікроконтролер можна прошити і через роз'єм для внутрішньосхемного програмування ICSP (In-Circuit Serial Programming), не звертаючи уваги на завантажувач.

Мова програмування Arduino – це видозмінена мова C/C++ із розширеним набором функцій.

Кожна програма для пристроїв Arduino містить дві основні функції:

- 1) void setup() – функція, що виконується о дин раз, після кожної подачі живлення або скидання плати Arduino;
- 2) void loop() – виконується постійно в циклі після виконання функції void setup().

Перед завантаженням програми в мікроконтролер слід спочатку вибрати тип плати Arduino та порт, до якого вона підключена.

Деякі функції для роботи з входам/виходами Arduino

pinMode(pin, value) – налаштування виводу з номером pin як дискретного входу (value=INPUT) або дискретного виходу (value=OUTPUT). Наприклад:

```
pinMode(13, OUTPUT);
```

digitalRead(pin) – функція зчитує із заданого входу значення HIGH або LOW.

digitalWrite(pin, value) – подає на цифровий вхід/вихід значення HIGH або LOW;

analogRead(pin) – зчитує величину напруги з аналогового входу і видає результат у вигляді цілого числа від 0 до 1023.

Світлодіод (LED) – це напівпровідниковий електронний компонент, призначений для світлової індикації подій, освітлення.

Будова світлового діода наведена на рис. 1.2.

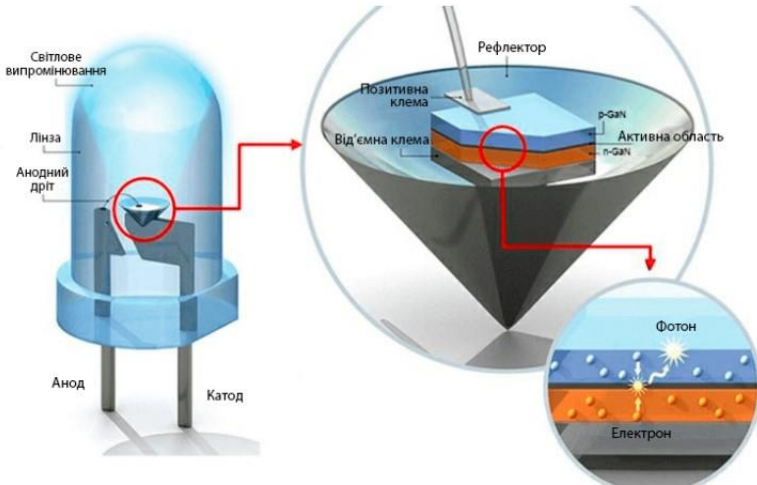


Рис. 1.2 Будова світлового діода

Будова і принцип роботи світлового діода добре висвітлені в [4]. Світлодіод може містити і більше одного кристалу, наприклад – три, що світять трьома основними кольорами. Такі світлодіоди називаються RGB-світлодіодами, можуть легко відтворювати різні кольори і мають ширші можливості для індикації подій.

План роботи

1. Ознайомитися з будовою та призначенням елементів плати Arduino Uno.
2. Ознайомитися з принципами програмування в середовищі Arduino IDE.
3. Навчитися програмно керувати різними світлодіодними індикаторами.

Порядок виконання роботи

1. Ознайомитися з теоретичними відомостями.
 2. Завантажити середовище Arduino IDE.
 3. Ознайомитись зі схемою підключення світлодіода (рис. 1.3.).
- Зібрати її на макетній платі. Використати

струмообмежувачий резистор на 220 Ом.

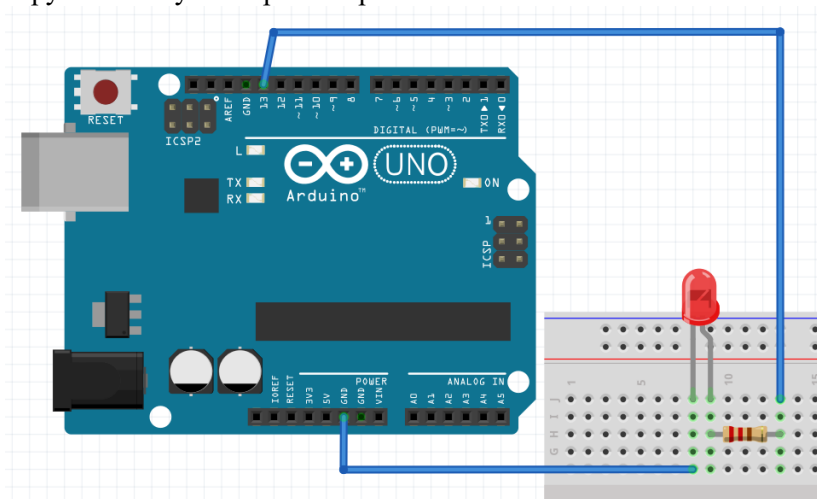


Рис. 1.3. Підключення світлодіода до Arduino Uno

У середовищі Arduino IDE написати програму, що запалюватиме світлодіод з частотою 1 Гц.

4. Скомпілювати програму та переконатися у відсутності помилок.

5. Підключити плату до USB-порту комп'ютера. У середовищі Arduino IDE вибрати тип плати та порт, до якого вона підключена. Завантажити створену програму в мікроконтролер. Переконатися в коректності роботи програми.

6. Ознайомитися зі схемою підключення світлодіода RGB до плати Arduino (рис. 1.4). Зібрати її на макетній платі. Використати резистори 220 Ом.

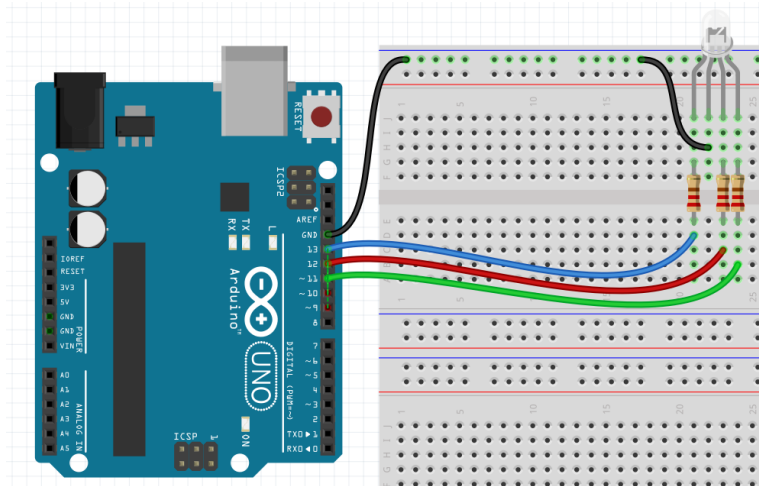


Рис. 1.4. Схема підключення світлодіода RGB до плати Arduino

7. Написати програму, що послідовно запалюватиме RGB-світлодіод різними кольорами.

Adafruit Arduino - RGB світлодіод

```
int redPin = 11;
int greenPin = 10;
int bluePin = 9;
void setup()
{
  pinMode(redPin, OUTPUT);
  pinMode(greenPin, OUTPUT);
  pinMode(bluePin, OUTPUT);
}
void loop()
{
  setColor(255, 0, 0); // червоний
  delay(1000);
  setColor(0, 255, 0); // зелений
```

```

delay(1000);
setColor(0, 0, 255); // синій
delay(1000);
setColor(255, 255, 0); // жовтий
delay(1000);
setColor(80, 0, 80); // фіолетовий
delay(1000);
setColor(0, 255, 255); // аквамарин
delay(1000);
}
void setColor(int red, int green, int blue)
{
#ifdef COMMON_ANODE
red = 255 - red;
green = 255 - green;
blue = 255 - blue;
#endif
analogWrite(redPin, red);
analogWrite(greenPin, green);
analogWrite(bluePin, blue);
}

```

Завантажити програму в мікроконтролер.

Поекспериментувати з часом світіння кожного кольору, змінюючи значення змінної delay.

9. Зробити висновки. Звіт повинен містити: титульний лист; тему, мету роботи; порядок виконання; створені програми; висновки.

Контрольні запитання

1. Яка функція налаштовує режим роботи піна на вхід?
2. Яка функція дозволяє зчитати стан цифрового входу?
3. Яка функція дозволяє змінити стан цифрового виходу?

Практична робота 2. Керування силовим навантаженням за допомогою платформи на базі контролера AtMega328P

Мета роботи: навчитись керувати силовим навантаженням за допомогою платформи на базі контролера AtMega328P.

Теоретичні відомості

Реле дозволяє керувати потужним навантаженням за допомогою мікроконтролера. Також забезпечує гальванічний захист.

Підключити на пряму до Arduino потужне навантаження, наприклад лампу освітлення або електронасос не вийде. Струм, який може протікати через виходи ATmega328, номінальний 20 мА максимальний 40 мА. Комутовати потужне навантаження можна за допомогою реле. До того ж, якщо навантаження живиться від змінного струму, наприклад 220 Вольт. Для підключення потужних навантажень до Arduino зазвичай використовують модулі реле. Один з таких модулів наведений на рис. 2.1.



Рис. 2.1. Модуль реле

План роботи

1. Навчитися підключати силове навантаження до контролера за допомогою модуля реле.

2. Навчитися керувати роботою модуля реле за допомогою команд по COM-порту.

Порядок виконання роботи

1. Підключити до плати Arduino модуль реле (рис. 2.2).

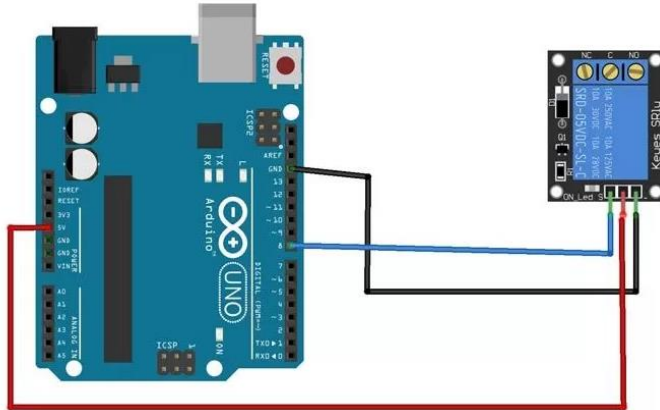


Рис. 2.2. Схема підключення модуля реле

2. До силових контактів модуля реле підключити довільне низьковольтне навантаження, наприклад, світлодіод через обмежувачий резистор 220 Ом.

3. Завантажити в мікроконтролер наступну програму:

```
void setup() {  
    // ініціалізуємо пін 9 як вихід.  
    pinMode(9, OUTPUT);  
}  
void loop() {  
    digitalWrite(8, HIGH); // вмикаємо сигнал  
високого рівня на пін 8  
    delay(5000); // чекаємо визначений час  
    digitalWrite(8, LOW); // вмикаємо сигнал  
низького рівня на пін 8  
    delay(5000); // чекаємо визначений час  
}
```

4. Поекспериментувати з тривалістю кожного стану модуля реле, змінюючи значення змінної `delay`.

5. Завантажити в мікроконтролер наступну програму:

```
void setup()
{
  Serial.begin(9600);
  pinMode(8, OUTPUT);
}
void loop()
{
  // отримуємо команди вмикання навантаження на
  // порту.
  String str = Serial.readString();
  // Якщо команда «1» то включаємо навантаження.
  if (str == «1») digitalWrite(8, HIGH);
  // Якщо команда «2» то виключаємо
  навантаження.
  if (str == «2») digitalWrite(8, LOW);
  // Чекаем 1 секунду.
  delay(1000);
}
```

6. Відкрити монітор послідовного порту. Ввести число 1. Спостерігати за зміною стану модуля реле. Ввести число 2. Спостерігати за зміною стану модуля реле.

7. Оформити звіт про виконання роботи. Звіт повинен містити: назву та мету роботи; тексти програм з коментарями; висновок про виконання роботи.

Контрольні запитання

1. Яке призначення модуля реле?
2. Чи дозволяє модуль реле комутувати прилади, щ оживляться від мережі 220 В?
3. Яка кількість цифрових пінів в Arduino Uno?
4. Який тип пінів використовується для підключення модуля реле?
5. Якою функцією можна отримати результат АЦП?

Практична робота 3. Звукова індикація подій зумером за допомогою платформи на базі контролера AtMega328P

Мета роботи: навчитися використовувати пасивний зумер для звукової індикації подій.

Теоретичні відомості

Найбільш поширеними і водночас простими засобами звукової індикації є п'єзоелектричні випромінювачі - зумери.

Їх широко застосовують у різних електронних пристроях, де потрібна звукова індикація. Їх ділять на пасивні та активні зумери.

Пасивний зумер містить лише п'єзоелектричний випромінювач, тому для звукової індикації на нього треба подавати сигнал звукової частоти.

Активний зумер містить крім п'єзоелектричного випромінювача ще й звуковий генератор, налаштований на певну частоту, тому для звукової індикації на нього достатньо просто подати напругу +5 В.

Технологічне виконання зумера показано на рис. 3.1, умовне позначення – на рис. 3.2.



Рис. 3.1. Технологічне виконання зумера



Рис. 3.2. Умовне позначення зумера

План роботи

1. Ознайомитися з принципом дії пасивного зумера.
2. Згенерувати з допомогою зумера звукові сигнали різної частоти.

Порядок виконання роботи

1. Ознайомитися з теоретичними відомостями.
 2. На монтажній платі скласти схему відповідно до рис. 3.3.
- 3.3.

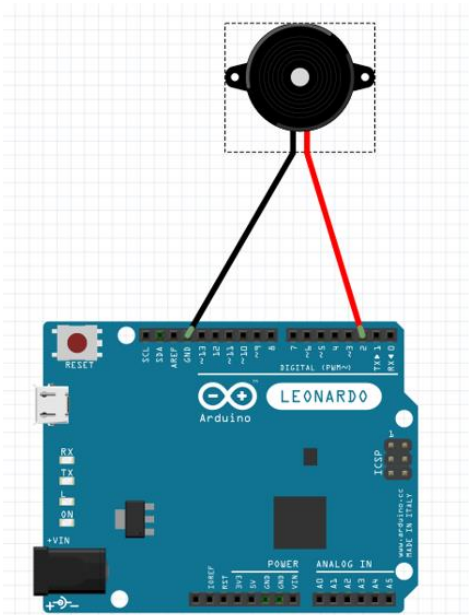


Рис. 3.3. Схема підключення зумера до плати Arduino

3. У середовищі Arduino IDE створити новий скетч (виводить почергово сигнал високого і низького рівня на цифровий вихід №2).

```
int buzzPin = 2;
void setup() {
    pinMode(buzzPin, OUTPUT); // установка
вивода 2 в режим виходу
}
void loop() {
    digitalWrite(buzzPin, HIGH); // перевод
вивода 2 в активний стан
    delay(500); // пауза півсекунди
    digitalWrite(buzzPin, LOW); // перевод
вивода 2 в неактивний стан
    delay(500); // пауза півсекунди
}
```

4. Скомпілювати програму та переконатися у відсутності помилок.

5. Підключити мікроконтролерну плату до USB-порта комп'ютера.

6. У середовищі Arduino IDE вибрати тип плати та порт, до якого вона підключена. Завантажити створену програму в мікроконтролер. Відмітити результати роботи.

7. У середовищі Arduino IDE створити новий скетч (виводить кожних 0,5 секунди звуковий сигнал частотою 100 Гц на цифровий вихід №2).

```
const byte piezoPin = 2;
void setup() {
    pinMode(piezoPin, OUTPUT); // налаштуємо вивод 2
на вихід
}
void loop() {
    tone(piezoPin, 100); // генеруємо звук з частотою 100
Гц
    delay(100); // пауза 100 мілісекунд
    noTone(piezoPin); // вимикаємо звук
    delay(900); // знову пауза 900 мс
```

}

Скомпілювати програму та переконатися у відсутності помилок.

8. Завантажити створену програму в мікроконтролер. Відмітити результати роботи.

9. На монтажній платі скласти схему з активним зумером відповідно до рис. 3.4.

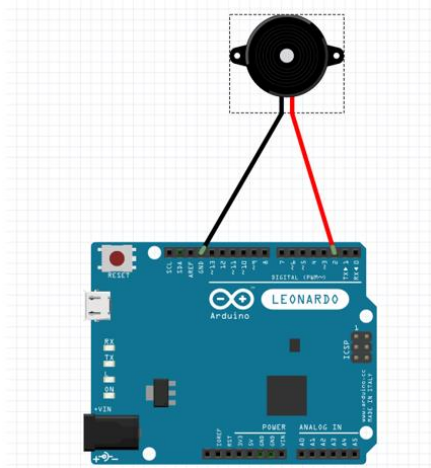


Рис. 3.4. Схема для вивчення роботи активного зумера

10. У середовищі Arduino IDE створити новий скетч (стандартний скетч, що виводить кожну 1 секунду сигнал високого рівня на цифровий вихід №13. При цьому потрібно червоний провід від зумера переключити на цифровий вихід №12).

```
// the setup function runs once when you press reset or power the board
```

```
void setup() {  
  // initialize digital pin LED_BUILTIN as an output.  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on
```

(HIGH is the voltage level)

```
delay(1000); // wait for a second
```

```
digitalWrite(LED_BUILTIN, LOW); // turn the LED off
```

by making the voltage LOW

```
delay(1000); // wait for a second
```

11. Завантажити створену програму в мікроконтролер. Відмітити результати роботи.

12. Зробити висновки. Звіт повинен містити: титульний лист; тему, мету роботи; порядок виконання; створені програми; висновки.

Контрольні запитання

1. Який принцип роботи активного зумера?
2. Який принцип роботи пасивного зумера?
3. Для чого потрібний звуковий генератор у активному зумері?
4. Який тип пінів використовується для підключення пасивного зумера?

Практична робота 4. Програмне керування яскравістю світлодіодних індикаторів з застосуванням платформи на базі контролера AtMega328P

Мета роботи: ознайомитися з способом програмного керування яскравістю світлодіодних індикаторів.

Теоретичні відомості

У машинах чи обладнанні, що використовуються в АПК, бувають випадки, коли потрібно не просто увімкнути чи вимкнути певний пристрій, а регулювати потужність на ньому, наприклад: з метою регулювання яскравості світлодіоду. У таких випадках застосовують метод широтно-імпульсної модуляції (ШІМ — англ. pulse-width modulation, PWM).

Основна ідея PWM полягає в тому, щоб швидко перемикає вихідний сигнал між високим та низьким станами на певній частоті, змінюючи не частоту, а відношення тривалості високого стану до загальної тривалості циклу, яке називають скважністю. Принцип роботи PWM неважко зрозуміти, розглянувши діаграму, показану на рис. 4.1.

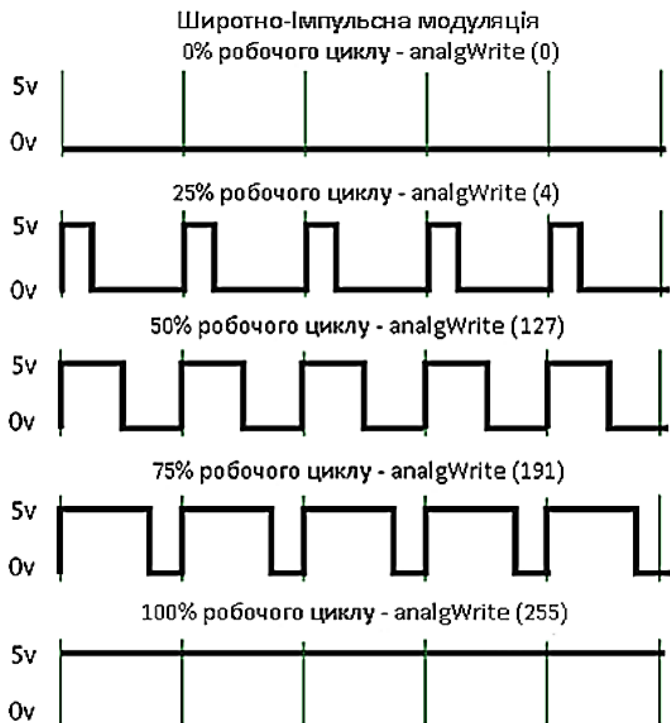


Рис. 4.1. Широтно-імпульсна модуляція (PWM)

Для дослідження принципу роботи PWM використаємо платформу Arduino. Ця платформа підтримує широкий спектр функцій, включаючи генерацію PWM-сигналів.

На Arduino для генерації PWM-сигналу використовуються піни, які позначені позначкою "PWM", наприклад, 3, 5, 6, 9, 10, 11 на більшості моделей. Для генерації PWM на Arduino ви можете використовувати функцію `analogWrite(pin, value)`, де `pin` - це номер піни з підтримкою PWM, а `value` - значення ширини імпульсу в діапазоні від 0 до 255. Значення 0 відповідає повному вимкненню виходу, а 255 - повному включенню.

Наприклад, якщо ви хочете згенерувати PWM-сигнал на піні 9 з половиною тривалості високого стану, то код може виглядати так:

```

int pwmPin = 9;

void setup() {
  // Налаштування піни як вихідного
  pinMode(pwmPin, OUTPUT);
}

void loop() {
  int pwmValue = 128; // Значення PWM від 0 до 255
  analogWrite(pwmPin, pwmValue); // Генерація PWM
}

```

Цей приклад встановлює вихідний пін 9 на половину свого максимального високого стану, що відповідає приблизно 50% яскравості для підключеного до нього пристрою, у нашому випадку - світлодіоду.

Отже, з допомогою PWM можна керувати, наприклад, яскравістю світлодіоду.

План роботи

1. Ознайомитися з принципом роботи PWM.
2. Реалізувати програмне керування яскравістю світлодіоду.
3. Реалізувати ручне керування яскравістю світлодіоду з використанням змінного резистору.

Порядок виконання роботи

1. Ознайомитися з теоретичними відомостями.
2. На монтажній платі скласти схему відповідно до рис. 4.2.

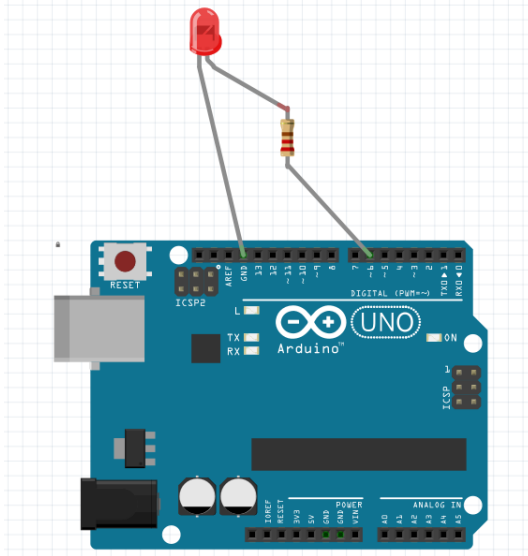


Рис. 4.2. Схема підключення світлодіоду

3. У середовищі Arduino IDE створити новий скетч (скетч, що виводить почергово сигнали різної скважності на цифровий вихід PWM №6).

```
#define LED_PIN 6
void setup()
{
  pinMode(LED_PIN, OUTPUT);
}
void loop()
{
  analogWrite(LED_PIN, 55);
  delay(500);

  analogWrite(LED_PIN, 130);
  delay(500);
  analogWrite(LED_PIN, 255);
  delay(500);
}
```

4. Завантажити створений скетч в мікроконтролер. Відмітити результати роботи.
5. На монтажній платі скласти схему відповідно до рис. 4.3.

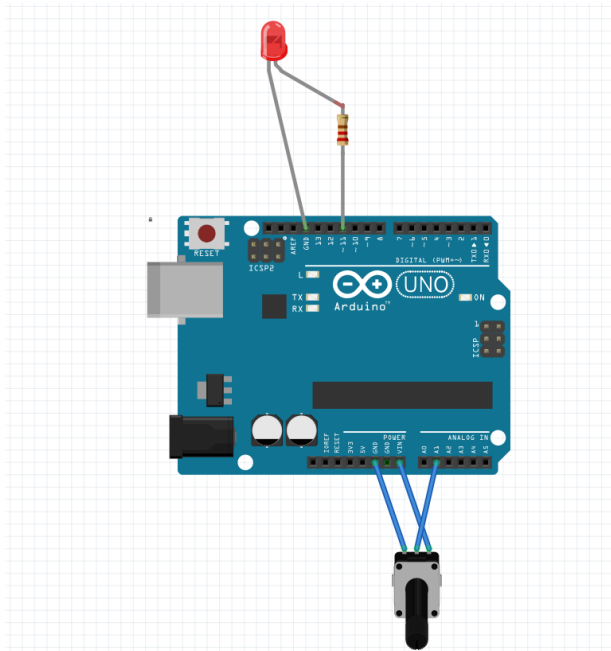


Рис. 4.3. Схема підключення для ручного регулювання яскравості світлодіоду

6. У середовищі Arduino IDE створити новий скетч (скетч, що змінює скважність на цифровому виході PWM №11 залежно від кута повороту ручки потенціометра (змінного резистора). При цьому потрібно провід від обмежуючого резистора переключити на цифровий вихід PWM №11 так, як показано на схемі).

```
void setup() {  
  pinMode(11, OUTPUT); //підключення світлодіоду  
  pinMode(A1, INPUT); // до входу A1 підключаємо  
  потенціометр  
}  
void loop() {
```



```
int val1 = analogRead(A1);
int val2 = val1 / 4;
analogWrite(11, val2);
}
```

7. Завантажити створений скетч в мікроконтролер. Відмітити результати роботи.

8. Оформити звіт про виконання роботи. Звіт повинен містити: назву та мету роботи; тексти програм з коментарями; висновок про виконання роботи.

Контрольні запитання

1. Що таке широтно-імпульсна модуляція (PWM)?
2. Що називають скважністю?
3. Яка змінного резистора на рис.4.3?
4. Який принцип роботи PWM?
5. Який пристрій використовують для генерації PWM-сигналу?

Практична робота 5. Контроль освітленості за допомогою за допомогою платформи на базі контролера AtMega328P

Мета роботи: Ознайомитися з принципом роботи і застосуванням датчика освітленості GL5516.

Теоретичні відомості

В сучасних машини і обладнанні в АПК часто виникає потреба забезпечити реакцію того чи іншого механізму на зміну освітленості. Для цього у склад відповідних систем автоматизації машини включають датчики освітленості.

Датчик освітленості GL5516, що його будемо використовувати в роботі, являє собою фоторезистор.

Фоторезистор – це напівпровідникова радіодеталь, яка змінює значення свого опору в залежності від світлового потоку, який потрапляє на чутливу зону. Чутлива зона датчика освітленості GL5516 виконана на основі сульфїду кадмію CdS. Фоторезистор не має p-n переходу, тому має однакову провідність незалежно від напрямку протікання струму. Загальний вигляд датчика освітленості GL5516 показано на рис. 5.1.



Рис. 5.1. Датчик освітленості GL5516

Характеристики датчика освітленості GL5516:

фоторезистор **чутливий до:** видимого світла;
максимальна напруга: 150 В;
максимальна потужність: 90 мВт;
опір при світлі (10 люкс): 5 - 10 кОм;
опір при відсутності світла: 500 кОм;
спектральний пік: 540 нм;
діаметр головки: 5 мм;
довжина ніжок: 32 мм.

Вивід результатів вимірювання освітленості реалізуємо на LCD-дисплей 1602A, показаний на рис. 5.2.



Рис. 5.2. LCD-дисплей 1602A

Цей LCD-дисплей призначений для виведення тексту. Дисплей можна використовувати для відображення параметрів вологості, температури, напруги, сили струму і т.д.

LCD-дисплей 1602A спроектований на базі контролера Hitachi HD44780. Контролер має однобайтні комірки пам'яті, вміст яких відображається на екрані. Даний LCD дисплей відображає 2 рядки по 16 символів. Роздільна здатність символів цього дисплею становить 5 x 8 пікселів.

План роботи

1. Ознайомитися з призначенням, будовою та принципами застосування датчика освітленості GL5516 .
2. Ознайомитися з схемою вимірювання освітленості.
3. Вивести поточні значення освітленості на LCD-дисплей 1602A.

Порядок виконання роботи

1. Ознайомитися з теоретичними відомостями.
2. На монтажній платі скласти схему відповідно до рис. 5.3.

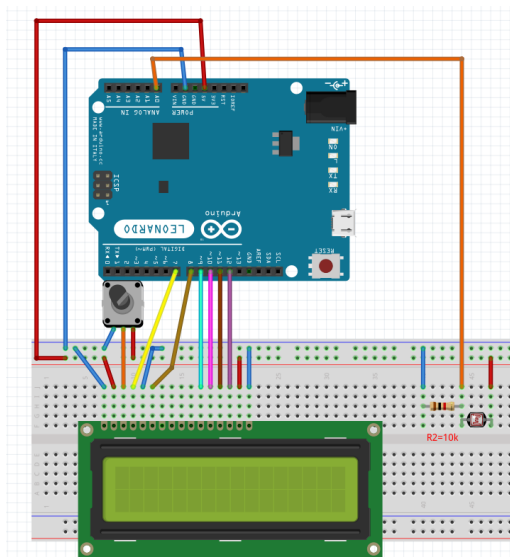


Рис. 5.3. Схема вимірювання освітленості

3. Створити новий скетч.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(7,8,9,10,11,12); //
```

призначаємо піни для роботи з дисплеєм

```
void setup() {  
  lcd.begin(16, 2);  
  lcd.setCursor(0,1); // встановлюємо курсор на  
позицію виводу тексту  
  lcd.write("LIGHT: "); // виводимо текст  
(освітленість (англ.))  
}  
void loop(){  
  int sensorValue = analogRead(A0); // зчитуємо  
аналоговий сигнал на піні A0  
  lcd.setCursor(7,1); // встановлюємо курсор на  
позицію виводу цифр  
  lcd.print(sensorValue); // виводимо значення  
(цифри)  
  delay(100);  
}
```

4. Завантажити створений скетч в мікроконтролер.

Відмітити результати роботи.

Контрольні запитання

1. Який принцип роботи датчика освітленості GL5516?
2. Яке призначення LCD-дисплею 1602A?
3. Яка сфера застосування датчика освітленості?
4. Що собою являє датчик освітленості GL5516?
5. До яких типів пінів підключають датчик освітленості

GL5516?

Практична робота 6. Вимірювання відстані за допомогою платформи на базі контролера AtMega328P

Мета роботи: розглянути принцип вимірювання відстані за допомогою ультразвукового датчика.

Теоретичні відомості

Для вимірювання відстані від певної частини вузла чи агрегату машини до інших об'єктів широко застосовують ультразвукові датчики відстані.

Принцип роботи ультразвукового датчика відстані заснований на явищі ехолокації, яке полягає у визначенні часу, за який згенерований датчиком ультразвуковий сигнал, відбитий від перешкоди, повернеться назад. Знаючи швидкість поширення звуку і час, можна легко знайти відстань до об'єкту.

У лабораторній роботі використовується ультразвуковий датчик відстані HC-SR04 (рис. 6.1).



Рис. 6.1. Ультразвуковий датчик відстані HC-SR04

Він включає в себе ультразвуковий передавач, приймач і вузол контролю. На передній частині HC-SR04 розташовано два ультразвукових датчики, перший з написом T (Transmitter) - це передавач ультразвукових хвиль (ТСТ40-16Т), а другий з написом R (Receive) - це приймач відбитих ультразвукових хвиль (ТСТ40-16R), по центру розташований кварцевий генератор на 27 МГц.

На покази датчика практично не впливають сонячне випромінювання і електромагнітні шуми.

Принцип роботи датчика наступний:

1. На вихід trig (тригер) посилаємо високий рівень протягом як мінімум 10мкс.

2. Після цього модуль посилає пакет ультразвукових імпульсів з частотою 40 кГц і приймає їх назад, якщо в зоні видимості є будь-які перешкоди.

3. Якщо сигнал повертається, модуль встановлює низький рівень на виході echo на 150мс. За часом, який минув з п.1 до низького рівня на виході echo можна розрахувати відстань до перешкоди за формулою:

$$\text{відстань} = (\text{time} * \text{sound velocity}) / 2,$$

де time - виміряне час імпульсу, sound velocity - швидкість звуку (340 м / с).

План роботи

1. Ознайомитися з принципом роботи ультразвукового датчика відстані HC-SR04.

2. Скомпонувати схему для вимірювання відстані на базі ультразвукового датчика з виводом результатів на LCD-дисплей 1602A та перевірити її роботу.

Порядок виконання роботи

1. Ознайомитися з теоретичними відомостями та принципом роботи ультразвукового датчика відстані HC-SR04.

2. На монтажній платі скласти схему відповідно до рис. 6.2.

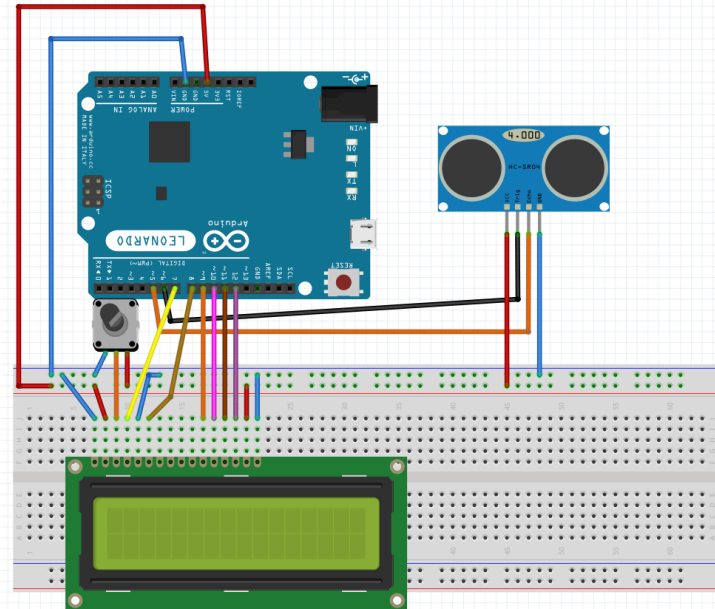


Рис. 6.2. Схема вимірювання відстані на базі ультразвукового датчика

3. Створити новий скетч.

```
#include <LiquidCrystal.h> // підключення LCD
бібліотеки
LiquidCrystal lcd (7,8,9,10,11,12); //
ініціалізація бібліотеки із зазначенням пінів підключення до
Arduino
int inputPin=5;
int outputPin=6;
void setup()
{
  lcd.begin(16, 2); // встановлення кількості символів
і рядків дисплея
  pinMode(inputPin, INPUT); // установка режимів
  pinMode(outputPin, OUTPUT); // роботи виводів
Arduino
  lcd.setCursor(0, 0); // установка курсору
```

```

    lcd.print("Distance:"); // вивод тексту на
    дисплей
}
void loop()
{
    // далі надсилання необхідних імпульсів на УЗ датчик
    digitalWrite(outputPin, LOW);
    delayMicroseconds(2);
    digitalWrite(outputPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(outputPin, LOW);
    int distance = pulseIn(inputPin, HIGH); //
    отримання даних з УЗ датчика
    distance=distance/58; // переведення значення в
    сантиметри
    lcd.setCursor(9, 0); // установка курсору
    lcd.print("  "); // виведення порожніх символів
    для видалення старих показань
    lcd.setCursor(9, 0); // установка курсору
    lcd.print(distance); // виведення значення відстані
    delay(100); // затримка перед наступним виконанням
}

```

4. Завантажити створений скетч в мікроконтролер. Відмітити результати роботи.

Контрольні запитання

1. Яке призначення ультразвукового датчика відстані HC-SR04?
2. Який принцип дії ультразвукового датчика відстані?
3. Яке фізичне явище використовується в датчику HC-SR04 для визначення відстані?
4. Яке призначення пінів датчику HC-SR04?
5. Для чого використовується пакет ультразвукових імпульсів?
6. Яке призначення LCD-дисплею 1602A?

Практична робота 7. Керування електродвигуном за допомогою ШИМ та платформи на базі контролера AtMega328P

Мета роботи: розглянути принцип керування електродвигуном постійного струму за допомогою силових ключів та платформи на базі контролера AtMega328P.

Теоретичні відомості

Широтно-імпульсну модуляцію часто застосовують для керування силовим навантаженням, що живиться від джерела постійного струму, як от: електродвигун мехатронного приводу, електродвигун систем вентиляції, потужний освітлювальний прилад, тощо.

У попередніх роботах вже розглядався принцип керування яскравістю LED за допомогою методу PWM. Проте, ми підключали світлодіод безпосередньо до цифрового піну навчальної платформи, так як максимальний струм, споживаний світлодіодом, значно менший за допустимий струм, що дає цифровий піл платформи.

У цій же роботі ми спробуємо покерувати електродвигуном постійного струму кулера діаметром 120 мм, що живиться від джерела постійного струму напругою 12 В та споживає струм до 200 мА, що у п'ять раз перевищує максимальний струм цифрового піну. Тому для керування двигуном у схему потрібно включити силовий елемент – MOSFET-транзистор – який здатний керувати потужними приладами при малих значеннях керуючих сигналів на його вході.

Принципова схема керування електродвигуном за допомогою PWM та MOSFET-транзистора показана на рис. 7.1.

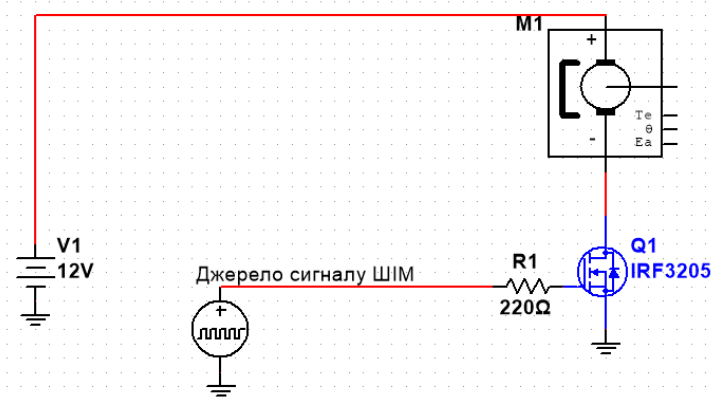


Рис. 7.1. Принципова схема керування електродвигуном за допомогою PWM та MOSFET-транзистора

У якості силового елемента використано потужний n-канальний MOSFET-транзистор IRF3205 з максимальним струмом на вихідному виводі Drain рівним 110 Ампер. Загальний вигляд, умовне позначення на схемах і розпіновка виводів IRF3205 показані на рис. 7.2.

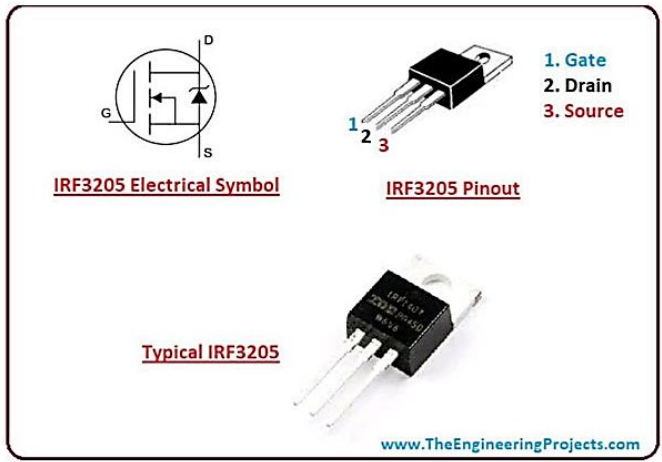


Рис. 7.2. MOSFET-транзистор IRF3205

План роботи

1. Ознайомитися з теоретичними відомостями.
2. Реалізувати програмне керування частотою обертання двигуна вентилятора.

Порядок виконання роботи

1. Ознайомитися з теоретичними відомостями.
2. На монтажній платі скласти схему відповідно до рис. 7.3.

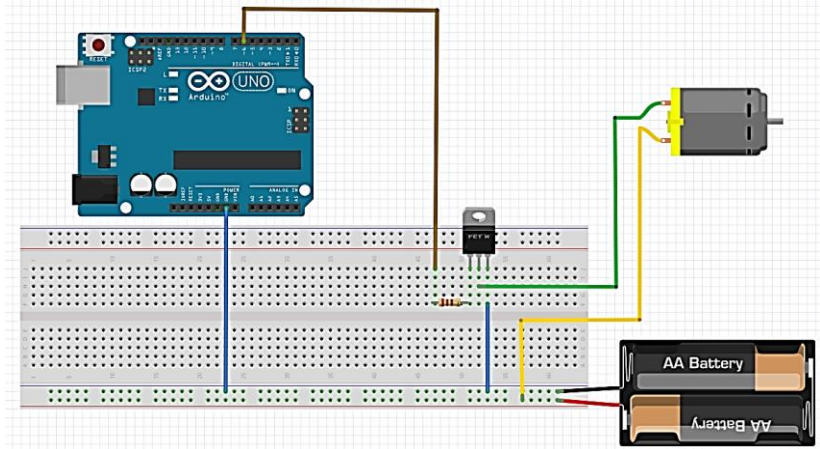


Рис. 7.3. Схема підключення для керування силовим навантаженням з допомогою MOSFET-транзистора IRF3205

3. У середовищі Arduino IDE створити новий скетч (скетч, що виводить по чергові сигнали різної скважності на цифровий вихід PWM №6).

```
#define LED_PIN 6
void setup()
{
  pinMode(LED_PIN, OUTPUT);
}
void loop()
{
  analogWrite(LED_PIN, 25);
  delay(2500);
}
```

```
analogWrite(LED_PIN, 130);  
delay(2500);  
  analogWrite(LED_PIN, 255);  
  delay(2500);  
}
```

4. Завантажити створений скетч в мікроконтролер. Відмітити результати роботи.

Контрольні запитання

1. Яке призначення ультразвукового датчика відстані HC-SR04?
2. Який принцип дії ультразвукового датчика відстані?
3. Яке фізичне явище використовується в датчику HC-SR04 для визначення відстані?
4. Яке призначення MOSFET-транзистора на схемі?
5. Для чого використовується саме цифровий пін PWM?

Практична робота 8. Керування пристроями за допомогою IR-приймача та платформи на базі контролера AtMega328P

Мета роботи: ознайомитися зі способами організації керування пристроями через IR-канал.

Теоретичні відомості

Нерідко для дистанційного керування різноманітними пристроями застосовують інфрачервоне (IR) випромінювання.

У цьому способі команди передаються за допомогою інфрачервоного сигналу від пультів дистанційного керування. Прийом сигналу здійснюється інфрачервоним фотодіодом, але частіше – за допомогою приймача інфрачервоного випромінювання, що являє собою IR-датчик та електронну схему, розміщені в одному корпусі як єдине ціле. Електронна схема дозволяє відсіяти паразитні інфрачервоні випромінювання від інших джерел.

У роботі використаємо IR-приймач VS1838B (рис. 8.1).

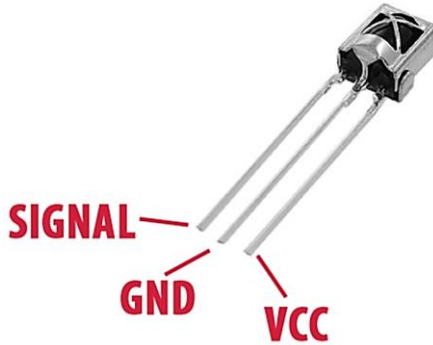


Рис. 8.1. IR-приймач VS1838B

Характеристики його наступні:

1. Робоча частота: 38кГц
2. Відстань прийому сигналу: до 20м
3. Робоча напруга: 2.7-6.0в
4. Споживаний струм: 0.4-1.5мА
5. Кут прийому сигналу: 90⁰

Команди для керування передають з допомогою IR-пульту керування, IR-випромінювач якого при натисканні на певну кнопку генерує певну послідовність імпульсів, яка сприймається IR-приймачем та відфільтровується від перешкод.

План роботи

1. Ознайомитися з теоретичними відомостями.
2. Реалізувати вивід кодів окремих кнопок IR-пульту на монітор COM-порту.

Порядок виконання роботи

1. Ознайомитися з теоретичними відомостями.
2. На монтажній платі скласти схему відповідно до рис. 8.2.

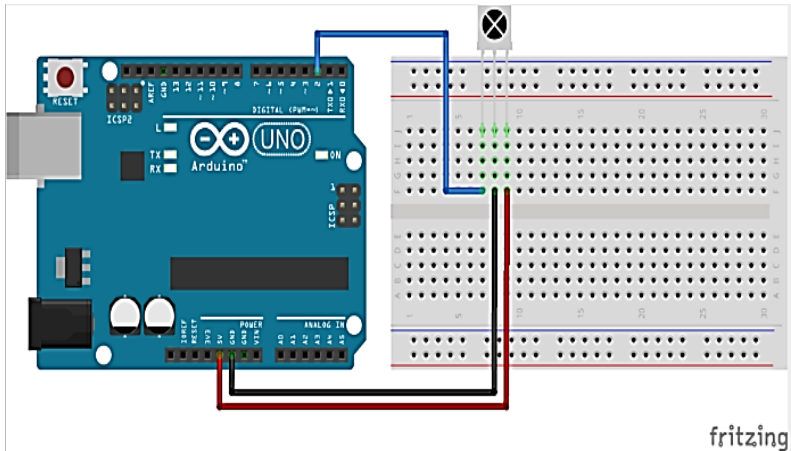


Рис. 8.2. Схема підключення IR-приймача VS1838B

3. У середовищі Arduino IDE створити новий скетч (скетч, що виводить на монітор COM-порту коди кнопок, що поступають від IR-приймача VS1838B на цифровий пін №2).

```
#include <IRremote.h>
int RESIEVE_PIN = 2;
IRrecv irrecv(RESIEVE_PIN);
decode_results res;
void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  irrecv.enableIRIn();
}
```

```
void loop() {
  // put your main code here, to run repeatedly:
  if (irrecv.decode(&res)) {
    Serial.print("0x");
    Serial.println(res.value, HEX);
    delay(50);
    irrecv.resume();
  }
}
```

4. Завантажити створений скетч в мікроконтролер. Зчитати декілька кодів кнопок. Відмітити результати роботи.

Контрольні запитання

1. Що таке IR-приймач?
2. Назвіть основні елементи IR-каналу керування.
3. Яка роль IR-пульту?
4. Скільки виводів має IR-приймач VS1838B, використаний у практичній роботі?
5. Чим відрізняється IR-приймач від IR-фотодіоду?
6. Як реалізоване зчитування кодів кнопок, що використано в цій роботі?
7. Яка бібліотека дає змогу працювати з IR-приймачем?

Список рекомендованої літератури

1. Ловейкін В. С., Ромасевич Ю. О., Крушельницький В.В. Мехатроніка : підручник. К., 2020. 404 с.
2. Margolis Michael. Arduino Cookbook. O'Reilly Media, 2011. 662 p.
3. Evans B. Arduino programming notebook. First edition. 2007. 38 p. URL: https://playground.arduino.cc/uploads/Main/arduino_notebook_v1-1.pdf.
4. Світлодіод – принцип роботи - BitKit: веб-сайт. URL: <https://bitkit.com.ua/svetodiiod> (дата звернення: 26.06.2023).
5. Паламар М. І. Проектування комп'ютеризованих вимірювальних систем і комплексів : навчальний посібник / М. І. Паламар, М. О. Стрембіцький, А. М. Паламар. Тернопіль : ТНТУ, 2018. 150 с.