

Лабораторна робота № 1

Моделювання цифрових логічних схем

Мета роботи: Ознайомлення з основними можливостями пакета програм автоматизованого проектування електронних схем Proteus. Моделювання роботи простих логічних схем.

Вступ

Система автоматизованого проектування Proteus розробляється компанією Labcenter Electronics. Proteus дозволяє проводити схемотехнічне моделювання, симуляцію роботи побудованих схем та створювати дизайн друкованих плат. У даному курсі лабораторних робіт використовується версія Proteus 8 Professional.

Даний пакет програм включає в себе такі підпрограми: ISIS і ARES. ISIS – це програмне забезпечення для синтезу і моделювання електронних схем в реальному часі. ARES використовується для розробки друкованих плат.

ISIS має велику кількість компонентів в бібліотеці, серед них: джерела, генератори, інструменти для аналізу (осцилоскоп, вольтметр, амперметр тощо), індикатори (probes) для моніторингу параметрів схеми в реальному часі, дискретні компоненти такі, як резистори, конденсатори, трансформатори, сенсори тощо. Імітація реальної поведінки електронних компонентів і схем в програмі ISIS базується на застосуванні SPICE-моделювання поведінки компонентів. Мова SPICE (Simulation Program with Integrated Circuit Emphasis) була розроблена в Каліфорнійському університеті Берклі на початку 70-х років минулого століття і стала де-факто основною для більшості програм, котрі симулюють поведінку електронних компонентів. У Proteus використовується не PSPICE (розроблена компанією MicroSim Corporation, яку у 1998 році поглинув OrCAD), як в інших подібних пакетах. ISIS ProSPICE базується на SPICE3F5.

ISIS підтримує як інтерактивну симуляцію в реальному часі, так і симуляцію за допомогою графіків, що відтворюють сигнали у визначених точках схеми зі встановленими там індикаторами. В будь-якому разі поведінка схеми вираховується за допомогою вбудованого у програму симулятора ProSPICE.

ARES пропонує моделювання друкованих плат. В якості автоматичного вбудованого трасувальника використовується програма ELECTRA Autorouter.

Запуск пакета

Proteus являється комерційним продуктом. Проте наявна безкоштовна демонстраційна версія. Вона володіє всіма функціями і можливостями платного пакету, але не дозволяє зберегти або роздрукувати результати роботи чи створити свій мікроконтролер.

Графічний ввід електронної схеми у Proteus

Proteus використовує стандартні засоби управління операційної графічної оболонки Microsoft Windows. Головне вікно містить робоче поле із нанесеною на нього сіткою для графічного конструювання схем, головного меню та кількох панелей інструментів. Графічний ввід цифрових схем розглянемо на конкретному прикладі:

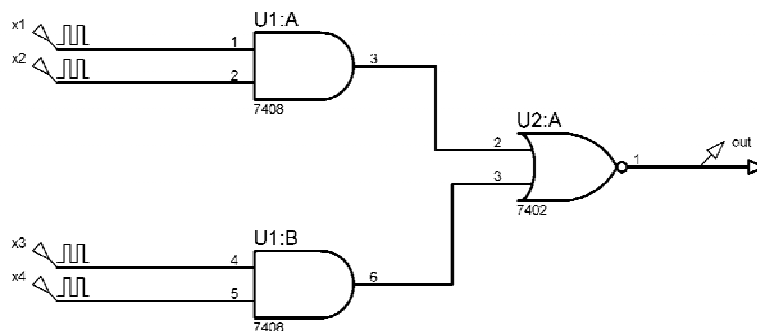


Рис. 1.1 Схема реалізації цифрової схеми 2I-2АБО-НІ на логічних елементах 2АБО-НІ та 2I.

На рис. 1.1 зображено ділянку робочого поля Proteus, що містить введену та відредаговану цифрову схему елемента 2I-2АБО-НІ. Для отримання схеми у такому вигляді слід провести наступну послідовність операцій:

- розмістити у необхідних місцях елементи схеми;
- задати параметри елементів схеми;
- з'єднати елементи схеми провідниками;
- задати назви вузлів схеми.

Розглянемо і прокоментуємо ці операції. Схема, що її наведено вище, містить лише стандартні елементи, наявні у вбудованих бібліотеках пакету. Для розміщення елемента на робочому полі його спочатку слід отримати з відповідної бібліотеки.

Отримати компоненти з бібліотеки можна кількома способами:

- У верхній частині **Object Selector** натиснути на кнопку **P**. Таким самим чином працює пункт **Pick parts from libraries** розділу головного меню **Library** (гаряча клавіша **P** на клавіатурі).

- Правою клавішею миші натиснути на порожню ділянку поля для побудови схеми і вибрати **Place – Component – From Libraries**.

Обидва способи приведуть до появи вікна пошуку елемента. Якщо назва елемента відома (наприклад, резистор - R), її можна безпосередньо набрати з клавіатури в поле **Keywords**.

Для безпосереднього перегляду наявних бібліотек елементів можна скористатися кнопкою **L** у верхній частині **Object Selector (Library – Library Manager** головного меню). Зазначимо, що в стандартних бібліотеках пакету використовуються умовні графічні позначення елементів, прийняті в західних країнах. Наводиться також повний список наявних елементів

Розглянемо конкретну схему на рис. 1.1. Вона містить два логічних елемента 2I (модель 74x08 з бібліотеки **74STD.lib**) та один логічний елемент 2АБО-НЕ (модель 74x02 з бібліотеки **74STD.lib**). Для задання вхідних сигналів використовується генератор тактових імпульсів **DCLOCK** (натисніть кнопку **Generator Mode** та виберіть дане джерело з списку).

Безпосередньо вибрані з бібліотек елементи після встановлення на робоче поле мають стандартні орієнтацію, назву та параметри.

Змінити орієнтацію елемента (наприклад, з горизонтальної на вертикальну) можна за допомогою іконок **Rotate** (повернути на 90 градусів) та **Mirror** (дзеркальне відображення), що розташовуються зліва від вікна **Device Selector**. Назва та параметри елемента змінюються подвійним клацанням мишею на відповідному елементі або командою **Edit Properties** контекстного меню елемента, після чого з'являється вікно із переліком параметрів.

Після встановлення елементів на робоче поле їх слід з'єднати провідниками у відповідності до принципової схеми. Для цього слід клацнути мишею на виводі елемента який буде під'єднуватися, а потім клацнути по одному разу в кожному місці, де провідник на принциповій схемі повертає під прямим кутом аж до досягнення виводу другого під'єднуваного елемента або провідника.

Після проведення усіх провідників схема є практично готовою до моделювання, але для зручності слід встановити конкретні назви для провідників. Для встановлення назви потрібно правою клавішею миші клацнути на відповідному провіднику і обрати **Place Wire Label** в контекстному меню (або використати **Wire Label Mode**), що призведе до появи вікна, у якому слід задати назву провідника.

Усі елементи на робочому полі мають унікальне значення (value) і характеристику (reference). Унікальна і послідовна анотація присвоюється кожному компоненту, якщо вручну її не перевизначити. Для зміни потрібно використати команду **Edit Properties** контекстного меню елемента. У діалоговому вікні **Edit Component**, яке з'явиться можна бачити, що є змога

змінити значення Part Reference та Part Value. У випадку елемента АБО-НЕ на рис.1.1 Part Reference=U2, Part Value=7402. Крім цього, може бути корисним, особливо у схемах із щільно розташованими елементами, приховати деякі дані, котрі вказуються поряд з елементом на робочому полі. Це також можна змінити у даному діалоговому вікні.

Окрім зазначених вище, при графічному конструюванні схеми можуть застосовуватися й інші команди, які будуть діяти для поточного (виділеного червоним кольором) елемента. Зокрема, у розділі **Edit** основного меню:

<i>Undo Changes</i>	– скасувати останню дію
<i>Redo Changes</i>	– відмінити скасування
<i>Cut to Clipboard</i>	– забрати елемент з робочого поля і запам'ятати його в буфері
<i>Copy to Clipboard</i>	– запам'ятати елемент у буфері
<i>Paste from Clipboard</i>	– встановити на робоче поле елемент з буфера
<i>Copy to Clipboard</i>	– запам'ятати елемент у буфері обміну Windows для подальшої його вставки в інший Windows – редактор (наприклад, Microsoft Word)
<i>Select All Objects</i>	– виділити всі елементи
<i>Clear Selection</i>	– відмінити виділення елементів
<i>Find/Edit Component</i>	– знайти елемент
<i>Align Objects</i>	– порівняти елементи один відносно іншого.

Після побудови схеми для можливості її подальшого використання слід зберегти її на диску. Для цього використовується команди **Save Project** та **Save Project As** розділу **File**. Раніше збережену схему можна відкрити командою **Open Project**. Повністю нову схему можна почати командою **New Project**.

Зазначимо, що для зручності найбільш часто застосовні операції продубльовано також у вигляді кнопок на панелях інструментів. Зокрема, у вигляді кнопок присутня більшість вищезазначених операцій. Для визначення операції слід навести

мишу на кнопку і потримати там вказівник миші, не натискаючи на її кнопку. В результаті з'явиться підказка.

Моделювання схеми у Proteus

Після завершення побудови схеми можна переходити до її моделювання. Для дослідження і аналізу даних емуляції електричних ланцюгів в ISIS є спеціальні інструменти, одним з яких є об'єкт графік, який використовується для перегляду результатів емуляції і отримання вимірів.

Графік – це об'єкт, який можна розмістити в проекті. Його призначення – керувати частковою симуляцією і відображати результати цієї симуляції. Вид аналізу, що виконується симуляцією, визначається типом розміщеного графіка. Частина проекту, яка симулюється, і дані, які відображені на графіку, визначаються такими об'єктами, як індикатори і генератори, які були додані на графік. Необхідно відзначити, що в один проект схеми може бути додано кілька об'єктів графік, розмір кожного з яких можна змінювати.

В ISIS є 14 видів аналізу даних моделювання. Для виконання аналізу необхідні такі об'єкти редактора як графік, генератори та індикатори. Для розміщення графіка потрібно використати **Graph Mode**, обрати потрібний тип графіка (наприклад, **DIGITAL** для цифрового аналізу перехідних процесів) і помістити елемент на робоче поле.

Види аналізу в ISIS і відповідні їм типи графіків наступні:

Аналоговий аналіз перехідних процесів	<i>ANALOGUE</i>
Цифровий аналіз перехідних процесів	<i>DIGITAL</i>
Аналіз перехідного процесу змішаного режиму	<i>MIXED</i>
Частотний аналіз	<i>FREQUENCY</i>
Аналіз розгортки на постійному струмі	<i>DC SWEEP</i>
Аналіз розгортки на змінному струмі	<i>AC SWEEP</i>
Аналіз передаточної кривої на постійному струмі	<i>TRANSFER</i>
Аналіз шумів	<i>NOISE</i>
Аналіз спотворень	<i>DISTORTION</i>

Аналіз Фур'є
Аудіо аналіз
Інтерактивний аналіз
Цифровий аналіз відповідності

FOURIER
AUDIO
INTERACTIVE
CONFORMANCE

Генератори використовуються для подачі тестових сигналів в досліджувану схему. Індикатори розміщуються в тих точках схеми, за якими ми хочемо спостерігати. Вибрати потрібний індикатор можна на панелі **PROBES**, яка може бути відкрита за допомогою натискання на кнопку **Probe Mode** на лівій панелі інструментів редактора ISIS. Вибрати потрібний генератор можна на панелі **GENERATORS**, яка відкривається за допомогою натискання на кнопку **Generator Mode** на лівій панелі інструментів редактора ISIS.

Індикатори і генератори необхідно додати на графік, налаштувати параметри аналізу і виконати аналіз за допомогою команди контекстного меню **Simulate Graph** (попередньо потрібно виділити за допомогою лівої кнопки миші в проєкті схеми потрібний графік).

Порядок дій при виконанні аналізу схеми може бути представлений у вигляді наступних кроків:

- Розміщення в робочому полі проєкту графіка відповідного типу.
- Розміщення і підключення до досліджуваної схеми генераторів для подачі тестових сигналів.
- Розміщення і підключення до досліджуваної схеми індикаторів. Індикатори підключаються до тих точок схеми, за якими ми хочемо спостерігати.
- Додавання генераторів та індикаторів на графік, для відображення даних які вони генерують/фіксують.
- Налаштування параметрів аналізу.

Для кожного аналізу можна налаштовувати певні параметри, змінні для аналізу, налаштування моделювання. Можливість налаштування опцій аналізу з'являється після розміщення відповідного аналізу графіка в робочому полі проєкту. Для того щоб відкрити вікно налаштувань, необхідно за допомогою лівої кнопки миші виділити графік, за допомогою правої кнопки миші

викликати контекстне меню і вибрати в ньому пункт **Edit Graph** або **Edit Properties**. Інтерфейс вікна налаштувань для кожного виду аналізу відрізняється. Задані параметри зберігаються разом з файлом схеми, тому навіть в тому випадку, коли файл буде відкритий на іншому комп'ютері, налаштування будуть використані ті ж.

Для відображення на графіку даних, які генерують/фіксують генератори та індикатори, необхідно додати йому ці об'єкти. При цьому графік повинен бути розміщений в робочому полі проекту. Кожен графік може відображати кілька кривих. Кожна крива відображає дані, асоційовані з одним генератором або індикатором. Для аналогових і змішаних типів графіків є можливість відобразити у вигляді окремої кривої дані, отримані в результаті виконання математичної функції. За замовчуванням назви кривих співпадають з назвами об'єктів, дані яких вони відображають. За необхідності назви кривих можна змінити.

Для додавання генераторів/індикаторів на графік необхідно за допомогою лівої кнопки миші виділити піктограму об'єкта на схемі і перетягнути її за допомогою миші на графік. В результаті чого на графіку відобразиться назва доданого генератора/індикатора. Також для додавання генераторів/індикаторів на графік можна скористатися командою **Add Traces**. Для того щоб викликати цю команду, необхідно за допомогою лівої кнопки миші виділити графік, на який передбачається додати об'єкти, за допомогою правої кнопки миші викликати контекстне меню і вибрати в ньому пункт **Add Traces**. В результаті цих дій буде відкрите діалогове вікно **Add Transient Trace**, яке містить наступні поля введення:

<i>Name</i>	– ім'я нової кривої;
<i>Probe P1–P4</i>	– вибір генераторів/індикаторів для відображення даних;
<i>Expression</i>	– в якості змінної для відображення даних може бути використано вираз;
<i>Trace Type</i>	– тип кривої: Analog (аналогова), Digital (цифрова), Phasor (фазовий вектор), Noise (шум)

Axis – вісь Y: Left (ліва), Right (права), Reference (опорна)

Після того як всі параметри у вікні **Add Transient Trace** налаштовані, потрібно натиснути кнопку ОК. В результаті чого обрані генератори/індикатори будуть додані у вікно графіка.

Моделювання цифрових логічних схем.

Цифрові інтегральні мікросхеми (ІМС) виконані за певною технологією (ТТЛ, КМОН тощо) випускаються у вигляді серій. У межах однієї серії ІМС мають однакові параметри (серед них напруга живлення, рівні логічних одиниці та нуля, гранична частота, здатність до навантаження та ін.) і є повністю сумісними між собою. Цифрові пристрої, виконані на ІМС лише однієї серії, не потребують внутрішніх схем узгодження і, як правило, найбільш стабільні у своїй роботі. Найбільш поширеними є ІМС, випущені за різними модифікаціями транзисторно-транзисторної логіки (ТТЛ). Серед ІМС, що випускалися у колишньому СРСР це мікросхеми серій 133 та 155 (власне ТТЛ), а також 555, 1531, 1533 та ін. (ТТЛ з діодами Шотткі - ТТЛШ). Мікросхеми усіх зазначених серій сумісні між собою за логічними рівнями. За стандартами, прийнятими у США, позначення ІМС, виконаних за ТТЛ-технологією, починається з числа 74 (74-а серія). Повне ж позначення виглядає як 74хNNN, де х - літери, що позначають особливості даної модифікації ТТЛ (і, іноді, фірму-виробника), зокрема, відсутність літери позначає просту ТТЛ-технологію, а позначення S або LS - технологію ТТЛШ. Цифри NNN означають номер моделі конкретної ІМС. Перелічимо деякі логічні ІМС 74-ї серії:

- 74х00 - базовий елемент 2І-НЕ
- 74х10 - логічний елемент 3І-НЕ
- 74х20 - логічний елемент 4І-НЕ
- 74х30 - логічний елемент 8І-НЕ
- 74х02 - логічний елемент 2АБО-НЕ
- 74х27 - логічний елемент 3АБО-НЕ
- 74х08 - логічний елемент 2І

74x32 - логічний елемент 2АБО

74x04 - інвертор (логічний елемент НЕ)

74x51 - логічний елемент 2І-2АБО-НЕ

74x86 - логічний елемент Виключне АБО на 2 входи

Для задання тестових вхідних сигналів використовуються генератори. Обрати потрібний генератор можна на панелі **GENERATORS** після натиснення на кнопку **Generator Mode** на лівій панелі інструментів. В ISIS існує 14 типів генераторів. Серед них варто виділити DC – джерело постійної напруги, SINE – генератор синусоїдальної напруги, PULSE – аналоговий імпульсний генератор, EXP – експоненціальний імпульсний генератор, DSTATE – встановлений логічний рівень, DEDGE – одиничний перехід від одного рівня до іншого, DPULSE – джерело одиничного цифрового тактового імпульсу, DCLOCK – джерело цифрового тактового сигналу, DPATTERN – джерело довільної послідовності логічних рівнів.

Після їхнього встановлення на схему слід задати параметри відповідного сигналу через пункт контекстного меню Edit Properties.

Лабораторне завдання

1. Уважно ознайомтесь з наведеним вище теоретичним матеріалом.

2. У Proteus введіть схему, зображену на рис 1.1. Проведіть симуляцію роботи цієї схеми.

3. Схема на рис. 1.1 фактично аналогом елемента 2І-2АБО-НЕ (74x51). Замініть в схемі дані елементи на схему 74x51. Проведіть симуляцію роботи цієї схеми, порівняйте її роботу з роботою попередньої схеми.

4. Зберіть схему «Виключне АБО» на елементах І-НЕ та повторіть для неї пункт 3.

5. Зберіть, та проведіть симуляцію роботи двох схем, згідно з Вашим варіантом. При проектуванні схем поведіть пошук оптимальних схем.

Варіанти завдання:

Варіант завдання визначається десятьма молодшими розрядами числа, даного викладачем, представленого в двійковій системі числення (h_{10}, h_9, \dots, h_1).

Наприклад: число 2345 перекладаємо у двійкову систему числення і отримуємо двійкове число:

$$2345_{(10)} = 100100101001_{(2)}.$$

Десять молодших розрядів отриманого двійкового числа, поданих у табл. 1.1, застосовуємо для визначення варіанта завдання в табл. 1.2 – 1.6.

Таблиця 1.1.

ВИЗНАЧЕННЯ ВАРІАНТА ЗАВДАННЯ

0	1	0	0	1	0	1	0	0	1
h_{10}	h_9	h_8	h_7	h_6	h_5	h_4	h_3	h_2	h_1

Відповідно до визначеного варіанта та таблиці істинності (табл. 1.2) визначається вихідна система з двох перемикальних функцій. Набір логічних елементів, які можна використовувати для побудови комбінаційних схем, надані в табл. 1.3. При емуляції роботи рекомендовано використовувати схеми 74-ї серії.

Таблиця 1.2.

ТАБЛИЦЯ ІСТИННОСТІ ПЕРЕМИКАЛЬНИХ ФУНКЦІЙ

x_4	x_3	x_2	x_1	f_1	f_2
0	0	0	0	1	1
0	0	0	1	h_7	1
0	0	1	0	1	1
0	0	1	1	0	0
0	1	0	0	x	0
0	1	0	1	0	0
0	1	1	0	1	x
0	1	1	1	x	x
1	0	0	0	1	h_4
1	0	0	1	0	0

1	0	1	0	0	0
1	0	1	1	<i>h1</i>	0
1	1	0	0	1	x
1	1	0	1	<i>h2</i>	<i>h5</i>
1	1	1	0	<i>h3</i>	<i>h6</i>
1	1	1	1	1	1

Таблиця 1.3.

ЕЛЕМЕНТНИЙ БАЗИС

<i>h10</i>	<i>h9</i>	<i>h8</i>	Логічні елементи
0	0	0	3І-НЕ, 2І
0	0	1	3І, 4І-НЕ
0	1	0	3АБО, 4І, НЕ
0	1	1	3І, 2АБО, НЕ
1	0	0	2АБО-НЕ, 4І
1	0	1	2І-НЕ, 4АБО
1	1	0	3АБО-НЕ, 3І
1	1	1	3І-НЕ, 2І-2АБО-НЕ

7. Контрольні питання:

1. В якій бібліотеці знаходиться логічні ІМС 74-ї серії?
2. Як записується досконала нормальна диз'юнктивна форма?
3. Як записується досконала нормальна кон'юнктивна форма?
4. Як провести симуляцію роботи Вашої схеми?
5. Як побудувати схему заданої функції на елементах 74х30?
6. В чому різниця між ДНФ, КНФ та ТДНФ?
7. Який Ви знаєте алгоритм запису таблично-заданої функції через базис Пірса?
8. Переведіть задане рівняння в базис І-НЕ, АБО-НЕ?

Лабораторна робота № 2

Дешифратори та демультимплексори

Мета роботи: Ознайомитись з принципом роботи та навчитися проектувати комбінаційні схеми: дешифратора, шифратора, демультимплексора, тощо.

Дешифратори та демультимплексори є функціональними цифровими пристроями і призначаються для виділення окремого виходу (дешифратори) або розподілу сигналу з одного входу по багатьох виходах (демультимплексори) в залежності від заданого коду. На рис. 2.1 зображено схему дешифратора на 2 кодових входи та 4 інверсні виходи на логічних елементах серії 74.

Для задання тестових вхідних сигналів використовуються стандартні генератори сигналів. Після встановлення на схему вхідних сигналів слід задати параметри відповідного сигналу. Список доступних типів генераторів включає генератор постійного струму (DC), генератор імпульсів (PULSE), генератор синусоїдальних коливань (SINE), генератори прямокутних імпульсів (DSTATE, DEDGE, DPULSE, DCLOCK, DPATTERN) тощо. На рис.2.1 використовуються джерела цифрового тактового імпульсу DCLOCK: U1:A(A) та U1:B(A).

У даній схемі використовується елемент «шина» (bus). Після її побудови слід визначити опис шини. В описі шини слід перерахувати назви усіх провідників, що входять у шину. Опис має вигляд $x[1..n]$, де x – це літера назви вузла, n – це індекс останньої назви. Через знак $|$ у такому ж форматі можна вказати назви, котрі починаються іншою літерою. Лінія, що під'єднується до шини, обов'язково повинна мати назву, що входить до опису шини.

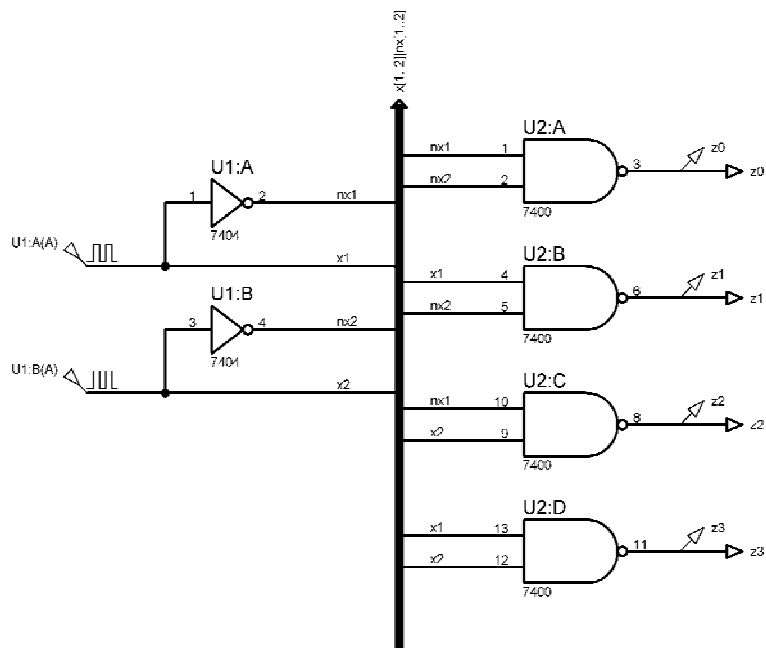


Рис. 2.1 Схема дешифратора

Для того, щоб провести цифрове моделювання даної схеми, використовується граф DIGITAL для цифрового аналізу перехідних процесів.

Для додавання генераторів/індикаторів на графік необхідно за допомогою лівої кнопки миші виділити піктограму об'єкта на схемі і перетягнути її за допомогою миші на графік. В результаті чого на графіку відобразиться назва доданого генератора/індикатора. Також для додавання генераторів/індикаторів на графік можна скористатися командою Add Traces контекстного меню. Отриманий граф дозволяє побачити часову діаграму у рівнях логічних одиниці та нуля для обраних сигналів.

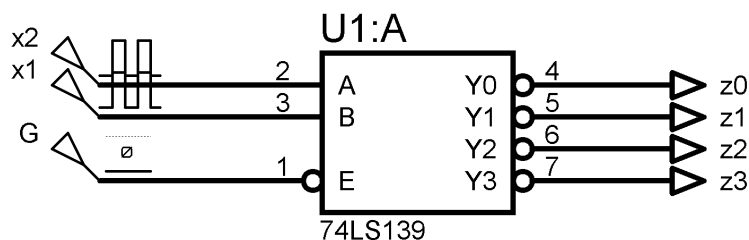


Рис. 2.2 Схема дешифратора/демультиплексора.

Стандартні серії цифрових ІМС містять деякі дешифратори та демультиплексори у інтегральному виконанні. Зазначимо, що, оскільки ці пристрої мають дуже схожі схеми, у вигляді ІМС як правило виробляється комбінована схема. Схема на рис. 2.2 функціонально повторює рис. 2.1. Інтегральна мікросхема 74LS139 - дешифратор\демультиплексор на 2 входи та 4 інверсні виходи. Крім 2-х кодових входів (вони позначаються літерами А-В), ці ІМС мають вхід синхронізації\демультиплексування G. При роботі в режимі дешифратора на цей вхід подається постійний нуль (вхід інверсний). У режимі демультиплексора на цей вхід подається сигнал, який необхідно розподілити по вихідних лініях.

На рис. 2.3 зображено схему демультиплексора на 2 адресні входи, 1 інформаційний вхід та 4 виходи даних з трьома станами. Третій стан виходу для цифрових схем означає встановлення великого вихідного опору схеми таким чином, що схема перестає впливати на своє навантаження. У даному демультиплексорі використані вентиля з трьома станами 74125. Ці схеми пропускають на вихід сигнал з інформаційного входу, якщо на керуючому вході встановлений нуль. У протилежному випадку вихід 74125 встановлюється в третій стан. Як джерело інформаційного сигналу використовується джерело G, частоту якого слід встановити такою, щоб спостерігати встановлення на відповідних виходах як нуля так і одиниці.

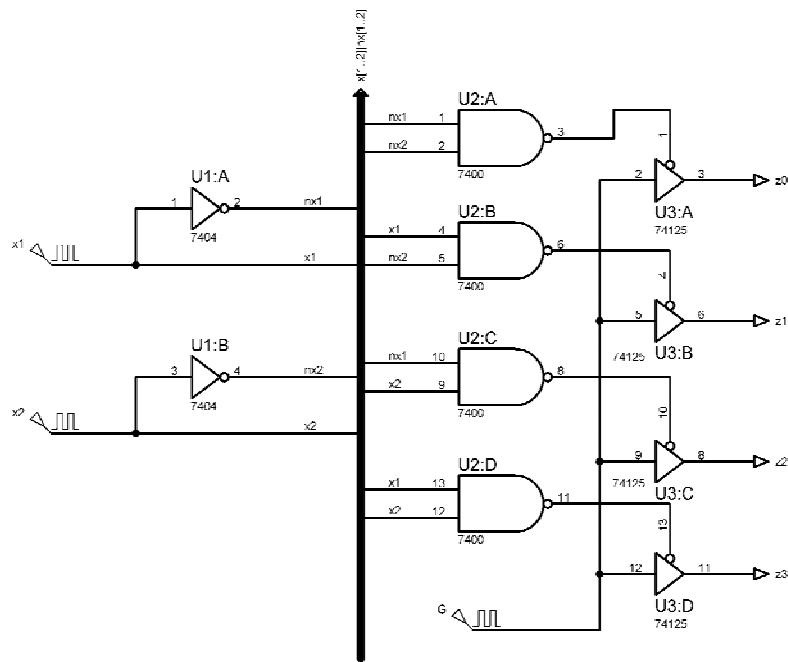


Рис. 2.3 Схема демультиплексора з трьома станами.

Лабораторне завдання

1. Введіть у Proteus схему рис. 2.1. Вважайте, що тривалість тактового імпульсу 100 мкс, його період 200 мкс. Затримка першого фронту повинна бути рівною тривалості імпульсу, тобто складати 100 мкс для тактових імпульсів. Тривалість, період та затримка імпульсу першого джерела повинна бути рівною відповідним параметрам для тактових імпульсів, для другого джерела усі ці параметри повинні бути вдвічі більшими (для третього - ще вдвічі більшими і т. д.). Таким чином можна перебрати усі можливі комбінації вхідних сигналів.

2. Виведіть графік залежності вхідних та вихідних цифрових сигналів дешифратора від часу та поясніть ці залежності.

3. Введіть у Proteus схему рис. 2.2., у якій застосовано інтегральний дешифратор\демультиплексор 74LS139. За аналогічних вхідних сигналів проведіть моделювання для цієї схеми у режимі дешифратора. Виведіть відповідні графіки для вхідних та вихідних сигналів та поясніть ці залежності.

4. Додайте до схеми рис. 2.2 джерело цифрового сигналу на вхід демультиплексування і проведіть моделювання схеми 74LS139 у режимі демультиплексора. Виведіть відповідні графіки для вхідних та вихідних сигналів та поясніть ці залежності.

5. Введіть у Proteus схему демультиплексора з трьома станами (рис. 2.3) і проведіть її моделювання зі вхідними сигналами, аналогічними до п.3. Виведіть графіки для вхідних та вихідних сигналів та поясніть ці залежності.

6 Згідно свого варіанту розробити схему мультиплексора або шифратора (кодера)

Варіанти завдань:

Таблиця 2.1.

$h1$	$h2$	$h3$	Схема	Кількість інформаційних входів
0	0	0	мультиплексор	4
0	0	1	мультиплексор	5
0	1	0	пріоритетний шифратор	6
0	1	1	шифратор	7
1	0	0	пріоритетний шифратор	4
1	0	1	мультиплексор	5
1	1	0	шифратор	6
1	1	1	шифратор	7

Таблиця 2.2.

$h4$	$h5$	$h6$	Логічні елементи
0	0	0	3І-НЕ
0	0	1	4І-НЕ
0	1	0	3АБО, 4І, НЕ
0	1	1	3І, 2АБО, НЕ
1	0	0	2АБО-НЕ, 4І
1	0	1	2І-НЕ, 4АБО
1	1	0	3АБО-НЕ
1	1	1	3І, 2АБО, НЕ

Проведіть моделювання роботи отриманої Вами схеми. Виведіть графіки для входних та вихідних сигналів та поясніть ці залежності.

7. Побудуйте демультиплексор, що має 8 виходів. При розробці схеми використовуйте мікросхеми 74LS139. Промодельюйте роботу Вашого демультиплексора.

8. Згідно свого варіанту розгляньте мікросхему та промодельюйте її роботу. Розробіть аналогічну схему на простих елементах 74 серії.

Варіанти завдань:

Таблиця 2.3.

$h7$	$h8$	$h9$	Мікросхема
0	0	0	74ls47
0	0	1	74ls257
0	1	0	74ls283
0	1	1	74LS138
1	0	0	74154
1	0	1	7483A
1	1	0	74HC238
1	1	1	74SL251

Контрольні питання:

1. Дайте визначення дешифратора.
2. Дайте визначення шифратора.

3. Як дешифратори і демультимплексор позначаються на схемах?
4. Чим відрізняється робота шифратора від пріоритетного шифратора?
5. Як за допомогою пріоритетного шифратора реалізувати схему шифратора?
6. Дайте визначення мультимплексора.
7. Як за допомогою демультимплексора реалізувати схему шифратора?
8. Яка максимальна кількість інформаційних входів у мультимплексора з трьома адресними входами?

Лабораторна робота № 3

Тригери

Мета роботи: Вивчити закони функціонування елементарних автоматів (тригерів), способи їх завдання; отримати навички складання таблиць переходів тригерних схем.

Тригер – цифровий автомат, який має два стійких стани 0 або 1 і призначений для зберігання одного біту даних. Стан тригера визначається сигналами на його входах. Під впливом вхідного сигналу тригер стрибкоподібно переходить з одного стійкого стану в інший. Тригери мають два виходи: прямий Q і інвертований P. Стан тригера визначається по прямому виходу.

За логічним функціонуванням розрізняють типи тригерів: RS, D, T, JK.

За способом запису інформації розрізняють асинхронні і синхронні тригери. Синхронні тригери мають спеціальний вхід синхронізації (тактовий) – C (від слова Clock).

За способом сприйняття тактових сигналів тригери діляться на статичні (керовані рівнем 0 або 1) і динамічні (керовані фронтом наростання або фронтом спаду).

Тригери є найпростішими цифровими автоматами з пам'яттю на 1 біт інформації. Наявність пам'яті еквівалентна наявності зворотнього зв'язку, що накладається на деяку комбінаційну цифрову схему. На рис. 3.1 зображено схему найпростішого RS-тригера на елементах 2I-НЕ серії 74 із зворотнім зв'язком.

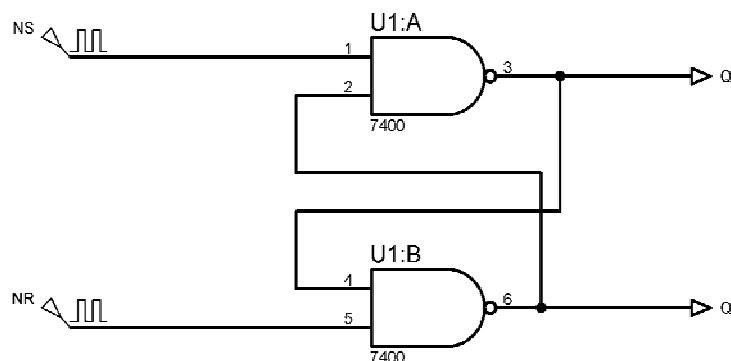


Рис. 3.1 Схема RS-тригера з інвертованими входами.

Даний тригер має інверсні входи встановлення 1 та 0 – відповідно NS та NR, а також прямий та інверсний інформаційні виходи – Q та \bar{Q} . Встановлення тригера в 1(0) відбувається при подачі нуля на NS(NR), при цьому протилежний вхід повинен бути встановлений в одиницю. При подачі одиниць на NS та NR разом тригер зберігає свій попередній стан. Подача двох нулів є забороненою комбінацією, при якій стан тригера буде невизначеним.

Для задання вхідних сигналів використовуються генератори цифрового тактового імпульсу DCLOCK. Після встановлення вхідних сигналів на схему слід задати параметри відповідного сигналу. При заданні нового періодичного цифрового сигналу можна задати його частоту (період), скважність (тривалість), початкове значення та затримку.

Для даної схеми тригера доцільно у початковий момент встановити режим зберігання інформації, потім у деякий момент подати імпульс встановлення 1, потім імпульс встановлення 0, після цього знову використати режим зберігання, і нарешті використати заборонений режим. Таким чином можна перебрати усі можливі режими роботи тригера.

Для того, щоб провести цифрове моделювання даної схеми, використовується граф DIGITAL з вкладки GRAPHS. У поле Digital Analysis переміщуються вхідні сигнали та індикатори напруги для вихідних сигналів. Це дозволяє побачити часову діаграму у рівнях логічних одиниці та нуля для обраних сигналів.

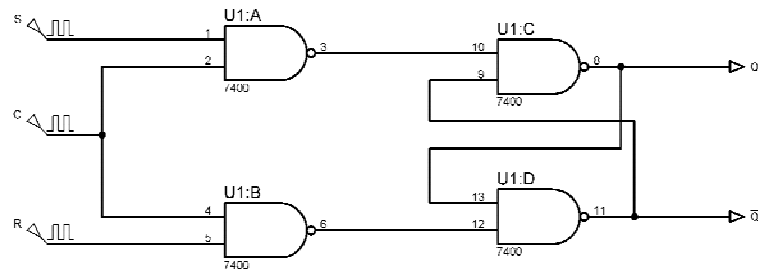


Рис. 3.2 Схема RS-тригера з синхронізуючим входом.

На рис. 3.2 зображено схему синхронного RS-тригера на елементах 2I-HE 7400. Даний тригер може змінювати свій стан лише при подачі одиниці на вхід синхронізації, коли ця схема працює як простий RS-тригер. Якщо ж на вхід синхронізації подається нуль, тригер зберігає свій стан незалежно від сигналів на R та S входах.

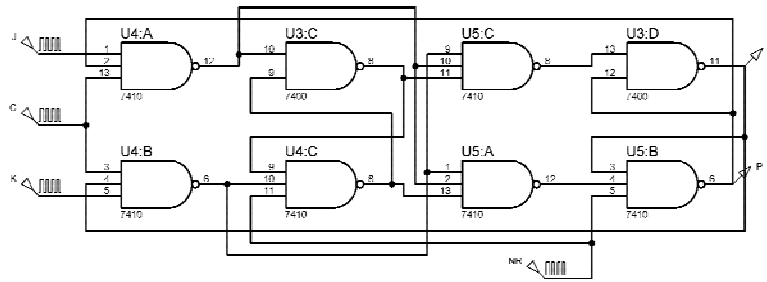


Рис. 3.3 Схема універсального двотактового JK-тригера.

На рис. 3.3 зображено схему універсального двотактового JK-тригера на елементах І-НЕ 7400 та 7410. Цей тригер при одночасній подачі одиниць на входи J та K переходить у лічильний режим, коли тригер змінює свій стан на протилежний при подачі імпульсу на вхід синхронізації C. У інших випадках ця схема працює як синхронний RS-тригер, причому перший простий тригер (Master, U3:C-U4:C) змінює стан по передньому фронту тактового імпульсу, а другий тригер (Slave, U3:D-U5:B) – по задньому фронту. Для початкового встановлення нуля в тригері передбачений інверсний вхід NR. Для аналізу різних режимів роботи схеми слід створити сигнали, що зображені на рис. 3.4.

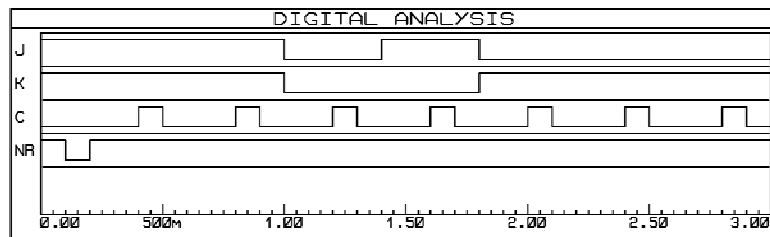


Рис. 3.4 Вхідні сигнали, що подаються на JK-тригер.

Стандартні серії цифрових ІМС містять значну кількість різноманітних тригерів у інтегральному виконанні. Зокрема,

інтегральна мікросхема 74107 - універсальний JK-тригер з входом скидання в нуль. Функціонально ця ІМС повторює схему рис. 3.3.

Лабораторне завдання

1. Введіть у Proteus схему рис. 3.1. Параметри входних імпульсів підберіть такими, щоб на виходах можна було спостерігати усі режими RS-тригера (встановлення 1, встановлення 0, зберігання, заборонений режим). Виведіть графік залежності входних та вихідних цифрових сигналів тригера від часу та поясніть ці залежності.

2. Замініть у схемі рис. 3.1. елементи І-НЕ на елементи АБО-НЕ (в цьому випадку входи тригера стануть не інверсними, а прямими – S та R). За аналогічних входних сигналів проведіть моделювання для цієї схеми. Виведіть відповідні графіки для входних та вихідних сигналів та поясніть ці залежності.

3. Введіть в Proteus схему синхронного RS-тригера (рис. 3.2). Входні пробні сигнали сконструюйте так, щоб на виходах можна було спостерігати усі режими RS-тригера (встановлення 1, встановлення 0, зберігання, заборонений режим, зберігання при відсутності синхронізації). Виведіть графік залежності входних та вихідних цифрових сигналів цього тригера від часу та поясніть ці залежності.

4. Введіть у Proteus схему універсального JK-тригера (рис. 3.3). Входні сигнали налаштуйте подібними до показаних на рис. 3.4, тобто так, щоб на виходах можна було спостерігати усі можливі режими тригера (скидання в нуль, лічильний режим, синхронне встановлення 1 та 0, зберігання, зберігання при відсутності синхронізації). Виведіть графік залежності входних та вихідних цифрових сигналів цього тригера від часу та поясніть ці залежності.

5. Повторіть моделювання п.4 для JK-тригера в інтегральному виконанні 74107.

6. Згідно Вашого варіанту розробити схему тригера та промоделюйте його роботу

Таблиця 3.1.

$h1$	$h2$	$h3$	Тип тригера, що треба розробити	На основі схем
0	0	0	D	74107
0	0	1	T	74109
0	1	0	JK	74НСТ74
0	1	1	D	74111
1	0	0	T	74НС74
1	0	1	JK	7474
1	1	0	D	74109
1	1	1	T	74107

Таблиця 3.2.

$h4$	Сигнал реагування тригера
0	передній фронт
1	задній фронт

7. Розробити, згідно свого варіанту тригер

Таблиця 3.3.

$h6$	$h7$	$h8$	Тип тригера, що треба розробити	тип
0	0	0	E	асинхронний
0	0	1	R	асинхронний
0	1	0	S	асинхронний
0	1	1	D	синхронний*
1	0	0	S	синхронний*
1	0	1	E	синхронний*
1	1	0	R	синхронний*
1	1	1	S	синхронний*

*- по рівню 1 синхронізуючого імпульсу.

Таблиця 3.4.

<i>h9</i>	Елементна база
0	3-I-Hi
1	3-Або-Hi

Контрольні питання:

1. Призначення тригера.
2. Дайте визначення тригера типу Т, D, JK, RS, R, S.
3. У чому відмінність тригера JK від RS-тригера?
4. Які є типи синхронних тригерів та який принцип їхньої роботи?
5. Напишіть таблицю переходів RS-, Т-, D-, JK – тригерів.
6. Як синхронний RS – тригер включити в режимі D-тригера?
7. Як синхронний JK – тригер включити в режимі Т-тригера?
8. Чим відрізняються синхронні тригери від асинхронних?

Лабораторна робота № 4**Регістри та лічильники**

Мета роботи: Вивчити принципи функціонування регістрів та лічильників, отримати навички роботи з регістрами та лічильниками.

Регістри та лічильники є цифровими пристроями з пам'яттю, що призначені для зберігання та перетворення інформації. На рис. 4.1 зображено схему зсувного реєстра на 3 розряди без схем читання та запису. Цей реєстр може застосовуватися як перетворювач послідовного коду (подається на вхід INFO) в паралельний (Знімається з виходів Q0-Q2). Побудований реєстр

на інтегральних JK-тригерах 74107, в яких інверсний вхід R застосовується для початкового встановлення в нуль, а інверсний вхід CLK – для синхронізації.

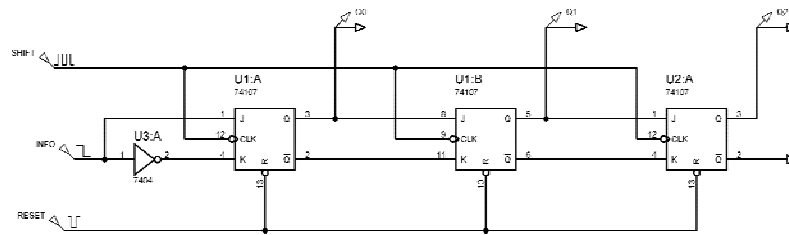


Рис. 4.1. Схема регістра зсуву.

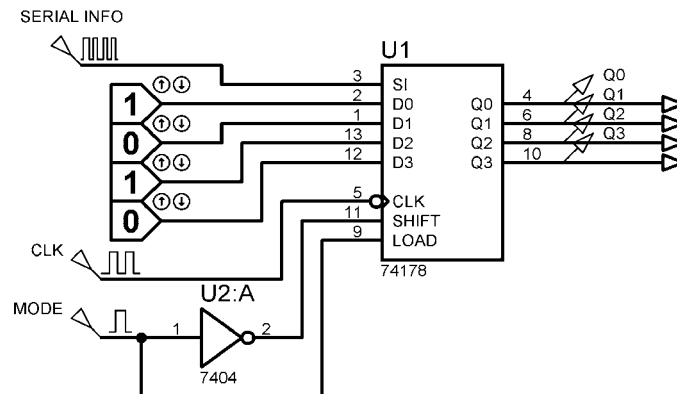


Рис. 4.2. Схема 4-розрядного регістра зсуву.

Серії інтегральних мікросхем містять деякі регістри в інтегральному виконанні. Так, ІМС 74178 являє собою інтегральний 4-розрядний регістр зсуву з можливістю паралельного запису та читання інформації. Призначення виводів 74178 таке: CLK – інверсний вхід синхронізації/зсуву, D0-D3 – входи паралельного запису інформації, SI – вхід послідовного запису інформації, LOAD – дозвіл паралельного запису (при подачі 1 на цей вхід у регістр записується інформація з входів A-D), SHIFT – дозвіл зсуву (при подачі 1 на цей вхід регістр буде зсувати інформацію по кожному імпульсу

на вході CLK). На рис. 4.2 зображено схему дослідження регістра 74178. Число для запису обране умовно, воно може бути яким завгодно.

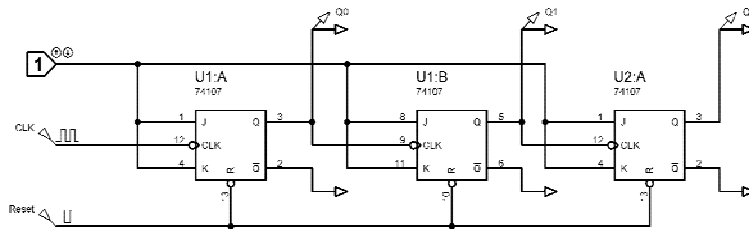


Рис. 4.3. Схема послідовного лічильника.

На рис. 4.3 зображено схему лічильника на двотактних JK-тригерах 74107. Такі тригери мають інверсні входи синхронізації і перекидаються по задньому фронті тактового імпульсу, тому при під'єднанні прямого виходу попереднього тригера до входу синхронізації наступного отримуємо лічильник, що лічить у прямому порядку від 0 до 7 (трирозрядний лічильник рис. 4.3)

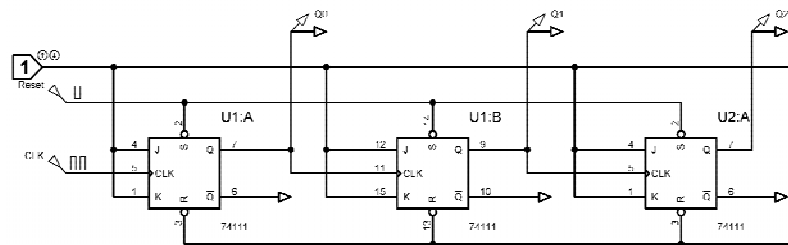


Рис. 4.4. Схема послідовного лічильника, що лічить в зворотному порядку.

На рис. 4.4 зображено схему лічильника на двотактних JK-тригерах 74111. Тригер 74111 відрізняється від 74107 тим, що входи синхронізації є прямими, і додатково введено інверсний вхід встановлення в одиницю. На схемі рис. 4.4. інверсний вихід кожного тригера під'єднаний до входу синхронізації наступного. Таким чином, утворюється лічильник, що лічить у

зворотному порядку від 7 до 0 (трирозрядний лічильник рис. 4.4)

Серії інтегральних мікросхем містять деякі лічильники в інтегральному виконанні. Наприклад, ІМС 74393 являє собою простий 4-розрядний двійковий лічильник, який має вхід встановлення нуля CLR та лічильний вхід, а також, звичайно, 4 інформаційні виходи QA-QD.

Лабораторне завдання

1. Введіть в програмі Proteus схему регістра зсуву (рис. 4.1.), SHIFT (тактові імпульси зсуву), INFO (вхідний послідовний код) та RESET (встановлення нуля) створіть такими, щоб встановити у регістр нуль, а потім записати в нього двійкове число 100. Виведіть графік залежності вхідних (Reset, Shift, Info) та вихідних (Q0-Q2) цифрових сигналів регістра від часу та поясніть ці залежності.

2. Перетворіть схему регістра зсуву рис. 4.1 таким чином, щоб зсув відбувався в лівий бік і був циклічним. Виведіть відповідні графіки для вхідних та вихідних сигналів та поясніть ці залежності.

3. Введіть у Proteus схему дослідження інтегрального регістра зсуву (рис. 4.2.). Джерела вхідних сигналів підберіть таким чином, щоб регістр послідовно виконав операції:

- а) паралельного запису числа $h1\ h2\ h3\ h4$;
- б) перетворення цього числа на послідовний код;
- в) послідовного запису числа $h5\ h6\ h7\ h8$;

Проведіть моделювання для цієї схеми. Виведіть відповідні графіки для вхідних та вихідних сигналів та поясніть ці залежності.

4. Введіть у Proteus схему лічильника (рис. 4.3.). Джерела вхідних сигналів підберіть таким чином, щоб лічильник спочатку був встановлений в нуль, а потім провів підрахунок 8 вхідних імпульсів. Виведіть відповідні графіки для вхідних та вихідних сигналів та поясніть ці залежності. Модернізуйте Ваш лічильник до лічильника з паралельним переносом.

Промоделюйте його роботу. Порівняйте отримані часові залежності.

5. Введіть у Proteus схему зворотного лічильника (рис. 4.4.). Джерела входних сигналів підберіть таким чином, щоб лічильник спочатку був встановлений в одиницю, а потім провів зворотний підрахунок до нуля. Виведіть відповідні графіки для входних та вихідних сигналів та поясніть ці залежності.

6. Введіть у Proteus схему дослідження простого інтегрального 4-розрядного лічильника 74393. Джерела входних сигналів підберіть таким чином, щоб лічильник спочатку був встановлений в нуль, а потім провів підрахунок 16 входних імпульсів. Виведіть відповідні графіки для входних та вихідних сигналів та поясніть ці залежності.

7. Додавши деякі логічні елементи, модифікуйте попередню схему таким чином, щоб отримати лічильник по модулю 10. Джерела входних сигналів підберіть таким чином, щоб лічильник спочатку був встановлений в нуль, а потім провів підрахунок 10 входних імпульсів. Виведіть відповідні графіки для входних та вихідних сигналів та поясніть ці залежності.

8. Введіть у Proteus схему згідно свого варіанту. Ознайомтесь з можливостями вашої схеми. Подайте входні сигнали, що характеризують роботу Вашої мікросхеми. Виведіть відповідні графіки для входних та вихідних сигналів та поясніть ці залежності.

Таблиця 4.1.

$h5$	$h4$	$h3$	схема
0	0	0	74164
0	0	1	74165
0	1	0	74166
0	1	1	74173
1	0	0	74177
1	0	1	74179
1	1	0	74160
1	1	1	74162

9. Розробіть схему, представлену в попередньому пункті. При розробці дозволяється використовувати будь-які логічні елементи та тригери 74-ї серії. Подайте вхідні сигнали, аналогічні до схеми, що досліджувалась в пункті 8. Порівняйте отримані епюри.

Контрольні питання:

1. На яких елементах побудовані регістри?
2. Призначення регістрів?
3. На яких елементах побудовані лічильники?
4. Нарисуйте схему синхронного лічильника з паралельним переносом.
5. За якими ознаками класифікують регістр?
6. Які мікрокоманди можуть реалізувати регістри?
7. Як працюють реверсивні лічильники?