

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних робіт з дисципліни
«Електроніка та мікросхемотехніка у системах управління та
діагностики» частина 2 для студентів спеціальності
G5 – «Електроніка, електронні комунікації, приладобудування
та радіотехніка» всіх форм навчання

Методичні вказівки до виконання лабораторних робіт з дисципліни «Електроніка та мікросхемотехніка у системах управління та діагностики» частина 2 для студентів спеціальності G5 – «Електроніка, електронні комунікації, приладобудування та радіотехніка» всіх форм навчання /Укл.: Л.С. Скрупська, О.А. Сахно, О.О. Каплієнко – Запоріжжя: НУЗП, 2026.– 51с.

Укладачі: Л.С. Скрупська, ст. викл.
О.А. Сахно, доцент, к. т. н.
О.О. Каплієнко, доцент, к. т. н.

Рецензент: В.В.Василевський, доцент, к. т. н.

Відповідальний
за випуск: Л.С. Скрупська, ст. викл.

Затверджено
на засіданні кафедри
«Електричні та
електронні апарати»
Протокол № __
від «24» 12 2025 р.

Затверджено НМК ЕТФ
Протокол № ____
від « » 2026 р.

ЗМІСТ

Вступ.....	4
Лабораторна робота № 8. Ознайомлення з архітектурою мікроконтролера та програмування портів вводу/виводу	5
Лабораторна робота № 9. Дослідження характеристик АЦП та ЦАП.....	13
Лабораторна робота № 10. Дослідження імпульсних джерел живлення та ШІМ-регулювання	18
Лабораторна робота № 11. Підключення та програмування різних типів сенсорів до мікроконтролера	23
Лабораторна робота № 12. Управління двигунами постійного струму та кроковими двигунами за допомогою мікроконтролера.....	29
Лабораторна робота № 13. Реалізація зв'язку між двома мікроконтролерами або між мікроконтролером та ПК через UART/SPI	34
Лабораторна робота № 14. Розробка простої діагностичної системи для вимірювання та відображення параметрів	40
Лабораторна робота № 15. Дослідження сучасних мікроконтролерів на базі IoT для систем управління.....	45
Список використаних джерел посилання	51

ВСТУП

Метою виконання лабораторних робіт є закріплення на практиці теоретичного матеріалу з дисципліни «Пристрої систем силової електроніки та перетворювальної техніки» частина 1.

Мета лабораторного практикуму — закріплення теоретичних знань та набуття практичних навичок з дослідження функціональних вузлів, схем та режимів роботи ключових напівпровідникових приладів, які формують основу сучасних систем силової електроніки.

Лабораторні роботи Частини 1 зосереджені на вивченні принципів побудови перетворювальної техніки, аналізі схем комутації та керування, а також використанні сучасного програмного забезпечення (схемотехнічного моделювання) для оцінки характеристик та діагностики роботи електронних пристроїв. Успішне виконання робіт є необхідною передумовою для розуміння складніших інтегрованих систем управління.

Методичні вказівки містять опис 8 лабораторних робіт за темами дослідження основи електронних компонентів і приладів та їх застосування в аналогових підсилювальних і фільтраційних схемах, напівпровідникових діодів, біполярних транзисторів, тиристорів, електронних підсилювачів, операційних підсилювачів, логічних елементів та фільтрів.

Студент зобов'язаний вивчити теоретичний матеріал, виконати практичні завдання та зробити звіт з лабораторної роботи, який буде мати основні розділи:

- назва лабораторної роботи;
- її мета;
- схеми експериментів, таблиці, графіки, розрахунки, висновки.

Звіт має бути оформлений у відповідності до СТП 1596 і захищений.

ЛАБОРАТОРНА РОБОТА № 8

Ознайомлення з архітектурою мікроконтролера та програмування портів вводу/виводу

Мета роботи

Вивчити архітектуру портів вводу/виводу (GPIO) мікроконтролера. Зрозуміти призначення регістрів спеціального призначення (DDR, PORT, PIN). Набути практичних навичок схемотехнічного моделювання мікроконтролерних систем та написання програм мовою C для керування зовнішніми пристроями (світлодіоди, кнопки) у середовищі **Electronics Workbench**.

1. Обладнання та програмне забезпечення

Робота виконується дистанційно на персональному комп'ютері.

1. **Програмне забезпечення для схемотехнічного моделювання: Electronics Workbench (EWB) / NI Multisim.**
 - *Примітка:* EWB зазвичай не містить вбудованого компілятора C. Програмування (лістинг коду) буде виконуватися окремо, а імітація логіки на портах — у середовищі EWB за допомогою логічних перемикачів і пробників для верифікації роботи.
2. **Об'єкт дослідження (Віртуальний):** Мікроконтролер ATmega328P (або еквівалентний 8-бітний МК, який можна імітувати на рівні портів).
3. **Елементи EWB:**
 - Логічні перемикачі (Logic Switches).
 - Логічні пробники (Logic Probes / LEDs).
 - Генератор імпульсів (Clock Source) або джерело живлення (+5V).

2. Завдання лабораторної роботи

1. **Керування виходом (Імітація блимання):** Імітувати програму, яка по черзі вмикає та вимикає світлодіод.
2. **Керування входом (Імітація опитування):** Імітувати роботу МК, який вмикає світлодіод тільки при натиснутій кнопці.
3. **Бітові зсуви ("Вогні, що біжать"):** Імітувати роботу

програми, що циклічно зсуває логічну одиницю по 8 виводах порту.

3. Теоретичні відомості

Порти вводу/виводу (GPIO) керуються трьома регістрами, які є комітками пам'яті всередині МК. Ми будемо імітувати їхню роботу, задаючи вхідні та вихідні дані вручну в EWB.

1. **DDR_x (Data Direction Register)**: Визначає, чи є пін **входом** (0) чи **виходом** (1).
2. **PORT_x (Data Register)**: Задає стан виходу.
3. **PIN_x (Input Register)**: Відображає реальний фізичний стан на вході.

Особливості імітації в EWB:

Оскільки EWB не емулює код C, ми будемо використовувати Логічні перемикачі та Логічні пробники як імітатори станів регістрів.

- **Для імітації виходу (Світлодіод)**: Ми налаштуємо вивід як $DDR_x = 1$. Логічний пробник у схемі буде відображати стан регістра $PORT_x$, який ми б змінювали в коді.
- **Для імітації входу (Кнопка)**: Ми налаштуємо вивід як $DDR_x = 0$. Логічний перемикач буде імітувати стан PIN_x (фізичне натискання кнопки).

4. Хід роботи

Завдання 1. Імітація програми блимання LED

Мета коду (який ми імітуємо):

1. Налаштувати **PB0** як вихід ($DDRB |= 1;$).
2. У циклі змінювати його стан: **УВИМКНУТИ** ($PORTB |= 1;$)
→**ВИМКНУТИ** ($PORTB \&= \sim 1;$).
3. **Складання схеми в EWB:**
 - Розмістіть на робочому полі 8-бітний порт (для імітації Port B), використовуючи 8 Логічних пробників (для виходів PB_0 до PB_7)(рис.1).
 - З'єднайте **PB0** (перший пробник) через струмообмежувальний резистор із джерелом живлення.

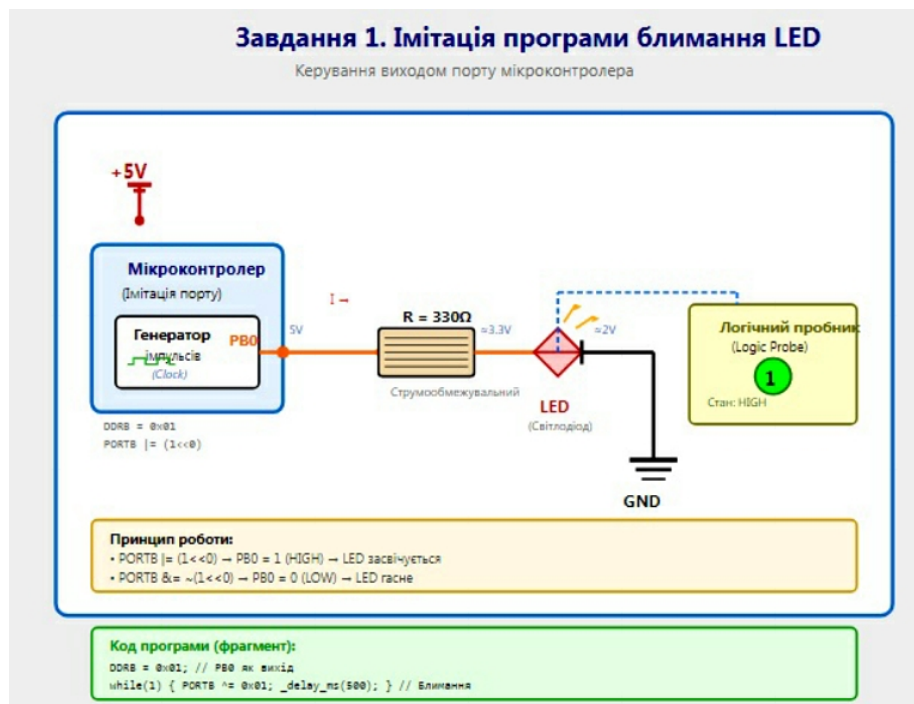


Рисунок 1 Схема EWB (імітація програми блимання світлодіода)

Опис схеми:

а) Джерело сигналу (Імітатор PORTB): Використовуємо **Логічний перемикач (Logic Switch)** або **Генератор імпульсів (Clock Source)**, підключений до джерела живлення (+5V). Цей елемент імітує біт PB0 регістра PORTB після його налаштування як вихід (DDRB=1).

б) Струмообмежувальний резистор: Резистор 330 Ом.

в) Навантаження: **Логічний пробник (Logic Probe)** або світлодіод.

г) Результат: При зміні стану перемикача на "1" (імітація команди $PORTB \mid = 1$), світлодіод засвічується. На "0" — гасне.

4. Імітація логіки (Безпосереднє керування):

- Оскільки EWB не виконує цикл, ви будете вручну імітувати два стани, які змінюються з часом.
- **Стан 1 (LED ON):** Запишіть в таблицю PORTB як 1 (що відповідає коду $PORTB \mid = 1$);
- **Стан 2 (LED OFF):** Запишіть в таблицю PORTB як 0 (що відповідає коду $PORTB \&= \sim 1$);

5. **Результат:** Підтвердьте, що при зміні логічного рівня (0 або 1) на піні, який імітує **PB0**, стан пробника (світлодіода) змінюється. Зафіксуйте цей принцип.

Завдання 2. Імітація опитування кнопки

Мета коду (який ми імітуємо):

1. Налаштувати **PB0** як вихід (LED).
2. Налаштувати **PD2** як вхід з Pull-up.
3. Перевіряти, чи PIN D є 0 (кнопка натиснута). Якщо так, увімкнути PORT B.
4. **Складання схеми в EWB:**
 - **Вихід (LED):** Вивід **PB0** (Логічний пробник).
 - **Вхід (Кнопка):** Вивід **PD2** (Логічний перемикач).
 - Кнопка підключена за схемою з внутрішнім Pull-up (тобто, коли кнопка не натиснута, вхід "1"; при натисканні (перемикач в 0), вхід "0"). (рис.2).

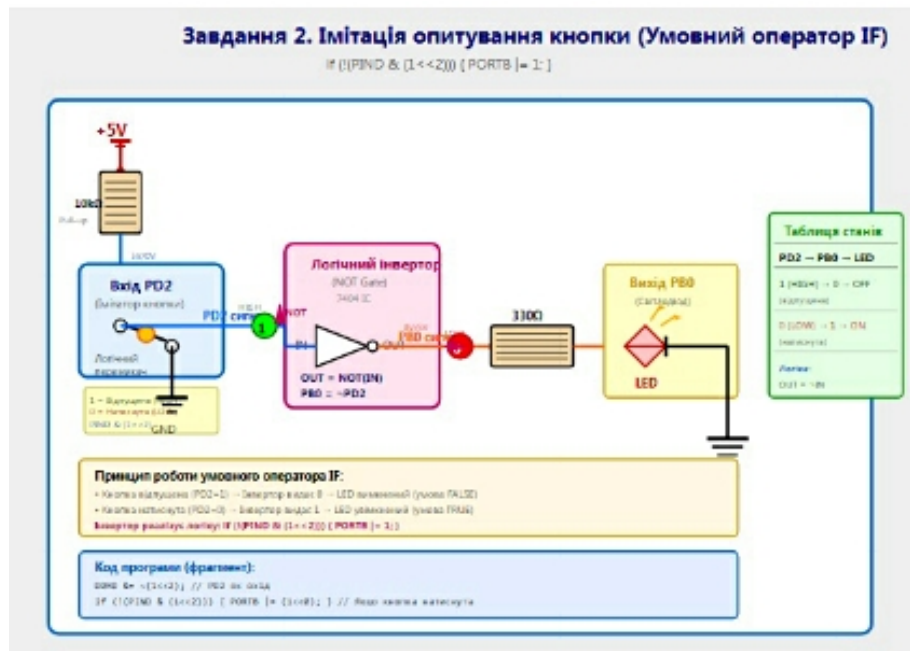


Рисунок 2 Схема EWB (імітація програми опитування кнопки).

Опис схеми:

- а) **Вхід (Імітатор кнопки PD2):** Використовуємо **Логічний перемикач (Logic Switch)**.

- Стан "1" імітує відпущену кнопку (через Pull-up).
- Стан "0" імітує натиснуту кнопку.
-

б) Умовний оператор (Імітатор IF): Використовуємо логічний елемент **Інвертор (NOT Gate)** (наприклад, з мікросхеми 7404). Інвертор перетворює "0" (натиснуто) на "1" і навпаки.

в) Вихід (LED PB0): До виходу інвертора підключаємо **Логічний пробник (світлодіод)** через струмообмежувальний резистор.

г) Результат:

- Коли перемикач **PD2** у стані **1** (кнопка відпущена) → Інвертор видає **0** → Світлодіод вимкнений.
- Коли перемикач **PD2** у стані **0** (кнопка натиснута) → Інвертор видає **1** → Світлодіод увімкнений.

5. Імітація логіки (Виконання умови):

- **Налаштування DDR/PORT:** Ми вважаємо, що в пам'яті МК встановлено: $DDRB = 1$ (PB0 - вихід) та $DDRD = 0$, $PORTD = 1$ (PD2 - вхід з Pull-up).
-
- **Перевірка:** Використовуючи логічний елемент **AND** або **NAND** (для імітації умови if), створіть схему, де вихід **PB0** буде логічною "1" тільки тоді, коли вхід **PD2** (стан кнопки) буде логічним "0".

-
- *Підказка:* Вихід = $\overline{PD2}$. Використовуйте інвертор.
-

6. Результат: Зафіксуйте стани перемикача (PD2) та індикатора (PB0) і заповніть таблицю.

7.

Ця таблиця відображає роботу вашої схеми в **Electronics Workbench**, що імітує умову if ($!(PIND \& (1 \ll 2))$). Вивід **PB0** (світлодіод) залежить від входу **PD2** (кнопка) після проходження через логіку (Інвертор).

Таблиця 1. Імітація опитування кнопки

Стан логічного перемикача (PD2) – Фізична кнопка	Логічний рівень на вході PD2 (PIN D)	Стан логічного елемента (Інвертор)	Стан світлодіода (Вихід PB0)	Коментар
1 (Відпущено)	1	0	0 (Вимкнений)	Умова PD2 не виконана
0 (Натиснуто)	0	1	1 (Увімкнений)	Умова PD2 виконана

Завдання 3. Імітація "Бігучих вогнів"

Мета коду (який ми імітуємо):

1. Налаштувати весь Порт D як вихід (DDRD = 0xFF).
2. У циклі зсувати одиницю від PD0 до PD7: PORTD = (1 << i);.
3. Складання схеми в EWB:
 - Використовуйте 8 Логічних пробників для відображення Порту D (PD₀ до PD₇).
 - Додаткові логічні елементи не потрібні.

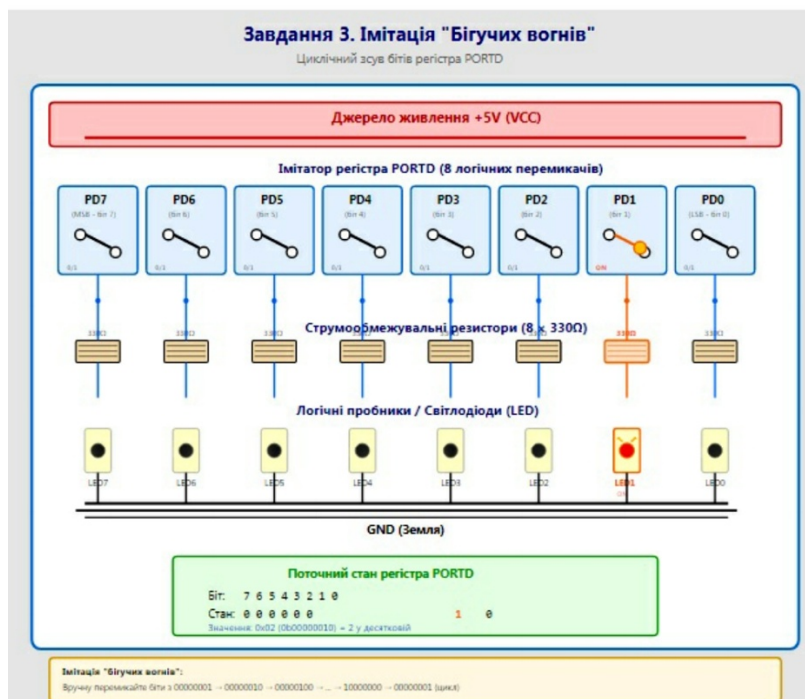


Рисунок 3 Схема імітації "Бігучих вогнів"

4. Імітація логіки (Ручний зсув):

- У цій частині ви будете вручну послідовно вводити стани регістра PORTD (відповідно до коду $PORTD = (1 \ll i)$);
- Наприклад, імітуємо 3-й крок ($i=2$): $PORTD = (1 \ll 2) = 00000100$. Ви вручну подаєте "1" тільки на 3-й пробник.

5. Результат: Заповніть таблицю станів, імітуючи зміну даних в регістрі PORTD і фіксуючи, який світлодіод світиться.

Ця таблиця відображає послідовну зміну значень 8-бітного регістра **PORTD**, що викликає ефект "бігучого вогню". Ви заповнюєте цю таблицю, вручну перемикаючи логічні пробники в EWB.

Таблиця 2. Імітація "Бігучих вогнів"

Крок (i)	Команда (Імітований код C)	Двійкове значення PORTD	Шістнадцяткове (Hex)	Активний світлодіод (Пробник)
0	$PORTD = (1 \ll 0)$;	0000 0001	0x01	PD ₀
1	$PORTD = (1 \ll 1)$;	0000 0010	0x02	PD ₁
2	$PORTD = (1 \ll 2)$;	0000 0100	0x04	PD ₂
3	$PORTD = (1 \ll 3)$;	0000 1000	0x08	PD ₃
4	$PORTD = (1 \ll 4)$;	0001 0000	0x10	PD ₄
5	$PORTD = (1 \ll 5)$;	0010 0000	0x20	PD ₅
6	$PORTD = (1 \ll 6)$;	0100 0000	0x40	PD ₆
7	$PORTD = (1 \ll 7)$;	1000 0000	0x80	PD ₇
8	$PORTD = (1 \ll 0)$;	0000 0001	0x01	PD ₀ (Повтор)

5. Вимоги до звіту

Звіт має бути надісланий у форматі PDF або DOCX і містити:

1. Назву, мету роботи.
2. Скріншоти схем із **Electronics Workbench** для Завдання 1 та Завдання 2 (із застосуванням логічних елементів для імітації if).
3. Заповнені таблиці істинності (Табл. 1, Табл. 2).
4. Лістинги програм на мові C (які імітувалися) з детальними коментарями, що пояснюють бітові операції.
5. Висновки.

6. Висновки до роботи

У висновках необхідно:

- Пояснити, як реалізується принцип керування виводами МК через маніпуляцію регістрами.
- Пояснити, як на рівні логічних елементів (в EWB) було імітовано умовний оператор if для керування світлодіодом кнопкою.
- Обґрунтувати важливість використання бітових зсувів для керування цілим портом (Завдання 3).

7. Контрольні питання

1. Яке двійкове значення потрібно записати в регістр DDRC, щоб виводи PC₇ та PC₆ були виходами, а PC₀ та PC₁ — входами?
2. Якщо ви налаштували ніжку як вхід (DDR_x = 0), що відбувається, коли ви записуєте в неї PORT_x = 1?
3. Поясніть, що означає бітова операція PORTB ^= (1 << 5); (операція XOR).
4. Чому пряме підключення світлодіода до ніжки мікроконтролера без резистора може призвести до пошкодження МК?
5. Чим відрізняється керування периферією через регістри (як у AVR) від використання високоуровневих бібліотечних функцій (як у Arduino)?

Список літератури

1. Бойко В. І., Жуйков В. Я., Зорі А. А. Цифрова схемотехніка та мікропроцесори. — К.: «Освіта України», 2016.
2. Ямненко Ю. С. Мікроконтролери та мікропроцесори в системах керування: Конспект лекцій. — К.: НТУУ «КПІ», 2020.
3. Жуйков В. Я., Лисенко О. І. Мікроконтролери: Навчальний посібник. — К.: НТУУ «КПІ», 2013.
4. Технічна документація (Datasheet) на мікроконтролер ATmega328P.

ЛАБОРАТОРНА РОБОТА № 9

Дослідження характеристик АЦП та ЦАП

Мета роботи

1. Вивчити принцип дії та структурну схему цифро-аналогового перетворювача на основі R-2R матриці.
2. Набути навичок моделювання роботи ЦАП та АЦП (у частині функціонального перетворення) у середовищі Electronics Workbench.
3. Експериментально визначити основні статичні характеристики перетворювачів: розрядність, крок квантування (LSB), функцію передачі.

1. Обладнання та програмне забезпечення

- **Програмне забезпечення:** Electronics Workbench (EWB) / NI Multisim.
- **Віртуальні вимірювальні прилади EWB:** Вольтметр, Логічні перемикачі (Logic Switches), Логічні пробники (Logic Probes).
- **Віртуальні компоненти:** Операційний підсилювач (Op-Amp), Прецизійні резистори (R, 2R), Джерело живлення (VCC), Джерело опорної напруги (V_{ref}).

2. Завдання лабораторної роботи

1. **Моделювання та дослідження ЦАП:** Скласти та дослідити схему 4-бітного ЦАП на основі R-2R матриці. Визначити експериментально крок квантування (LSB) та функцію передачі.
2. **Дослідження характеристики АЦП:** Імітувати функцію передачі 3-бітного АЦП та визначити його похибку квантування.

3. Теоретичні відомості

3.1. Цифро-аналоговий перетворювач (ЦАП)

ЦАП перетворює N-бітний цифровий код D у відповідну аналогову напругу V_{out} . Найпоширенішою є схема на основі **R-2R матриці**.

Вихідна напруга V_{out} визначається як сума вагових внесків

кожного біта:

$$V_{out} = V_{ref} \cdot \sum_{i=0}^{N-1} b_i \cdot 2^{i-N}$$

де V_{ref} — опорна напруга, N — розрядність (кількість біт), b_i — значення i -го біта (0 або 1).

Крок квантування (LSB, Least Significant Bit): Це мінімальна зміна вихідної напруги, яка виникає при зміні молодшого біта (b_0):

$$LSB = \frac{V_{ref}}{2^N}$$

3.2. Аналого-цифровий перетворювач (АЦП)

АЦП виконує зворотню функцію, перетворюючи аналогову напругу V_{in} у N -бітний цифровий код.

Функція передачі АЦП: Являє собою залежність вихідного цифрового коду від вхідної аналогової напруги. На відміну від ЦАП, вона має ступінчастий вигляд, оскільки діапазон вхідних напруг (сходинок) відображається в одне цифрове значення.

Похибка квантування: Неминуча похибка, що виникає через дискретність вихідного коду, становить $\pm 0.5 \cdot LSB$.

Хід роботи

Завдання 1. Дослідження 4-бітного ЦАП (R-2R)

1. Складання схеми в EWB:

- Використовуйте чотири пари резисторів R та $2R$. Рекомендовані номінали: $R=10$ кОм, $2R=20$ кОм.
- Налаштуйте опорну напругу $V_{ref} = +10$ В.
- Як елементи керування бітами ($b_0 - b_3$, де b_3 — MSB) використовуйте 4 **Логічні перемикачі (Logic Switches)**, які підключатимуть відповідний резистор $2R$ або до V_{ref} , або до GND (землі).
- На виході схеми встановіть Вольтметр. (рис. 1)

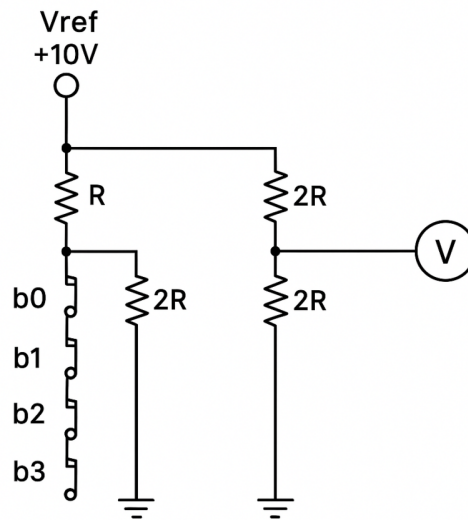


Рисунок 1 Схема дослідження 4-бітного ЦАП (R-2R).

2. Визначення теоретичного LSB: Розрахуйте теоретичний крок квантування для $N=4$ та $V_{ref}=10$ В.

$$LSB_{теор} = f_{рас} 10 \text{ В } 2^4 = f_{рас} 10 / 16 = 0.625 \text{ В}$$

3. Експериментальні виміри:

Заповніть Таблицю 1, послідовно вводячи всі 16 двійкових кодів від 0000 до 1111 за допомогою логічних перемикачів і фіксуючи значення V_{out} з Вольтметра.

Таблиця 1. Функція передачі ЦАП (4-біт, $V_{ref}=10$ В)

Десятковий код	Двійковий код ($b_3b_2b_1b_0$)	$V_{out, теор}$ (В)	$V_{out, експ}$ (В)	$\Delta V = V_{експ} - V_{теор}$ (В)
0	0000	0.000		
1	0001	0.625		
2	0010	1.250		
3	0011	1.875		
...		
15	1111	9.375		

Завдання 2. Імітація дослідження функції передачі АЦП (3-біт)

Оскільки повноцінний АЦП складно моделювати в EWB, ми імітуємо його функцію передачі за допомогою компараторів, що

відповідає Flash ADC.

1. Налаштування порогів (R-Ladder): Для 3-бітного АЦП ($N=3$) потрібно $2^3 - 1 = 7$ компараторів. Виберіть $V_{ref} = 8$ В. Тоді $LSB = 8$ В / $8 = 1$ В. Пороги спрацювання компараторів: 0.5 LSB, 1.5 LSB, 2.5 LSB, 3.5 LSB...

2. Визначення зон перетворення: Заповніть Таблицю 2, яка визначає, за якої вхідної напруги V_{in} має відбутися перетворення у відповідний цифровий код.

Таблиця 2. Функція передачі АЦП (3-біт, $V_{ref}=8$ В)

Десятковий код	Двійковий код	Мінімальна V_{in}	Максимальна V_{in}	V_{in} для тесту (Середина зони)	$V_{out, експ}$
0	000	0 В	0.5 В	0.25 В	
1	001	0.5 В	1.5 В	1.0 В	
2	010	1.5 В	2.5 В	2.0 В	
3	011	2.5 В	3.5 В	3.0 В	
4	100	3.5 В	4.5 В	4.0 В	
5	101	4.5 В	5.5 В	5.0 В	
6	110	5.5 В	6.5 В	6.0 В	
7	111	6.5 В	8.0 В	7.25 В	

3. Експеримент (тест у EWB):

- За допомогою джерела напруги (DC Voltage Source) встановлюйте значення V_{in} (наприклад, 0.25 В, 1.0 В, 2.0 В тощо).
- На макеті Flash ADC (або його логічній імітації) перевіряйте, чи збігається цифровий код (на логічних пробниках) з теоретичним значенням V_{out} для вибраної V_{in} .
- Визначте найбільшу відмінність між V_{in} і аналоговим еквівалентом отриманого коду.

Вимоги до звіту

Звіт повинен містити:

1. Титульний аркуш.
2. Мету роботи та завдання.

3. Короткий огляд теоретичних відомостей.
4. **Принципові схеми 4-бітного ЦАП (R-2R) та імітаційної схеми АЦП, виконані у EWB.**
5. Розрахунок теоретичного LSB для ЦАП.
6. Заповнені **Таблиці 1 та 2** з експериментальними даними.
7. **Графік функції передачі ЦАП** (V_{out} від двійкового коду).
8. **Висновки** до роботи.

Висновки до роботи

У висновках необхідно узагальнити отримані результати: чи підтвердився теоретичний крок квантування (LSB) для ЦАП, які основні джерела похибок (наприклад, неточність віртуальних резисторів у EWB), та чи відповідає ступінчаста функція передачі АЦП теоретичним очікуванням.

Контрольні питання

1. Яке призначення операційного підсилювача у схемі ЦАП R-2R?
2. Поясніть, що таке **диференціальна нелінійність (DNL)** та **інтегральна нелінійність (INL)** АЦП/ЦАП.
3. Чому у схемі R-2R використовуються резистори лише двох номіналів, а не зважені резистори (R, 2R, 4R...)?
4. Що таке похибка квантування і який її максимальний розмір?
5. Як зміниться крок LSB, якщо у 8-бітному ЦАП збільшити опорну напругу V_{ref} вдвічі?

Список літератури

1. Павленко В. О. *Промислова електроніка*. Підручник. — К.: Видавництво "Каравела", 2018. — 520 с.
2. Шаповал С. П., Шаповал М. С. *Мікропроцесорні системи: Елементи та пристрої*. Навчальний посібник. — Львів: Видавництво "Львівська політехніка", 2021. — 356 с.
3. Мокін Б. І., Мокін В. Б. *Мікропроцесорні системи управління та діагностики*. Підручник. — Вінниця: ВНТУ, 2019. — 450 с.
4. Водозазкін В. М., Кожухар М. В. *Електроніка та мікросхемотехніка*. Навчальний посібник. — Харків: Видавництво "ХАІ", 2020. — 310 с.

ЛАБОРАТОРНА РОБОТА № 10

Дослідження імпульсних джерел живлення та ШІМ-регулювання

Мета:

1. Вивчити принцип дії та структурні особливості основних типів імпульсних DC-DC перетворювачів (Buck та Boost).
2. Набути навичок моделювання та дослідження характеристик імпульсних перетворювачів у середовищі **Electronics Workbench (EWB)**.
3. Експериментально дослідити залежність вихідної напруги від коефіцієнта заповнення (D) ШІМ-сигналу.
4. Визначити коефіцієнт корисної дії (ККД) модельованих перетворювачів.

1. Обладнання та програмне забезпечення

- **Програмне забезпечення для схемотехнічного моделювання:** Electronics Workbench 5.12 або NI Multisim (EWB).
- **Віртуальні вимірювальні прилади EWB:** Осцилограф, Вольтметр (AC/DC), Амперметр (AC/DC), Джерело живлення (DC Voltage Source).
- **Віртуальні компоненти:** N-канальний MOSFET-транзистор (як ключ), ідеальний діод, індуктивність, конденсатори, джерело імпульсного сигналу (для ШІМ).

2. Завдання лабораторної роботи

1. Зібрати та дослідити схему **понижувального (Buck) DC-DC перетворювача** з параметрами $V_{in} = 12 \text{ В}$, $V_{out} = 5 \text{ В}$.
2. Зібрати та дослідити схему **підвищувального (Boost) DC-DC перетворювача** з параметрами $V_{in} = 5 \text{ В}$, $V_{out} = 12 \text{ В}$.
3. Визначити ККД обох перетворювачів при номінальному навантаженні.

3. Теоретичні відомості

Імпульсні перетворювачі постійної напруги (DC-DC) використовують ключовий елемент (зазвичай, MOSFET), який працює

в режимі високочастотного перемикання під керуванням сигналу ШІМ (Широтно-Імпульсна Модуляція).

Коефіцієнт заповнення (Duty Cycle, D):

$$D = \frac{t_{on}}{T} = \frac{t_{on}}{t_{on} + t_{off}}$$

де t_{on} — час замкненого ключа, T — період ШІМ.

Формули для вихідної напруги (в ідеальному безперервному режимі провідності):

- **Buck-перетворювач (понижувальний):**

$$V_{out} = V_{in} \cdot D$$

- **Boost-перетворювач (підвищувальний):**

$$V_{out} = V_{in} / (1 - D)$$

Коефіцієнт корисної дії (ККД):

$$\eta = \frac{P_{out}}{P_{in}} \cdot 100\% = \frac{V_{out} \cdot I_{out}}{V_{in} \cdot I_{in}} \cdot 100\%$$

Хід роботи

4.1. Дослідження понижувального перетворювача (Buck)

Схема: Зберіть схему Buck-перетворювача в EWB (рис.1), використовуючи:

- $V_{in} = 12$ В.
- MOSFET-ключ (керований імпульсним генератором).
- Діод Шотткі.
- Індуктивність L (наприклад, 100 мкГн).
- Конденсатор C (наприклад, 100 мкФ).
- Навантаження $R_{нав}$ (наприклад, 10 Ом).
- Частота ШІМ $f = 50$ кГц.

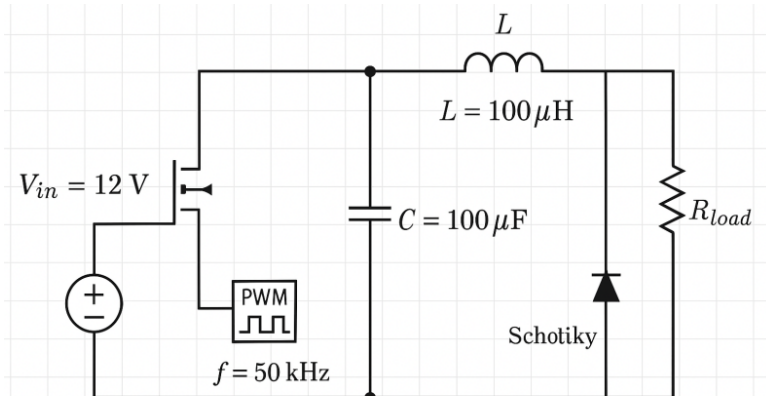


Рисунок 1. Схема понижувального перетворювача

Завдання 1: Визначення залежності V_{out} від D

1. Розрахуйте теоретичне значення коефіцієнта заповнення $D_{теор}$ для $V_{out} = 5$ В:
 $D_{теор} = V_{out} / V_{in} = 5 \text{ В} / 12 \text{ В} \approx 0.42$
2. Встановіть D відповідно до Таблиці 1 та виміряйте V_{out} , I_{in} та I_{out} .

Таблиця 1. Характеристики Виск-перетворювача ($V_{in}=12$ В)

№	Коефіцієнт заповнення D	Розрахункова $V_{out, теор}$ (В)	Виміряна $V_{out, експ}$ (В)	I_{in} (А)	I_{out} (А)	P_{in} (Вт)	P_{out} (Вт)	ККД η (%)
1	0.25							
2	0.42 ($V_{out} \approx 5$ В)	5.0						
3	0.60							
4	0.80							

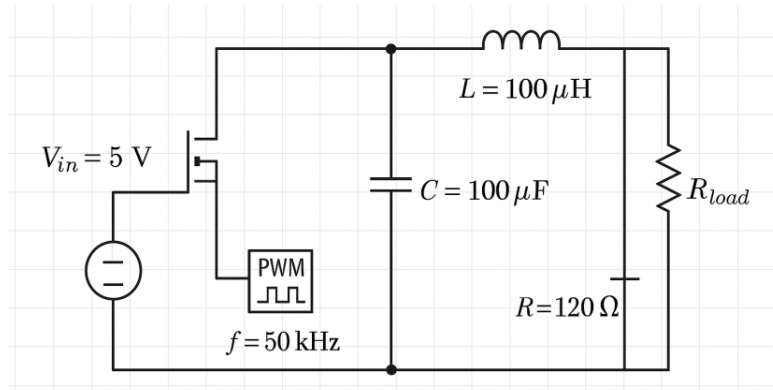
Завдання 2: Дослідження пульсацій

1. Встановіть $D = 0.42$ ($V_{out} \approx 5$ В).
2. Використовуючи осцилограф, виміряйте амплітуду пульсацій $V_{пульс}$ на виході V_{out} та зафіксуйте форму сигналу.

4.2. Дослідження підвищувального перетворювача (Boost)

Схема: Зберіть схему Boost-перетворювача в EWB. (рис.2))

- $V_{in} = 5$ В.
- MOSFET-ключ (керований імпульсним генератором).
- Діод Шотткі.
- Індуктивність L (наприклад, 100 мкГн).
- Конденсатор C (наприклад, 100 мкФ).
- Навантаження $R_{нав}$ (наприклад, 120 Ом).
- Частота ШІМ $f = 50$ кГц.



Риснок 2 Схема підвищувального перетворювача

Завдання 3: Визначення залежності V_{out} від D

1. Розрахуйте теоретичне значення коефіцієнта заповнення $D_{теор}$ для $V_{out} = 12$ В:

$$12 \text{ В} = \frac{5 \text{ В}}{1 - D_{теор}} \Rightarrow 1 - D_{теор} = 5/12 \approx 0.417 \Rightarrow D_{теор} \approx 0.583$$

2. Встановіть D відповідно до Таблиці 2 та виміряйте V_{out} , I_{in} та I_{out} .

Таблиця 2. Характеристики Boost-перетворювача ($V_{in}=5$ В)

№	Коефіцієнт заповнення D	Розрахункова $V_{out, теор}$ (В)	Виміряна $V_{out, експ}$ (В)	I_{in} (А)	I_{out} (А)	P_{in} (Вт)	P_{out} (Вт)	ККД η (%)
1	0.30							
2	0.58 ($V_{out} \approx 12$ В)	11.97						
3	0.70							
4	0.85							

Завдання 4: Графічний аналіз

1. Побудуйте графіки залежності V_{out} від D для Buck та Boost перетворювачів на основі даних Таблиць 1 та 2.

Вимоги до звіту

Звіт повинен містити:

1. Титульний аркуш та мету роботи.
2. Опис використаного обладнання та програмного забезпечення.

3. Принципові схеми Buck та Boost перетворювачів (з EWB), що використовувалися для моделювання.
4. Заповнені Таблиці 1 та 2 з усіма розрахунковими та вимірними значеннями, включаючи ККД.
5. Графік залежності $V_{out} = f(D)$ для обох перетворювачів.
6. Аналіз результатів та висновки щодо точності відповідності експериментальних даних теоретичним формулам.

Висновки до роботи

Студенти повинні сформулювати висновки, в яких:

- Порівняти теоретичні та експериментальні значення V_{out} для Buck та Boost перетворювачів.
- Проаналізувати вплив коефіцієнта заповнення D на вихідну напругу.
- Пояснити, чому ККД у моделюванні (EWB) є високим (близьким до 100%) порівняно з реальним пристроєм.
- Оцінити рівень пульсацій вихідної напруги та запропонувати шляхи його зниження.

Контрольні питання

1. Яку роль виконує індуктивність L у схемі Buck-перетворювача?
2. Поясніть, чому Boost-перетворювач завжди видає напругу, вищу за вхідну.
3. Що таке "безперервний режим провідності" (CCM) і як його забезпечити?
4. Як зміна частоти перемикання ШІМ вплине на розміри індуктивності та конденсатора?
5. Назвіть основні джерела втрат енергії в реальних імпульсних

джерелах живлення.

6. Наведіть приклад використання Buck-Boost перетворювача.

Список літератури українських видавництв

1. Бойко О.І., Шпак В.П. (2020). Промислова електроніка та мікропроцесорна техніка. Київ: НТУУ «КПІ».
2. Загірняк М.В., Щукін І.С. (2018). Основи електроніки та

- мікросхемотехніки. Львів: Видавництво Львівської політехніки.
3. Нетушил А.В. (2019). Електротехніка, електроніка та мікросхемотехніка. Харків: ХНУР.
 4. Шевчук Б. А. (2021). Силова електроніка. Вінниця: ВНТУ.

ЛАБОРАТОРНА РОБОТА № 11

Підключення та програмування різних типів сенсорів до мікроконтролера

Мета

1. Вивчити принципи узгодження аналогових сигналів сенсорів (схеми посилення та зміщення) для подачі на вхід АЦП мікроконтролера.
2. Набути навичок моделювання роботи аналого-цифрового перетворювача (АЦП) та схем керування цифровими інтерфейсами (I²C) у середовищі **Electronics Workbench (EWB)**.
3. Дослідити вплив параметрів АЦП (розрядність, опорна напруга) на точність вимірювання.

1. Обладнання та програмне забезпечення

- **Програмне забезпечення для схемотехнічного моделювання:** Electronics Workbench 5.12 або NI Multisim.
- **Віртуальні вимірювальні прилади EWB:** Осцилограф, Вольтметр DC, Функціональний генератор/Імпульсний генератор, Логічний аналізатор.
- **Віртуальні компоненти:**
 - Операційні підсилювачі (ОП) (наприклад, LM741).
 - Прецизійні резистори.
 - Компонент АЦП (якщо доступний в EWB) або імітація його роботи логічним конвертером.
 - Логічні елементи та цифрові регістри (для імітації цифрових інтерфейсів).
 - Джерело живлення, що регулюється (для імітації аналогового сенсора).

2. Завдання лабораторної роботи

1. Розробити та дослідити схему узгодження сигналу для типового аналогового сенсора (наприклад, LM35), забезпечивши перетворення вихідної напруги (0 В ... 1 В) у діапазон АЦП мікроконтролера (0 В...5 В).
2. Визначити максимальну похибку квантування, виходячи з розрядності АЦП.
3. Змоделювати базову передачу даних цифровим інтерфейсом I²C між мікроконтролером (Master) та сенсором (Slave) у режимі запису.

3. Теоретичні відомості

3.1. Аналого-цифрове перетворення

Роздільна здатність АЦП (R): Кількість біт, які використовує АЦП (наприклад, 10 біт). **Опорна напруга (V_{ref}):** Максимальна напруга, яку може виміряти АЦП. **Крок квантування (LSB):** Мінімальна зміна напруги, яку може розрізнити АЦП.

$$LSB = V_{ref} / 2^R$$

Якщо $R=10$ біт та $V_{ref}=5$ В, то $LSB = 5 \text{ В} / 1024 \approx 4.88 \text{ мВ}$.

Схема узгодження: Необхідна для масштабування, зміщення або посилення сигналу сенсора, щоб він повністю охоплював діапазон $0 \dots V_{ref}$ АЦП.

3.2. Інтерфейс I²C (Inter-Integrated Circuit)

I²C — двопровідний послідовний інтерфейс, що використовується для зв'язку між інтегральними схемами:

- **SCL (Serial Clock):** Лінія синхронізації (тактування), керується Master (мікроконтролером).
- **SDA (Serial Data):** Двонаправлена лінія даних.
- Всі пристрої підключаються до шини через підтягуючі резистори (Pull-up resistors) до лінії живлення.

4. Хід роботи

4.1. Дослідження аналогового інтерфейсу (Сенсор температури LM35)

Вхідні дані:

- Сенсор LM35 має чутливість $10 \text{ мВ}/^\circ\text{C}$. Діапазон вимірювання

0°C ... 100°C відповідає 0 В ... 1 В.

- АЦП мікроконтролера: $V_{ref} = 5 \text{ В}$, $R = 10 \text{ біт}$.

Завдання 1: Розрахунок параметрів схеми посилення

- Визначте необхідний коефіцієнт посилення (K) для переведення діапазону 0 В ... 1 В у 0 В ... 5 В:

$$K = \frac{V_{ref,max}}{V_{sens,max}} = \frac{5 \text{ В}}{1 \text{ В}} = 5$$

- Сконструйте схему неінвертуючого підсилювача на ОП з коефіцієнтом $K=5$. (рис. 1)

$$K = 1 + \frac{R_2}{R_1}$$

Наприклад, якщо $R_1 = 10 \text{ кОм}$, то $R_2 = (K-1) \cdot R_1 = 4 \cdot 10 \text{ кОм} = 40 \text{ кОм}$.

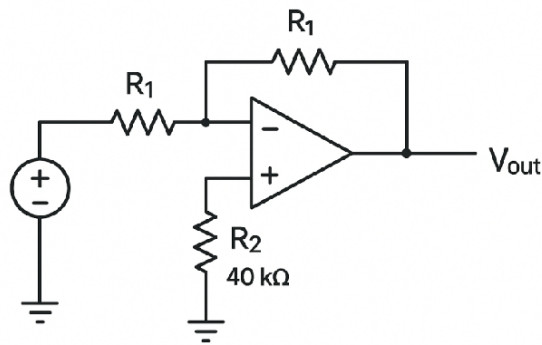


Рисунок 1 Схеми узгодження сигналу для LM35 (Неінвертуючий підсилювач, $K=5$)

Завдання 2: Моделювання та вимірювання

- Зберіть схему, яка включає джерело живлення (імітація сенсора), неінвертуючий підсилювач та АЦП (або вольтметр для контролю вхідної напруги АЦП). (рис.2)

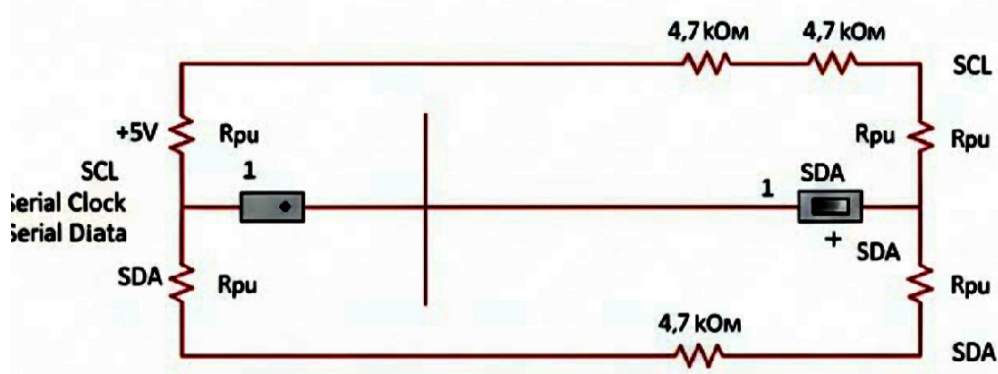


Рисунок 2 Схема імітації обміну даними по шині I²C (Стартова послідовність)

Використовуйте джерело живлення для імітації температури за Таблицею 1:

Таблиця 1. Аналогове перетворення (LM35 → АЦП)

T (°C)	$V_{\{in, sens\}}$ (В)	$V_{out, amp}$ (В) (Повинно бути)	$V_{out, amp}$ (В) (Експ.)	Значення АЦП (10-біт) $N = V_{out}/V_{ref} \cdot 1024$
0	0.00	0.00		0
25	0.25	1.25		256
50	0.50	2.50		512
75	0.75	3.75		768
100	1.00	5.00		1024

4.2. Дослідження цифрового інтерфейсу (Імітація I²C)

Завдання 3: Моделювання передачі даних Оскільки EWB не дозволяє написати код мікроконтролера, ми імітуємо логіку передачі 8-бітного слова даних за допомогою логічного аналізатора та імпульсних генераторів.

- Зберіть схему, що імітує дві лінії I²C (SCL та SDA) з підтягуючими резисторами. Використовуйте логічні перемикачі для ручного формування стартової послідовності (Start Condition) та бітів адреси/даних.

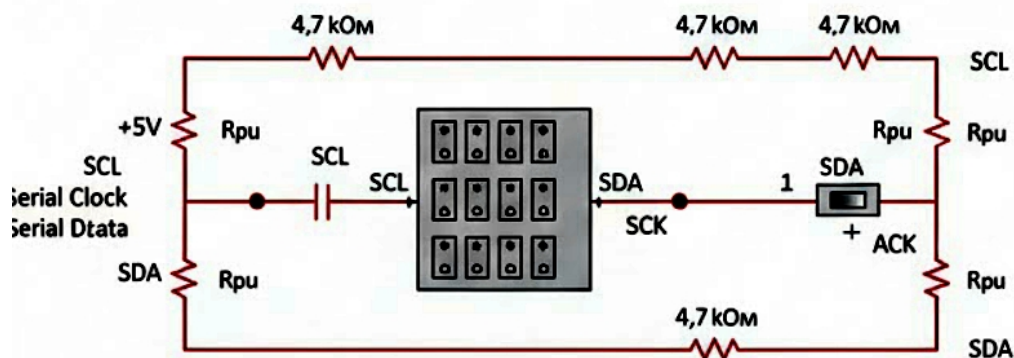


Рисунок 3 Схема імітації обміну даними по шині I²C (Передача байта)

- Використовуйте осцилограф або логічний аналізатор для фіксації сигналів.

Послідовність, що моделюється (запис 0x55 до сенсора з адресою 0x20):

1. **START:** SDA переходить з HIGH на LOW, поки SCL залишається HIGH.
2. **Адреса Slave (7 біт):** 0010 000
3. **Біт R/W (Запис):** 0
4. **АСК (Підтвердження від Slave):** Master звільняє SDA, Slave опускає SDA до LOW.
5. **Дані (8 біт):** 0101 0101 (0x55)
6. **STOP:** SCL переходить з LOW на HIGH, потім SDA переходить з LOW на HIGH.

Таблиця 2. Контроль логіки I²C

Етап	SCL (Тактування)	SDA (Дані)	Примітка
Очікування	HIGH	HIGH	Шина вільна
START	HIGH	HIGH → LOW	MASTER ініціює
Адреса (біт 7)	LOW → HIGH	(Адресний біт)	
...	
Біт R/W	LOW → HIGH	(0 - Запис)	
АСК/NACK	LOW → HIGH	(LOW - АСК)	Slave підтверджує
Дані (0x55)	Послідовність	0101 0101	
STOP	HIGH	LOW → HIGH	MASTER завершує

Вимоги до звіту

Звіт повинен містити:

1. Титульний аркуш, мету роботи та завдання.
2. Розрахунки коефіцієнта посилення K для аналогової схеми.
3. Принципову схему **неінвертуючого підсилювача** з номіналами резисторів (EWB).
4. Принципову схему імітації **I²C-шини** (EWB).
5. Заповнену Таблицю 1 з розрахунковими та експериментальними значеннями.
6. Форму сигналу на лініях SCL та SDA (з логічного аналізатора/осцилографа) для підтвердження стартової послідовності.

7. Висновки та відповіді на контрольні питання.

Висновки до роботи

Студенти повинні сформулювати висновки, в яких:

- Підтвердити успішність узгодження діапазонів напруг для аналогового сенсора та АЦП (наприклад, що 100°C успішно трансформується у 5 В).
- Оцінити значення LSB та пояснити, як розрядність АЦП впливає на точність кінцевого вимірювання.
- Підтвердити розуміння логіки роботи шини I²C, зокрема, умови START, STOP та механізм ACK.

Контрольні питання

1. Якщо розрядність АЦП збільшити з 8 до 12 біт (при $V_{ref}=5$ В), як зміниться мінімальна похибка квантування?
2. Поясніть призначення підтягуючих резисторів на лініях SCL та SDA в інтерфейсі I²C.
3. Яка перевага використання інструментального підсилювача порівняно зі звичайним неінвертуючим ОП у схемах узгодження сенсорів?
4. Чому струмова петля 4-20 мА є кращою за напругу 0-10 В для передачі сигналів на великі відстані?
5. Що таке "холодний спай" у термопарі та навіщо його компенсувати?

Список літератури

1. Павленко В.В., Ковальчук Л.М. *Мікропроцесорні системи та інтерфейси*. Київ: Видавництво "Освіта України", 2021, 348 с.
2. Лазаренко В.М., Петров А.І. *Електронні компоненти та пристрої*. Харків: ХНУР/2019, 412 с.
3. Бойко О.І., Шпак В.П. *Промислова електроніка та мікропроцесорна техніка*. Київ: НТУУ «КПІ» / 2020, 475 с.
4. Шевчук Б. А. *Цифрова та аналогова схемотехніка*. Вінниця: ВНТУ / 2021, 301 с.

ЛАБОРАТОРНА РОБОТА № 12

Управління двигунами постійного струму та кроковими двигунами за допомогою мікроконтролера

Мета

1. Вивчити принцип дії та схемотехнічну реалізацію мостової схеми **Н-Міст** для зміни напрямку обертання двигуна постійного струму.
2. Набути навичок моделювання генерації сигналу **Широтно-Імпульсної Модуляції (ШІМ)** та дослідити її вплив на ефективну напругу.
3. Дослідити послідовність керуючих імпульсів, необхідних для покрокового обертання біполярного крокового двигуна.

1. Обладнання та програмне забезпечення

- **Програмне забезпечення для схемотехнічного моделювання:** Electronics Workbench 5.12 або NI Multisim.
- **Віртуальні вимірювальні прилади EWB:** Осцилограф, Вольтметр DC/AC, Функціональний генератор/Імпульсний генератор, Логічний аналізатор.
- **Віртуальні компоненти:**
 - MOSFET-транзистори або VJT-транзистори (для побудови Н-Моста).
 - Захисні діоди (Flyback Diodes) для індуктивних навантажень.
 - Двигун постійного струму (DC Motor) як індуктивне навантаження.
 - Логічні елементи (AND, NOT) та цифрові генератори для імітації керування фазами крокового двигуна.

2. Завдання лабораторної роботи

1. Розробити та дослідити схему Н-Міст на чотирьох транзисторах для керування двигуном постійного струму.
2. Виміряти ефективну напругу на двигуні при різних значеннях коефіцієнта заповнення ШІМ-сигналу.
3. Смоделювати послідовність керуючих сигналів для біполярного крокового двигуна в режимі повного кроку.

3. Теоретичні відомості

3.1. Керування швидкістю: ШІМ

ШІМ (PWM) дозволяє регулювати середню потужність, що подається на двигун, змінюючи коефіцієнт заповнення (Duty Cycle, D).

$$D = \frac{T_{on}}{T_{period}}$$

$$V_{avg} = V_{peak} \cdot D$$

де T_{on} — час увімкненого стану, T_{period} — загальний період імпульсу, V_{peak} — напруга живлення.

3.2. Керування напрямком: H-Міст

H-Міст — це схема, яка дозволяє змінювати полярність навантаження (двигуна). Вмикаючи діагональні пари транзисторів, ми керуємо напрямком струму:

- **Напрямок 1 (Вперед):** Вмикаються ключі T_1 і T_4 .
- **Напрямок 2 (Назад):** Вмикаються ключі T_2 і T_3 .

Важливо: Одночасне вмикання вертикальних пар (T_1/T_2 або T_3/T_4) спричиняє коротке замикання джерела живлення, чого слід уникати.

3.3. Керування кроковим двигуном

Біполярний кроковий двигун має дві обмотки (фази А та В). Керування здійснюється зміною полярності струму в обмотках.

Таблиця 1. Послідовність повного кроку (4 кроки)

Крок	Фаза А	Фаза В	Кут (відносний)
1	+	+	0°
2	-	+	90°
3	-	-	180°
4	+	-	270°

(Знаки "+" та "-" означають напрямок струму в обмотці).

4. Хід роботи

4.1. Дослідження керування швидкістю за допомогою ШІМ

1. Побудова генератора ШІМ: Використовуйте Функціональний генератор (Pulse Generator) EWB для генерації прямокутних імпульсів з фіксованою частотою

(наприклад, 1 кГц) та регульованим коефіцієнтом заповнення D.

2. **Схема:** Підключіть вихід генератора ШІМ через ключ (наприклад, MOSFET-транзистор) до індуктивного навантаження (імітація двигуна) (рис.1).

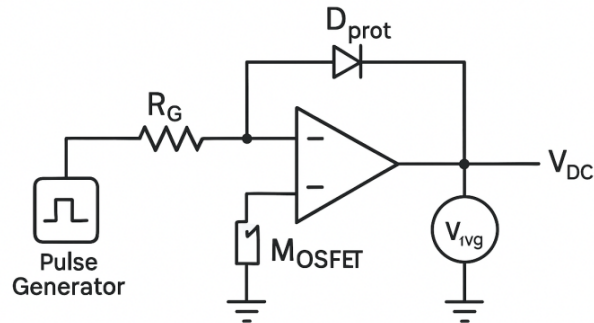


Рисунок 1 Схема управління швидкістю двигуна за допомогою ШІМ

Ключові елементи схеми:

1. **Генератор ШІМ (Pulse Generator):** Імітує вихід мікроконтролера, генеруючи прямокутні імпульси з регульованим коефіцієнтом заповнення (Duty Cycle).
 2. **Резистор R_G (Gate Resistor):** Обмежує струм керування затвором MOSFET (хоча для EWB це менш критично, ніж для реальних схем).
 3. **MOSFET-транзистор:** Використовується як ключ, який відкривається і закривається ШІМ-сигналом.
 4. **Двигун (Motor):** Імітується індуктивним навантаженням (індуктор + резистор) або віртуальним компонентом DC Motor в EWB.
 5. **Захисний діод (D_{prot}):** Підключається паралельно обмотці двигуна (індуктивному навантаженню) у зворотному напрямку. Він захищає MOSFET від високовольтного індуктивного кидка (проти-ЕРС), який виникає при вимкненні струму.
 6. **Осцилограф/Вольтметр:** Використовується для вимірювання форми сигналу (Осцилограф) та середньої напруги (V_{avg}) на двигуні (Вольтметр DC).
3. **Вимірювання:** Підключіть **Вольтметр DC** паралельно
 4. навантаженню для вимірювання V_{avg} (середньої напруги)

та **Осцилограф** для контролю форми сигналу.

Таблиця 2. Залежність середньої напруги від коефіцієнта заповнення

№	D (Duty Cycle)	T_{on} / T_{period}	V_{avg} (Розрахункове, $V_{peak}=12\text{ В}$)	V_{avg} (Експ.)
1	0.1	10%	1.2 В	
2	0.25	25%	3.0 В	
3	0.5	50%	6.0 В	
4	0.75	75%	9.0 В	
5	1.0	100%	12.0 В	

4.2. Дослідження керування напрямком: Н-Міст

1. **Складання Н-Моста:** Зберіть Н-Міст на чотирьох MOSFET-транзисторах. Всі індуктивні навантаження потребують **захисних діодів** (Flyback Diodes), встановлених паралельно транзисторам у зворотному напрямку. (рис. 2)

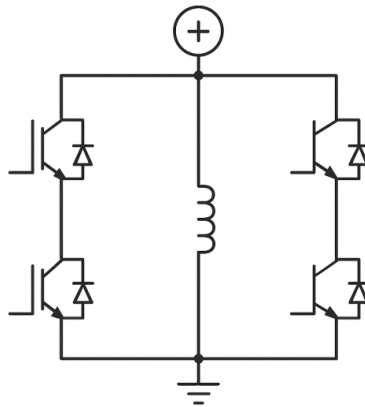


Рисунок 2 Схеми Н-Міста на чотирьох MOSFET-транзисторах

2. **Керування:** Використовуйте чотири логічні перемикачі (або генератори імпульсів), які імітують керуючі сигнали I_1, I_2, I_3, I_4 з мікроконтролера.
3. **Тестування:** Виміряйте напругу на двигуні (напрямок струму) у двох режимах:

Таблиця 3. Режими роботи Н-Моста

Режим	I_1	I_2	I_3	I_4	Напруга на двигуні	Напрямок обертання
Очікування	LOW	LOW	LOW	LOW	0 В	Вимкнено
Вперед	HIGH	LOW	LOW	HIGH	$+V_{DC}$	1
Назад	LOW	HIGH	HIGH	LOW	$-V_{DC}$	2
Заборонений	HIGH	HIGH	LOW	LOW	Коротке замикання!	Не тестувати

4.3. Імітація керування кроковим двигуном

1. **Схема імітації:** Оскільки в EWB немає інтегрованого крокового двигуна, ми імітуємо дві обмотки індукторами L_A та L_B та використовуємо два окремих Н-Мости (по одному на кожну фазу).
2. **Керування обмотками:** Використовуйте два незалежних генератори імпульсів (або логічні перемикачі) для формування послідовності струмів, що відповідає Таблиці 1 (Повний крок).
3. **Аналіз:** Використовуйте **Логічний аналізатор** для фіксації сигналів на входах Н-Мостів, щоб підтвердити правильність послідовності керування фазами А та В.

Вимоги до звіту

Звіт повинен містити:

1. Титульний аркуш, мету роботи та завдання.
2. Схему **ШІМ-керування** двигуном (п. 4.1).
3. Схему **Н-Міст** на транзисторах (п. 4.2).
4. Заповнену **Таблицю 2** з розрахунковими та експериментальними значеннями V_{avg} .
5. Заповнену **Таблицю 3** з результатами тестування напрямку обертання Н-Моста.
6. Графік форми ШІМ-сигналу та осцилограму V_{avg} при $D=50\%$.
7. Висновки та відповіді на контрольні питання.

Висновки до роботи

Студенти повинні сформулювати висновки, в яких:

- Підтвердити лінійну залежність середньої напруги V_{avg} від коефіцієнта заповнення D (на основі Таблиці 2).
- Продемонструвати розуміння логіки Н-Моста для інверсії

- полярності напруги на навантаженні.
- Обґрунтувати важливість використання захисних діодів для індуктивних навантажень.
- Підтвердити розуміння послідовності керування фазами крокового двигуна.

Контрольні питання

1. Поясніть, чому для керування швидкістю двигуна постійного струму краще використовувати ШІМ, а не регулювати напругу за допомогою резистора.
2. Яку функцію виконує **Dead Time** (час паузи) у схемах керування Н-Мостом на високих частотах?
3. Чому крокові двигуни є незамінними для систем точного позиціонування без використання енкодерів?
4. Назвіть основну перевагу біполярного крокового двигуна порівняно з уніполярним.
5. Поясніть явище **проти-ЕРС** і те, як вона виникає при вимкненні індуктивного навантаження.

Список літератури

1. Павленко В.В., Ковальчук Л.М. (2021). *Мікропроцесорні системи та інтерфейси*. Київ: Видавництво "Освіта України". (348 с.)
2. Лазаренко В.М., Петров А.І. (2019). *Електронні компоненти та пристрої*. Харків: ХНУР. (412 с.)
3. Бойко О.І., Шпак В.П. (2020). *Промислова електроніка та мікропроцесорна техніка*. Київ: НТУУ «КПІ». (475 с.)
4. Шевчук Б. А. (2021). *Цифрова та аналогова схемотехніка*. Вінниця: ВНТУ. (301 с.)

ЛАБОРАТОРНА РОБОТА № 13

Реалізація зв'язку між двома мікроконтролерами або між мікроконтролером та ПК через UART/SPI

Мета

1. Вивчити принцип формування та прийому асинхронного

- послідовного кадру даних (UART) у режимі "точка-точка".
2. Набути навичок моделювання синхронного послідовного обміну даними за протоколом SPI.
 3. Проаналізувати вплив параметрів зв'язку (швидкість, біти даних) на цілісність передачі.

1. Обладнання та програмне забезпечення

- **Програмне забезпечення для схемотехнічного моделювання:** Electronics Workbench 5.12 або NI Multisim.
- **Віртуальні вимірювальні прилади EWB:** Осцилограф, Логічний аналізатор (Logic Analyzer), Перемикачі, Індикатори.
- **Віртуальні компоненти:**
 - **Генератори тактових імпульсів:** для імітації синхронного тактування.
 - **Регістри зсуву (Shift Registers):** для паралельно-послідовного та послідовно-паралельного перетворення даних (імітація роботи UART/SPI).
 - **Логічні елементи:** AND, OR, NOT (для формування логіки протоколу).

2. Завдання лабораторної роботи

1. Змоделювати передачу 8-бітного слова даних через UART з використанням стартового та стопового бітів.
2. Використовуючи Логічний аналізатор, зафіксувати часову діаграму передачі.
3. Реалізувати схему обміну даними за протоколом SPI між віртуальними Master та Slave пристроями.
4. Проаналізувати синхронізацію даних на лініях MOSI та MISO під керуванням SCLK.

3. Теоретичні відомості

3.1. Асинхронний зв'язок UART

UART є найпростішим і найпоширенішим інтерфейсом. Ключовою особливістю є **відсутність лінії тактування**. Синхронізація досягається завдяки тому, що обидва пристрої працюють на однаковій, заздалегідь узгодженій швидкості (baud rate).

Передача кадру: Починається з переходу від неактивного

HIGH-рівня до активного LOW-рівня (Стартовий біт), далі йдуть біти даних (LSB першим), потім опціональний біт парності та завершується Стоповим бітом (HIGH).

3.2. Синхронний зв'язок SPI

SPI — повнодуплексний, Master-Slave інтерфейс. Усі пристрої синхронізуються за допомогою спільної лінії тактування SCLK, яку генерує Master.

- **MOSI (Master Out Slave In):** Дані йдуть від Master до Slave.
- **MISO (Master In Slave Out):** Дані йдуть від Slave до Master.
- **CS/SS (Chip Select):** Лінія, яка активує потрібний пристрій Slave (зазвичай активний LOW).

4. Хід роботи

4.1. Моделювання передачі даних через UART

4.1.1. Схема передавача (TX):

1. Зберіть схему, де 8-бітний паралельний вхід (наприклад, 8 логічних перемикачів або вихід 8-бітного лічильника) підключається до входу 8-бітного **регістра зсуву (Shift Register)**. (рис.1)
2. Налаштуйте логіку, що додає Стартовий біт (LOW) перед даними та Стоповий біт (HIGH) після даних.
3. Вихід регістра зсуву (TX) підключіть до Осцилографа або Логічного аналізатора.

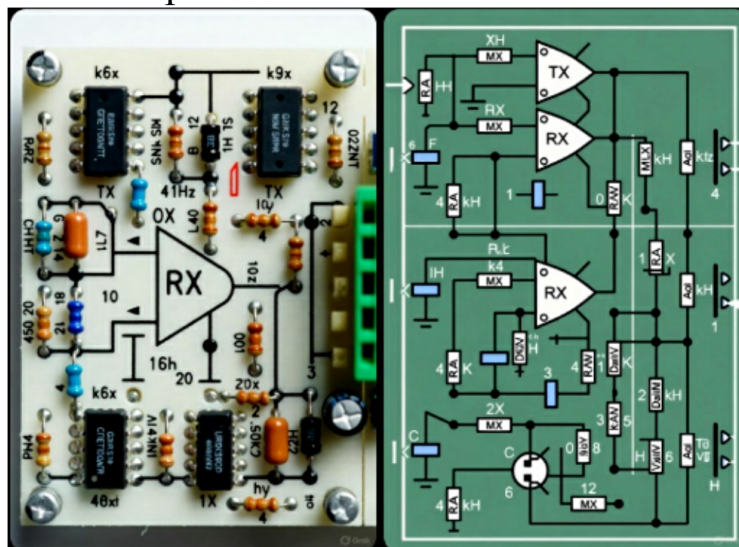


Рисунок 1 Схема передачі даних через UART для 8-бітного паралельного входу

4.1.2. Схема приймача (RX) та синхронізація:

1. Схема приймача має ідентичний регістр зсуву.
2. Для імітації синхронізації, використайте два незалежних тактових генератори з однаковою номінальною частотою (наприклад, 10 кГц) для TX та RX.
3. У реальних UART-схемах для синхронізації тактовий сигнал приймача запускається після виявлення Стартового біта і далі вибірка даних відбувається по **середині бітового інтервалу**.(рис.2)

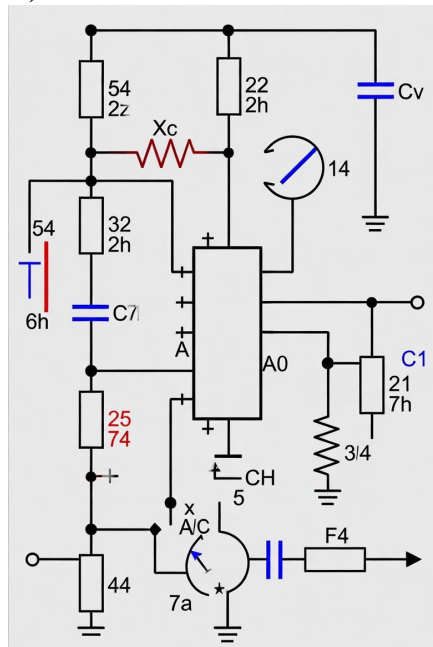


Рисунок 2 Схема приймача (RX) та синхронізація

4.1.3. Тестування та аналіз:

1. Встановіть передавальні дані (наприклад, двійковий код 10101010).
2. Запустіть моделювання та зафіксуйте осцилограму.

Таблиця 1. Аналіз кадру UART (Дані: 10101010)

№ Біта	Тип біта	Логічний рівень	Тривалість (мкс)	Призначення
1	Стартовий	LOW (0)	100	Сигналізація початку кадру
2-9	Дані (LSB →MSB)	0, 1, 0, 1, 0, 1, 0, 1	100 кожен	Передача корисної інформації
10	Стоповий	HIGH (1)	100	Сигналізація кінця кадру

4.2. Моделювання обміну даними через SPI

4.2.1. Схема Master-Slave:

1. Складіть схему, що включає два 8-бітних **реєстри зсуву** (один Master, інший Slave).
2. **Master (передавач):** Має Генератор тактових імпульсів, який генерує сигнал **SCLK**. (рис. 3)
3. **З'єднання:**
 - SCLK: спільна лінія тактування від Master до Slave.
 - MOSI: Вихід Master (реєстр зсуву) \rightarrow Вхід Slave (реєстр зсуву).
 - MISO: Вихід Slave \rightarrow Вхід Master.
 - CS (Chip Select): Додайте логічний перемикач для імітації вибору Slave-пристрою (активний LOW).

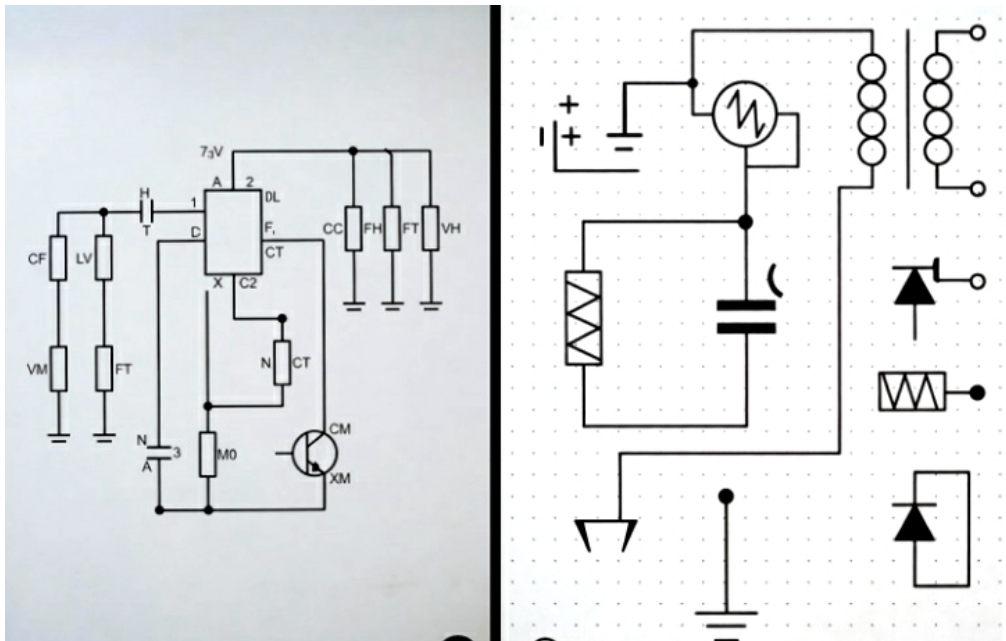


Рисунок 3 Схема SPI Master-Slave обміну

4.2.2. Тестування та аналіз (Передача Master \rightarrow Slave):

1. Встановіть початкові дані на Master. Встановіть CS у LOW.
2. Передайте 8 імпульсів SCLK.
3. За допомогою Логічного аналізатора відстежте синхронну зміну сигналів на SCLK та MOSI.
4. Проаналізуйте, як дані були занесені у реєстр Slave.

Таблиця 2. Аналіз синхронізації SPI (CPOL=0, CPHA=0)

№ Біта	SCLK (Імпульс)	MOSI (Master →Slave)	Значення даних, що передається
1	↑	D ₇	D ₇
2	↑	D ₆	D ₆
...
8	↑	D ₀	D ₀

Примітка: При CPOL=0 та CPHA=0 дані фіксуються по передньому (висхідному) фронту SCLK.

Вимоги до звіту

Звіт повинен містити:

1. Титульний аркуш, мету роботи та завдання.
2. Схему **UART-передавача** та приймача, змодельовану в EWB (п. 4.1).
3. Схему **SPI Master-Slave** обміну (п. 4.2).
4. Заповнену **Таблицю 1** з аналізом бітів кадру UART.
5. Заповнену **Таблицю 2** з аналізом синхронізації даних SPI.
6. Часову діаграму (скріншот Логічного аналізатора) передачі одного байта через UART.
7. Висновки та відповіді на контрольні питання.

Висновки до роботи

Студенти повинні сформулювати висновки, в яких:

- Порівняти необхідність синхронізації в UART та SPI.
- Пояснити, як Стартовий і Стоповий біти допомагають приймати дані в асинхронному режимі.
- Обґрунтувати, чому SPI має потенційно вищу швидкість передачі даних, ніж UART, та чому для нього потрібно більше ліній.

Контрольні питання

1. Чому в протоколі UART необхідно, щоб тактові частоти приймача та передавача були максимально наближені? Яка допустима відносна похибка?
2. Поясніть, у чому полягає відмінність між режимами SPI CPOL=0, CPHA=0 та CPOL=1, CPHA=1.

3. Які переваги має повнодуплексний зв'язок SPI над напівдуплексним зв'язком I²C?
4. Поясніть, що таке "**колізія**" в асинхронних системах і як її уникають у UART.
5. Як у схемах на основі EWB можна імітувати біт парності для контролю цілісності даних UART?

Список літератури

1. Павленко В.В., Ковальчук Л.М. (2021). *Мікропроцесорні системи та інтерфейси*. Київ: Видавництво "Освіта України". (348 с.)
2. Лазаренко В.М., Петров А.І. (2019). *Електронні компоненти та пристрої*. Харків: ХНУР. (412 с.)
3. Бойко О.І., Шпак В.П. (2020). *Промислова електроніка та мікропроцесорна техніка*. Київ: НТУУ «КПІ». (475 с.)
4. Шевчук Б. А. (2021). *Цифрова та аналогова схемотехніка*. Вінниця: ВНТУ. (301 с.)

ЛАБОРАТОРНА РОБОТА № 14

Розробка простої діагностичної системи для вимірювання та відображення параметрів

Мета

1. Набути навичок моделювання системи збору та обробки даних (SCOD) у схемотехнічному середовищі EWB.
2. Вивчити процес імітації аналого-цифрового перетворення (АЦП) та масштабування даних.
3. Реалізувати логіку діагностики на основі порогових значень та механізм візуалізації результатів.

1. Обладнання та програмне забезпечення

- **Програмне забезпечення для схемотехнічного моделювання:** Electronics Workbench 5.12 або NI Multisim.

- **Віртуальні вимірювальні прилади EWB:** Осцилограф, Вольтметр DC/AC.
- **Віртуальні компоненти:**
 - **Аналогові джерела:** Джерело синусоїдального сигналу (для імітації сенсора).
 - **Операційні підсилювачі (ОП):** Для імітації кондиціонування сигналу.
 - **Компаратори:** Для імітації порогової діагностики.
 - **Цифрові лічильники та дешифратори:** Для імітації роботи АЦП та відображення даних на сегментних індикаторах.
 - **Регістри:** Для імітації цифрової пам'яті.

2. Завдання лабораторної роботи

1. Спроекувати та зібрати схему кондиціонування сигналу, що масштабує вхідну напругу в діапазоні ± 1 В до однополярного діапазону 0-5 В.
2. Смоделювати 4-бітовий АЦП за допомогою дискретизації та перетворення рівня.
3. Запрограмувати (змоделювати) логіку, яка імітує діагностичну функцію: спрацювання індикатора помилки, якщо виміряний параметр перевищує 80% від максимального значення.
4. Відобразити виміряний цифровий код на 7-сегментному індикаторі.

3. Теоретичні відомості

3.1. Імітація АЦП в EWB

Оскільки EWB 5.12 не має складного програмованого АЦП, його функцію можна імітувати, використовуючи комбінацію аналогових та цифрових компонентів:

1. **Дискретизація:** Використання тактового сигналу для фіксації аналогової напруги через рівні проміжки часу.
 2. **Компараторна логіка:** Використання масиву компараторів для визначення того, в якому з 2^N діапазонів знаходиться вхідна напруга (для $N=4$ потрібно $2^4=16$ компараторів). Це імітація **Flash-АЦП**.
- **Роздільна здатність (для 4 біт, $V_{ref}=5$ В):**

$$\text{Роздільна здатність} = \frac{V_{ref}}{2^N} = \frac{5 \text{ В}}{16} = 0.3125 \text{ В/крок}$$

3.2. Діагностика та порогові значення

Діагностика – це порівняння виміряного значення з граничними параметрами.

- **Максимальний код (для 4 біт):** $1111_2 = 15_{10}$.
- **Діагностичний поріг (80%):** $15 \cdot 0.8 = 12$. Якщо цифровий код ≥ 12 (1100_2), спрацьовує помилка (OVERLOAD).

4. Хід роботи

4.1. Етап 1: Кондиціонування сигналу

Мета: Перетворення вхідного сигналу ± 1 В у діапазон 0 - 5 В.

1. Складіть схему на основі двох операційних підсилювачів. Перший ОП використовуйте як диференційний підсилювач, щоб позбутися шуму, якщо необхідно.
2. Другий ОП використовуйте як інвертуючий або неінвертуючий підсилювач, налаштувавши його коефіцієнт посилення K_U на 2.5 (щоб перетворити діапазон 2 В у 5 В).
3. Додайте **зсув рівня** за допомогою джерела живлення 2.5 В та резисторного дільника для зміщення всього сигналу вгору.
 - *Фінальний діапазон:* $V_{\text{вхід}} \in [-1 \text{ В}, +1 \text{ В}] \rightarrow V_{\text{вихід ОП}} \in [0 \text{ В}, 5 \text{ В}]$.
4. **Схема кондиціонування:**

Етап 1: Кондиціонування сигналу

Мета: Перетворення вхідного сигналу ± 1 В у діапазон 0 - 5 В.

5. Складіть схему на основі двох операційних підсилювачів. Перший ОП використовуйте як диференційний підсилювач, щоб позбутися шуму, якщо необхідно.
6. Другий ОП використовуйте як інвертуючий або неінвертуючий підсилювач, налаштувавши його коефіцієнт посилення K_U на 2.5 (щоб перетворити діапазон 2 В у 5 В).
7. Додайте **зсув рівня** за допомогою джерела живлення 2.5 В та резисторного дільника для зміщення всього сигналу вгору.
 - *Фінальний діапазон:* $V_{\text{вхід}} \in [-1 \text{ В}, +1 \text{ В}] \rightarrow V_{\text{вихід ОП}} \in [0 \text{ В}, 5 \text{ В}]$.

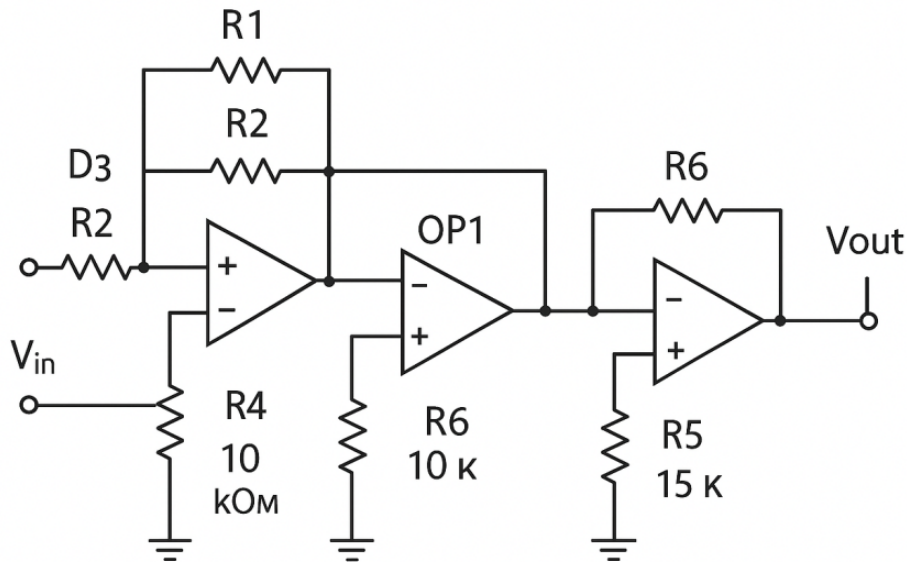
8. **Схема кондиціонування (рис.1):**

Рисунок 1 Схема кондиціонування сигналу

4.2. Етап 2: Імітація АЦП (4-бітове перетворення)

1. Створіть масив з 15 компараторів. На позитивні входи компараторів подайте напруги від 0.3125 В до 4.6875 В (крок 0.3125 В) через прецизійний резисторний дільник від $V_{ref}=5$ В.
2. На інвертуючі входи всіх компараторів подайте вихідний сигнал з Етапу 1.
3. Вихід компараторів підключіть до **кодера пріоритету** (Priority Encoder), який перетворює унітарний код компараторів на 4-бітовий двійковий код.

4.3. Етап 3: Діагностика та Відображення

1. **Діагностичний поріг:** Вихідні 4 біти з кодера пріоритету (D3, D2, D1, D0) підключіть до логічної схеми, що реалізує умову $D_{код} \geq 12$ (1100_2).
 - Умова: $(D3 \cdot D2) + (D3 \cdot D2 \cdot D1) + (D3 \cdot D2 \cdot D1 \cdot D0)$ або простішу логіку для коду 1100_2 .
2. Підключіть вихід цієї логічної схеми до червоного індикатора **OVERLOAD**.
3. Цифровий код (D3-D0) підключіть до двійково-десятькового дешифратора (BCD-to-7-segment decoder) та до 7-сегментного індикатора для візуалізації виміряного значення. (Рис.2)

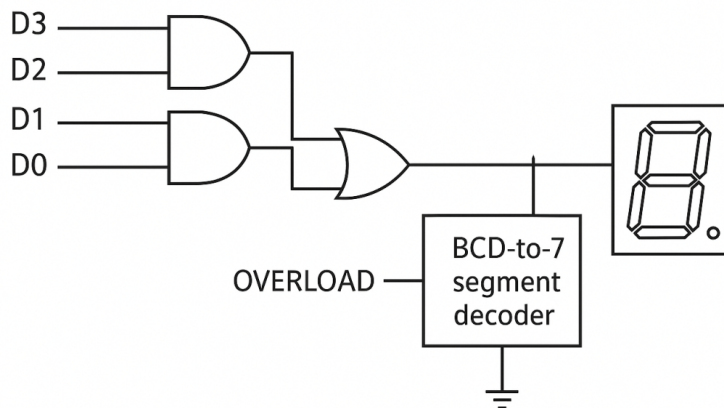


Рисунок 2 Принципова Схема Логіки Діагностики

Таблиця 1. Калібрування та діагностика (4-бітовий АЦП)

$V_{\text{вихід}}$ (± 1 В)	$V_{\text{вихід ОП}}$ (0-5 В)	Цифровий код (D3-D0)	Код (10)	$V_{\text{код}}$ (В)	Діагноз (OVERLOAD)
-1.0	0.0	0000	0	0.0	НІ
0.0	2.5	1000	8	2.5	НІ
+0.6	4.0	1100	12	3.75	ТАК
+1.0	5.0	1111	15	4.6875	ТАК

Вимоги до звіту

Звіт повинен містити:

1. Титульний аркуш, мету роботи та завдання.
2. **Принципову схему** кондиціонування сигналу (Етап 1) з номіналами резисторів та ОП.
3. **Принципову схему** логіки діагностики (Етап 3) з обґрунтуванням логічної функції.
4. Заповнену **Таблицю 1** з тестовими значеннями.
5. Скріншот EWB, що демонструє роботу схеми при $V_{\text{вихід}}=+0.6$ В (з активним індикатором OVERLOAD).
6. Висновки та відповіді на контрольні питання.

Висновки до роботи

Студенти повинні сформулювати висновки, в яких:

- Пояснити, як був реалізований процес АЦП за допомогою компараторів та кодера.
- Обґрунтувати важливість кондиціонування сигналу (зсуву та

- масштабування) для коректної роботи АЦП.
- Описати, як цифровий код використовується для прийняття діагностичного рішення.

Контрольні питання

1. Поясніть, що таке **роздільна здатність** АЦП і як вона змінюється, якщо V_{ref} зменшити вдвічі при незмінній розрядності.
2. Яку роль виконує **кодер пріоритету** у Flash-АЦП і чому він необхідний?
3. Для чого потрібен **зсув рівня** (level shifting) у схемі кондиціонування, якщо вхідний сигнал є двополярним (\pm)?
4. Опишіть логічну функцію, яка б реалізувала діагностичний поріг $D_{код} \geq 10$ (1010_2) для 4-бітного коду.
5. Чому у реальних мікроконтролерах частота дискретизації АЦП є обмежуючим фактором для діагностики високочастотних сигналів?

Список літератури

1. Литвиненко А.Г., Сокол Є.І. (2022). *Мікропроцесорна техніка: Архітектура, програмування, застосування*. Київ: КПІ ім. Ігоря Сікорського. (450 с.)
2. Шевчук Б. А. (2021). *Цифрова та аналогова схемотехніка*. Вінниця: ВНТУ. (301 с.)
3. Павленко В.В., Ковальчук Л.М. (2021). *Мікропроцесорні системи та інтерфейси*. Київ: Видавництво "Освіта України". (348 с.)

ЛАБОРАТОРНА РОБОТА № 15

Дослідження сучасних мікроконтролерів на базі IoT для систем управління

Мета Ознайомитися з сучасними тенденціями в електроніці, такими як інтеграція Інтернету речей (IoT), та реалізувати просту систему управління на базі мікроконтролера з бездротовим з'єднанням

(моделювання).

Ознайомитися, як мікросхемотехніка еволюціонує до компактних, енергоефективних пристроїв для діагностики та контролю.

1. Обладнання та програмне забезпечення

- **Програмне забезпечення для схемотехнічного моделювання:** Electronics Workbench 5.12 (або NI Multisim).
- **Віртуальні компоненти EWB:**
 - **Генератори та логічні джерела:** Тактовий генератор (Clock), джерела живлення, логічні перемикачі (для імітації сенсорів).
 - **Цифрові елементи:** JK-тригери, D-тригери, регістри, дешифратори.
 - **Комбінаційна логіка:** Елементи AND, OR, NOT, XOR.
 - **Прилади відображення:** Світлодіоди (LED) для відображення стану (Sleep, Active, Data Ready).

2. Завдання лабораторної роботи

Спроекувати та змоделювати в EWB **функціональний блок мікроконтролера IoT**, який виконує такі ключові функції:

1. **Моделювання Енергоефективності (Режим "Sleep/Wake"):** Створити лічильник стану (state machine), який переводить пристрій між режимами "Сон" (Sleep) та "Активний" (Active) з низьким робочим циклом (duty cycle).
2. **Збір Сенсорних Даних:** Моделювати збір даних з трьох цифрових сенсорів (S_2 , S_1 , S_0).
3. **Логіка Діагностики та Управління:** Розробити логічний вузол, який активує **Керуючий Вихід (Control)** лише у випадку, якщо сенсорні дані відповідають коду 101_2 (**5**).
4. **Моделювання "Бездротового" З'єднання:** Створити вихід **"Data Ready"** (сигнал готовності даних), який активується лише в режимі "Active" після успішного збору даних.

3. Теоретичні відомості

3.1. Концепція System-on-Chip (SoC) в IoT

Сучасні пристрої IoT інтегрують на одному кристалі:

обчислювальне ядро (CPU/MCU), пам'ять, аналогово-цифрові перетворювачі (АЦП), периферійні інтерфейси та, найголовніше, **радіомодуль** (Wi-Fi, Bluetooth).

3.2. Енергоефективність (Sleep/Wake Cycle)

Головна вимога до IoT — мінімальне споживання енергії. Мікроконтролери проводять до 99% часу в режимі **Sleep** (глибокий сон), де активний лише внутрішній годинник (Watchdog Timer). Вони переходять у режим **Active** лише на короткий проміжок часу для:

1. Збору даних із сенсорів.
2. Обробки даних.
3. Передачі даних через бездротовий канал.

3.3. Реалізація Станційного Автомата ("Sleep/Wake")

Режим роботи пристрою може бути описаний як **скінченний автомат (State Machine)**, реалізований на лічильниках або тригерах.

Стан	Опис	Вихід (Споживання)
00 (Sleep)	Мінімальне споживання. Логіка вимкнена.	Низьке
01 (Wake Up)	Ініціалізація, вмикання периферії.	Середнє
10 (Active)	Збір, обробка даних, активація логіки.	Високе
11 (Transmit/Ready)	Сигналізація готовності даних.	Середнє

4. Хід роботи

4.1. Етап 1: Моделювання Станційного Автомата (Sleep/Wake Logic) (рис.1)

Реалізуйте 2-бітовий лічильник на D-тригерах або JK-тригерах, який переходить між 4 станами (00, 01, 10, 11, 00, ...).

1. Використовуйте тактовий генератор (Clock) для синхронізації переходів.
2. Визначте логічний вихід "**Active Mode**" (логічна 1), який активується лише в станах **10** (Active) та **11** (Transmit).
3. Використовуйте світлодіод (LED) для індикації режиму "Active Mode".

SLEEP/WAKE LOGIC

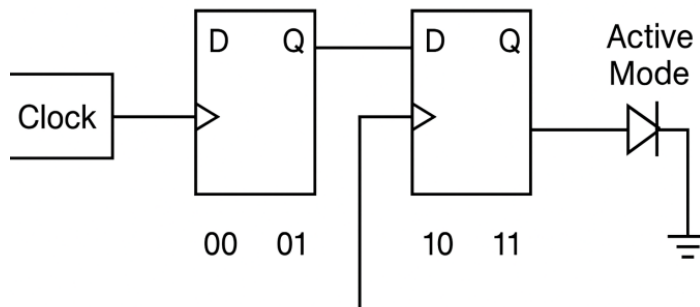


Рисунок 1 Принципова схема реалізації 2-бітового станційного автомата (Sleep/Wake Logic)

4.2. Етап 2: Збір та Діагностика Сенсорних Даних

1. Підключіть три логічні перемикачі (Logic Switches) як імітатори сенсорних входів: S_2 (MSB), S_1 , S_0 (LSB).
2. Розробіть комбінаційну логіку для керуючого виходу **Control** (C) за умовою:

$$C = 1, \text{ якщо } (S_2 S_1 S_0) = 101_2 (5)$$

Логічна функція: $C = S_2 \cdot S_1 \cdot S_0$

3. Підключіть LED для індикації C.

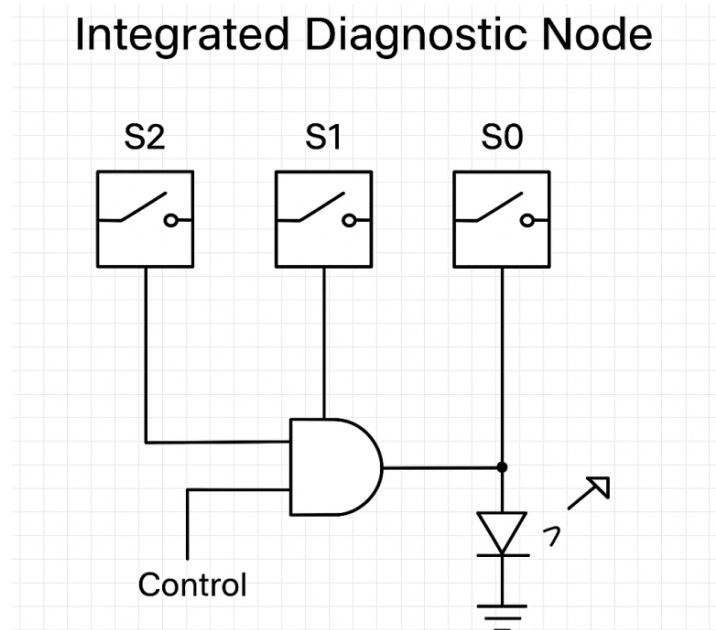


Рисунок 2 Принципова схема інтегрованого вузла діагностики (C та D)

4.3. Етап 3: Інтеграція та Моделювання "Бездротового" З'єднання

Вихід "Data Ready" (D) моделює активацію радіомодуля (що є енергоємним).

Логічна умова для D: Сигнал D має активуватися лише тоді, коли:

1. Пристрій перебуває у режимі **Active** (стан 10 або 11).
2. І спрацювала керуюча логіка $C=1$.

Фінальна Логічна Функція D: $D = (\text{Active Mode}) \cdot C$

1. Реалізуйте функцію D за допомогою логічного елемента AND.
2. Підключіть світлодіод для індикації D.

Таблиця 1. Тестування інтегрованої системи

Заповніть таблицю, змінюючи стани сенсорів та спостерігаючи за виходами C та D у різних режимах роботи (Active Mode, Sleep Mode).

Час (Стан Автомата)	Active Mode	$S_2 S_1 S_0$	Логіка C (101 ₂)?	Вихід D (Data Ready)
Sleep (00)	0	101	1	0
Active (10)	1	101	1	1
Active (10)	1	011	0	0
Transmit (11)	1	101	1	1
Sleep (00)	0	001	0	0

Вимоги до звіту

Звіт повинен містити:

1. Титульний аркуш, мету роботи та завдання.
2. **Принципову схему** реалізації 2-бітового станційного автомата (Sleep/Wake Logic) з позначенням компонентів.
3. **Принципову схему** інтегрованого вузла діагностики (C та D) з обґрунтуванням логічних функцій.
4. Заповнену **Таблицю 1** з тестовими значеннями.
5. Скріншот EWB, що демонструє роботу схеми, коли **стан автомата = Active (10)**, а **сенсори встановлені на 101₂**. На скріншоті має бути активний індикатор **Data Ready (D=1)**.

6. Висновки та відповіді на контрольні питання.

Висновки до роботи

Студенти мають підтвердити, що було успішно змодельовано:

- Енергоефективний цикл роботи IoT-пристрою (Sleep/Wake).
- Логіку діагностики, що активує керуючий сигнал лише за точно заданою умовою.
- Функціональну інтеграцію, де робота одного вузла (радіомодуля) жорстко залежить від стану іншого вузла (стану живлення та результатів діагностики).

Контрольні питання

1. Поясніть, як змінюється енергоспоживання (в реальному МК) при переході з режиму "Sleep" в режим "Active". Який компонент МК залишається активним навіть у "глибокому сні"?
2. Що таке **SoC (System-on-Chip)** у контексті IoT-пристроїв і чому ця архітектура є ключовою для мініатюризації?
3. Наведіть приклад, як би ви реалізували функцію "Wake Up" за допомогою зовнішнього переривання (наприклад, натискання кнопки) у моделі EWB.
4. Якби керуючий вихід ($\$C\$$) вимагав активації, якщо код був **непарним** (001_2 , 011_2 , 101_2 , 111_2), якою була б його найпростіша логічна функція?
5. Поясніть, чому ми вводимо проміжний логічний стан C (Control) перед активацією D (Data Ready). Який принцип IoT-діагностики це імітує?

Список літератури

1. Литвиненко А.Г., Сокол Є.І. (2022). *Мікропроцесорна техніка: Архітектура, програмування, застосування*. Київ: КПІ ім. Ігоря Сікорського.
2. Шевчук Б. А. (2021). *Цифрова та аналогова схемотехніка*. Вінниця: ВНТУ.
3. Павленко В.В., Ковальчук Л.М. (2021). *Мікропроцесорні системи та інтерфейси*. Київ: Видавництво "Освіта України".

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ПОСИЛАННЯ

1. Бойко В. І., Жуйков В. Я., Зорі А. А. Цифрова схемотехніка та мікропроцесори. — К.: «Освіта України», 2016.
2. Ямненко Ю. С. Мікроконтролери та мікропроцесори в системах керування: Конспект лекцій. — К.: НТУУ «КПІ», 2020.
3. Жуйков В. Я., Лисенко О. І. Мікроконтролери: Навчальний посібник. — К.: НТУУ «КПІ», 2013.
4. Загірняк М.В., Щукін І.С. (2018). Основи електроніки та мікросхемотехніки. Львів: Видавництво Львівської політехніки.
5. Нетушил А.В. (2019). Електротехніка, електроніка та мікросхемотехніка. Харків: ХНУР. Шевчук Б. А. (2021). Силова електроніка. Вінниця: ВНТУ
6. Павленко В.В., Ковальчук Л.М. *Мікропроцесорні системи та інтерфейси*. Київ: Видавництво "Освіта України", 2021, 348 с.
7. Лазаренко В.М., Петров А.І. *Електронні компоненти та пристрої*. Харків: ХНУР/2019, 412 с.
8. Бойко О.І., Шпак В.П. *Промислова електроніка та мікропроцесорна техніка*. Київ: НТУУ «КПІ» / 2020, 475 с.
9. Шевчук Б. А. *Цифрова та аналогова схемотехніка*. Вінниця: ВНТУ / 2021, 301 с.
10. Литвиненко А.Г., Сокол Є.І. (2022). *Мікропроцесорна техніка: Архітектура, програмування, застосування*. Київ: КПІ ім. Ігоря Сікорського. (450 с.)