

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних робіт

з дисципліни «Вбудовані комп'ютерні системи»

для студентів спеціальності F7 «Комп'ютерна інженерія»

за освітньою програмою «Спеціалізовані комп'ютерні системи»

всіх форм навчання

Методичні вказівки до лабораторних робіт з дисципліни «Вбудовані комп'ютерні системи» для студентів спеціальності F7 «Комп'ютерна інженерія» за освітньою програмою «Спеціалізовані комп'ютерні системи» всіх форм навчання/ Укл.: С.С. Грушко, А.В. Тіменко, Н.А. Куликовська – Запоріжжя: НУ «Запорізька політехніка», 2025. – 37 с.

Укладачі:

С.С. Грушко, к.т.н., доцент
А.В. Тіменко, ст. викладач
Н.А. Куликовська, ст. викладач

Рецензент:

С.Ю. Скрупський, к.т.н., доцент

Відповідальний
за випуск:

С.С. Грушко, к.т.н., доцент

Затверджено:

на засіданні кафедри
«Комп'ютерні системи та мережі»
Протокол № 2
від 29.08.2025 р.

Рекомендовано до видання
НМК факультету КНТ
Протокол № 3
від 25.09.2025 р.

ЗМІСТ

	С.
Вступ	5
1 Лабораторна робота № 1 Моделювання окремих модулів у Proteus	6
1.1 Створення та налаштування проєкту в ISIS	6
1.2 Вибір та розміщення компонентів схеми	7
1.3 З'єднання елементів схеми	9
1.4 Підключення статичних сигналів	9
1.5 Підключення динамічних сигналів	10
1.6 Візуалізація результатів моделювання	12
1.7 Проведення моделювання	15
1.8 Перевірка результатів та налагодження	16
1.8 Зміст звіту	17
1.8 Контрольні питання	17
2 Лабораторна робота № 2 Дослідження моделі годинника реального часу	18
2.1 Теоретичні відомості	18
2.2 Хід роботи	22
2.3 Зміст звіту	23
2.4 Контрольні питання	23
3 Лабораторна робота № 3 Дослідження моделі пристрою керування двигуном постійного струму	24
3.1 Теоретичні відомості	24
3.2 Хід роботи	26
3.3 Зміст звіту	29
3.4 Контрольні питання	29
4 Лабораторна робота № 4 Дослідження моделі пристрою керування кроковим двигуном	30
4.1 Теоретичні відомості	30
4.2 Хід роботи	33
4.3 Зміст звіту	35

4.4 Контрольні питання.....	35
Перелік джерел посилання	36

ВСТУП

Методичні вказівки призначені для проведення лабораторних робіт з дисципліни «Вбудовані комп'ютерні системи» і розроблені відповідно до навчальної програми підготовки студентів спеціальності F7 «Комп'ютерна інженерія» за освітньою програмою «Спеціалізовані комп'ютерні системи».

Вбудовані комп'ютерні системи є невід'ємною частиною сучасних технологічних рішень і широко застосовуються в промисловості, транспорті, медицині, побутовій техніці та інших галузях. Розуміння принципів роботи таких систем, володіння навичками їх проєктування та налагодження є важливими компетенціями сучасного інженера-програміста.

Лабораторні роботи виконуються з використанням сучасного програмного середовища Proteus ISIS, яке дозволяє моделювати роботу електронних схем і мікропроцесорних систем без необхідності використання реального обладнання. Це забезпечує безпечне та ефективне освоєння матеріалу, можливості експериментування з різними конфігураціями схем та детального аналізу їх роботи.

При виконанні лабораторних робіт студенти набувають практичних навичок аналізу та синтезу цифрових схем, налагодження роботи вбудованих систем, інтерпретації часових діаграм, роботи з технічною документацією та оформлення результатів дослідження.

1 ЛАБОРАТОРНА РОБОТА №1

МОДЕЛЮВАННЯ ОКРЕМИХ МОДУЛІВ У PROTEUS

Мета роботи - набуття практичних навичок моделювання та налагодження роботи окремих модулів та пристроїв у середовищі Proteus ISIS, включаючи конфігурування компонентів, підключення джерел сигналів та візуалізацію результатів моделювання.

1.1 Створення та налаштування проєкту в ISIS

Для початку роботи необхідно створити структуру проєкту:

– завантажте Proteus Professional Demonstration
(<https://www.labcenter.com/downloads/>)

– створіть на диску папку проєкту з ім'ям не більше 8 латинських символів;
– запустіть середовище ISIS Proteus через відповідну піктограму на робочому столі;

– у головному меню виберіть: «Файл → Зберегти проєкт»;
– у діалоговому вікні збереження оберіть створену папку та введіть ім'я файлу (таке саме, як ім'я папки);

– середовище автоматично присвоїть розширення .dsn.

Інтерфейс середовища ISIS включає:

– горизонтальне меню з вертикальними піктограмами швидкого доступу (ліворуч);

– поле креслення (центральна частина);

– закладки бібліотек елементів (знизу зліва);

– кнопки управління моделюванням (знизу зліва).

Рис. 1.1 ілюструє основні елементи інтерфейсу ISIS.

1.2 Вибір та розміщення компонентів схеми

Послідовність дій для вибору необхідних компонентів [1]:

- клацніть лівою кнопкою миші по закладці «Р»;
- у вікні «Pick Devices» (рис. 1.2) в полі «Категорія» виберіть «TTL 74 Series»;
- подвійним клацанням лівої кнопки миші оберіть потрібні мікросхеми зі списку.

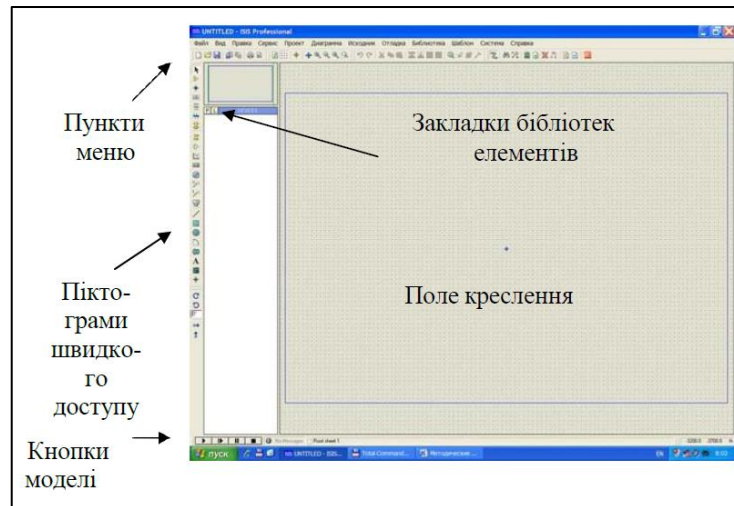


Рисунок 1.1 - Призначення елементів інтерфейсу в Proteus

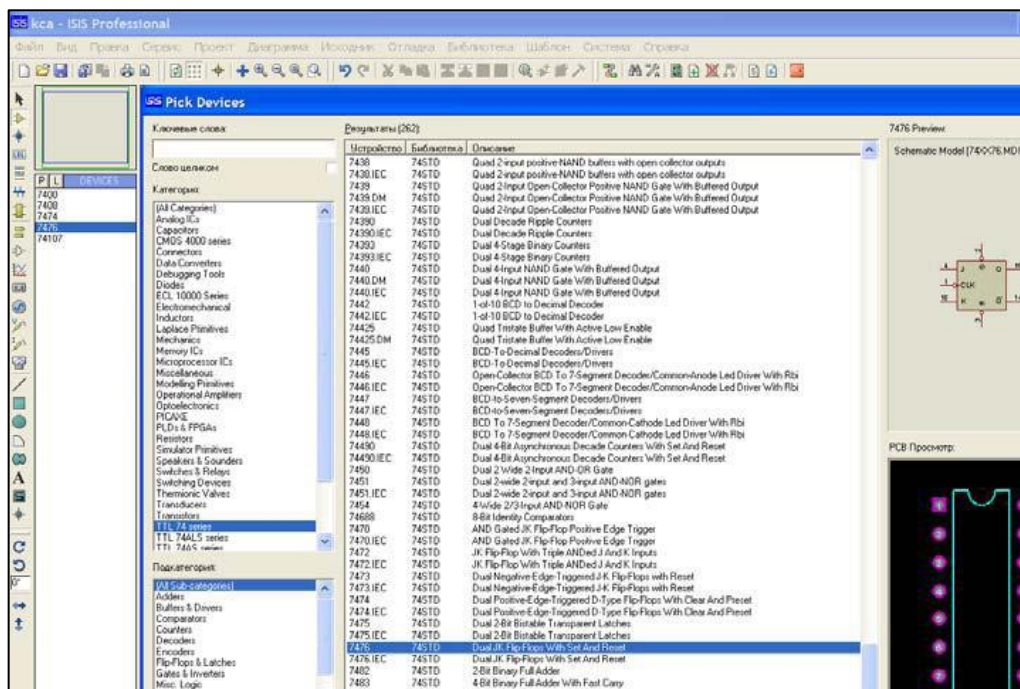


Рисунок 1.2 - Вибір необхідних елементів схеми

Після вибору:

- обрані елементи (табл.1.1) з'являться в лівій колонці вікна проекту;
- для видалення непотрібного елемента виділіть його правою кнопкою миші та натисніть «DEL».

Для розміщення компонентів виконайте наступні дії:

- клацанням лівої кнопки миші виберіть елемент зі списку;
- перемістіть курсор на поле креслення;
- подвійним кліком лівої кнопки зафіксуйте місце розташування елемента (рис. 1.3).

Таблиця 1.1 - Відповідність логічних елементів мікросхемам 74-ої серії

Логічний елемент	Мікросхема	Примітка
2І-НІ	7400	Два входи, інверсний вихід
2І	7408	Два входи, прямий вихід
2АБО-НІ	7402	Два входи, інверсний вихід
2АБО	7432	Два входи, прямий вихід
JK-тригер	7476	Тригер зі входами J, K
D-тригер	7474	Тригер з одним інформаційним входом

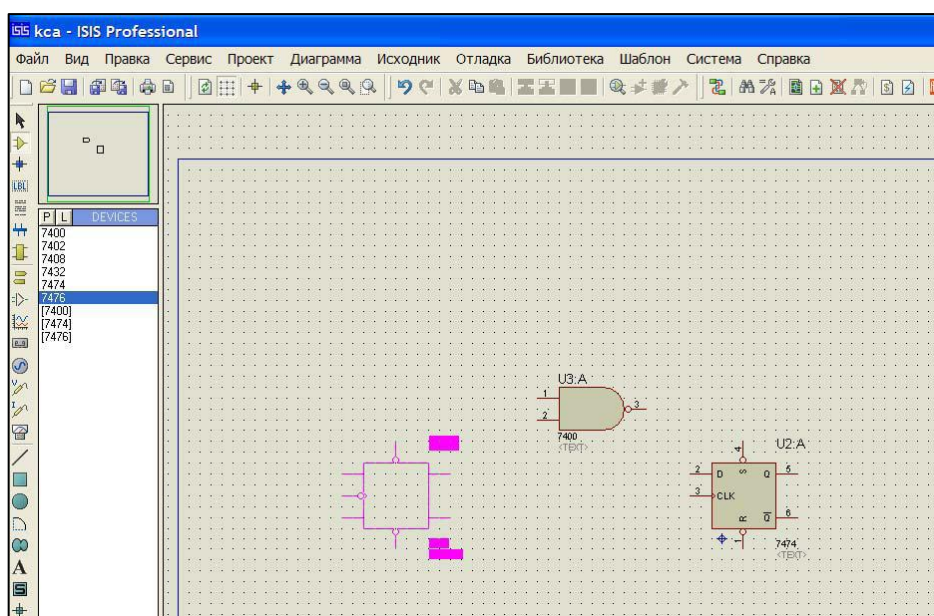


Рисунок 1.3 - Розташування елементів схеми в полі креслення

1.3 З'єднання елементів схеми

Для з'єднання контактів елементів:

- підведіть курсор до необхідного контакту елемента;
- при наближенні з'явиться червоний квадратик біля контакту (рис. 1.4);
- клацніть лівою кнопкою миші для фіксації початку провідника;
- ведіть провідник до потрібної точки схеми;
- для зміни напрямку клацніть один раз лівою кнопкою в точці повороту;
- завершіть з'єднання подвійним клацанням у кінцевій точці.

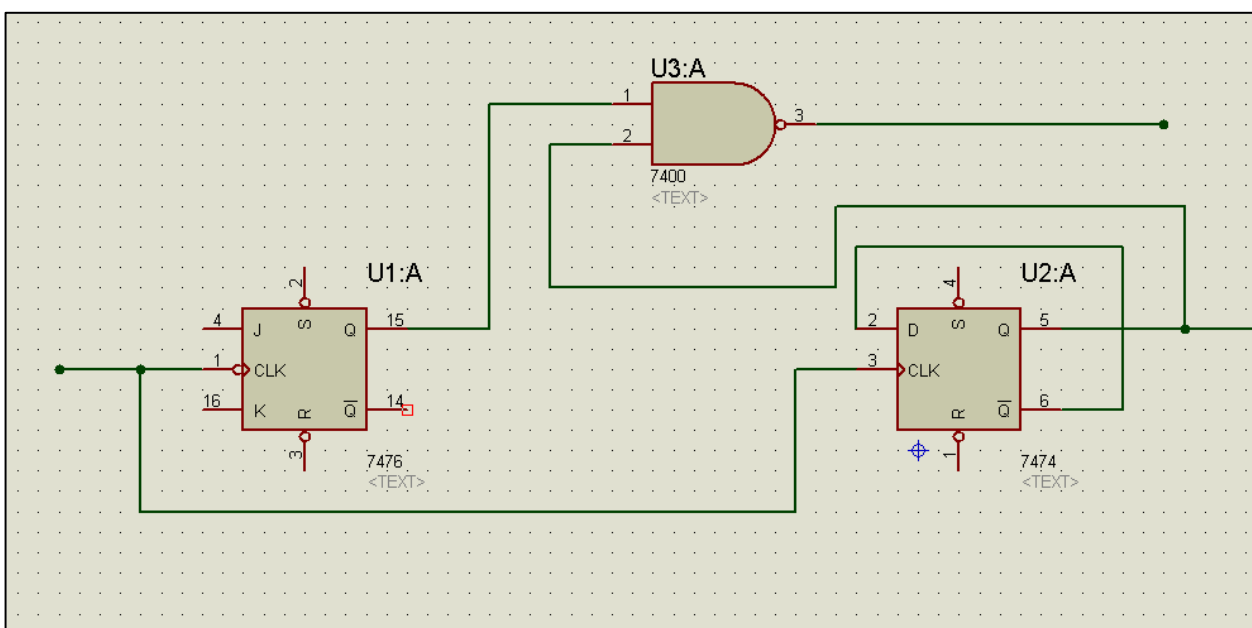


Рисунок 1.4 - З'єднання елементів схеми на полі креслення

ВАЖЛИВО: при з'єднанні контактів різних елементів провідники автоматично формують електричне з'єднання у вузлах перетину.

1.4 Підключення статичних сигналів

Для подачі постійних логічних рівнів на входи схеми використовуються спеціальні елементи – термінали:

- у лівому вертикальному меню виберіть піктограму «Термінали» (рис. 1.5);

- для логічної «1» оберіть термінал «POWER»;
- для логічного «0» оберіть термінал «GROUND»;
- розмістіть термінали на полі креслення аналогічно звичайним елементам;
- з'єднайте їх з відповідними контактами схеми.

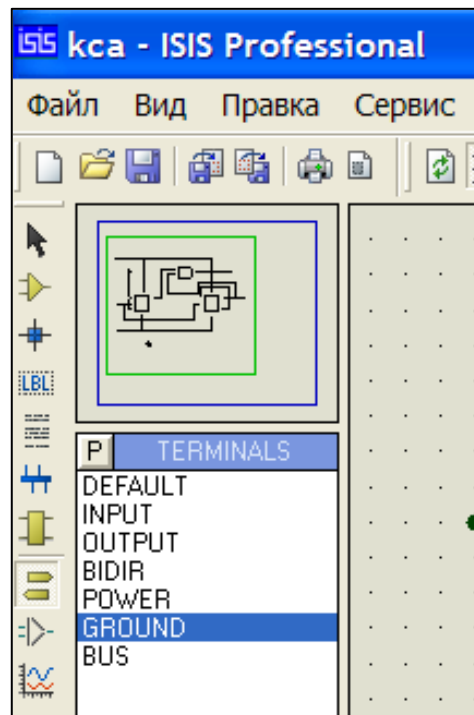


Рисунок 1.5 - Вибір терміналів «0» і «1»

На рис. 1.4 термінали використані для подачі сигналів на входи установки початкового стану тригерів та статичного рівня на входи J, K.

1.5 Підключення динамічних сигналів

Для імітації змінних вхідних сигналів (тактових імпульсів, керуючих сигналів) використовуються генератори.

Послідовність дій:

- у лівій вертикальній панелі оберіть бібліотеку генераторів (рис. 1.6);
- виберіть тип «DPATTERN» (цифровий шаблон);
- розмістіть генератор на схемі.

Для налаштування параметрів:

- двічі клацніть лівою кнопкою по зображенню генератора;
- у вікні властивостей «Digital Pattern Generator Properties» (рис. 1.7) виконайте налаштування.

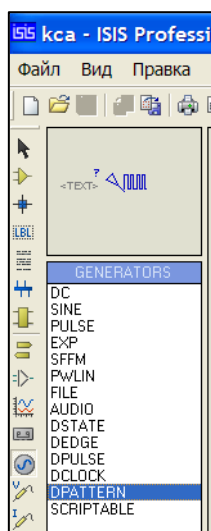


Рисунок 1.6 - Вибір генераторів



Рисунок 1.7 – Назви вікон для конфігурування генераторів вхідних сигналів

Для генератора тактових імпульсів (CLK):

- вид сигналу: «Цифровий → Шаблон»;
- встановіть прапорець «Рівномірний таймінг»;
- тривалість імпульсу (Pulse width): 50m (50 мілісекунд).

Для генератора перепаду (одиничний фронт):

- вид сигналу: «Цифровий → Один фронт»;
- полярність: позитивна; – тривалість фронту: 1m.

За необхідності введіть зрозуміле ім'я генератора в полі «Ім'я генератора».

ПРИМІТКА: для моделювання роботи комбінаційних цифрових автоматів з 5 входами використовуйте 5 генераторів з періодами, що відрізняються вдвічі: 50m, 100m, 200m, 400m, 800m відповідно.

Підключіть вихідні контакти генераторів до відповідних входів схеми стандартним способом (рис. 1.8).

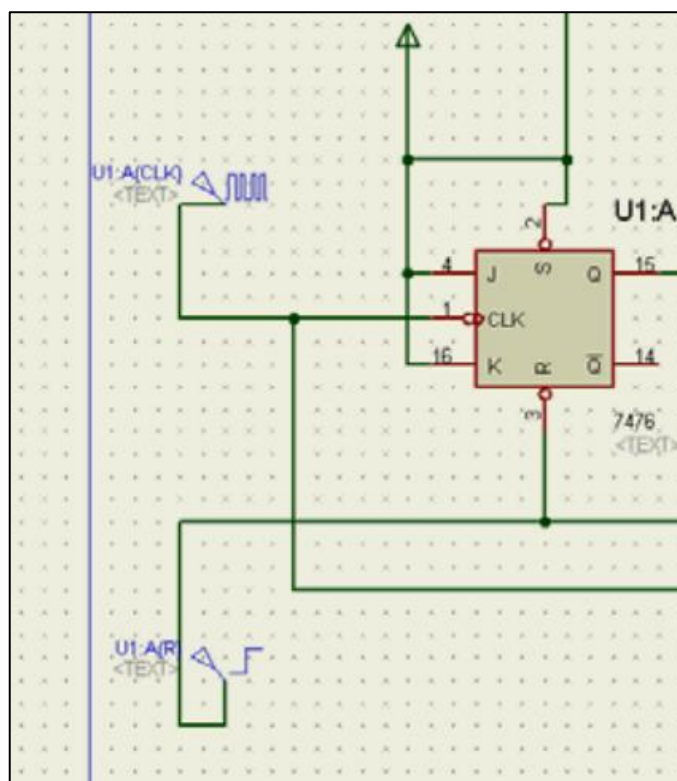


Рисунок 1.8 - Підключення генераторів

1.6 Візуалізація результатів моделювання

Логічні пробники дозволяють спостерігати стан окремих сигналів схеми.

Послідовність підключення:

- клацніть по закладці «Р»;
- у полі «Ключові слова» вікна «Pick Devices» введіть: LOGICPROBE (BIG);
- додайте компонент до проєкту подвійним клацанням;
- розмістіть пробники біля точок схеми, стан яких потрібно контролювати;
- підключіть входи пробників до відповідних ланцюгів.

Рекомендована послідовність підключення пробників:

- сигнал CLK; – вхідні сигнали (у порядку зростання періоду);
- вихідні сигнали.

Схема з підключеними пробниками показана на рис. 1.9.

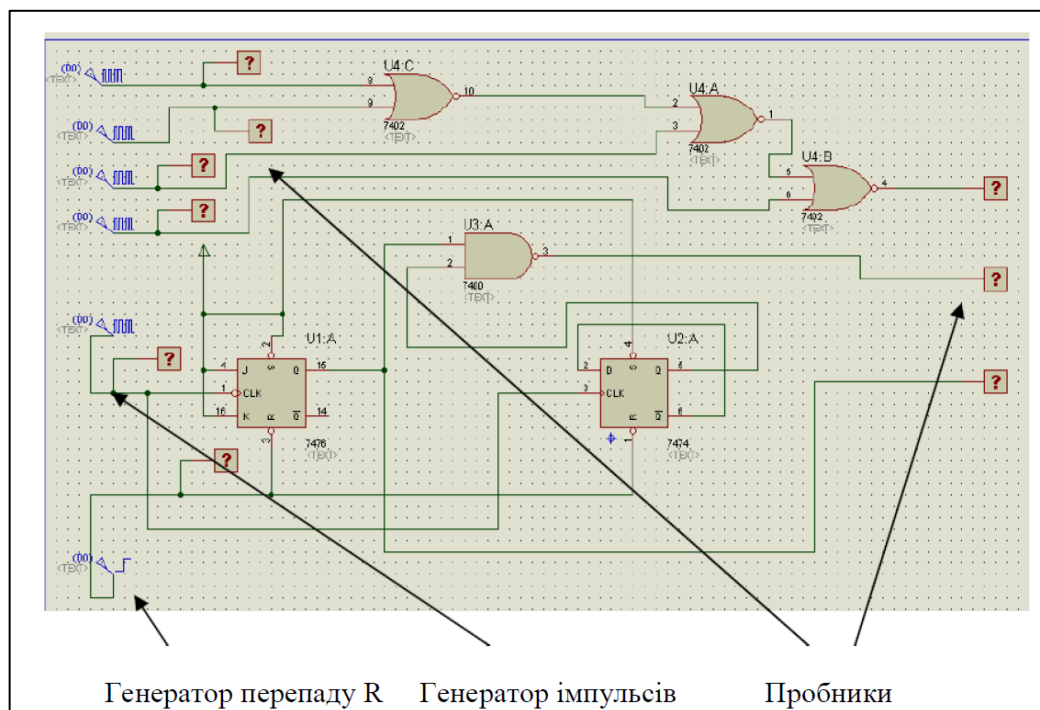


Рисунок 1.9 - Схема пристрою з підключеними генераторами і пробниками, підготовлена до моделювання

Для спостереження за роботою схеми в безперервному режимі використовується графічний аналізатор.

Послідовність дій:

- у лівій панелі виберіть піктограму «Діаграма»;
- оберіть тип «GRAPHS → DIGITAL» (рис. 1.10);
- на полі креслення, утримуючи ліву кнопку миші, накресліть рамку аналізатора;
- завершіть розміщення повторним кліком.

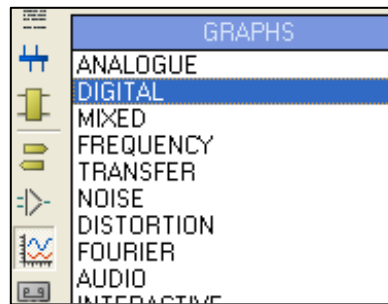


Рисунок 1.10 - Вибір графічного аналізатора

Для підключення щупів:

- у боковій панелі виберіть «Щуп напруги»;
- клацніть на проводі, який потрібно контролювати;
- щуп автоматично підключиться.

Послідовність налаштування:

- клацніть правою кнопкою по зображенню аналізатора;
- виберіть «Додати трасу»;
- у вікні «Add Transient Trace» у рядку «Щуп P1» виберіть потрібний провід зі списку;
- повторіть для всіх необхідних сигналів.

ВАЖЛИВО: назви трас можуть містити тільки латинські символи.

Для налаштування відображення:

- клацніть лівою кнопкою по аналізатору;
- у вікні «Діаграма перехідного процесу» встановіть параметр «Кін. час» рівним 2s (або іншому значенню залежно від потреб).

Результуюча схема з аналізатором показана на рис. 1.11.

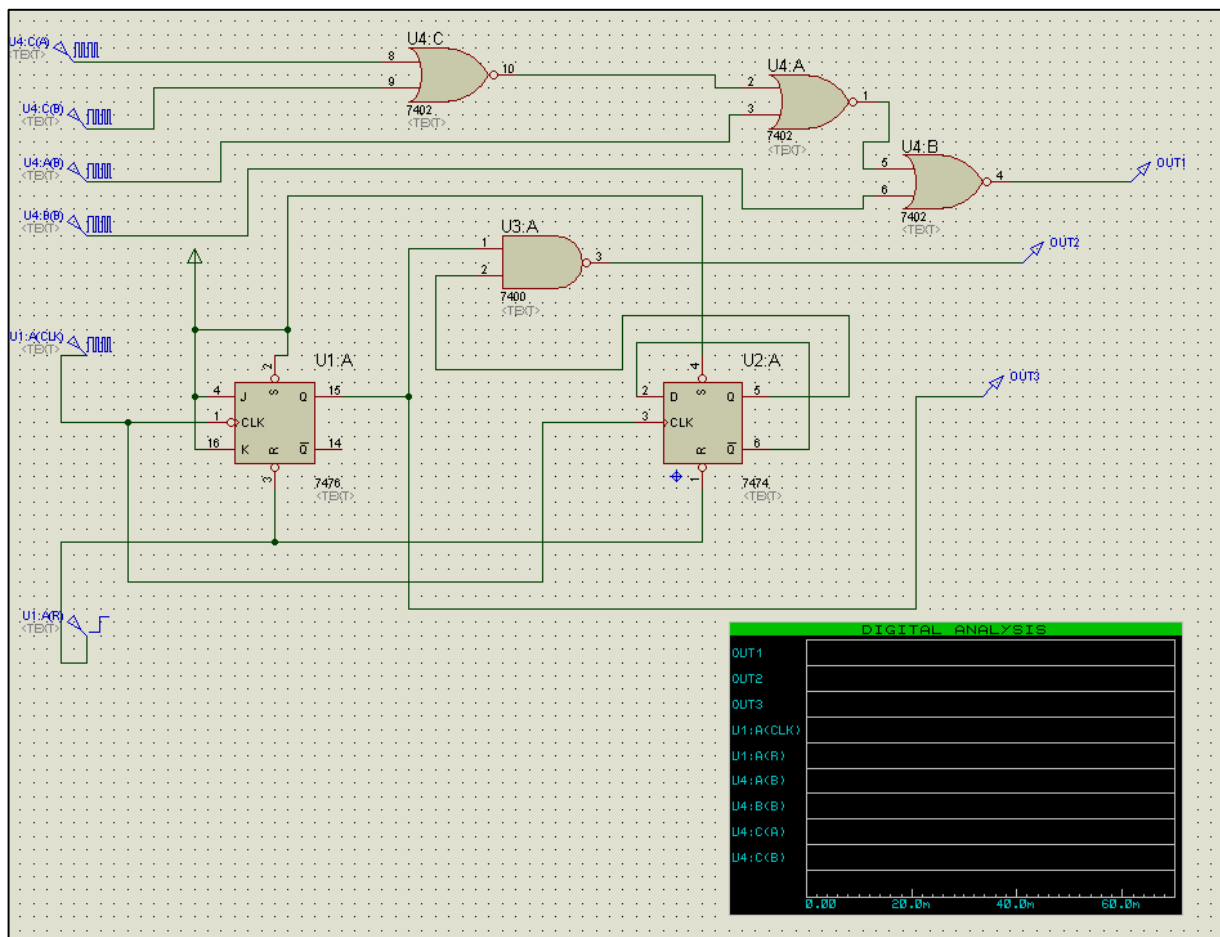


Рисунок 1.11 - Схема пристрою з підключеними генераторами і аналізатором, підготовлена до моделювання

1.7 Проведення моделювання

Перед початком моделювання:

- у головному меню виберіть: System → Set Animation Options;
- встановіть параметр Single Step Time = 51m.

Послідовність дій для покрокового моделювання:

- встановіть необхідні початкові значення вхідних сигналів на генераторах;
- натисніть піктограму «Крок» у лівому нижньому куті (рис. 1.12);
- спостерігайте за зміною стану пробників (замість «?» з'являться логічні рівні);
- кожне натискання кнопки «Крок» відповідає одному тактовому імпульсу CLK.



Рисунок 1.12 - Піктограми керування моделюванням

ПРИМІТКА: в процесі моделювання біля контактів мікросхем автоматично з'являються «маленькі пробники», проте їх розміри незручні для детального аналізу.

Для безперервного моделювання:

- натисніть клавішу «Пропуск» (або кнопку відтворення);
- на діаграмі відобразяться часові діаграми обраних сигналів (рис. 1.13);
- для детального перегляду подвійним клацанням розгорніть вікно діаграми;
- утримуючи ліву кнопку миші, пересувайте вертикальний маркер для зчитування значень сигналів у конкретні моменти часу;
- у лівій частині діаграми відобразатимуться рівні сигналів: H (висока «1») або L (низький «0»).

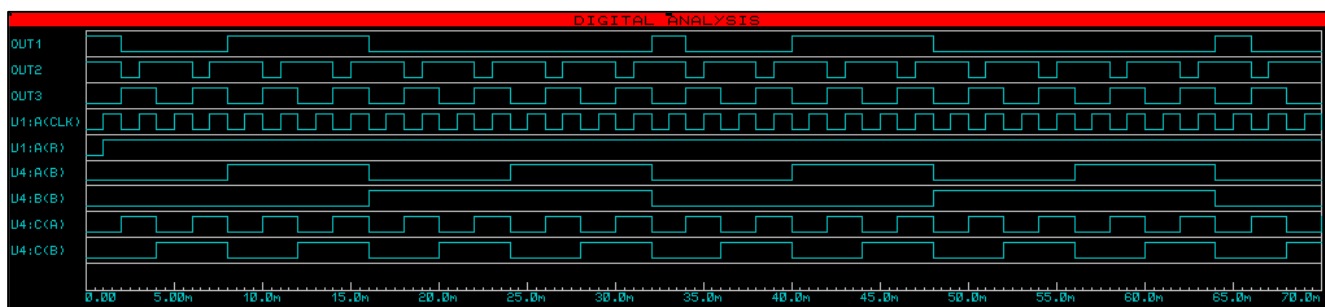


Рисунок 1.13 - Результати моделювання на графічному аналізаторі

1.8 Перевірка результатів та налагодження

Порівняйте отримані на діаграмах або пробниках результати з очікуваними згідно таблиці функціонування пристрою:

- переконайтеся у правильності послідовності станів вихідних сигналів;
- перевірте відповідність часових співвідношень;

– проаналізуйте реакцію на всі можливі комбінації вхідних сигналів.

При виявленні невідповідностей:

- зупиніть моделювання (кнопка з червоним колом);
- перейдіть до редагування схеми;
- внесіть необхідні зміни в з'єднання або параметри елементів;
- повторіть моделювання.

1.9 Зміст звіту

1. Мета роботи.
2. Скрін робочої схеми.
3. Скрін графічного аналізатора з результатами моделювання, аналіз часових діаграм.
4. Висновки до роботи.
5. Відповіді на контрольні питання.

1.10 Контрольні питання

1. Які типи логічних елементів доступні в бібліотеці TTL 74 Series?
2. Яка різниця між терміналами POWER та GROUND?
3. Для чого використовується генератор типу DPATTERN?
4. Як налаштувати тривалість імпульсів генератора?
5. Яка різниця між логічними пробниками та графічним аналізатором?
6. Як інтерпретувати позначення H та L на діаграмах?
7. Які кроки необхідні для виправлення помилок у схемі?

2 ЛАБОРАТОРНА РОБОТА №2

ДОСЛІДЖЕННЯ МОДЕЛІ ГОДИННИКА РЕАЛЬНОГО ЧАСУ

Мета роботи - ознайомлення із принципами реалізації лічильників реального часу, проведення моделювання роботи схеми лічильника.

2.1 Теоретичні відомості

2.1.1 Поняття годинника реального часу

Годинник реального часу (Real-Time Clock, RTC) – це електронний пристрій, який веде відлік поточного часу навіть при вимкненому основному живленні системи. RTC працює від батарейки та використовується в комп'ютерах, смартфонах, системах автоматизації та пристроях інтернету речей.

2.1.2 Способи реалізації RTC

Існує декілька основних способів створення годинників реального часу. Найпростіший – використання спеціалізованих мікросхем (DS1307, DS3231, PCF8563), які містять кварцовий резонатор 32768 Гц, живляться від батарейки CR2032 зі споживанням 1-2 мкА та підключаються через інтерфейси I²C або SPI. Також можна використовувати вбудовані модулі RTC в мікроконтролерах (STM32, ESP32, ATmega328P) з підключенням зовнішнього кварца 32.768 кГц, або реалізувати годинник програмно через таймери мікроконтролера. У пристроях з мережевим підключенням застосовується синхронізація через протокол NTP, GPS або GSM/LTE модулі.

2.1.3 Принцип роботи базового лічильника

Основою годинника є кварцовий резонатор на 32.768 кГц (2^{15} Гц). Цю частоту послідовно ділять навпіл п'ятнадцять разів, отримуючи на виході 1 Гц – один імпульс за секунду. Секундні імпульси надходять на каскад лічильників: лічильник секунд рахує від 0 до 59, при переповненні генерує сигнал переносу на лічильник

хвилин, який також рахує до 59 і передає сигнал лічильнику годин (0-23 для 24-годинного формату). Лічильник днів враховує різну кількість днів у місяцях та високосні роки.

Точність RTC залежить від стабільності кварцового резонатора. Звичайний кварц має похибку 20-100 ppm (1-5 хвилин на місяць), а мікросхеми з температурною компенсацією (DS3231) – 2-3 ppm (менше хвилини на рік).

2.1.4 Опис схеми годинника реального часу

Представлена схема являє собою повнофункціональний цифровий годинник реального часу з можливістю ручного встановлення часу. Схема складається з кількох функціональних блоків, які працюють разом для забезпечення точного відліку та відображення часу.

У лівій верхній частині схеми розташований генератор тактових імпульсів U10:A, який є елементом логіки HI (інвертор). У поєднанні з резистором R1 та конденсатором він утворює простий RC-генератор, що створює періодичні імпульси – основу для всієї схеми годинника. Частота генератора визначається номіналами резистора та конденсатора, а у реальній схемі годинника ця частота повинна бути точно підібрана або використовуватись кварцовий резонатор для забезпечення стабільності та точності.

Для перетворення високої частоти генератора в секундні імпульси використовуються лічильники-ділники U9 та U5 – мікросхеми типу 7493, які є 4-розрядними двійковими лічильниками і можуть ділити вхідну частоту на 2, 4, 8 або 16. Мікросхема U9 приймає сигнал безпосередньо від генератора і ділить його частоту, а вихідний сигнал надходить на мікросхему U5, яка продовжує ділення частоти. У результаті послідовного ділення на виході отримуємо імпульси з частотою 1 Гц – рівно один імпульс за секунду, який є "серцебиттям" годинника.

Лічильник секунд складається з двох декадних BCD-лічильників (Binary-Coded Decimal): U73 рахує одиниці секунд від 0 до 9, а U11 рахує десятки секунд від 0 до 5. Коли лічильник одиниць досягає значення 9 і приходить наступний імпульс, він скидається в 0 і генерує сигнал переносу на лічильник десятків. Коли

лічильник десятків показує 5, а одиниць – 9 (тобто 59 секунд), при наступному імпульсі обидва лічильники скидаються і генерується сигнал переносу на лічильник хвилин.

Структура блоку хвилин повністю аналогічна блоку секунд: U5 (нижній) рахує одиниці хвилин від 0 до 9, а U6 рахує десятки хвилин від 0 до 5. Цей блок приймає імпульси переносу від лічильника секунд, і кожен такий імпульс означає, що минуло 60 секунд і потрібно додати одну хвилину. При досягненні 59 хвилин лічильники скидаються і генерується сигнал переносу на лічильник годин.

Лічильник годин має дещо іншу структуру, оскільки потрібно рахувати не до 59, а до 23 у 24-годинному форматі або до 12 у 12-годинному форматі. U72 рахує одиниці годин від 0 до 9, а U11 рахує десятки годин від 0 до 2 для 24-годинного формату. У схемі реалізована додаткова логіка для автоматичного скидання лічильників при досягненні граничного значення (23:59 або 12:59), що досягається за допомогою логічних елементів U7:A та U7:B, які аналізують стан лічильників і формують сигнал скидання в потрібний момент.

Для відображення часу використовується велика LED-матриця U1, розташована у верхній правій частині схеми. Це семисегментний індикатор, який показує всі шість цифр часу: дві цифри годин, дві цифри хвилин і дві цифри секунд. Між лічильниками та індикатором знаходяться декодери-драйвери U2, U3, U4, U13, U14, U15 – мікросхеми типу 7447 або 7448, які перетворюють двійково-десятковий код (BCD) з лічильників у сигнали для управління сегментами індикатора. Кожний декодер відповідає за одну цифру на дисплеї. Наприклад, якщо лічильник одиниць секунд має значення 7 (код 0111 у двійковій системі), декодер U15 активує потрібні сегменти індикатора так, щоб на ньому відобразилась цифра «7».

У лівій частині схеми розташовані дві кнопки для ручного встановлення часу: SW1 «SET MINS» – кнопка встановлення хвилин та SW2 «SET HOURS» – кнопка встановлення годин. Кожна кнопка підключена через резистор (R2 для хвилин, R1 для годин), що запобігає дребезгу контактів. При натисканні кнопки генерується імпульс, який через логічні елементи U10:C (для хвилин) та U10:B (для годин)

надходить на відповідні лічильники. Кожне натискання кнопки «SET MINS» збільшує значення хвилин на одиницю, при досягненні 59 хвилин наступне натискання скидає лічильник у 0. Аналогічно працює кнопка "SET HOURS" для годин. Логічні елементи U7:A та U7:B необхідні, щоб під час ручного встановлення часу нормальний підрахунок імпульсів від генератора був тимчасово відключений, що запобігає конфліктам.

У схемі присутні конденсатори, які забезпечують фільтрацію живлення та стабільну роботу цифрових мікросхем – вони відфільтровують високочастотні завади та стабілізують напругу живлення. Також використовуються резистори підтяжки на входах логічних елементів, які необхідні для підтримки визначеного логічного рівня на входах, коли кнопки не натиснуті.

Протягом однієї хвилини в схемі відбувається наступне: генератор U10:A виробляє високочастотні імпульси, які через ділянки U9 та U5 перетворюються в секундні імпульси. Кожну секунду лічильники U73 (одиниці секунд) та U11 (десятки секунд) збільшують своє значення на одиницю, декодери U15 та U14 перетворюють ці значення в сигнали для індикатора, і ми бачимо, як змінюються цифри секунд на дисплеї: 01, 02, 03... 58, 59. Коли обидва лічильники секунд показують максимальне значення (5 і 9), наступний секундний імпульс скидає обидва лічильники секунд у 0, генерує сигнал переносу на лічильник хвилин, і лічильник одиниць хвилин збільшується на 1. На індикаторі ми бачимо, як час змінюється, наприклад, з 12:34:59 на 12:35:00. Аналогічний процес відбувається при переході від 59 хвилин до нової години.

Перевагами даної схеми є проста і зрозуміла структура, яка добре демонструє принципи цифрової схемотехніки, можливість ручного встановлення часу без необхідності чекати, використання стандартних широко доступних мікросхем серії 74xx, а також наочне відображення всіх розрядів часу на одному індикаторі. До особливостей можна віднести те, що схема не зберігає час при вимкненні живлення (відсутня батарея резервного живлення), точність залежить від стабільності RC-генератора, яка може бути недостатньою для довготривалого використання, та відсутність функції встановлення секунд.

2.2 Хід роботи

1. Ознайомитись із інформацією про різні способи реалізації годинників реального часу.
2. Зібрати схему за рис. 2.1. Провести моделювання її роботи.
3. Описати всі компоненти схеми у звіті, розібратись з принципом роботи схеми.

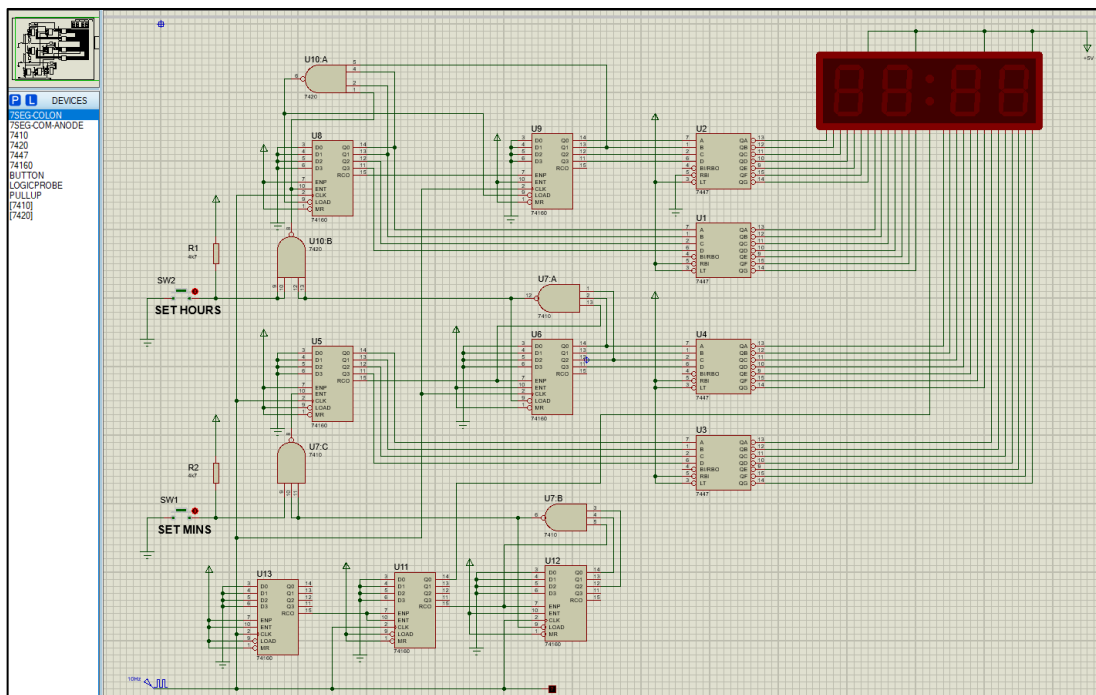


Рисунок 2.1 – Схема годинника реального часу

4. Роботу продемонструвати викладачу, за результатами виконання оформити і захистити звіт.

2.3 Зміст звіту

1. Мета роботи.
2. Скріншоти роботи схеми з поясненнями.
3. Висновки, в яких слід відобразити, що було зроблено в ході роботи, які навички отримані, які труднощі виникли та як їх було подолано, практичне значення вивченого матеріалу.
4. Відповіді на контрольні питання.

2.4 Контрольні питання

1. Що таке годинник реального часу і для чого він призначений?
2. Які існують способи реалізації RTC? У чому переваги та недоліки кожного?
3. Чому для RTC використовується саме частота 32768 Гц?
4. Як працює каскад дільників частоти?
5. Яким чином лічильник секунд передає сигнал лічильнику хвилин?
6. Як RTC враховує різну кількість днів у місяцях?
7. Що таке високосний рік і як RTC його враховує?
8. Які інтерфейси використовуються для підключення RTC-мікросхем?
9. Від чого залежить точність годинника реального часу?

3 ЛАБОРАТОРНА РОБОТА №3

ДОСЛІДЖЕННЯ МОДЕЛІ ПРИСТРОЮ КЕРУВАННЯ ДВИГУНОМ ПОСТІЙНОГО СТРУМУ

Мета роботи - ознайомлення із принципом керування двигуном постійного струму, дослідження роботи пристрою керування двигуном постійного струму з використанням перемикачів та реле.

3.1 Теоретичні відомості

3.1.1 Двигун постійного струму

Двигун постійного струму (ДПС) – це електрична машина, яка перетворює електричну енергію постійного струму в механічну енергію обертання. ДПС широко використовуються в робототехніці, автоматизації, електротранспорті та побутовій техніці завдяки простоті керування швидкістю та напрямком обертання [2, 3].

Основні характеристики двигунів постійного струму:

- напруга живлення – визначає номінальну робочу напругу двигуна;
- номінальний струм – струм, який споживає двигун при номінальному навантаженні;
- швидкість обертання – залежить від прикладеної напруги;
- крутний момент – зусилля, яке може створити двигун на валу;
- напрямок обертання – визначається полярністю підключення.

3.1.1 Принцип керування напрямком обертання

Напрямок обертання двигуна постійного струму визначається полярністю напруги, прикладеної до його обмоток. Якщо змінити полярність живлення (поміняти плюс і мінус місцями), двигун почне обертатися в протилежному напрямку. Це основний принцип реверсування двигуна.

Для зміни напрямку обертання використовуються різні схеми перемикування:

- механічні перемикачі та реле – прості та надійні рішення для нечастого перемикання;
- Н-міст (H-bridge) на транзисторах або мікросхемах – для швидкого електронного керування;
- контакторні схеми – для потужних промислових двигунів.

3.1.2 Схема з реле для реверсування двигуна

Реле – це електромагнітний перемикач, який дозволяє керувати потужним навантаженням за допомогою слабого керуючого сигналу. У схемі керування двигуном реле використовується для зміни полярності напруги, що подається на двигун.

Принцип роботи реле полягає в тому, що при подачі напруги на котушку реле виникає електромагнітне поле, яке притягує якір і перемикає контакти. Після зняття напруги з котушки пружина повертає контакти у вихідне положення.

Існує кілька основних варіантів організації керування двигуном постійного струму:

- схема з одним реле – найпростіший варіант, де реле перемикає полярність живлення двигуна. При увімкненні реле двигун обертається в одну сторону, при вимкненні – в іншу або зупиняється;

- схема з двома реле – забезпечує три режими роботи: обертання за годинниковою стрілкою, обертання проти годинникової стрілки та зупинка. Кожне реле відповідає за один напрямок обертання;

- схема з реле та перемикачами – комбінує автоматичне та ручне керування. Перемикачі дозволяють оператору безпосередньо задавати режим роботи, а реле забезпечують необхідну комутацію силових кіл;

- схема з кількома двигунами – дозволяє керувати декількома двигунами незалежно або синхронно, що необхідно в складних механізмах та роботах.

Представлена на рис. 3.1 схема є класичним прикладом керування одним двигуном постійного струму за допомогою двох перемикачів та одного реле. Схема містить наступні основні компоненти:

BAT1 – джерело живлення постійного струму (батарея), яке забезпечує електричну енергію для роботи двигуна. Напруга батареї визначає максимальну швидкість обертання двигуна;

M – двигун постійного струму, який є навантаженням схеми. Напрямок обертання двигуна залежить від полярності напруги, поданої на його виводи;

SW1 – перемикач верхнього рівня, який відповідає за один напрямок керування двигуном. При замиканні цього перемикача активується реле RV1;

SW2 – перемикач нижнього рівня, який забезпечує альтернативний спосіб керування. При замиканні цього перемикача двигун підключається до живлення іншим способом;

RV1 – реле з перемикаючими контактами, яке виконує функцію зміни полярності напруги, що подається на двигун. Котушка реле живиться через перемикач SW1, а силові контакти реле перемикають напрямок струму через двигун.

Принцип роботи схеми наступний: коли обидва перемикачі SW1 та SW2 розімкнені, двигун знеструмлений і не обертається. При замиканні SW1 спрацьовує реле RV1, його контакти перемикаються, і двигун підключається до джерела живлення з однією полярністю, обертаючись у визначеному напрямку. При замиканні SW2 (за умови, що SW1 розімкнений) двигун підключається до джерела живлення з протилежною полярністю, що викликає обертання в протилежному напрямку.

Важливо відзначити, що одночасне замикання обох перемикачів може призвести до короткого замикання або неправильної роботи схеми, тому в реальних пристроях передбачаються блокування, які запобігають такій ситуації.

3.2 Хід роботи

1. Ознайомитися з теорією і розібратися з принципом роботи двигунів постійного струму.

2. Зібрати схему за рис. 3.1 в Proteus. Перевірити роботу. Якщо джерело живлення буде під'єднано до кола з мотором, то мотор буде обертатися.

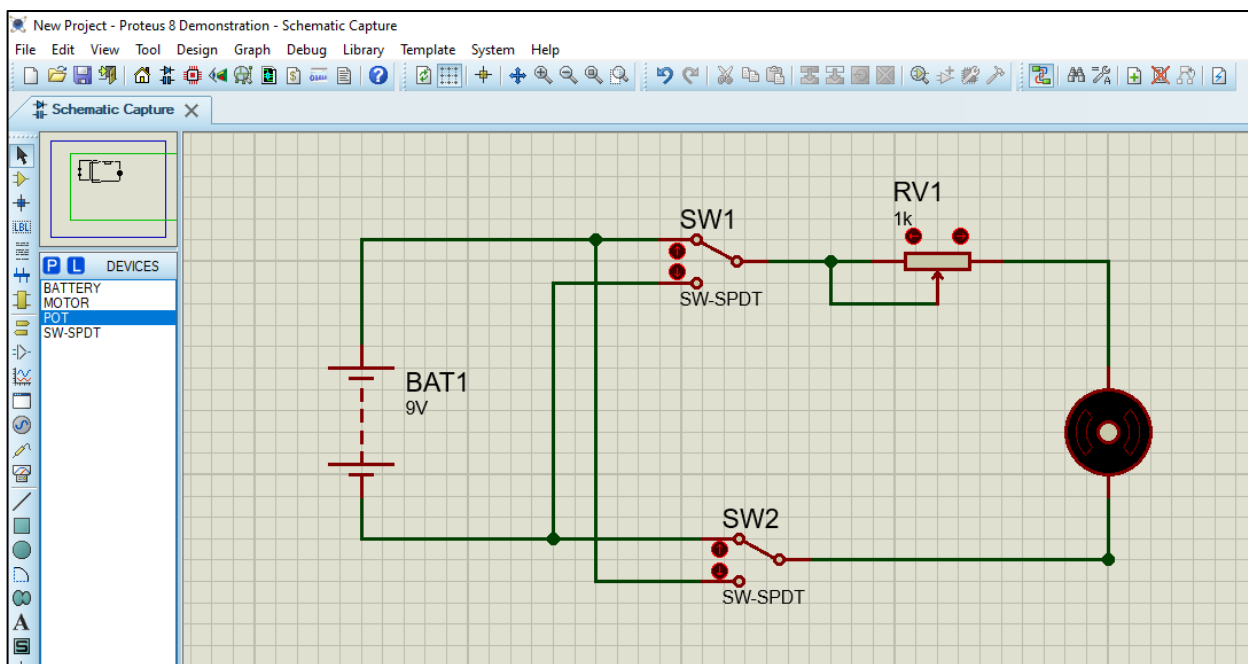


Рисунок 3.1 – Схема керування двигуном постійного струму

3. Провести моделювання роботи схеми у різних режимах:

- режим 1: Обидва перемикачі розімкнені – двигун не обертається;
- режим 2: Замкнений SW1, розімкнений SW2 – спостерігайте напрямок обертання двигуна;
- режим 3: Розімкнений SW1, замкнений SW2 – зафіксуйте зміну напрямку обертання;
- режим 4: Спробуйте замкнути обидва перемикачі одночасно – що відбувається?

Для кожного режиму зробіть скріншот та запишіть спостереження про напрямок обертання двигуна та стан реле.

4. Зберіть схему згідно з вашим варіантом завдання в табл. 3.1 (варіант обирається за номером у журналі).

Таблиця 3.1 – Варіанти завдань

Варіант	Завдання
1, 9, 17	Схема містить 3 мотори, 2 з яких обертаються за годинниковою стрілкою, а третій - проти. Всі мотори обертаються з однаковою швидкістю.
2, 10, 18	Схема містить 3 мотори, 2 з яких обертаються проти годинникової стрілки, а третій - за. Всі мотори обертаються з однаковою швидкістю.
3, 11, 19	Схема містить 3 мотори, 2 з яких обертаються за годинниковою стрілкою, а третій - проти. Третій мотор має меншу швидкість обертання.
4, 12, 20	Схема містить 3 мотори, 2 з яких обертаються проти годинникової стрілки, а третій - за. Другий і третій мотори мають меншу швидкість обертання.
5, 13, 21	Схема містить 4 мотори, 2 з яких обертаються за годинниковою стрілкою, а 2 - проти. Всі мотори обертаються з однаковою швидкістю.
6, 14, 22	Схема містить 4 мотори, перший і четвертий обертаються за годинниковою стрілкою, а другий і третій - проти. Третій і четвертий мотори мають меншу швидкість обертання.
7, 15, 23	Схема містить 3 мотори, 2 з яких обертаються проти годинникової стрілки, а третій - за. Всі мотори обертаються з різною швидкістю.
8, 16, 24	Схема містить 4 мотори, перший і четвертий обертаються за годинниковою стрілкою, а другий і третій - проти. Третій і четвертий мотори мають меншу швидкість обертання.

4. Роботу продемонструвати викладачу, за результатами виконання оформити і захистити звіт.

3.3 Зміст звіту

1. Мета роботи.
2. Скріншоти моделювання роботи схеми в різних режимах з поясненнями.
3. Скріншот схеми заданого варіанту з позначеннями всіх компонентів.
4. Висновки, в яких слід відобразити що було зроблено в ході роботи, які навички отримані, чи відповідає робота схеми заданому варіанту, які труднощі виникли та як їх було подолано.
5. Відповіді на контрольні питання.

3.4 Контрольні питання

1. Що таке двигун постійного струму і як він працює?
2. Від чого залежить напрямок обертання двигуна постійного струму?
3. Які способи зміни напрямку обертання двигуна ви знаєте?
4. Що таке реверсування двигуна?
5. Як працює реле? Опишіть його будову та принцип дії.
6. Які переваги та недоліки використання реле для керування двигунами?
7. Що таке H-міст і як він працює?
8. Чому в схемі з рисунка 3.1 не можна одночасно замикати обидва перемикачі?
9. Як можна змінити швидкість обертання двигуна постійного струму?
10. Що таке широтно-імпульсна модуляція (ШІМ) і як її використовують для керування двигунами?

ЛАБОРАТОРНА РОБОТА №4

ДОСЛІДЖЕННЯ МОДЕЛІ ПРИСТРОЮ КЕРУВАННЯ КРОКОВИМ ДВИГУНОМ

Мета роботи – ознайомлення із принципами керування кроковим двигуном, дослідження роботи пристрою керування кроковим двигуном та вивчення особливостей його підключення і конфігурації.

4.1 Теоретичні відомості

Крокові двигуни [2, 3], крім відмінностей у загальній конструкції, відрізняються ще й схемою включення обмоток. Є декілька варіантів їх конфігурацій, залежно від якої двигуни поділяються на уніполярні (англ. «Unipolar») і біполярні (англ. «Bipolar»).

Уніполярні крокові двигуни, так само як і біполярні, мають дві обмотки, і кожна з них має центральне відведення. Залежно від необхідного напрямку магнітного поля, в роботу включається відповідна половина обмотки, що досягається простим перемиканням ключів і істотно спрощує схему драйвера. Це дозволяє змінювати напрямок магнітного поля, створюваного обмоткою, перемиканням її половинок. Подібний механізм дозволяє в якості керуючої системи використовувати найпростіший уніполярний драйвер з чотирма ключами.

Як правило, уніполярний двигун має 6 виводів, але середні виводи обмоток можуть бути об'єднані в середині самого двигуна, тому такий двигун може мати й 5 виводів. Таким чином, якщо вам в руки потрапив невідомий двигун з шість чи п'ятьма виводами – це гарантовано уніполярний кроковий двигун.

З біполярними кроковими двигунами справа йде трохи інакше. Дані двигуни мають тільки одну обмотку в одній фазі. Щоб змінювати напрямок магнітного поля з метою змінити напрямок струму в обмотці керуюча схема біполярного двигуна повинна бути набагато складніша. Цього можна досягти за допомогою мостової схеми (H-bridge).

До того ж, для спрощення завдання можна придбати кілька драйверних чіпів, які вам допоможуть. Біполярні крокові двигуни, на відміну від уніполярних мають два виводи на одну фазу, жоден з яких не є спільним.

Біполярні крокові двигуни трохи складніше в управлінні, але при схожих габаритах, біполярний двигун здатний забезпечити більший момент, в порівнянні з уніполярним. Однак уніполярний двигун, на противагу біполярному, простіше в експлуатації, і цілком згодиться для приводу пристроїв з невеликою потужністю – побутова техніка (пральна машина, холодильник), магнітофони тощо.

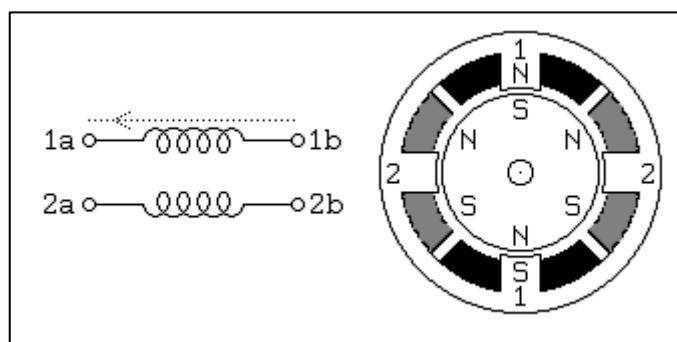


Рисунок 4.1 – Модель біполярного крокового двигуна

Біполярний кроковий двигун по суті такий самий принцип роботи, як уніполярний кроковий двигун, за винятком того, що протилежні котушки з'єднані паралельно, і тому пристрій має лише чотири клеми.

Двигун повинен керуватися за допомогою схеми, яка може пропускати струм через кожен пару котушок у будь-якому напрямку. Це досягається за допомогою драйвера H-Bridge для кожної котушки – найпростіше це можна реалізувати за допомогою мікросхем драйвера моста, таких як L298 або L6201/2/3.

Властивості та теорія роботи моделі біполярного крокового двигуна майже ідентичні властивостям моделі однополярного крокового двигуна.

Представлена на рис. 4.1 схема є системою керування біполярним кроковим двигуном за допомогою спеціалізованого драйвера L297 (контролер) та силового драйвера L298 (H-міст). Схема дозволяє керувати напрямком обертання, швидкістю та режимом роботи крокового двигуна.

Основні компоненти схеми:

– U1 (L297) – мікросхема-контролер крокового двигуна, яка генерує послідовність керуючих сигналів для обмоток двигуна. Ця мікросхема визначає логіку роботи та послідовність перемикавання обмоток залежно від режиму кроку (повний крок, напівкрок);

– U2 (L298) – подвійний Н-міст (силовий драйвер), який підсилює керуючі сигнали від L297 і забезпечує достатній струм для живлення обмоток крокового двигуна. Містить чотири виходи (OUT1-OUT4) для підключення чотирьох обмоток біполярного крокового двигуна;

– U1 (CLOCK) – генератор тактових імпульсів, який визначає швидкість обертання двигуна. Частота імпульсів прямо пропорційна швидкості обертання – чим вища частота, тим швидше обертається двигун.

Перемикачі (SW) – три перемикачі для керування режимами роботи:

– перший перемикач керує сигналом ENABLE (увімкнення/вимкнення драйвера);

– другий перемикач керує сигналом HALF/FULL (вибір режиму: повний крок або напівкрок);

– третій перемикач керує сигналом CONTROL (напрямок обертання – за годинниковою стрілкою або проти).

R1, R2 – резистори SENSE1 та SENSE2, які використовуються для вимірювання струму в обмотках двигуна. Вони підключені між виводами SENSE мікросхеми L298 і землею, забезпечуючи зворотний зв'язок по струму для захисту від перевантаження.

Принцип роботи схеми.

Генератор тактових імпульсів U1(CLOCK) постійно генерує імпульси, які надходять на вхід CLOCK мікросхеми L297. Кожен імпульс викликає перехід крокового двигуна на один крок (або напівкрок, залежно від налаштування).

Мікросхема L297 аналізує стан керуючих входів і генерує відповідну послідовність сигналів на чотирьох виходах (IN1, IN2, IN3, IN4). Ці сигнали визначають, які обмотки двигуна повинні бути активовані в даний момент часу.

Послідовність перемикання обмоток створює обертове магнітне поле, яке змушує ротор двигуна обертатися покроково.

Сигнали з виходів L297 надходять на входи мікросхеми L298, яка підсилює їх до рівня, достатнього для керування обмотками двигуна. L298 містить чотири силові ключі (два H-мости), які можуть пропускати струм до 2А на канал. Виходи OUT1-OUT4 мікросхеми L298 підключені безпосередньо до чотирьох обмоток біполярного крокового двигуна.

Режими роботи:

– режим ENABLE (увімкнення). При замиканні відповідного перемикача на вхід ENABLE подається високий рівень напруги (+5V), що активує драйвер. Двигун починає обертатися відповідно до тактових імпульсів. При розімкненні перемикача драйвер вимикається, і двигун зупиняється, утримуючи поточну позицію;

– режим HALF/FULL (напівкрок/повний крок). Цей вхід визначає режим кроку двигуна. При високому рівні (перемикач замкнений на +5V) L297 працює в режимі напівкроку – двигун робить вдвічі більше кроків на один оберт, що забезпечує плавніше обертання та вищу точність позиціонування. При низькому рівні (перемикач на землі) працює режим повного кроку – двигун робить стандартну кількість кроків, що забезпечує більший крутний момент;

– режим CONTROL (напрямок обертання). Стан цього входу визначає напрямок обертання двигуна. При високому рівні двигун обертається в одному напрямку (наприклад, за годинниковою стрілкою), при низькому – в протилежному (проти годинникової стрілки). L297 змінює послідовність активації обмоток залежно від цього сигналу.

4.2 Хід роботи

1. Ознайомитися з теорією і розібратися з принципом роботи крокових двигунів.

2. Зібрати схему за рис. 4.2.

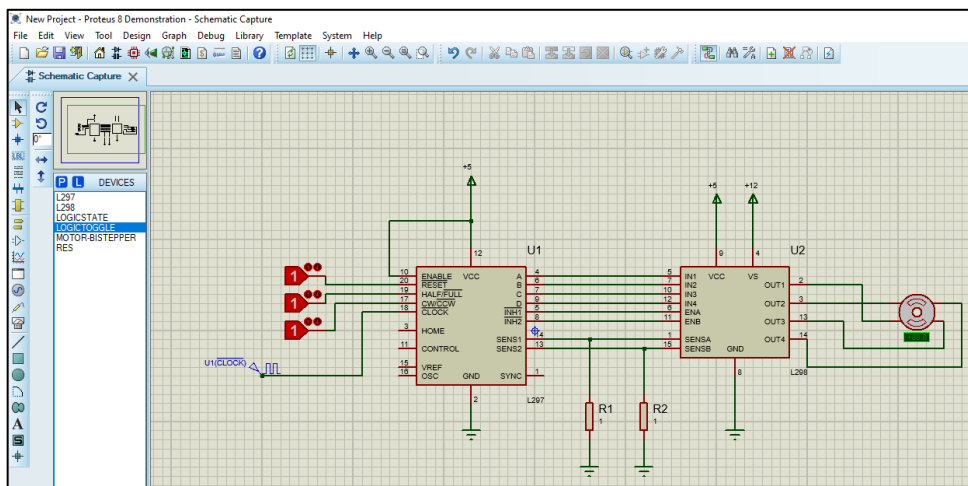


Рисунок 4.2 – Схема керування кроковим двигуном

3. Запустити і дослідити роботу схеми в різних режимах. Для кожного режиму зробити детальні записи спостережень, скріншоти та описати поведінку двигуна.

4. Знайти технічну документацію (datasheet) до мікросхем L297 та L298. Це можна зробити через пошук в Інтернеті, наприклад "L297 datasheet PDF" або на сайтах виробників електронних компонентів.

З документації вибрати та занести у звіт наступну інформацію:

Для L297:

- призначення мікросхеми;
- напруга живлення;
- опис всіх виводів (призначення кожного піна);
- внутрішня структурна схема;
- часові діаграми роботи в режимах повного та напівкроку;
- таблиці послідовностей перемикання виходів;
- типові схеми включення.

Для L298:

- призначення мікросхеми;
- максимальний струм на канал;
- діапазон робочих напруг;
- опис виводів;
- внутрішня схема Н-мосту;

- принцип роботи захисту по струму;
- типові схеми включення;
- рекомендації щодо вибору резисторів SENSE.

5. Роботу продемонструвати викладачу, за результатами виконання оформити і захистити звіт.

4.3 Зміст звіту

1. Мета роботи.
2. Скріншоти моделювання роботи схеми в різних режимах з поясненнями.
3. Витяги з datasheet мікросхем L297 та L298.
4. Висновки, в яких слід відобразити що було зроблено в ході роботи, що було зроблено в ході лабораторної роботи, які знання та навички отримані, чи досягнута мета роботи, які особливості роботи крокових двигунів та їх драйверів виявлені.
- 5 Відповіді на контрольні питання.

4.4 Контрольні питання

1. Що таке кроковий двигун і чим він відрізняється від звичайного двигуна постійного струму?
2. Які основні типи крокових двигунів існують? У чому їх різниця?
3. Що таке уніполярний кроковий двигун? Скільки виводів він має?
4. Що таке біполярний кроковий двигун? Чому для нього потрібен Н-міст?
5. Який кроковий двигун (уніполярний чи біполярний) забезпечує більший крутний момент при однаковому розмірі? Чому?
6. Що таке режим повного кроку (full step)? Опишіть послідовність активації обмоток.
7. Що таке режим напівкроку (half step)? Які його переваги порівняно з повним кроком?
8. Від чого залежить швидкість обертання крокового двигуна?

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Волканін Є.Є. Методичні матеріали до лабораторних занять із навчальної дисципліни «Мікроконтролери пристроїв електромеханічних систем» – Кременчук. 2023. – 30 с.
2. Valvano, J. W. Embedded Systems: Introduction to ARM®Cortex™-M Microcontrollers. 2014. 594 p.
3. Yeadon, W. H., & Yeadon, A. W. Handbook of Small Electric Motors. New York: McGraw-Hill. 2003. 1137 p.