

Міністерство освіти і науки України

Луцький національний технічний університет
(повне найменування вищого навчального закладу)

Факультет робототехніки та штучного інтелекту
(повне найменування факультету)

Кафедра електроніки, фізики та СМАРТ-систем
(повне найменування кафедри)

**КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «БАКАЛАВР»**

**ПРОЄКТУВАННЯ МІКРОКОНТРОЛЕРНОЇ СИСТЕМИ
ШИРОТНО -ІМПУЛЬСНОГО КЕРУВАННЯ
НАВАНТАЖЕННЯМ НА БАЗІ ARDUINO UNO R3**

**DESIGN OF A MICROCONTROLLER-BASED PULSE-WIDTH
MODULATION LOAD CONTROL SYSTEM BASED ON
ARDUINO UNO R3**

спеціальність 171 Електроніка
(шифр і назва спеціальності)
освітня програма «Електроніка»
(назва освітньої програми)

Виконав: здобувач вищої освіти групи ЕЛ-41
Антосюк Роман Тарасович

(підпис)

Керівник: к.ф.-м.н., доцент
Хвищун Микола Вячеславович

(підпис)

Кваліфікаційну роботу
допущено до захисту
«__» червня 2026 р.
Гарант освітньої програми:
к.т.н., доцент
Мороз Сергій Анатолійович

(підпис)

Луцьк – 2026 року

Факультет комп'ютерних та інформаційних технологій
Кафедра електроніки та телекомунікацій
Ступінь вищої освіти бакалавр
Галузь знань: 17 «Електроніка, автоматизація та електронні комунікації»
Спеціальність: 171 «Електроніка»
Освітня програма: «Електроніка»

ЗАТВЕРДЖУЮ
Завідувач кафедри

«__» _____ 20__ р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Антосюку Роману Тарасовичу

-
1. Тема кваліфікаційної роботи: *Проектування мікроконтролерної системи широтно-імпульсного керування навантаженням на базі Arduino Uno 3*
керівник роботи: *к.ф.-м.н., доцент Хвищун М.В.*
затверджені наказом закладу вищої освіти від «31» грудня 2025 р. № 559/01-02
-
2. Строк подання здобувачем роботи: 04.06.2026 р.
-
3. Вихідні дані до роботи: *мікроконтролерна платформа – Arduino Uno 3 на базі ATmega328P; силовий ключ – MOSFET-транзистор; метод керування – широтно-імпульсна модуляція (PWM); тип навантаження – електричне навантаження постійного струму*
-
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
Вступ
Розділ 1. Аналітичний огляд систем широтно-імпульсного керування
Розділ 2. Схемне проектування та принцип роботи системи
Розділ 3. Розрахункова апаратно-рограмна частина
Розділ 4. Охорона праці та безпека
Висновки
-
5. Перелік графічного матеріалу
Презентація POWERPOINT (12 слайдів)
-
-
-
-
-
-
-
-
-
-

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
<i>Розділ 1. Аналітичний огляд систем широтно-імпульсного керування</i>	<i>Хвищун М. В., доцент</i>		
<i>Розділ 2. Схемне проектування та принцип роботи системи</i>	<i>Хвищун М. В., доцент</i>		
<i>Розділ 3. Розрахункова апаратно-рограмна частина</i>	<i>Хвищун М. В., доцент</i>		
<i>Розділ 4. Охорона праці та безпека</i>	<i>Заблоцький В. Ю., доцент</i>		
<i>Нормоконтроль</i>	<i>Селепина Й. Р., доцент</i>		
<i>Гарант ОП</i>	<i>Мороз С. А., доцент</i>		
<i>Показник запозичень тексту</i>	<i>%</i>		
<i>Академічна доброчесність</i>	<i>Хвищун М. В., доцент</i>		

7. Дата видачі завдання «31» грудня 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	<i>Розділ 1. Аналітичний огляд систем широтно-імпульсного керування</i>	до 01.03.2026 р.	
2.	<i>Розділ 2. Схемне проектування та принцип роботи системи</i>	до 15.03.2026 р.	
3.	<i>Розділ 3. Розрахункова апаратно-рограмна частина</i>	до 01.04.2026 р.	
4.	<i>Розділ 4. Охорона праці та безпека</i>	до 10.04.2026 р.	
5.	<i>Висновки</i>	до 20.04.2026 р.	
6.	<i>Формування списку використаних джерел</i>	до 30.04.2026 р.	
7.	<i>Оформлення пояснювальної записки та ілюстративного матеріалу</i>	до 10.05.2026 р.	
8.	<i>Нормоконтроль</i>	до 20.05.2026 р.	
9.	<i>Інструментальна перевірка на академічний плагіат</i>	до 30.05.2026 р.	
10.	<i>Представлення кваліфікаційної роботи до захисту</i>	до 04.06.2026 р.	

Здобувач вищої освіти _____ Роман АНТОСЮК

Керівник роботи _____ Микола ХВИЩУН

АНОТАЦІЯ

Антосюк Р. Т. Проєктування мікроконтролерної системи широтно-імпульсного керування навантаженням на базі Arduino Uno R3. Рукопис. Кваліфікаційна робота бакалавра ОП «Електроніка» спеціальності «Електроніка». Луцький національний технічний університет. Луцьк, 2026. 58 с.

Кваліфікаційна робота бакалавра складається зі вступу, чотирьох розділів, висновків, списку використаних джерел.

У першому розділі бакалаврської роботи проведено аналіз сучасних систем широтно-імпульсного керування навантаженням. Виконано порівняльний аналіз платформ Arduino Uno, STM32 та ESP32, а також силових ключів для реалізації систем керування. Обґрунтовано вибір Arduino Uno 3 та MOSFET-транзистора для реалізації системи PWM-керування.

У другому розділі виконано схемне проєктування системи широтно-імпульсного керування на базі Arduino Uno 3. Розроблено структурну, функціональну та принципову електричну схеми, описано принцип роботи системи. Для створення схем використано середовище EasyEDA.

У третьому розділі проведено розрахунок параметрів PWM-сигналу, силового ключа та потужності навантаження. Розроблено програмне забезпечення для Arduino Uno у середовищі Arduino IDE та проведено дослідження енергоефективності й стабільності роботи системи.

У четвертому розділі розглянуто питання охорони праці та безпеки під час експлуатації мікроконтролерної системи.

У висновках підсумовано результати роботи. Встановлено, що розроблена система PWM-керування на базі Arduino Uno 3 забезпечує ефективне регулювання потужності, високу енергоефективність та простоту реалізації.

Ключові слова: Arduino Uno 3, PWM, широтно-імпульсна модуляція, MOSFET, керування навантаженням, мікроконтролерна система.

ABSTRACT

Antosiuk R. Design Of A Microcontroller-Based Pulse-Width Modulation Load Control System Based On Arduino Uno R3. Manuscript. Qualification work for bachelor's degree in the Educational Program «Electronics», specialty «Electronics». Lutsk National Technical University. Lutsk, 2026. 58 p.

The bachelor's qualification work consists of an introduction, four chapters, conclusions, and a list of references.

In the first chapter, an analysis of modern pulse-width modulation load control systems was carried out. A comparative analysis of Arduino Uno, STM32, and ESP32 platforms, as well as power switching elements for control systems implementation, was performed. The choice of Arduino Uno 3 and a MOSFET transistor for the PWM control system was substantiated.

In the second chapter, the circuit design of the pulse-width modulation control system based on Arduino Uno 3 was developed. Structural, functional, and schematic electrical diagrams were created, and the operating principle of the system was described. The EasyEDA environment was used for schematic development.

The third chapter presents the calculation of PWM signal parameters, power switch parameters, and load power. Software for Arduino Uno was developed in the Arduino IDE environment, and studies of the system's energy efficiency and operational stability were carried out.

The fourth chapter considers occupational safety and operational safety issues related to the use of the microcontroller-based control system.

The conclusions summarize the obtained results. It was established that the developed PWM control system based on Arduino Uno 3 provides effective power regulation, high energy efficiency, and simplicity of implementation.

Keywords: Arduino Uno 3, PWM, Pulse-Width Modulation, MOSFET, Load Control, Microcontroller System

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1 АНАЛІТИЧНИЙ ОГЛЯД СИСТЕМ ШИРОТНО-ІМПУЛЬСНОГО КЕРУВАННЯ.....	9
1.1 Принципи широтно-імпульсної модуляції (PWM)	9
1.2 Огляд мікроконтролерних платформ (Arduino, STM32, ESP32).....	11
1.3 Силкові ключі для керування навантаженням	16
1.4 Функціональні можливості Arduino Uno 3	20
РОЗДІЛ 2 СХЕМНЕ ПРОЄКТУВАННЯ ТА ПРИНЦИП РОБОТИ СИСТЕМИ	24
2.1 Структурна та функціональна схема системи керування.....	24
2.2 Розробка принципової електричної схеми та схеми силового керування навантаженням.....	27
2.3 Формування PWM-сигналу та принцип роботи таймерів ArduinoUno 3 ..	29
2.4 Алгоритм роботи системи широтно – імпульсного керування	33
РОЗДІЛ 3 РОЗРАХУНКОВА АПАРАТНО-ПРОГРАМНА ЧАСТИНА	37
3.1 Розрахунок основних параметрів DC – DC перетворювача	37
3.2 Розрахунок індуктивності дроселя та параметрів LC-фільтра	40
3.3 Програмна реалізація системи PWM-керування.....	42
РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА	49
4.1 Аналіз небезпечних факторів при роботі з силовою електронікою.....	49
4.2 Ергономіка та безпечна організація робочого місця	52
ВИСНОВКИ.....	55
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	56

ВСТУП

Актуальність теми. У сучасних умовах розвитку електроніки, автоматизованих систем керування, вбудованих пристроїв та технологій Інтернету речей (IoT) особливої актуальності набуває підвищення енергоефективності та точності керування електричними навантаженнями. Значна кількість сучасних електронних систем потребує ефективного регулювання потужності для забезпечення стабільної роботи, зменшення теплових втрат та підвищення надійності функціонування. Одним із найбільш ефективних методів керування електричним навантаженням є широтно-імпульсна модуляція (PWM), яка дозволяє змінювати середнє значення напруги або струму без значних втрат енергії [1].

Широтно-імпульсне керування широко застосовується у системах керування електродвигунами, світлодіодному освітленні, імпульсних джерелах живлення, системах автоматизації та робототехніці. Основною перевагою PWM є робота силових елементів у ключовому режимі, що забезпечує високий коефіцієнт корисної дії та мінімальні втрати потужності. На відміну від аналогових методів регулювання, PWM дозволяє реалізувати ефективне цифрове керування із високою точністю та стабільністю параметрів системи.

Для реалізації систем широтно-імпульсного керування широко використовуються сучасні мікроконтролерні платформи, серед яких Arduino Uno 3, STM32 та ESP32. Однією з найбільш доступних та популярних платформ є Arduino Uno 3, яка базується на мікроконтролері ATmega328P та має вбудовані апаратні таймери для генерації PWM-сигналів. Перевагами Arduino Uno є простота програмування, доступність елементної бази, велика кількість бібліотек та можливість швидкого створення прототипів електронних систем [2, 3].

Для реалізації ефективного керування навантаженням у системах PWM доцільним є використання MOSFET-транзисторів як силових ключів. Вони забезпечують високу швидкодію, низькі втрати потужності та надійну роботу в

імпульсному режимі. Поєднання мікроконтролера Arduino Uno та MOSFET-транзистора дозволяє створити енергоефективну систему керування навантаженням із можливістю регулювання потужності в широкому діапазоні.

Метою кваліфікаційної роботи є проектування та дослідження мікроконтролерної системи широтно-імпульсного керування навантаженням на базі Arduino Uno 3.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- провести аналіз сучасних методів керування електричним навантаженням;
- дослідити принципи формування PWM-сигналів;
- виконати порівняльний аналіз мікроконтролерних платформ;
- обґрунтувати вибір Arduino Uno 3 та силового MOSFET-ключа;
- розробити структурну, функціональну та принципову схеми системи;
- реалізувати алгоритм PWM-керування;
- виконати розрахунок параметрів силової частини;
- розробити програмне забезпечення для Arduino Uno 3.

Об'єктом дослідження є мікроконтролерна система керування електричним навантаженням.

Предметом дослідження є методи та алгоритми широтно-імпульсного керування навантаженням на базі мікроконтролерної платформи Arduino Uno 3.

Практичне значення роботи полягає у розробці енергоефективної системи PWM-керування на базі Arduino Uno 3, яка може бути використана у системах автоматизації, керуванні електродвигунами, світлодіодних системах, вбудованих електронних пристроях та навчально-дослідницьких проєктах.

У процесі виконання роботи інструменти штучного інтелекту застосовувалися виключно як допоміжний засіб для редагування текстових матеріалів та опрацювання літературних джерел. Усі отримані результати були перевірені автором із дотриманням принципів академічної доброчесності.

РОЗДІЛ 1

АНАЛІТИЧНИЙ ОГЛЯД СИСТЕМ ШИРОТНО-ІМПУЛЬСНОГО КЕРУВАННЯ

1.1 Принципи широтно-імпульсної модуляції (PWM)

Широтно-імпульсна модуляція (PWM, Pulse Width Modulation) є одним із найпоширеніших методів керування електричною потужністю в сучасних електронних системах. Суть методу полягає у формуванні імпульсного сигналу прямокутної форми з постійною частотою та змінною тривалістю імпульсів, що дозволяє ефективно регулювати середнє значення напруги або струму на навантаженні [4].

Принцип дії ШІМ базується на періодичному перемиканні сигналу між двома рівнями (логічний «1» і «0» або високий/низький рівень напруги) з високою частотою. При цьому середнє значення вихідної напруги визначається співвідношенням часу увімкненого стану до загального періоду сигналу, що називається коефіцієнтом заповнення (duty cycle).

Коефіцієнт заповнення визначається як:

$$D = \frac{t_{on}}{T} 100\%, \quad (1.1)$$

де t_{on} – тривалість імпульсу (час увімкнення), T – період сигналу.

Зміна коефіцієнта заповнення дозволяє керувати середнім значенням напруги на навантаженні без зміни амплітуди сигналу. Наприклад, при коефіцієнті заповнення 50% середнє значення напруги становить половину максимальної.

На рисунку 1.1 подано графі PWM-сигнал при коефіцієнті заповнення 50 %. Основна ідея PWM полягає у тому, що замість плавної зміни аналогової напруги використовується швидке перемикання джерела живлення. Завдяки інерційності навантаження (наприклад, двигуна або світлодіода) формується усереднений ефект, еквівалентний аналоговому сигналу.

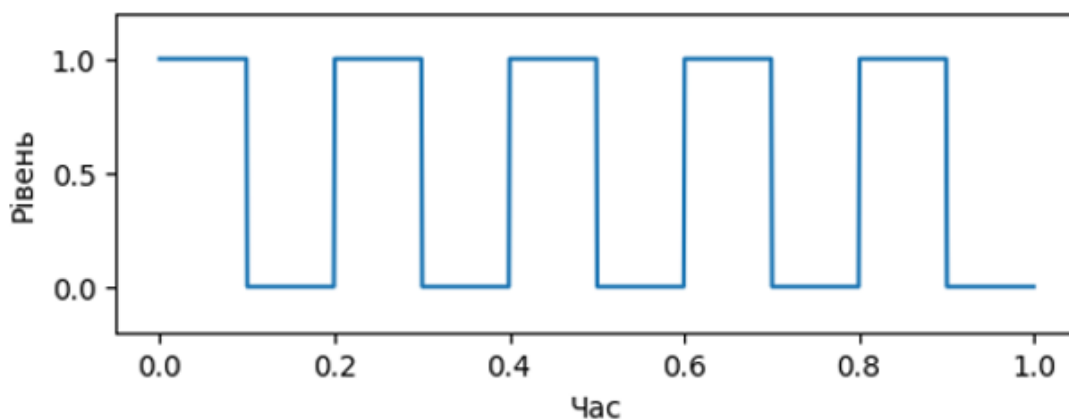


Рисунок 1.1 – PWM-сигнал при коефіцієнті заповнення 50 %

До ключових параметрів широтно-імпульсної модуляції належать:

- період сигналу (T) – час одного повного циклу;
- частота (f) – величина, обернена до періоду $f=1/T$;
- ширина імпульсу (t_{on}) – час активного стану сигналу;
- коефіцієнт заповнення (D) – визначає рівень вихідної потужності.

Зі збільшенням коефіцієнта заповнення зростає середня потужність, що передається на навантаження.

У сучасних мікроконтролерах генерація PWM-сигналу здійснюється за допомогою таймерів та лічильників. Лічильник підраховує тактові імпульси, а при досягненні заданого значення змінює стан вихідного сигналу. Таким чином формується необхідна тривалість імпульсу [5].

Як правило, використовуються такі режими роботи PWM:

- Fast PWM – швидке формування сигналу з високою частотою;
- Phase Correct PWM – симетричний сигнал для зменшення гармонік;
- Center-aligned PWM – імпульси симетричні відносно центру періоду.

Ці режими дозволяють оптимізувати характеристики сигналу залежно від конкретного застосування.

Розглянемо принцип керування потужністю. PWM дозволяє ефективно керувати потужністю за рахунок зміни тривалості включення ключового елемента (транзистора). При цьому втрати енергії мінімальні, оскільки ключ працює або в режимі повного відкриття, або повного закриття.

Середнє значення напруги на навантаженні визначається за формулою (1.2):

$$U_{avg} = D \cdot U_{max}, \quad (1.2)$$

де U_{max} – максимальна напруга джерела.

Таким чином, зміна коефіцієнта заповнення дозволяє плавно регулювати вихідні параметри без значних втрат енергії.

Основними перевагами широтно-імпульсної модуляції є:

- висока енергоефективність;
- можливість цифрової реалізації;
- висока точність керування;
- простота інтеграції у мікроконтролерні системи;
- універсальність застосування (двигуни, LED, джерела живлення).

Отже, широтно-імпульсна модуляція є ефективним методом цифрового керування електричною потужністю у сучасних електронних системах. Використання PWM дозволяє забезпечити високу енергоефективність, точність регулювання та мінімальні втрати потужності. Завдяки можливості реалізації за допомогою мікроконтролерів Arduino Uno 3 та інших вбудованих платформ, PWM широко застосовується у системах автоматизації.

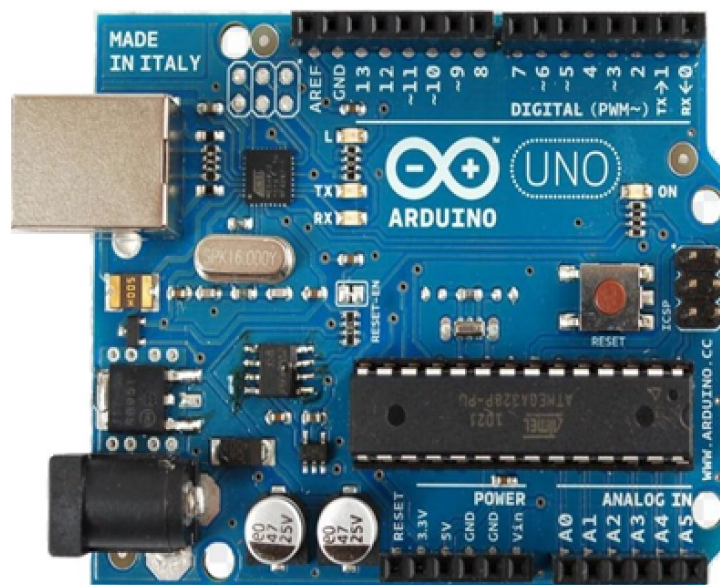
1.2 Огляд мікроконтролерних платформ (Arduino, STM32, ESP32)

Мікроконтролерні платформи є основою сучасних вбудованих систем керування. Для реалізації PWM-керування доцільно порівняти Arduino Uno 3, STM32 та ESP32 за продуктивністю, кількістю каналів PWM, складністю програмування та можливостями інтеграції з периферією.

Arduino Uno 3 є зручною платформою для навчальних і прототипних систем завдяки відкритій архітектурі, наявності апаратних PWM-виходів,

простому програмуванню в Arduino IDE та широкій бібліотечній підтримці [6].

На рисунку 1.2 подано зовнішній вигляд Arduino Uno3.



Рисунку 1.2 – Зовнішній вигляд Arduino Uno3 [6]

Основні характеристики Arduino Uno 3:

- мікроконтролер: ATmega328P;
- тактова частота: 16 МГц;
- 14 цифрових входів / виходів (6 PWM);
- 6 аналогових входів;
- напруга живлення: 5 В.

На рисунку 1.3 подано архітектуру та структурну схему Arduino Uno 3.

На рисунку подано структурну схему мікроконтролера ATmega328P, який є основним елементом плати Arduino Uno 3. У центральній частині схеми зображено процесорне ядро AVR, регістри, флеш-пам'ять, SRAM, EEPROM, тактовий генератор і службові блоки керування. Праворуч показані периферійні модулі мікроконтролера: таймери, PWM-блоки, UART, SPI, ADC та цифрові порти вводу/виводу.

Особливу роль у системі PWM-керування відіграють таймери та блоки PWM, які формують імпульсні сигнали для керування навантаженням. Через

цифрові виводи мікроконтролер передає керуючий сигнал на силовий ключ, наприклад MOSFET-транзистор.

Також схема демонструє наявність аналогових входів, які можуть використовуватися для зчитування сигналів із потенціометра або датчиків, що дозволяє змінювати коефіцієнт заповнення PWM-сигналу.

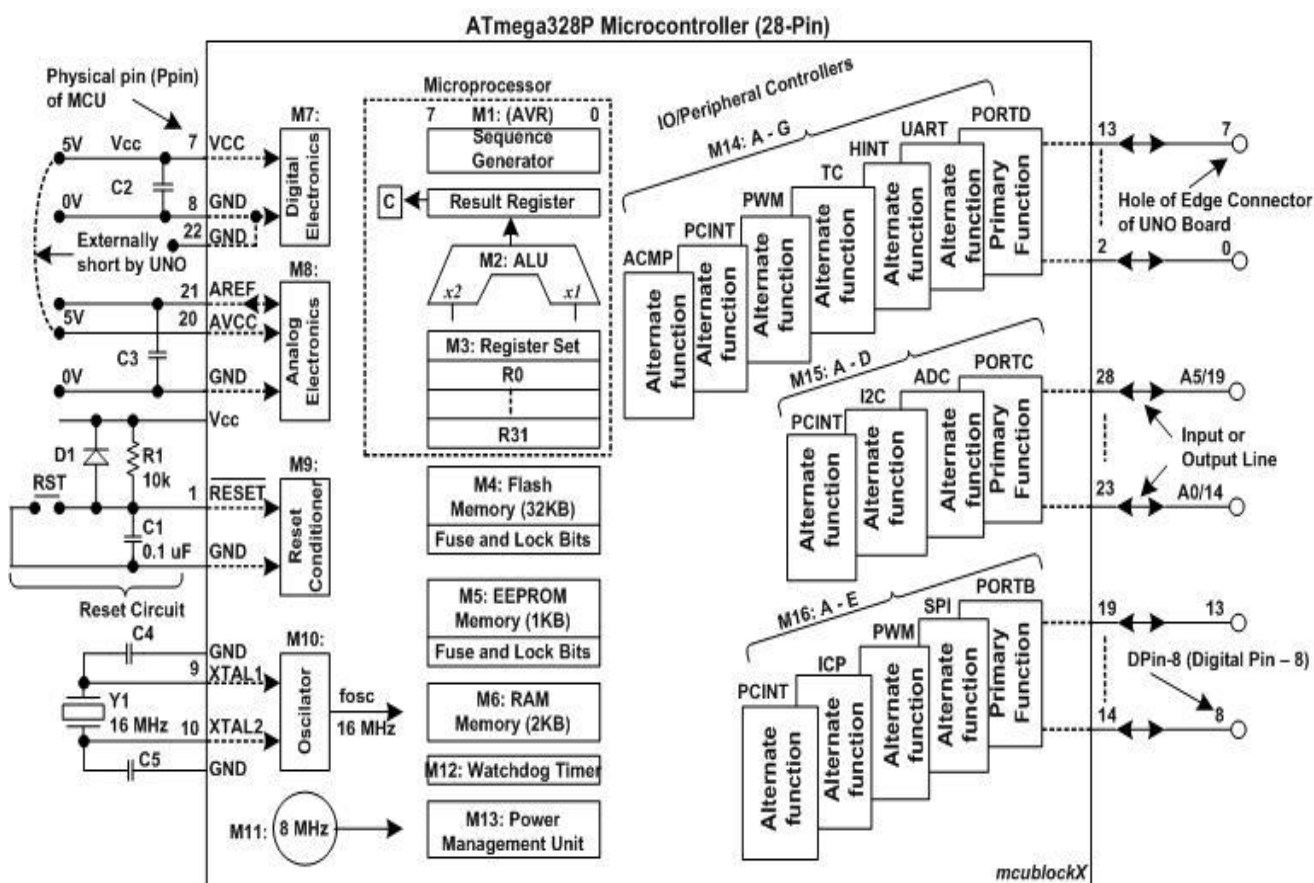


Рисунок 1.3 – Архітектура та структурна схема Arduino Uno 3 [6]

Платформи STM32F103C8T6 Blue Pill побудовані на ядрах ARM Cortex- M і мають розвинені таймери, тому придатні для промислових систем керування, де потрібна висока точність PWM та робота в реальному часі [8]. На рисунку 1.4 подано зовнішній вигляд STM32F103C8T6 Blue Pill, а на рисунку 1.5 подано структуру плати STM32 F103C8T6 Blue Pill.

STM32 широко застосовується у складних системах керування електроприводами та силовій електроніці.

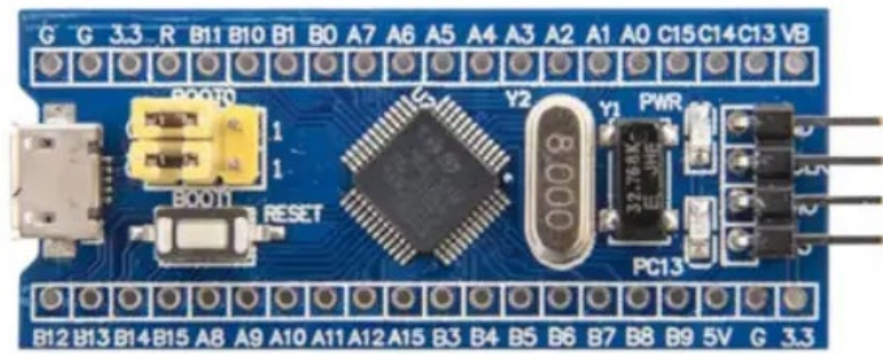


Рисунок 1.4 – Зовнішній вигляд STM32 F103C8T6 Blue Pill [8]

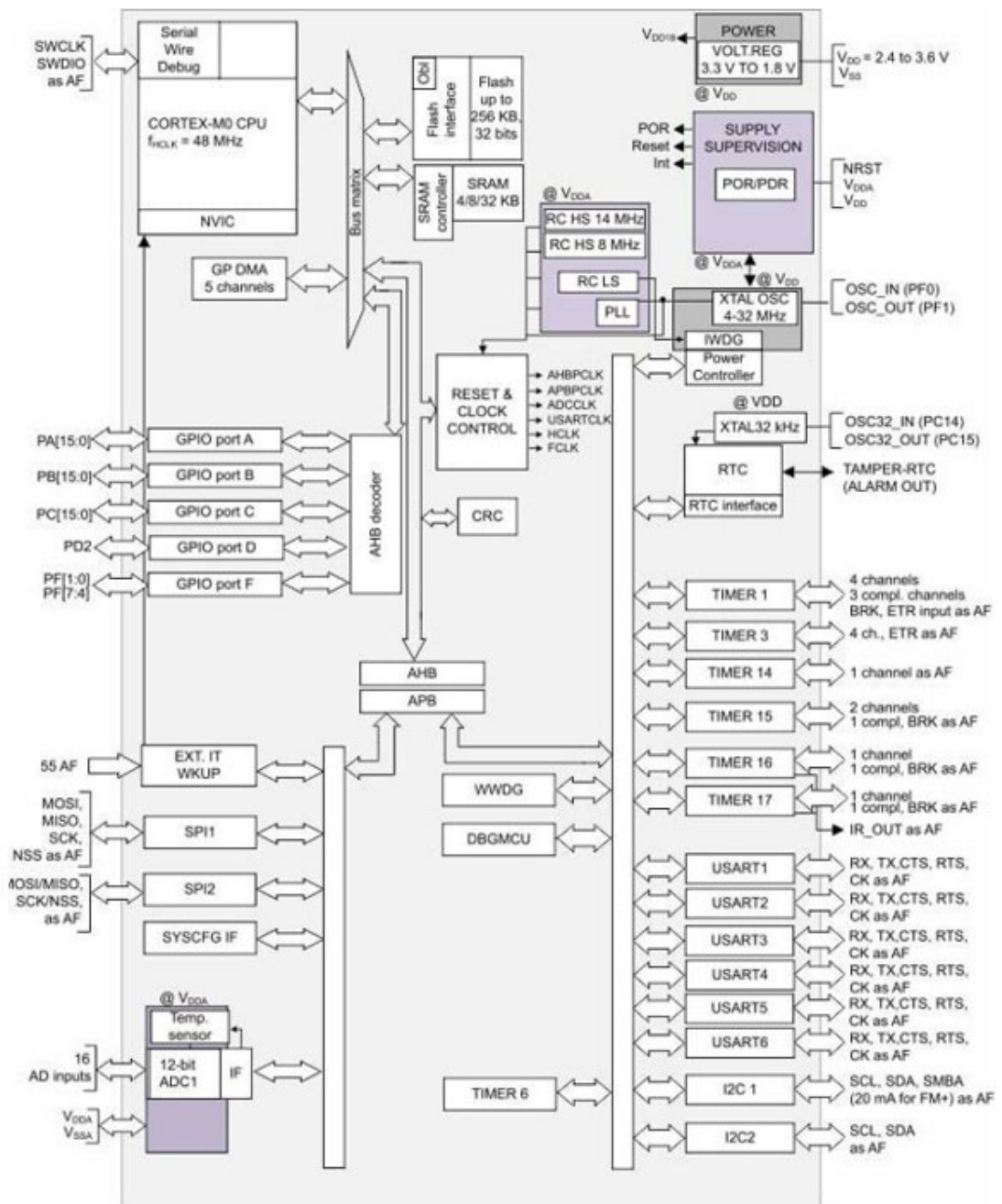


Рисунок 1.5 – Структура та приклад плати STM32 F103C8T6 Blue Pill [8]

ESP32 є сучасною платформою з підтримкою Wi-Fi та Bluetooth, значною кількістю GPIO і каналів PWM. Її доцільно застосовувати в IoT-системах, однак для даної роботи пріоритетом є простота реалізації, тому основною платформою обрано Arduino Uno 3 [9].

На рисунку 1.6 подано плату ESP32 та її функціональні можливості.

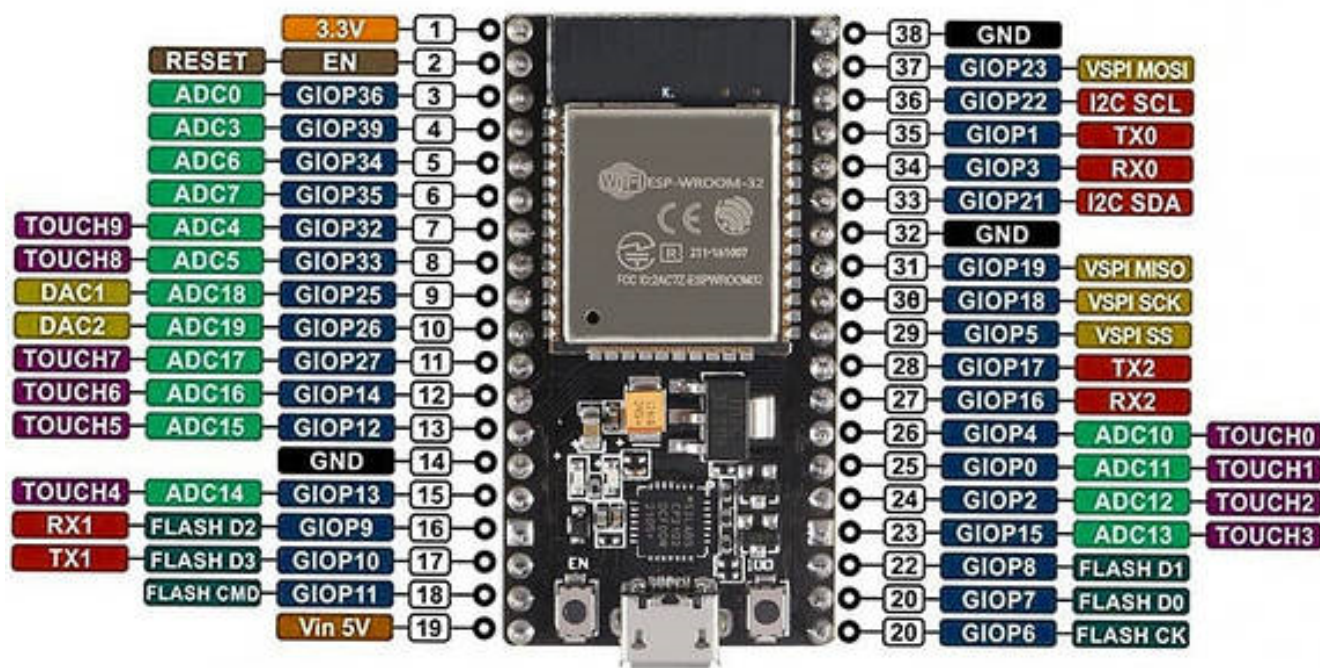


Рисунок 1.6 – Плата ESP32 та її функціональні можливості [9]

Узагальнено ESP32 має двоядерний процесор Xtensa, тактову частоту до 240 МГц, вбудовані бездротові інтерфейси та підтримку багатьох PWM-каналів, що робить її ефективною для дистанційного керування навантаженням.

Проведений аналіз показує, що:

- Arduino Uno 3 є оптимальним вибором для навчальних і прототипних систем;
- STM32 забезпечує максимальну продуктивність і точність PWM;
- ESP32 є найкращим варіантом для IoT-рішень.

У таблиці 1.1 подано порівняння мікроконтролерних платформ.

Таблиця 1.1 – Порівняння мікроконтролерних платформ

Параметр	Arduino Uno	STM32	ESP32
Ядро	AVR	ARM Cortex-M	Xtensa
Частота	16 МГц	до 400 МГц	до 240 МГц
PWM	6 каналів	розширені	до 16 каналів
Комунікації	базові	розширені	Wi-Fi, Bluetooth
Складність	низька	висока	середня
Застосування	навчання	промисловість	IoT

З огляду на мету даної роботи, доцільним є використання платформи Arduino Uno 3, яка забезпечує достатній функціонал для реалізації системи широтно-імпульсного керування при мінімальній складності реалізації.

1.3 Силові ключі для керування навантаженням

Силові ключі забезпечують комутацію струму між джерелом живлення та навантаженням. У системах PWM-керування найбільш доцільними є MOSFET-транзистори, оскільки вони працюють у ключовому режимі, мають високу швидкодію та низькі втрати потужності [10].

У розробленій системі керування використовується N-канальний MOSFET, на затвор якого через обмежувальний резистор подається PWM-сигнал із Arduino Uno 3. Підтягувальний резистор забезпечує закритий стан транзистора за відсутності сигналу, а захисний діод застосовується для індуктивного навантаження [11].

На рисунку 1.7 показано підключення N-канального MOSFET-транзистора FQP30N06L для керування індуктивним навантаженням (соленоїдом або двигуном) за допомогою Arduino. Цифровий сигнал з Arduino подається на затвор транзистора через резистор R1 (220 Ом), який обмежує струм керування та захищає вихід мікроконтролера. Резистор R2 (100 кОм) використовується як підтягуючий до землі, щоб забезпечити закритий стан транзистора при

відсутності сигналу. При подачі логічного сигналу MOSFET відкривається, і струм проходить через навантаження L1 від джерела живлення 12 V до землі через транзистор. Це забезпечує вмикання соленоїда або двигуна. Діод D1 (1N4002) виконує функцію захисту від імпульсів самоіндукції, які виникають під час вимикання індуктивного навантаження. Він запобігає пошкодженню MOSFET-транзистора та підвищує надійність роботи схеми.

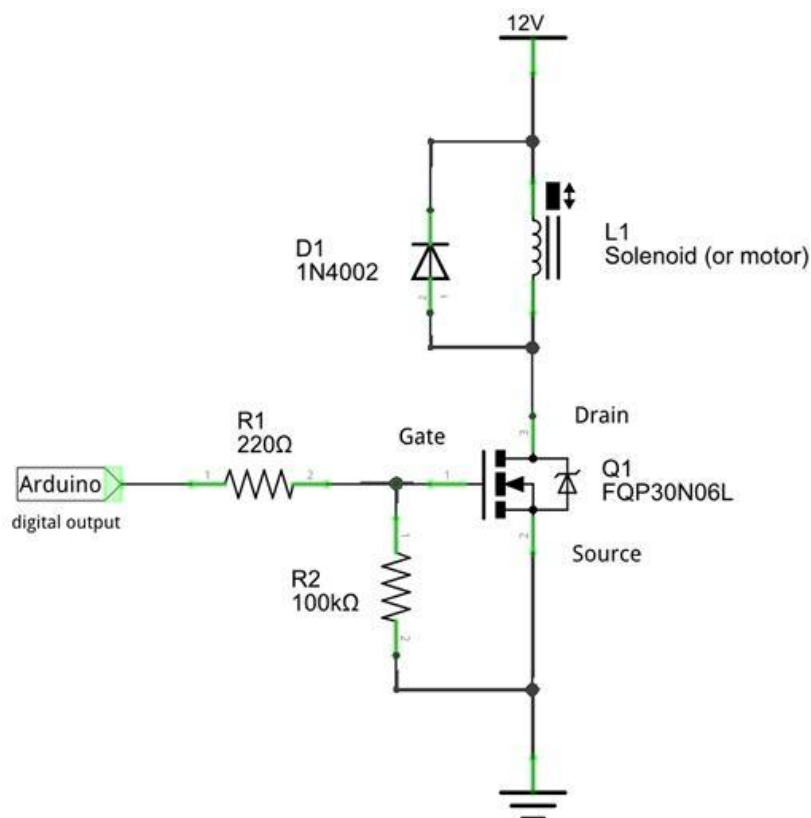


Рисунок 1.7 – Схема підключення MOSFET-транзистора [11]

MOSFET широко використовується у схемах керування двигунами, LED та джерелах живлення.

Біполярні транзистори та реле можуть використовуватися у простих схемах, однак вони мають суттєві обмеження: BJT потребує базового струму та має більші втрати, а реле через механічний принцип роботи непридатне для високочастотного PWM-керування [12].

На рисунку 1.8 подано схему підключення біполярного транзистора до Arduino.

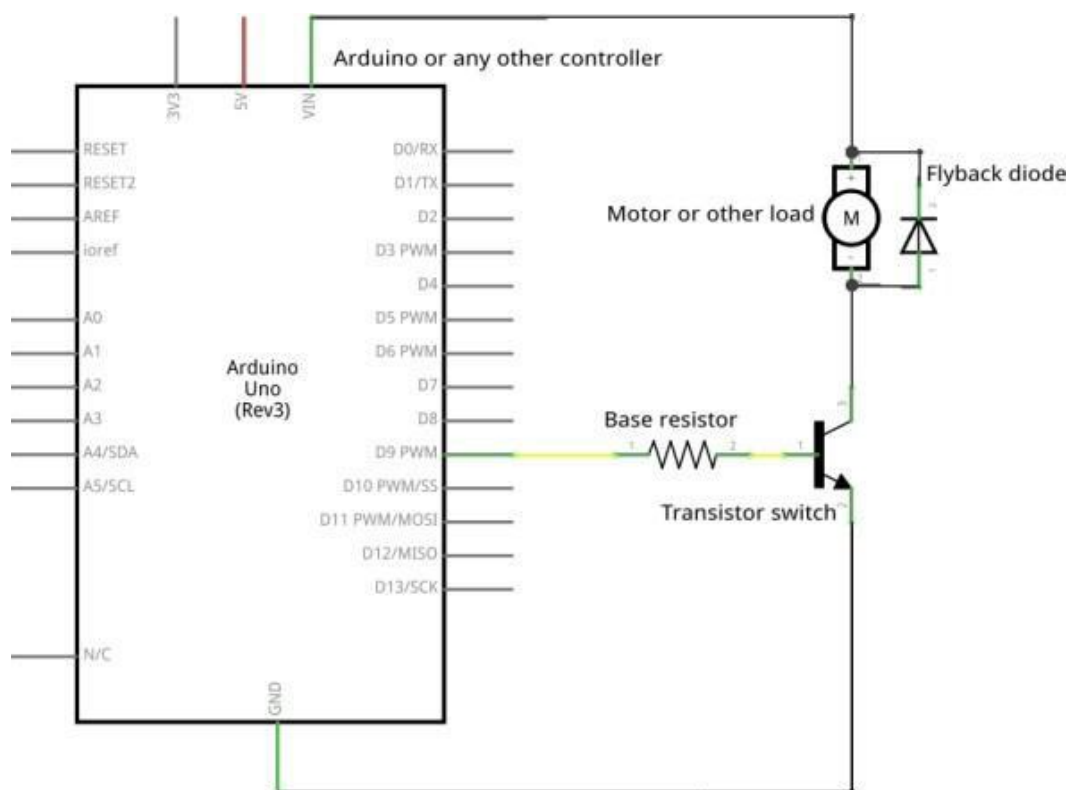


Рисунок 1.8 – Схема підключення біполярного-транзистора як ключа [12]

Порівняльний аналіз показує, що MOSFET-транзистори є найбільш ефективними для реалізації систем широтно-імпульсного керування. Вони забезпечують високу швидкодію, низькі втрати енергії та стабільність роботи системи.

На рисунку 1.9 показано принцип підключення MOSFET-транзистора до Arduino Uno для керування електричним навантаженням за допомогою PWM – сигналу.

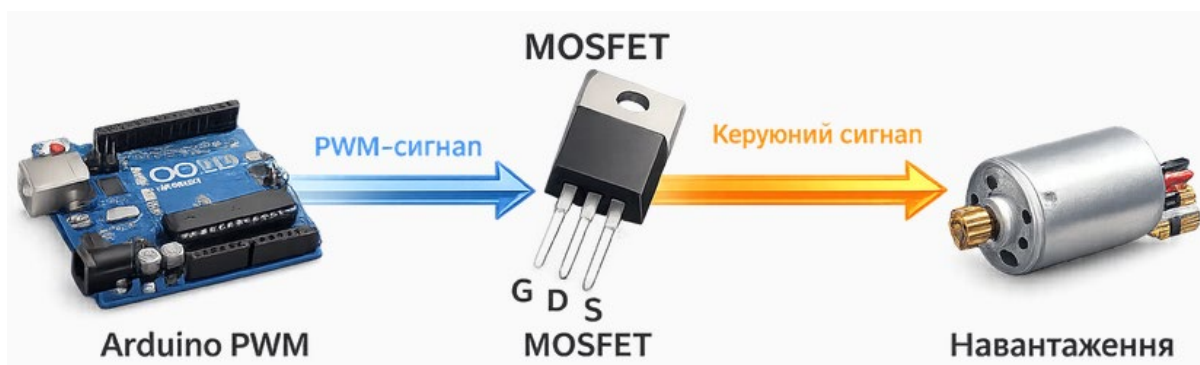


Рисунок 1.9 – MOSFET як силовий ключ у схемі PWM-керування

У таблиці 1.2 наведено порівняння функціоналу силових ключів

Таблиця 1.2 – Порівняння силових ключів

Параметр	MOSFET	BJT	Реле
Тип керування	Напруга	Струм	Котушка
Швидкодія	Висока	Середня	Низька
Енергоефективність	Висока	Низька	Середня
PWM	Так	Обмежено	Ні
Надійність	Висока	Середня	Низька (знос)
Застосування	PWM, двигуни	Прості схеми	Високі навантаження

Як видно з рисунка 1.10, MOSFET-транзистори мають найвищу енергоефективність та є найбільш придатними для PWM-керування.

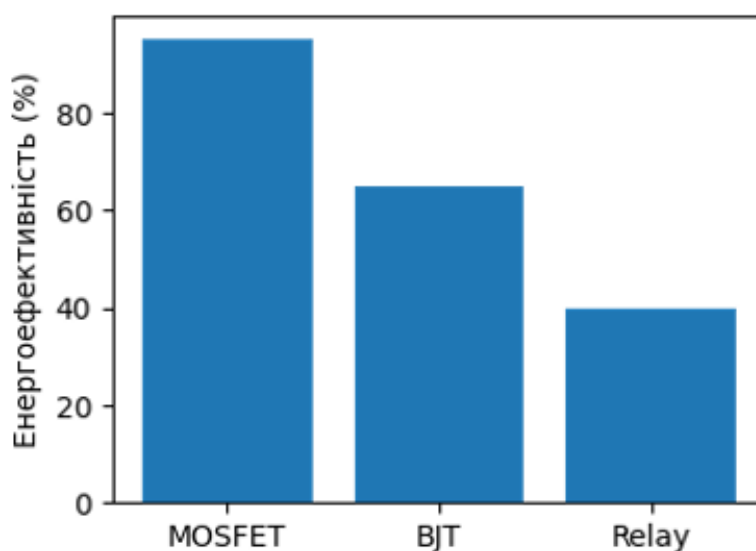


Рисунок 1.10 – Порівняння енергоефективності силових ключів

Проведений аналіз показує, що для систем широтно-імпульсного керування найбільш доцільним є використання MOSFET-транзисторів, які забезпечують високу швидкодію, низькі втрати енергії та ефективну роботу в режимі PWM. Біполярні транзистори можуть застосовуватись у простих схемах,

тоді як реле доцільно використовувати лише для дискретного керування без PWM.

1.4 Функціональні можливості Arduino Uno 3

Arduino Uno 3 містить мікроконтролер ATmega328P, цифрові входи/виходи, аналогові входи, таймери та інтерфейси зв'язку. Наявність апаратних таймерів дозволяє формувати PWM-сигнали без значного навантаження на процесор [13].

Розглянемо таймери та PWM-можливості Arduino Uno 3.

На рисунку 1.11 відображено формування PWM-сигналу в Arduino Uno 3.

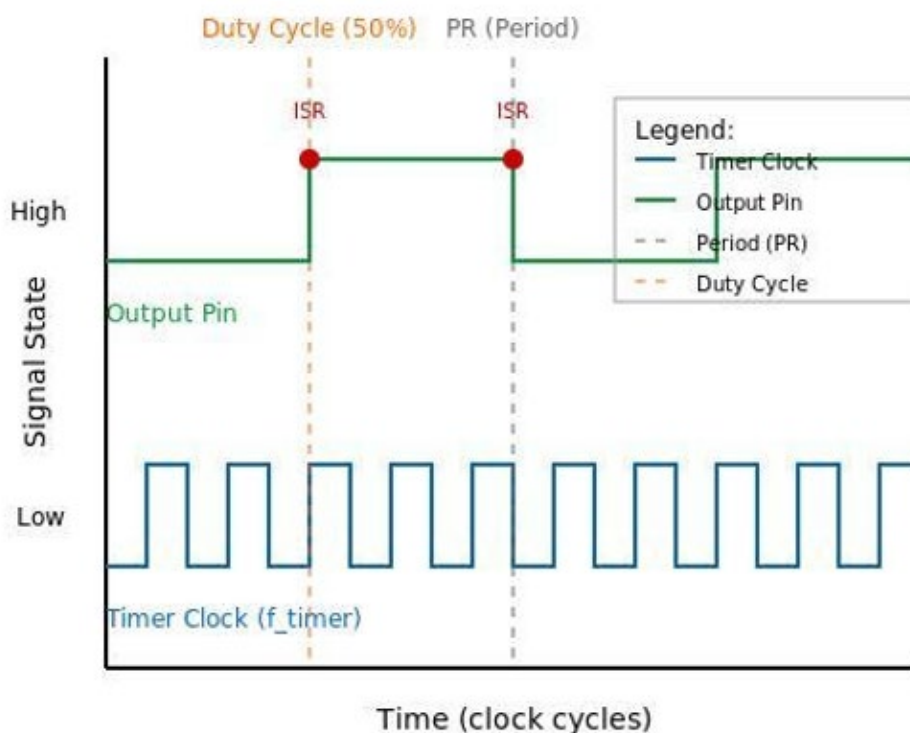


Рисунок 1.11 – Формування PWM-сигналу в Arduino Uno 3 [13]

Arduino Uno 3 програмується за допомогою середовища Arduino IDE, яке підтримує мову програмування, засновану на C/C++.

Розглянемо функціональні можливості Arduino Uno 3, які подано на рисунку 1.12.

Arduino Uno 3 у системі виконує функції зчитування сигналів із керуючих елементів або датчиків, обробки даних, формування PWM-сигналу, керування виконавчим пристроєм через MOSFET-ключ та відображення стану системи на індикаторах.

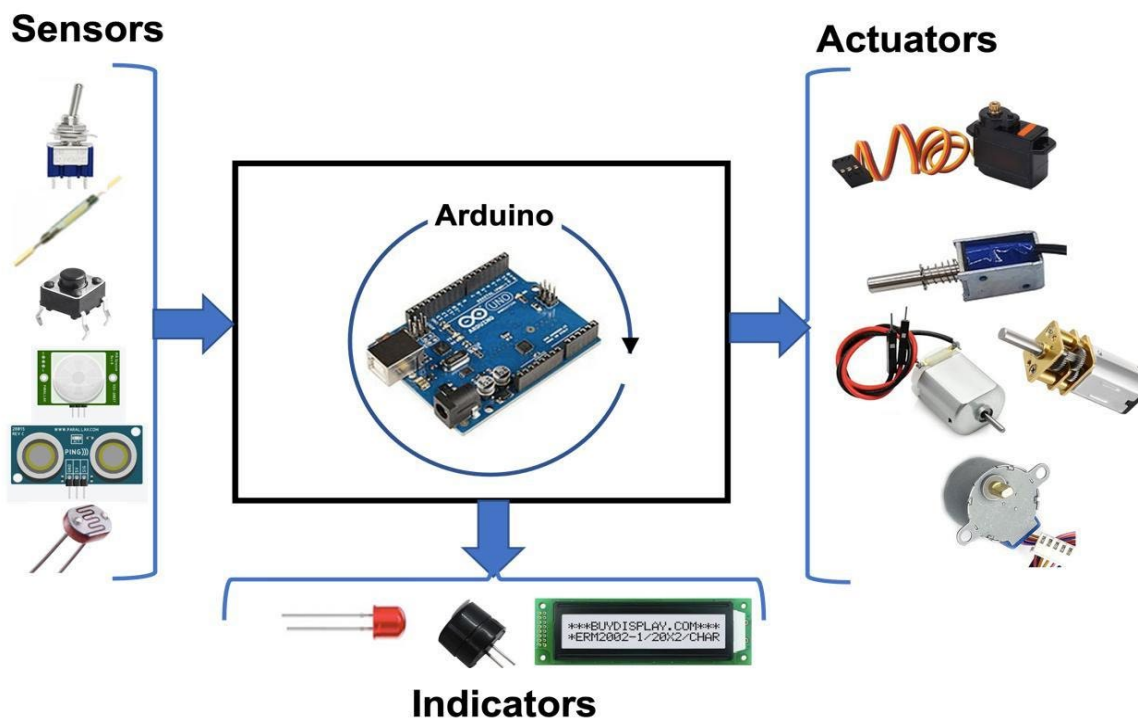


Рисунок 1.12 – Функціональні можливості Arduino Uno 3

Проведений аналіз показує, що Arduino Uno 3 має достатні апаратні та програмні можливості для реалізації систем широтно-імпульсного керування навантаженням. Завдяки наявності апаратних таймерів, підтримки PWM та простоти програмування, дана платформа є доцільною для використання у даній кваліфікаційній роботі.

Порівняльний аналіз існуючих рішень показує, що PWM-системи на мікроконтролерах мають найкраще співвідношення енергоефективності, точності та гнучкості реалізації порівняно з аналоговими, релейними та фазовими регуляторами [14].

У таблиці 1.3 подано порівняльний аналіз рішень систем керування.

Таблиця 1.3 – Порівняння існуючих систем керування

Тип системи	Енергоефективність	Складність	Точність	Можливість PWM	Застосування
Аналогова	Низька	Низька	Висока	Ні	Лінійні регулятори
Релейна	Середня	Дуже низька	Низька	Ні	Побутові системи
Фазова	Середня	Середня	Середня	Обмежено	АС-навантаження
PWM (МК)	Висока	Середня	Висока	Так	Вбудовані системи

З таблиці видно, що PWM-керування має найкраще співвідношення ефективності, точності та гнучкості реалізації.

На рисунку 1.13 подана структурна схема системи PWM-керування навантаженням.

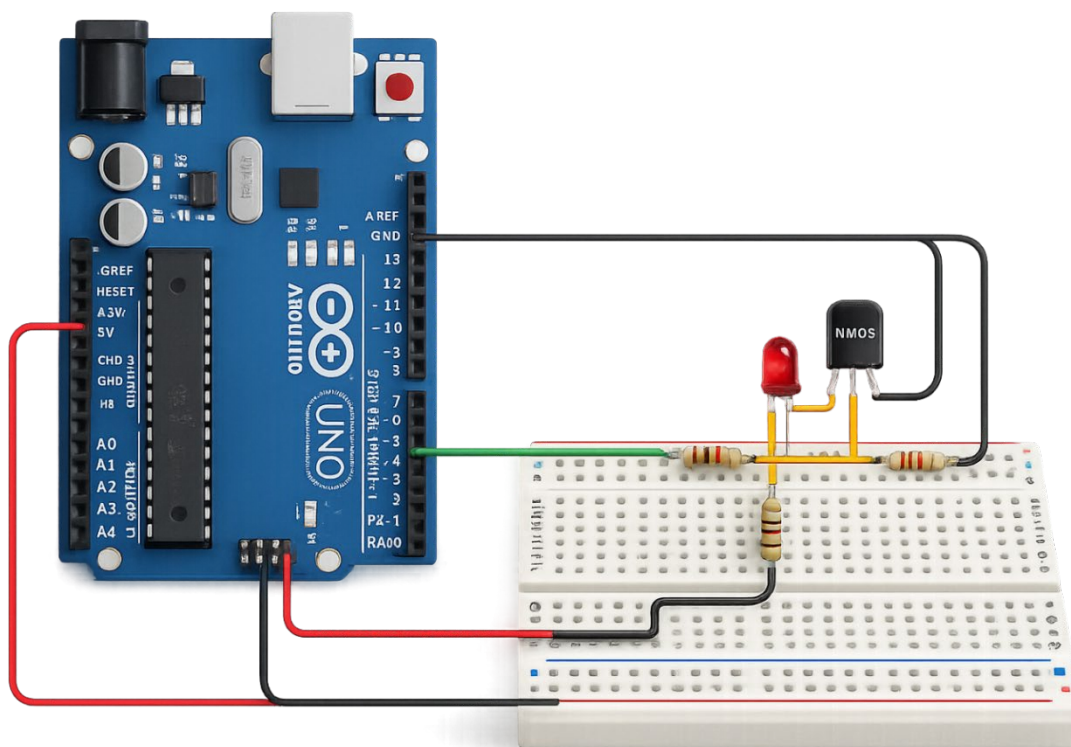


Рисунок 1.13 – Структурна схема системи PWM-керування навантаженням

Типова структура системи включає мікроконтролер Arduino Uno 3, блок формування PWM-сигналу, драйвер або ланцюг керування затвором, силовий MOSFET-ключ, джерело живлення, захисні елементи та навантаження.

PWM дозволяє ефективно керувати потужністю без використання громіздких аналогових схем.

Проведений аналіз існуючих рішень показав, що найбільш ефективним методом керування електричним навантаженням є широтно-імпульсна модуляція, реалізована на базі мікроконтролера. Обґрунтовано вибір платформи Arduino Uno 3, яка забезпечує достатні функціональні можливості для реалізації системи керування при мінімальній складності та вартості. Отримані висновки є основою для подальшого проєктування схемотехнічної реалізації системи керування у наступних розділах кваліфікаційної роботи.

Запропонована структурна схема системи є оптимальною для реалізації задачі керування навантаженням із використанням PWM.

РОЗДІЛ 2

СХЕМНЕ ПРОЄКТУВАННЯ ТА ПРИНЦИП РОБОТИ СИСТЕМИ

2.1 Структурна та функціональна схема системи керування

Проектування системи PWM-керування передбачає розробку структурної, функціональної та принципової схем, які визначають проходження сигналів, взаємодію апаратних вузлів і спосіб керування потужністю навантаження [15].

Основною задачею системи є регулювання потужності електричного навантаження шляхом зміни коефіцієнта заповнення PWM-сигналу. Реалізація такого підходу дозволяє забезпечити плавне керування навантаженням при мінімальних втратах енергії та високій стабільності роботи системи.

Структурна схема системи PWM-керування наведена на рисунку 2.1.

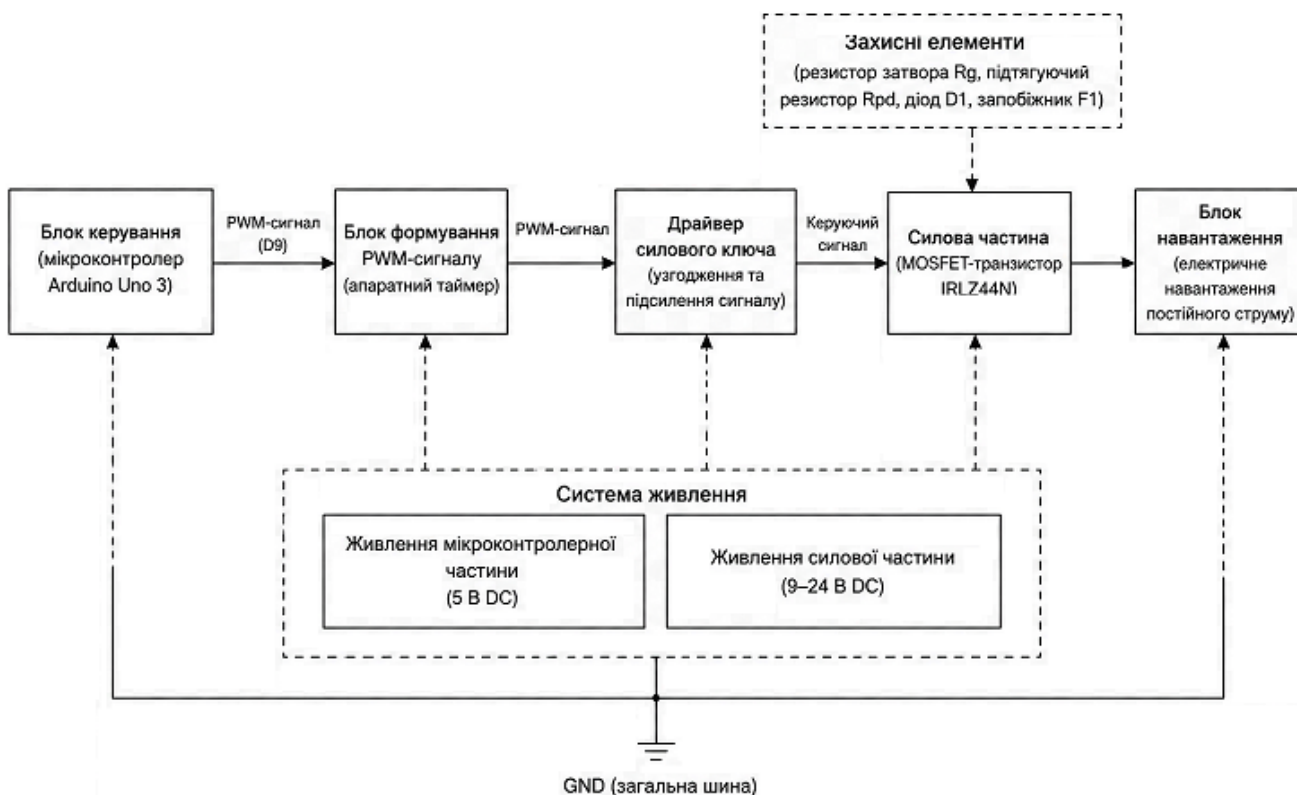


Рисунок 2.1 – Структурна схема системи широтно-імпульсного керування навантаженням [15]

Центральним елементом системи є мікроконтролер Arduino Uno 3, який виконує функцію формування PWM-сигналу та керування силовим MOSFET-

транзистором.

Сигнал керування формується програмно та подається на PWM-вихід D9. Далі через резистор затвора сигнал надходить на керуючий електрод MOSFET-транзистора, який працює у ключовому режимі.

Силова частина системи забезпечує комутацію струму навантаження та відокремлена від сигнальної частини мікроконтролера. Такий підхід дозволяє підвищити надійність системи та мінімізувати вплив імпульсних завад на роботу Arduino Uno 3.

Розглянемо опис функціональних блоків системи. Як видно з таблиці 2.1, кожен функціональний блок виконує окрему задачу, що дозволяє реалізувати модульну структуру системи керування.

Таблиця 2.1 – Функціональні блоки системи керування

Блок системи	Основна функція
Arduino Uno 3	Формування PWM-сигналу
PWM-вихід D9	Передача керуючого сигналу
MOSFET-транзистор	Комутація струму навантаження
Захисний діод	Захист від самоіндукції
Джерело живлення	Живлення системи
Навантаження	Виконавчий елемент
Система охолодження	Відведення тепла

Функціональна схема проходження сигналів показує послідовність роботи системи: живлення Arduino Uno 3, ініціалізація PWM-виходу, формування PWM-сигналу, керування затвором MOSFET, комутація струму та зміна середньої потужності навантаження [16].

У таблиці 2.2 подані основні компоненти та їх призначення. Інформація з даної таблиці дає краще уявлення про структуру та принцип роботи системи.

Розглянемо функціональну схему проходження сигналів, яка наведена на рисунку 2.2.

Таблиця 2.2 – Основні компоненти та їх призначення

Позначення	Компонент	Призначення
A1	Arduino Uno3	Мікроконтролер
VT1	MOSFET	Силовий ключ
R1	Резистор (100...220 Ом)	Обмеження струму затвора
R2	Резистор (10 кОм)	Підтягування затвора до GND
VD1	Діод	Захист від перенапруги
HL1	Світлодіод	Індикація роботи
M1	Навантаження	Об'єкт керування
C1	Конденсатор 4700 мкФ	Фільтрація живлення



Рисунок 2.2 – Функціональна схема проходження сигналів у системі PWM-керування

Принцип роботи системи полягає в тому, що Arduino Uno 3 після запуску формує PWM-сигнал заданої частоти, подає його на затвор MOSFET, а MOSFET

змінює тривалість увімкненого стану навантаження відповідно до коефіцієнта заповнення.

До переваг розробленої системи належать простота схемної та програмної реалізації, висока енергоефективність, низькі втрати потужності, швидкодія, цифровий характер керування та можливість масштабування.

Розроблена структура системи є оптимальною для реалізації задач PWM-керування навантаженням у вбудованих електронних системах.

Отже, у підрозділі розроблено структурну та функціональну схему системи широтно-імпульсного керування навантаженням на базі Arduino Uno 3. Визначено взаємозв'язок між основними функціональними блоками системи, описано принцип проходження сигналів та особливості взаємодії апаратних вузлів. Використання MOSFET-транзистора у ключовому режимі дозволяє забезпечити ефективне керування потужністю навантаження при мінімальних енергетичних втратах. Отримані результати є основою для подальшого схемного проєктування та розробки принципової електричної схеми системи.

2.2 Розробка принципової електричної схеми системи PWM-керування навантаженням

Принципова електрична схема відображає електричні з'єднання Arduino Uno 3, MOSFET-транзистора, резисторів керування, захисного діода, джерела живлення та навантаження [17].

У схемі силове навантаження комутується MOSFET-транзистором, а мікроконтролер подає на затвор лише керуючий PWM-сигнал. Такий поділ сигнальної та силової частин підвищує надійність і захищає Arduino від перевантаження.

Для забезпечення стабільного закритого стану MOSFET використовується підтягуючий резистор R_{pd} , який підключений між затвором та землею. Це дозволяє уникнути випадкового відкриття транзистора при запуску системи або відсутності керуючого сигналу.

Принципова електрична схема силового керування наведена на рисунку 2.3. Як видно з рисунка 2.3, навантаження підключене до стоку MOSFET-транзистора, а витік підключений до загальної шини GND. Така конфігурація є типовою для низькобічного (low-side) керування навантаженням.

Для індуктивних навантажень у схемі використовується захисний діод D1, який компенсує імпульси самоіндукції при вимиканні навантаження та запобігає пошкодженню MOSFET-транзистора. У системі також використовується плавкий запобіжник F1, який забезпечує захист від короткого замикання та перевантаження.

Розглянемо особливості реалізації системи керування навантаженням. Однією з основних переваг розробленої системи є модульний принцип побудови.

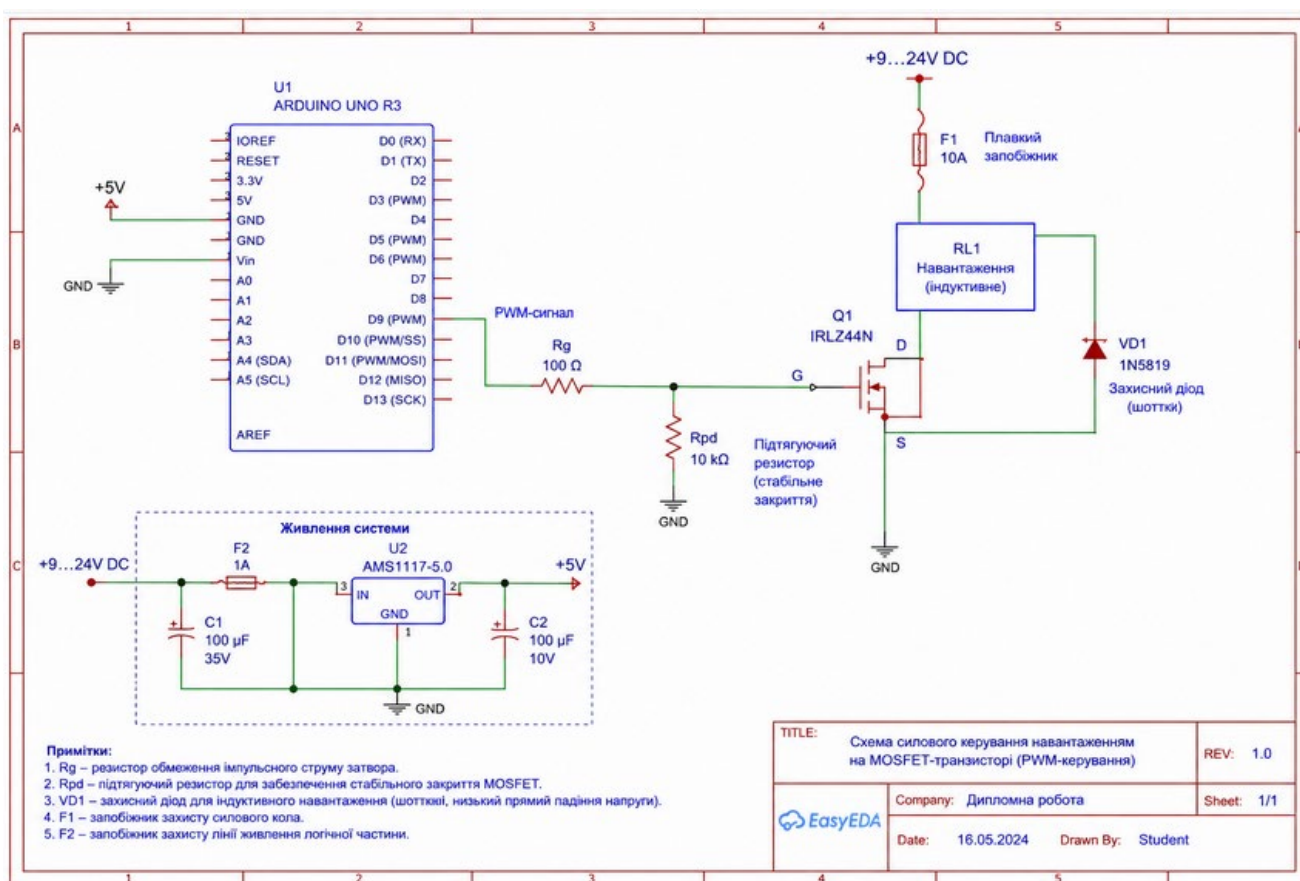


Рисунок 2.3 – Принципова електрична схема силового керування навантаженням на MOSFET-транзисторі

Важливою особливістю системи є використання MOSFET-транзистора

IRLZ44N, який має низький опір відкритого каналу та забезпечує ефективну роботу в імпульсному режимі.

При збільшенні коефіцієнта заповнення збільшується тривалість відкритого стану MOSFET, а відповідно – середня потужність навантаження.

При зменшенні коефіцієнта заповнення потужність зменшується.

Такий принцип роботи забезпечує ефективне цифрове керування електричним навантаженням при високому коефіцієнті корисної дії. Розроблена принципова схема забезпечує ефективне керування навантаженням за допомогою PWM. Використання MOSFET як силового ключа дозволяє мінімізувати втрати енергії, а застосування Arduino Uno 3 спрощує реалізацію системи та забезпечує гнучкість керування.

2.3 Формування PWM-сигналу та принцип роботи таймерів Arduino Uno 3

Arduino Uno 3 побудована на ATmega328P, який має апаратні таймери для генерації PWM-сигналів. У роботі використовується вивід D9, пов'язаний із Timer1, що забезпечує стабільне формування керуючого сигналу для MOSFET [18]. У таблиці 2.3 перераховано PWM-виводи Arduino Uno 3.

Таблиця 2.3 – PWM-виводи Arduino Uno 3

Таймер	PWM-виводи	Розрядність
Timer0	D5, D6	8 біт
Timer1	D9, D10	16 біт
Timer2	D3, D11	8 біт

У розробленій системі PWM-сигнал формується на виводі D9, який апаратно пов'язаний із Timer1. Використання саме 16-бітного таймера дозволяє отримати більш високу точність регулювання коефіцієнта заповнення та забезпечити стабільну роботу системи керування.

Принцип формування PWM-сигналу базується на циклічному підрахунку

імпульсів таймером. Після запуску таймер починає рахувати тактові імпульси від системного генератора частотою 16 МГц. При досягненні заданого значення лічильника відбувається зміна стану вихідного сигналу. Таким чином формується прямокутний сигнал із необхідною тривалістю імпульсів.

Коефіцієнт заповнення PWM визначається як відношення часу активного стану сигналу до повного періоду згідно формули (2.1):

$$D = \frac{t_{on}}{T} 100\%, \quad (2.1)$$

де t_{on} – тривалість імпульсу;

T – період сигналу.

Зміна коефіцієнта заповнення дозволяє регулювати середню потужність, що подається на навантаження. Наприклад, при коефіцієнті заповнення 25 % навантаження отримує приблизно чверть максимальної потужності, а при 75 % – близько трьох чвертей.

Середнє значення вихідної напруги визначається за формулою (2.2):

$$U_{avg} = D \cdot U_{max}, \quad (2.2)$$

де U_{max} – максимальна напруга джерела живлення;

D – коефіцієнт заповнення.

Однією з важливих особливостей Arduino Uno 3 є можливість апаратної генерації PWM-сигналу за допомогою функції `analogWrite()`. Дана функція дозволяє швидко змінювати коефіцієнт заповнення без необхідності безпосередньої роботи з регістрами таймера.

Приклад формування PWM-сигналу наведено в програмі:

```
const int pwmPin = 9;
void setup() {
    pinMode(pwmPin, OUTPUT);
}
```

```
void loop() {
    analogWrite(pwmPin, 128);
}
```

У системі можуть використовуватися режими Fast PWM та Phase Correct PWM. Fast PWM забезпечує високу частоту та просту реалізацію, а Phase Correct PWM формує симетричний сигнал і зменшує гармонічні спотворення.

Значення коефіцієнта заповнення безпосередньо визначає тривалість відкритого стану MOSFET-транзистора. При малому коефіцієнті заповнення навантаження отримує невелику середню потужність, а при збільшенні значення PWM зростає тривалість проходження струму через навантаження. Такий спосіб регулювання є енергоефективним, оскільки силовий ключ більшу частину часу перебуває або у повністю відкритому, або у закритому стані.

Вибір конкретного режиму PWM залежить від типу навантаження та вимог до якості керуючого сигналу. Для світлодіодного навантаження важливо уникнути помітного мерехтіння, тому частота сигналу повинна бути достатньо високою. Для двигунів постійного струму додатково враховують інерційність механічної частини, рівень шуму та нагрівання силового ключа. У розробленій системі використання апаратної генерації PWM дає змогу підтримувати стабільну частоту сигналу незалежно від виконання основного програмного циклу. У таблиці 2.4 наведено порівняння режимів PWM.

Таблиця 2.4 – Порівняння режимів PWM

Параметр	Fast PWM	Phase Correct PWM
Частота	Вища	Нижча
Симетрія сигналу	Ні	Так
Гармоніки	Більші	Менші
Швидкодія	Висока	Середня
Застосування	PWM-керування	Прецизійні системи

Частота PWM-сигналу визначається параметрами таймера та дільником

частоти (prescaler). Для Timer1 частота PWM може бути розрахована за формулою (2.3):

$$f_{\text{PWM}} = f_{\text{clk}} / (N \cdot (1 + \text{TOP})), \quad (2.3)$$

де f_{clk} – тактова частота мікроконтролера;

N – коефіцієнт дільника;

TOP – максимальне значення таймера.

Зміна значення prescaler дозволяє регулювати частоту PWM-сигналу залежно від типу навантаження. Для світлодіодних систем зазвичай використовують частоти понад 500 Гц, тоді як для керування двигунами можуть застосовуватись частоти декілька кілогерц.

У розробленій системі PWM-сигнал використовується для керування MOSFET-транзистором IRLZ44N, який працює у ключовому режимі. PWM-сигнал надходить із виводу D9 через резистор R_g на затвор MOSFET-транзистора. Зміна коефіцієнта заповнення PWM призводить до зміни середньої потужності, що подається на навантаження. Таким чином реалізується плавне регулювання швидкості двигуна, яскравості світлодіодів або потужності іншого електричного навантаження.

Важливою перевагою використання апаратних таймерів є мінімальне навантаження на процесорне ядро мікроконтролера. Генерація PWM-сигналу відбувається апаратно, тому центральний процесор може паралельно виконувати інші задачі:

Отже, використання апаратних таймерів Arduino Uno 3 дозволяє ефективно реалізувати систему широтно-імпульсного керування навантаженням із високою точністю та енергоефективністю. Генерація PWM-сигналу за допомогою Timer1 забезпечує стабільну роботу силового MOSFET-ключа та можливість плавного регулювання потужності навантаження. Завдяки простоті програмування та наявності вбудованих апаратних модулів Arduino Uno 3 є ефективною платформою для реалізації систем PWM-керування у електронних пристроях.

2.4 Алгоритм роботи системи широтно-імпульсного керування

Алгоритм роботи системи визначає послідовність дій мікроконтролера: запуск, ініціалізація PWM-виводу, зчитування керуючого параметра, розрахунок коефіцієнта заповнення, формування PWM-сигналу та керування MOSFET-ключем.

Для підвищення надійності алгоритм може містити програмні обмеження максимального коефіцієнта заповнення. Це особливо важливо для індуктивних навантажень і двигунів постійного струму, де під час запуску можливий підвищений пусковий струм.

У разі перевищення встановленого порогу програма обмежує PWM-рівень або повністю вимикає силовий ключ, що зменшує ризик перегріву MOSFET та пошкодження джерела живлення.

Під час практичної реалізації алгоритму доцільно передбачити початкове встановлення коефіцієнта заповнення на нульовому рівні. Це унеможливорює випадкове ввімкнення навантаження одразу після подачі живлення. Після ініціалізації мікроконтролер переходить до основного циклу, у якому виконується зчитування керуючого параметра, його масштабування до діапазону 0...255 та передавання отриманого значення у функцію формування PWM-сигналу.

Основою системи є мікроконтролер Arduino Uno 3, який формує керуючий PWM-сигнал на основі встановлених параметрів коефіцієнта заповнення. Сформований сигнал подається на затвор MOSFET-транзистора IRLZ44N, який працює у ключовому режимі та здійснює комутацію електричного навантаження.

Після подачі живлення Arduino Uno 3 налаштовує GPIO-виводи, параметри таймерів і початкове значення PWM. У циклі програма оновлює коефіцієнт заповнення та за потреби контролює аварійні режими.

Після завершення ініціалізації система переходить у безперервний цикл

роботи. У циклі мікроконтролер постійно зчитує параметри керування. Джерелом вхідних даних може бути потенціометр, кнопки керування, датчики або інші зовнішні сигнали. Отримані аналогові або цифрові значення використовуються для визначення необхідного режиму роботи системи.

На наступному етапі виконується обробка отриманих даних та розрахунок коефіцієнта заповнення PWM-сигналу. Аналогове значення перетворюється у цифровий діапазон від 0 до 255, який використовується у функції `analogWrite()` для формування широтно-імпульсного сигналу. При цьому виконується перевірка допустимих меж значення коефіцієнта заповнення для забезпечення стабільної роботи системи.

Розраховане значення записується у регістр порівняння таймера OCR1A. Після оновлення регістрів апаратний таймер автоматично формує PWM-сигнал із заданим коефіцієнтом заповнення. Частота сигналу залишається сталою, а змінюється лише тривалість активного імпульсу.

PWM-сигнал із виводу D9 подається через резистор на затвор MOSFET-транзистора. При високому рівні сигналу транзистор відкривається, і струм проходить через навантаження. При низькому рівні сигналу MOSFET закривається, а струм припиняється. Таким чином шляхом швидкого перемикавання транзистора змінюється середнє значення потужності, що подається на навантаження. У процесі роботи система постійно виконує контроль свого стану. Перевіряються можливі аварійні ситуації:

Після завершення ініціалізації мікроконтролер переходить у основний цикл роботи програми. У циклі постійно виконується формування PWM-сигналу та оновлення параметрів керування відповідно до заданого алгоритму.

На рисунку 2.5 зображено алгоритм роботи системи широтно-імпульсного керування.

Однією з важливих особливостей алгоритму є використання апаратної генерації PWM.

Це дозволяє мінімізувати навантаження на центральне ядро мікроконтролера та забезпечити стабільність параметрів сигналу незалежно від

виконання інших програмних задач.

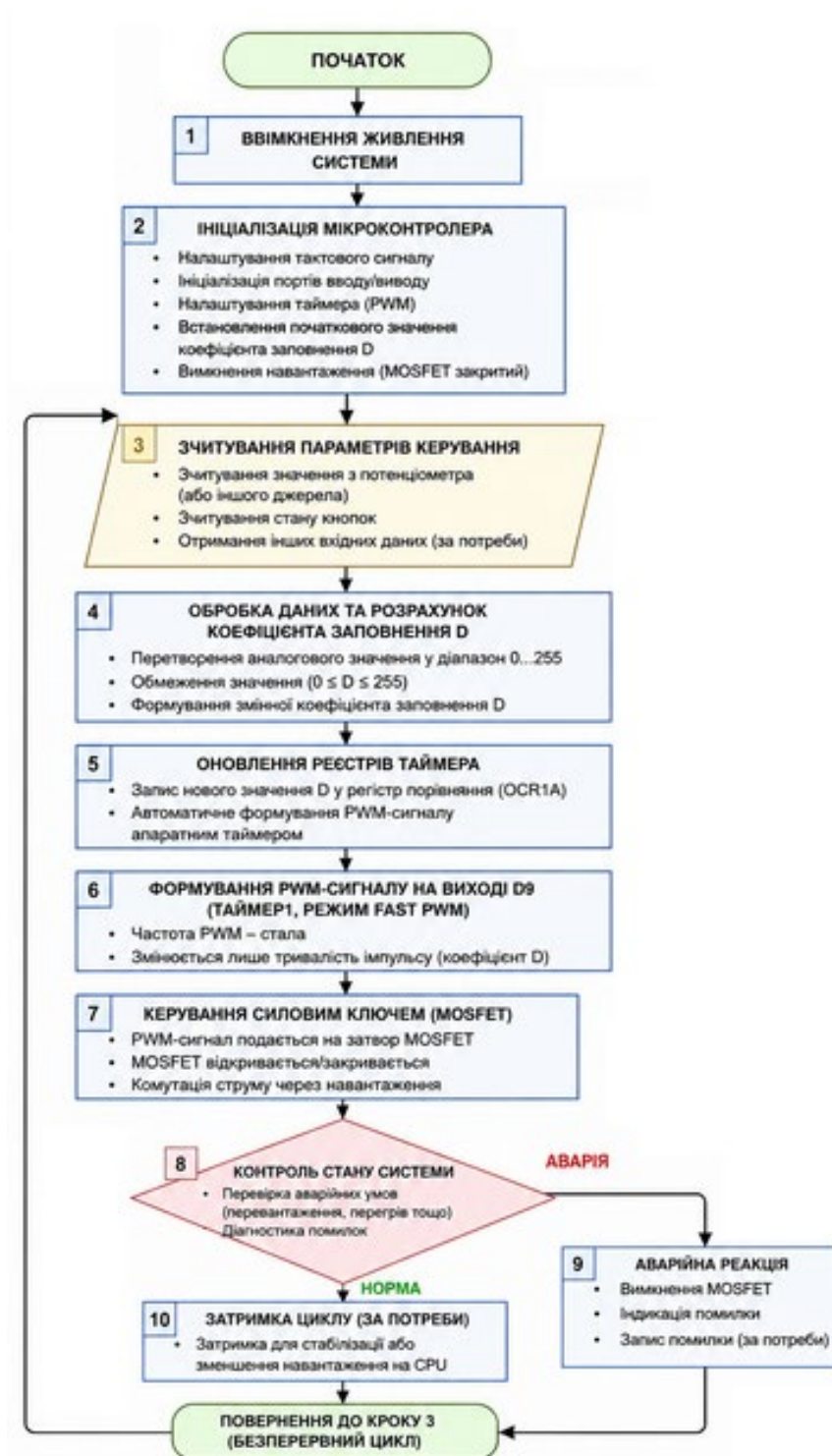


Рисунок 2.5 – Алгоритм роботи системи широтно-імпульсного керування

У системах керування електричними навантаженнями важливим параметром є частота PWM-сигналу. Занадто низька частота може викликати

пульсації струму та шум, тоді як надто висока призводить до збільшення втрат на перемикання транзистора. Тому частота PWM обирається з урахуванням типу навантаження та параметрів силового MOSFET-транзистора.

Якщо система працює у нормальному режимі, цикл повторюється знову зі зчитування параметрів керування. За необхідності може виконуватись коротка затримка для стабілізації роботи або зменшення навантаження на процесор.

У випадку виникнення аварійної ситуації система переходить у режим захисту. MOSFET-транзистор вимикається, навантаження відключається, а користувач отримує сигнал про помилку. За необхідності інформація про аварію може записуватись у пам'ять або передаватись через інтерфейс зв'язку.

Таким чином, алгоритм роботи системи широтно-імпульсного керування забезпечує ефективне регулювання потужності електричного навантаження шляхом зміни коефіцієнта заповнення PWM-сигналу. Використання апаратних таймерів Arduino Uno 3 та MOSFET-транзистора IRLZ44N дозволяє реалізувати енергоефективну систему керування із високою стабільністю роботи та мінімальними втратами енергії.

РОЗДІЛ 3

РОЗРАХУНКОВА АПАРАТНО-ПРОГРАМНА ЧАСТИНА

3.1 Розрахунок параметрів PWM-керування навантаженням

Розрахунок параметрів PWM-керування є важливим етапом проектування системи керування електричним навантаженням, оскільки саме від параметрів PWM-сигналу залежить стабільність роботи системи, рівень втрат потужності та ефективність керування навантаженням. У даному підрозділі виконано розрахунок основних параметрів PWM-керування для системи на базі Arduino Uno 3 та MOSFET-транзистора IRLZ44N.

Для розрахунку параметрів системи використано технічні характеристики елементів системи, наведені у таблиці 3.1.

Таблиця 3.1 – Вихідні параметри системи PWM-керування

Параметр	Позначення	Значення
Напруга живлення	$U_{\text{жив}}$	12 В
Тактова частота АТmega328Р	f_{clk}	16 МГц
Коефіцієнт дільника	N	64
Максимальне значення таймера	TOP	255
Опір навантаження	R_{H}	12 Ом
MOSFET-транзистор	—	IRLZ44N
Опір відкритого каналу	$R_{\text{DS(on)}}$	0,022 Ом

Проведемо розрахунок частоти PWM-сигналу, яка визначається параметрами апаратного таймера мікроконтролера згідно формули (3.1):

$$f_{\text{PWM}} = f_{\text{clk}} / (N(1 + \text{TOP})), \quad (3.1)$$

де f_{clk} – тактова частота мікроконтролера;

- N – коефіцієнт дільника частоти;
- TOP – максимальне значення таймера.

Підставимо значення: $f_{\text{PWM}} = 16\,000\,000 / (64 \cdot (255 + 1)) = 967 \text{ Гц}$.

Отримане значення частоти забезпечує стабільне PWM-керування навантаженням та нормальну роботу MOSFET-транзистора.

Здійснимо розрахунок періоду PWM-сигналу. Період PWM-сигналу визначається як величина, обернена до частоти: $T = 1/967 = 1,024 \text{ мс}$. Таким чином, тривалість одного циклу PWM-сигналу становить приблизно 1 мс.

Проведемо розрахунок коефіцієнта заповнення PWM, який визначає співвідношення часу активного стану сигналу до повного періоду PWM згідно формули (3.2):

$$D = \frac{t_{\text{on}}}{T} 100\%. \quad (3.2)$$

Для Arduino Uno значення PWM задається у межах від 0 до 255. Коефіцієнт заповнення визначається згідно аформули (3.3):

$$D = \frac{\text{PWM}}{255} 100\%. \quad (3.3)$$

Розрахуємо коефіцієнт заповнення для декількох значень PWM.

Для PWM = 64: $D = 64/255 \cdot 100\% \approx 25\%$.

Для PWM = 128: $D = 128/255 \cdot 100\% \approx 50\%$.

Для PWM = 192: $D = 192/255 \cdot 100\% \approx 75\%$.

Розрахунок середньої напруги на навантаженні. Середня напруга на навантаженні залежить від коефіцієнта заповнення PWM-сигналу і визначається згідно формули (3.4):

$$U_{\text{avg}} = D \cdot U_{\text{max}}. \quad (3.4)$$

Для коефіцієнта заповнення 25 %: $U_{avg}=0,25 \cdot 12=3$ В.

Для коефіцієнта заповнення 50 %: $U_{avg}=0,5 \cdot 12=6$ В.

Для коефіцієнта заповнення 75 %: $U_{avg}=0,75 \cdot 12=9$ В.

Розрахуємо струму навантаження, який визначається згідно закону Ома формула (3.5):

$$I = \frac{U}{R}. \quad (3.5)$$

Для напруги 12 В та навантаження 12 Ом: $I=12/12=1$ А.

Для режиму 50 % PWM: $I=6/12=0,5$ А.

Потужність навантаження визначається згідно (3.6):

$$P=U \cdot I. \quad (3.6)$$

Для максимального режиму: $P=12 \cdot 1=12$ Вт.

Для режиму 50 % PWM: $P=6 \cdot 0,5=3$ Вт.

Втрати потужності на MOSFET у відкритому стані визначаються згідно формули (3.7):

$$P_{loss}=I^2 \cdot R_{DS(on)}. \quad (3.7)$$

Підставимо значення: $P_{loss}=12 \cdot 0,022=0,022$ Вт.

Отримані втрати є незначними, що підтверджує високу ефективність використання MOSFET IRLZ44N у системі PWM-керування.

У таблиці 3.2 подано результати розрахунків.

Проведені розрахунки показують, що система PWM-керування забезпечує плавне регулювання потужності навантаження шляхом зміни коефіцієнта заповнення сигналу.

Таблиця 3.2 – Основні параметри PWM-керування

PWM	Коефіцієнт заповнення	Середня напруга	Струм	Потужність
64	25 %	3 В	0,25 А	0,75 Вт
128	50 %	6 В	0,5 А	3 Вт
192	75 %	9 В	0,75 А	6,75 Вт
255	100 %	12 В	1 А	12 Вт

Отримані значення підтверджують ефективність використання MOSFET-транзистора IRLZ44N та апаратного PWM-керування на базі Arduino Uno 3.

3.2 Розрахунок силовій частини системи керування

У цьому підрозділі виконується розрахунок силовій частини системи PWM-керування навантаженням. Основними елементами силовій частини є MOSFET-транзистор IRLZ44N, навантаження постійного струму, захисний діод, резистор у колі затвора та джерело живлення.

Вихідні дані для розрахунку подані в таблиці 3.3.

Таблиця 3.3 – Вихідні параметри силовій частини

Параметр	Позначення	Значення
Напруга живлення	$U_{\text{жив}}$	12 В
Опір навантаження	$R_{\text{н}}$	12 Ом
MOSFET-транзистор	VT1	IRLZ44N
Опір відкритого каналу	$R_{\text{DS(on)}}$	0,022 Ом
Напруга керування затвором	U_{GS}	5 В
Резистор затвора	R_{g}	100 Ом
Підтягуючий резистор	R_{pd}	10 кОм
Захисний діод	VD1	1N5819 / 1N4007

Розрахунок струму навантаження. Струм навантаження визначається за законом Ома згідно формули (3.5). Підставимо значення: $I_H = 12/12 = 1$ А.

Отже, максимальний струм навантаження становить 1 А.

Розрахунок потужності навантаження. Потужність, що споживається навантаженням, визначається: $P_H = U_{\text{жив}} \cdot I_H = 12$ Вт.

Отже, максимальна потужність навантаження становить 12 Вт.

Розрахунок втрат потужності на MOSFET. MOSFET-транзистор у відкритому стані має малий опір каналу $R_{DS(on)}$. Втрати потужності на транзисторі визначаються: $P_{VT} = I^2 \cdot R_{DS(on)} = 0,022$ Вт.

Отримане значення є дуже малим, тому MOSFET IRLZ44N може працювати без значного нагрівання при струмі навантаження 1 А.

Розрахунок спаду напруги на MOSFET. Спад напруги на відкритому MOSFET визначається: $U_{DS} = I_H \cdot R_{DS(on)} = 1 \cdot 0,022 = 0,022$ В.

Спад напруги становить лише 0,022 В, що підтверджує високу ефективність силового ключа.

Розрахунок ККД силової частини. Коефіцієнт корисної дії силової частини можна оцінити за формулою (3.8):

$$\eta = (P_H / (P_H + P_{VT})) \cdot 100\% . \quad (3.8)$$

$$\text{Звідси, } \eta = (12 / (12 + 0,022)) \cdot 100\% \approx 99,8\%.$$

Отже, ККД силової частини становить приблизно 99,8 %.

Розрахунок резистора затвора MOSFET. Резистор R_g встановлюється між PWM-виходом Arduino та затвором MOSFET. Він обмежує імпульсний струм заряду ємності затвора та захищає вихід мікроконтролера. При напрузі керування 5 В та резисторі 100 Ом максимальний імпульсний струм становить:

$$I_g = U_{\text{PWM}} / R_g = 5 / 100 = 0,05 \text{ А} = 50 \text{ мА}.$$

Оскільки цей струм має короткочасний імпульсний характер, резистор 100 Ом є допустимим для захисту GPIO-виходу Arduino.

Розрахунок підтягуючого резистора. Підтягуючий резистор R_{pd}

підключається між затвором MOSFET та землею. Його призначення – забезпечити закритий стан транзистора при відсутності PWM-сигналу.

Струм через резистор: $I_{pd} = U_{GS} / R_{pd} = 5 / 10000 = 0,0005 \text{ A} = 0,5 \text{ mA}$.

Отже, резистор 10 кОм не створює значного навантаження на вихід Arduino та забезпечує стабільне закриття MOSFET. У таблиці 3.4 подано результати розрахунку силової частини.

Таблиця 3.4 – Результати розрахунку силової частини

Параметр	Результат
Струм навантаження	1 А
Потужність навантаження	12 Вт
Втрати на MOSFET	0,022 Вт
Падіння напруги на MOSFET	0,022 В
ККД силової частини	99,8 %
Струм через резистор затвора	50 мА
Струм через підтягуючий резистор	0,5 мА

Проведені розрахунки показали, що силова частина системи PWM-керування має високий коефіцієнт корисної дії та незначні втрати потужності. MOSFET-транзистор IRLZ44N забезпечує ефективну комутацію навантаження при струмі 1 А, а використання резисторів R_g підвищує стабільність та надійність роботи схеми. Захисний діод VD1 забезпечує захист транзистора при роботі з індуктивними навантаженнями.

3.3 Програмна реалізація системи PWM-керування

Для керування навантаженням використовується функція `analogWrite()`, яка задає значення PWM у діапазоні від 0 до 255. Значення 0 відповідає вимкненому навантаженню, 255 – повному ввімкненню, а проміжні значення забезпечують плавне регулювання середньої потужності. Такий підхід дозволяє змінювати режим роботи системи без зміни апаратної частини, що є важливою

перевагою мікроконтролерного керування.

Програмна частина системи реалізується у середовищі Arduino IDE мовою, сумісною з C/C++. Основою програми є два обов'язкові блоки: `setup()` та `loop()`.

У блоці `setup()` виконується налаштування PWM-виводу як виходу, встановлення початкового рівня сигналу та, за потреби, запуск послідовного порту для налагодження.

У блоці `loop()` постійно виконується зчитування керуючого сигналу, обробка значення та формування PWM-сигналу на виводі D9.

Програмна реалізація системи PWM-керування є завершальним етапом розробки мікроконтролерної системи керування навантаженням. Вона забезпечує формування керуючого PWM-сигналу, зміну коефіцієнта заповнення, передавання сигналу на силовий MOSFET-ключ та контроль режимів роботи навантаження. У даній роботі програмна частина реалізована для плати Arduino Uno 3, яка є зручною платформою для створення навчальних і прикладних систем керування завдяки простоті програмування, наявності PWM-виходів і підтримці середовища Arduino IDE.

У попередніх підрозділах було обґрунтовано використання Arduino Uno 3 як основного керуючого пристрою та MOSFET-транзистора як силового ключа для комутації навантаження. Відповідно до розрахунків, PWM-сигнал доцільно подавати з цифрового виводу D9, оскільки цей вивід підтримує апаратну генерацію PWM. Силовий ключ працює за схемою low-side switching, коли навантаження підключається до джерела живлення, а MOSFET комутує його нижнє плече відносно загальної шини GND.

Основним завданням програмного забезпечення є формування прямокутного імпульсного сигналу зі змінним коефіцієнтом заповнення. Для цього використовується функція `analogWrite(pin, value)`, де `pin` – номер PWM-виводу, а `value` – числове значення від 0 до 255. Значення 0 відповідає вимкненому навантаженню, 255 – повному ввімкненню, а проміжні значення забезпечують регулювання середньої потужності. Для Arduino Uno 3 PWM

підтримується на виводах 3, 5, 6, 9, 10 та 11; типова частота для більшості PWM-виводів становить близько 490 Гц, а для виводів 5 і 6 – близько 980 Гц .

Програмна реалізація складається з таких основних етапів: ініціалізація виводів мікроконтролера, встановлення початкового значення PWM, зчитування керуючого сигналу, перетворення вхідного значення у діапазон 0...255, подавання PWM-сигналу на MOSFET-ключ і повторення циклу керування. Така структура дозволяє забезпечити плавне регулювання потужності навантаження без складної апаратної реалізації. На рисунку 3.6 подано блок-схему програмної реалізації системи PWM-керування.

На рисунку 3.6 наведено алгоритм роботи програмної частини системи PWM-керування. Після запуску Arduino Uno виконується ініціалізація PWM-виводу D9, який використовується для керування затвором MOSFET-транзистора. Далі програма зчитує керуючий сигнал, наприклад із потенціометра або кнопок, перетворює його у цифрове значення PWM у межах від 0 до 255 і передає це значення у функцію `analogWrite()`.

У результаті на виході D9 формується імпульсний сигнал із відповідним коефіцієнтом заповнення. MOSFET-ключ змінює тривалість увімкненого стану навантаження, що забезпечує регулювання його середньої потужності.

Для практичної реалізації можна використати код програми поданої нижче у лістингу 2.1.

Лістинг 2.1 – Код програми для практичної реалізації

```
const int pwmPin = 9;           // PWM-вихід Arduino Uno
const int controlPin = A0;      // аналоговий вхід для потенціометра
int sensorValue = 0;
int pwmValue = 0;
void setup() {
    pinMode(pwmPin, OUTPUT);
    analogWrite(pwmPin, 0);
}
void loop() {
    sensorValue = analogRead(controlPin);
    // Перетворення діапазону 0-1023 у діапазон 0-255
    pwmValue = map(sensorValue, 0, 1023, 0, 255);
```

```
// Формування PWM-сигналу на виводі D9
analogWrite(pwmPin, pwmValue);
delay(10);
}
```

Кінець лістингу 2.1

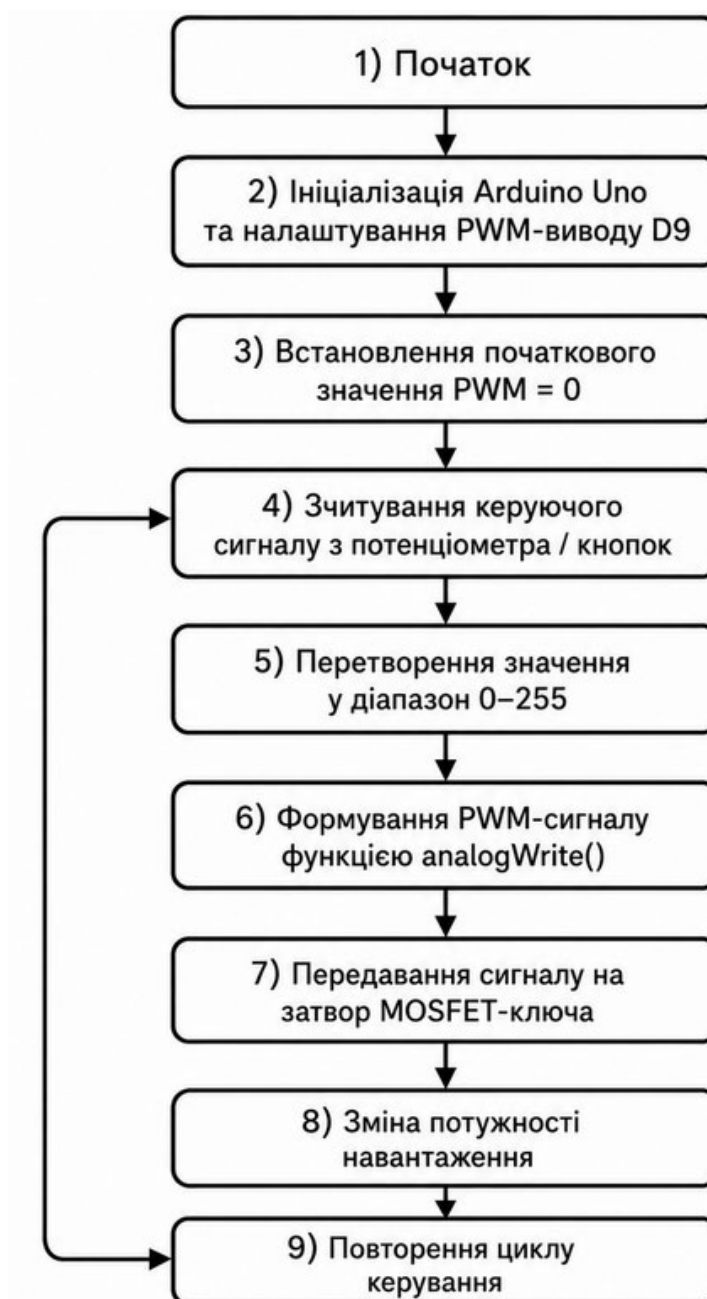


Рисунок 3.6 – Блок-схема програмної реалізації системи PWM-керування

У наведеному коді аналоговий вхід A0 використовується для зчитування напруги з потенціометра або іншого керуючого елемента. Оскільки аналого-цифровий перетворювач Arduino Uno 3 формуює значення в діапазоні

від 0 до 1023, це значення перетворюється у діапазон 0...255, який використовується функцією `analogWrite()`. Після цього PWM-сигнал подається на вивід D9, підключений до затвора MOSFET-ключа через резистор. Саме такий підхід відповідає попередньо розробленій схемі силового керування навантаженням.

Рисунок 3.7 відображає логіку взаємодії програмної та апаратної частин системи.



Рисунок 3.7 – Логічна структура програмного забезпечення системи

Керуючий елемент формує вхідний сигнал, який зчитується мікроконтролером. Далі програмний алгоритм обробляє отримане значення та

задає відповідний коефіцієнт заповнення PWM-сигналу. Сформований сигнал надходить на MOSFET-ключ, який виконує силову комутацію навантаження.

Таким чином, програмна частина не подає живлення на навантаження безпосередньо, а лише формує керуючий сигнал для силового каскаду.

Для захисту системи та підвищення стабільності роботи доцільно передбачити програмне обмеження максимального коефіцієнта заповнення. Наприклад, якщо навантаження має значну потужність або під час запуску виникає підвищений струм, максимальне значення PWM можна обмежити не 255, а, наприклад, 220...230.

Це зменшує навантаження на MOSFET-транзистор і джерело живлення:
`pwmValue = constrain(pwmValue, 0, 230); analogWrite(pwmPin, pwmValue);`

Також у програмі можна реалізувати плавний запуск навантаження. Це особливо корисно для двигунів постійного струму або інших інерційних навантажень, оскільки різке ввімкнення може спричинити значний пусковий струм (лістинг 2.2).

Лістинг 2.2 – Код плавного запуску навантаження

```
for (int i = 0; i <= pwmValue; i++) {
    analogWrite(pwmPin, i);
    delay(5);
}
```

Кінець лістингу

Застосування плавного наростання PWM-сигналу дозволяє зменшити електричні та теплові перевантаження силового ключа. Крім того, програмне керування дає змогу легко змінювати режим роботи системи без перероблення апаратної частини. Це є однією з головних переваг мікроконтролерного PWM-керування порівняно з аналоговими або релейними способами регулювання.

Узагальнено програмна реалізація системи забезпечує такі функції:

- ініціалізацію PWM-виводу Arduino Uno;
- зчитування вхідного керуючого сигналу;
- масштабування значення до діапазону 0...255;

- формування PWM-сигналу з потрібним коефіцієнтом заповнення;
- передавання керуючого сигналу на MOSFET-ключ;
- плавну зміну потужності навантаження;
- можливість програмного обмеження максимального рівня потужності.

Таким чином, програмна частина є зв'язувальною ланкою між користувачем, мікроконтролером і силовою частиною пристрою. Вона забезпечує гнучкість керування, стабільність роботи та можливість подальшого вдосконалення системи, наприклад шляхом додавання датчиків струму, температурного контролю або автоматичного регулювання за зворотним зв'язком.

Отже, у підрозділі розроблено програмну реалізацію системи PWM-керування навантаженням на базі Arduino Uno 3. Визначено основні етапи роботи програми: ініціалізація PWM-виводу, зчитування керуючого сигналу, перетворення його у значення коефіцієнта заповнення та формування PWM-сигналу за допомогою функції `analogWrite()`. Показано, що використання PWM-виводу D9 дозволяє ефективно керувати MOSFET-ключем, який здійснює силову комутацію навантаження.

Запропонований алгоритм забезпечує плавне регулювання потужності навантаження, можливість програмного обмеження максимального рівня PWM та реалізацію плавного запуску. Це підвищує стабільність роботи системи, зменшує пускові перевантаження та відповідає загальній структурі розробленої мікроконтролерної системи широтно-імпульсного керування.

РОЗДІЛ 4

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА

4.1 Електробезпека та захист від короткого замикання в мікроконтролерній системі керування

Під час розробки та експлуатації системи PWM-керування необхідно враховувати електричні ризики: коротке замикання, перевантаження MOSFET, перегрів силового ключа, неправильну полярність підключення та імпульсні перенапруги [19].

Окрему увагу слід приділяти тепловому режиму MOSFET. Навіть за високої енергоефективності PWM-керування частина енергії розсіюється у вигляді тепла через опір відкритого каналу та втрати під час перемикавання. Якщо температура корпусу транзистора зростає, опір каналу може збільшуватися, що призводить до додаткових втрат. Тому для стабільної роботи системи необхідно контролювати нагрівання та за потреби застосовувати радіатор або обмеження максимального PWM-рівня.

Коротке замикання у силовому колі є найбільш небезпечним аварійним режимом, оскільки струм може різко перевищити допустимі значення для MOSFET-транзистора, друкованих доріжок і джерела живлення. Тому при проєктуванні необхідно передбачати захист на рівні схеми, монтажу та програмного алгоритму. До базових заходів належать використання запобіжника, правильний вибір перерізу провідників, перевірка полярності живлення та застосування елементів із запасом по струму.

Захист силової частини забезпечується правильним вибором MOSFET, резистором затвора, підтягувальним резистором, захисним діодом для індуктивного навантаження, запобіжником та контролем температури силових елементів.

Схема на рисунку 4.1 демонструє використання MOSFET у ключовому режимі. Резистор затвора обмежує імпульсний струм керування, підтягувальний

резистор запобігає випадковому відкриванню транзистора, а захисний діод пригнічує імпульси самоіндукції.

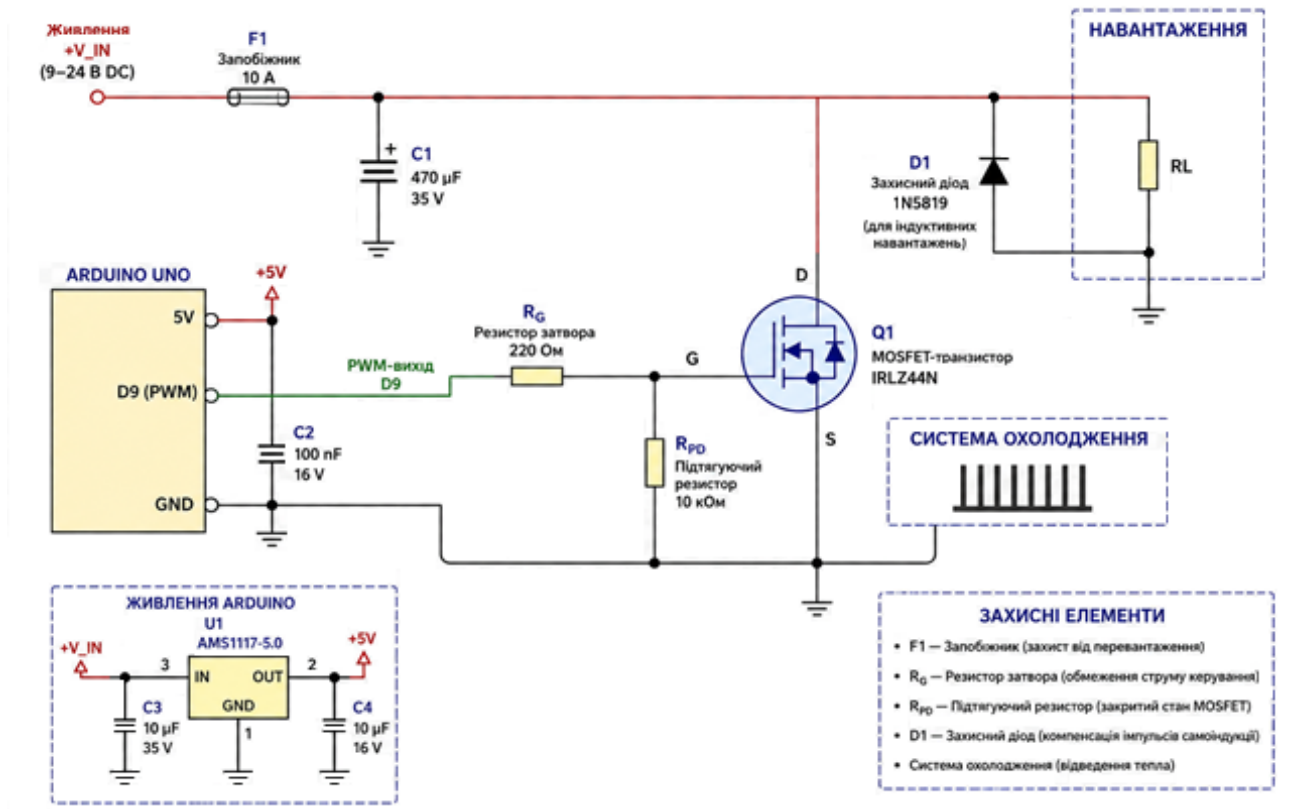


Рисунок 4.1 – Схема силового керування навантаженням на MOSFET-транзисторі [19]

У таблиці 4.1. подано потенційно небезпечні електричні параметри системи.

Таблиця 4.1 – Потенційно небезпечні електричні параметри

Елемент системи	Робоча напруга	Небезпечний фактор
Arduino Uno	5 В	Перевантаження портів
MOSFET	12–24 В	Перегрів, пробій
Джерело живлення	220/12 В	Ураження струмом
Навантаження	До 10 А	Коротке замикання

Для захисту Arduino Uno 3 необхідно використовувати стабілізоване джерело живлення, спільну землю GND, правильну полярність підключення та не перевищувати допустимий струм GPIO-виводів.

У таблиці 4.2 подано основні засоби захисту системи.

Таблиця 4.2 – Основні засоби захисту системи

Захисний елемент	Призначення
Плавкий запобіжник	Захист від перевантаження
TVS-діод	Захист від перенапруги
RC-фільтр	Зменшення імпульсних завад
Резистор затвора	Обмеження струму PWM
Радіатор	Охолодження MOSFET

У таблиці 4.3 подано теплові режими MOSFET.

Таблиця 4.3 – Теплові режими MOSFET

Струм навантаження	Температура без радіатора	Температура з радіатором
1 А	35 °C	30 °C
3 А	68 °C	45 °C
5 А	96 °C	58 °C
8 А	>120 °C	74 °C

Під час PWM-керування MOSFET нагрівається внаслідок провідникових і комутаційних втрат, тому доцільно застосовувати силові елементи із запасом по струму, радіатор, вентиляцію та періодичний контроль температури.

На рисунку 4.2 подано залежність температури MOSFET від коефіцієнта заповнення PWM.

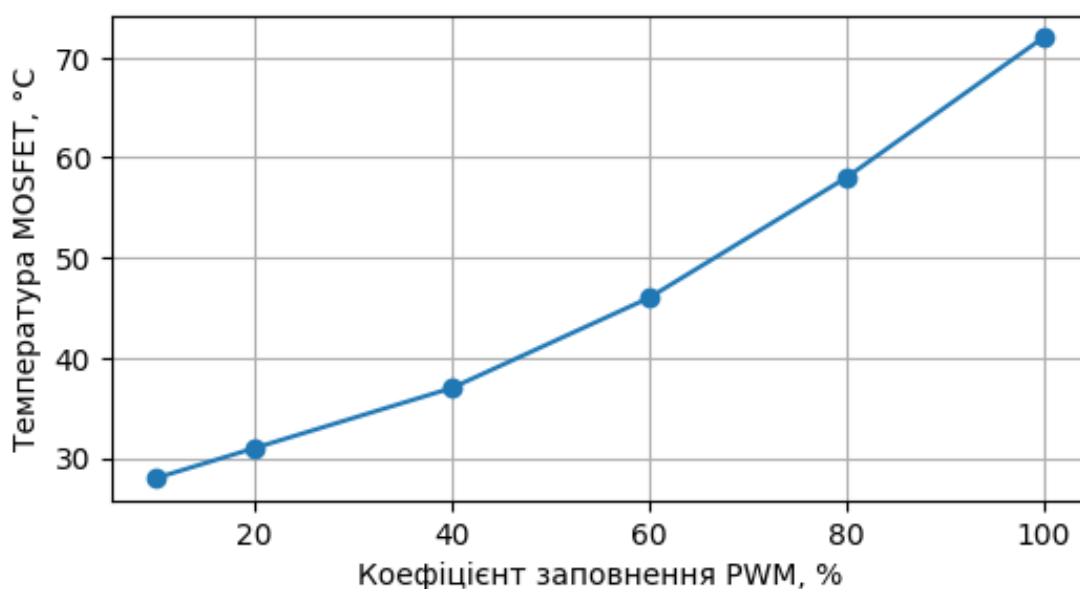


Рисунок 4.2 – Залежність температури MOSFET від коефіцієнта заповнення PWM

Особливу увагу необхідно приділяти правильному підключенню джерела живлення та дотриманню полярності, оскільки помилки можуть призвести до пошкодження мікроконтролера та силових елементів.

Отже, електробезпека системи PWM-керування забезпечується поєднанням схемних, апаратних і програмних засобів захисту, що зменшують ризик короткого замикання, перегріву та пошкодження мікроконтролера.

4.2 Ергономіка та безпечна організація робочого місця

Правильна організація робочого місця під час розробки, програмування та налагодження системи PWM-керування знижує ризик травмування, помилок монтажу, перевтоми оператора та пошкодження електронних компонентів [20]. Для тривалої роботи за комп'ютером необхідно дотримуватися ергономічних вимог: забезпечити достатнє освітлення, правильне положення монітора, підтримувати зручну посадку та робити короткі перерви. Під час пайки слід використовувати вентиляцію або витяжку, оскільки випари флюсу можуть негативно впливати на самопочуття. Дотримання цих вимог підвищує безпеку та

якість виконання роботи.

Під час роботи з мікроконтролерними платами та силовими елементами важливо підтримувати порядок на робочій поверхні. Проводи живлення та сигнальні лінії повинні бути розташовані так, щоб виключити випадкове замикання або механічне пошкодження. Вимірювальні прилади, паяльне обладнання та блоки живлення розміщують у доступних, але безпечних зонах. Це полегшує налагодження схеми та зменшує ймовірність помилок під час підключення.

Робоче місце повинно мати достатню площу для розміщення Arduino Uno 3, макетної або друкованої плати, лабораторного блока живлення, вимірювальних приладів, паяльного обладнання, комп'ютера та документації.

Для безпечної роботи необхідні достатнє освітлення, вентиляція, відсутність сторонніх металевих предметів, упорядковане розташування кабелів і використання засобів антистатичного захисту.

На рисунку 4.3 показано організацію робочого місця для проєктування мікроконтролерних систем.



Рисунок 4.3 – Організація робочого місця для проєктування мікроконтролерних систем

Під час монтажу та тестування слід виконувати роботи при вимкненому живленні, використовувати антистатичний браслет, ізольований інструмент, контролювати температуру паяльника та робити перерви під час тривалої роботи за комп'ютером.

Отже, дотримання ергономічних, електробезпечних і пожежних вимог забезпечує безпечну та ефективну роботу з мікроконтролерною системою PWM-керування.

ВИСНОВКИ

1. Проведено аналіз сучасних методів керування електричним навантаженням і встановлено, що найбільш ефективним для мікроконтролерних систем є PWM-керування, яке забезпечує точне регулювання потужності при мінімальних втратах енергії.

2. Досліджено принципи формування PWM-сигналів та показано, що зміна коефіцієнта заповнення дозволяє плавно змінювати середню потужність навантаження без зміни амплітуди напруги.

3. Виконано порівняльний аналіз Arduino Uno 3, STM32 та ESP32. Обґрунтовано вибір Arduino Uno 3 завдяки наявності PWM-виходів, простоті програмування та достатнім можливостям для навчально-прикладної системи.

4. Обґрунтовано використання MOSFET-транзистора як силового ключа, оскільки він має високу швидкодію, низькі втрати потужності та надійно працює в режимі PWM-керування.

5. Розроблено структурну, функціональну та принципову електричну схеми системи PWM-керування навантаженням на базі Arduino Uno 3, а також визначено взаємодію мікроконтролера, силового ключа, джерела живлення та навантаження.

6. Реалізовано алгоритм роботи системи, що передбачає ініціалізацію PWM-виходу, формування керуючого сигналу, подачу його на затвор MOSFET та зміну потужності навантаження відповідно до коефіцієнта заповнення.

7. Виконано розрахунок параметрів PWM-сигналу, режимів роботи MOSFET-ключа та потужності навантаження, що підтвердило правильність вибору елементної бази.

8. Розроблено програмне забезпечення для Arduino Uno 3 у середовищі Arduino IDE, яке забезпечує генерацію PWM-сигналу та керування силовим ключем за допомогою функції `analogWrite()`.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Rashid M. H., Krein P. T. Power Electronics Handbook. 5th ed. Oxford: Butterworth-Heinemann, 2023. 1664 p. URL: <https://surl.lt/pfaaqy> (дата звернення: 27.02.2026).
2. Arduino SA. Arduino Uno Rev3 Documentation. Turin: Arduino SA, 2026. URL: <https://docs.arduino.cc/hardware/uno-rev3> (дата звернення: 20.05.2026).
3. Microchip Technology Inc. ATmega328P Automotive Complete Datasheet. Chandler: Microchip Technology Inc., 2024. 660 p. URL: <https://www.microchip.com/en-us/product/atmega328p> (дата звернення: 20.02.2026).
4. Arduino SA. PWM Signal on Arduino. Turin: Arduino SA, 2025. URL: <https://docs.arduino.cc/learn/microcontrollers/analog-output> (дата звернення: 20.02.2026).
5. Arduino SA. analogWrite() – Arduino Reference. Turin: Arduino SA, 2025. URL: <https://docs.arduino.cc/language-reference/en/functions/analog-io/analogWrite/> (дата звернення: 27.03.2026).
6. Margolis M., Jepson B., Weldin N. Arduino Cookbook. 3rd ed. Sebastopol: O'Reilly Media, 2020. 800 p. URL: <https://www.oreilly.com/library/view/arduino-cookbook-3rd/9781491903513/> (дата звернення: 27.03.2026).
7. Monk S. Programming Arduino: Getting Started with Sketches. 3rd ed. New York: McGraw-Hill Education, 2022. 192 p. URL: <https://surl.lu/jfevwh> (дата звернення: 27.03.2026).
8. STMicroelectronics. RM0008 Reference Manual: STM32F101xx, STM32F102xx, STM32F103xx, STM32F105xx and STM32F107xx advanced Arm-based 32-bit MCUs. Geneva : STMicroelectronics, 2021. URL: <https://surl.lu/wtzfqv> (дата звернення: 27.03.2026).
9. Espressif Systems. ESP32 Technical Reference Manual. Version 5.7.

Shanghai: Espressif Systems, 2026. URL: <https://surl.li/ajamjo> (дата звернення: 27.03.2026).

10. Erickson R. W., Maksimović D. Fundamentals of Power Electronics. 3rd ed. Cham: Springer, 2020. 1081 p. URL: <https://surl.li/hzfgfg> (дата звернення: 27.03.2026).

11. Texas Instruments. MOSFET power losses and how they affect power-supply efficiency. Dallas: Texas Instruments, 2016. URL: <https://www.ti.com/lit/pdf/slyt664> (дата звернення: 17.04.2026).

12. Texas Instruments. AN-558 Introduction to Power MOSFETs and Their Applications. Dallas: Texas Instruments, 2011. URL: <https://www.ti.com/lit/pdf/snva008> (дата звернення: 17.04.2026).

13. Horowitz P., Hill W. The Art of Electronics. 3rd ed. Cambridge: Cambridge University Press, 2015. 1220 p. URL: <https://www.cambridge.org/9780521809269> (дата звернення: 27.05.2026).

14. Mohan N., Undeland T. M., Robbins W. P. Power Electronics: Converters, Applications, and Design. 3rd ed. Hoboken: John Wiley & Sons, 2003. 824 p. URL: <https://surl.li/vvzbjv> (дата звернення: 17.04.2026).

15. EasyEDA. EasyEDA Std User Guide. Shenzhen: EasyEDA, 2025. URL: <https://docs.easyeda.com/> (дата звернення: 20.04.2026).

16. EasyEDA. Schematic Capture. Shenzhen: EasyEDA, 2025. URL: <https://docs.easyeda.com/en/Introduction/Schematic-Capture/> (дата звернення: 20.04.2026).

17. Arduino SA. Getting Started with Arduino IDE 2. Turin: Arduino SA, 2024. URL: <https://docs.arduino.cc/software/ide-v2/tutorials/getting-started-ide-v2> (дата звернення: 20.04.2026).

18. Arduino SA. Arduino Language Reference. Turin: Arduino SA, 2025. URL: <https://docs.arduino.cc/language-reference/> (дата звернення: 25.05.2026).

19. Про затвердження Правил охорони праці під час експлуатації

електронно-обчислювальних машин: Наказ Держгірпромнагляду України від 26.03.2010 № 65. База даних «Законодавство України». Верховна Рада України. URL: <https://zakon.rada.gov.ua/go/z0293-10> (дата звернення: 25.05.2026).

20. Про затвердження Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці: Наказ Держнаглядохоронпраці України від 26.01.2005 № 15. База даних «Законодавство України». Верховна Рада України. URL: <https://zakon.rada.gov.ua/go/z0231-05> (дата звернення: 25.05.2026).