

## Лабораторна робота № 7

### Регістри

**Мета роботи** Ознайомитись з принципом роботи різних типів регістрів.

#### Теоретичні відомості

Регістри – пристрої, котрі призначені для прийому, зберігання та передачі даних, представлених в двійковому коді. Інформація в регістрі зберігається в вигляді числа або слова, представлених у вигляді комбінацій нулів та одиниць. За допомогою регістрів можна здійснити операції перетворення кодів (паралельного в послідовний і навпаки).

Основною ознакою, за якою розрізняють регістри є спосіб запису інформації:

- паралельний – інформація записується одночасно в усі тригери регістра;
- послідовний – інформація записується спочатку в перший тригер, а потім передається в другий і т.п.;
- універсальний – працює в будь-якому з режимів.

Послідовні регістри бувають звичайні зсувні та реверсивні. В звичайних регістрах інформація зсувається в одну сторону, а в реверсивних – в обидві.

#### Паралельний регістр

У паралельному регістрі інформація записується паралельно на всі входи. Прикладом такого регістра є паралельний регістр на тригерах D-типу (рис. 7.1).

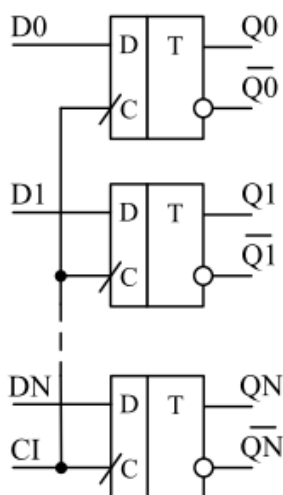


Рис. 7.1. Принципова схема паралельного регістра

У паралельному регістрі в момент приходу синхроімпульсу інформація одночасно запам'ятовується зі всіх входів D, та зберігається в регістрі до моменту приходу наступного синхроімпульсу. На рис. 7.2 показані часові діаграми роботи паралельного регістру.

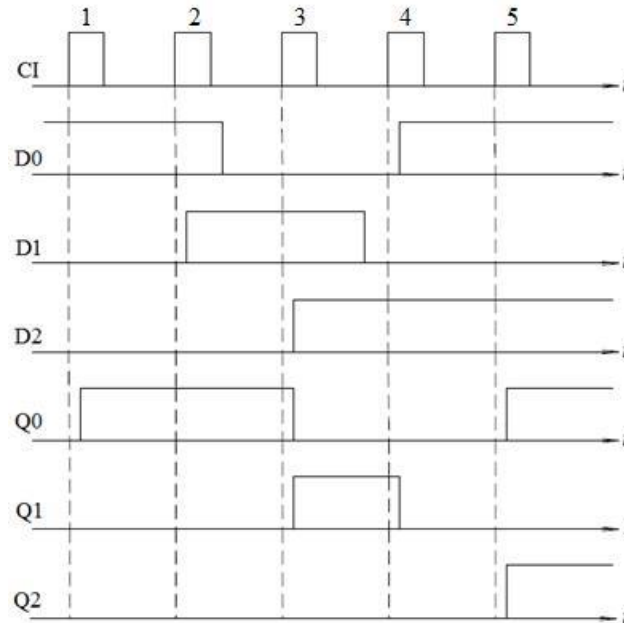


Рис. 7.2. Часові діаграми роботи паралельного регістра

### Послідовний регістр

У послідовному регістрі інформація спочатку записується в перший тригер, а потім передається в другий і т.д. На рис. 7.3 приведено приклад принципової схеми послідовного регістру.

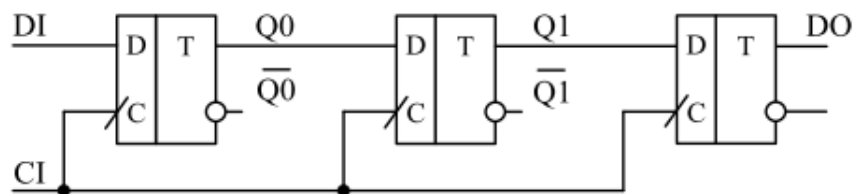


Рис. 7.3. Принципова схема послідовного регістра

На рис. 7.3 введені наступні позначення: DI – data input (вхід даних), DO – data output (вихід даних). На рис. 7.4 приведені часові діаграми роботи послідовного регістра.

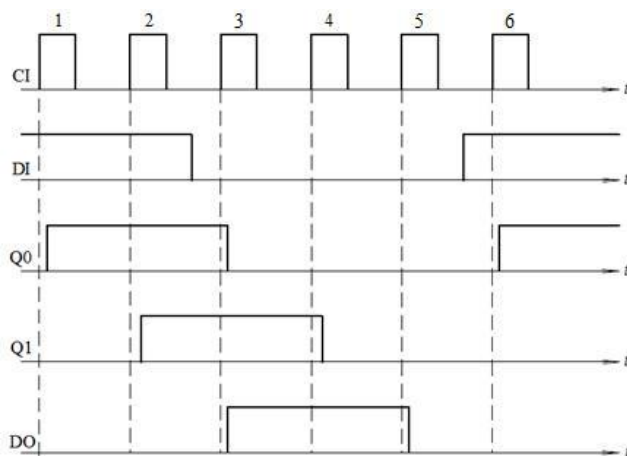


Рис. 7.4. Часові діаграми роботи послідовного регістра

Так як дані зсуваються в часі (кожен тригер зсуває дані на один такт), то такий регістр ще називається зсувним.

Реверсивний зсувний регістр дозволяє зсувати інформацію вправо чи вліво, в залежності від значень керуючих сигналів на додатковому вході. На рис. 7.5 наведена принципова схема реверсивного зсувного регістра з використанням цифрових перемикачів. В даній схемі сигнал V керує перемиканням регістру і в залежності від його значення змінюється сторона зсуву інформації.

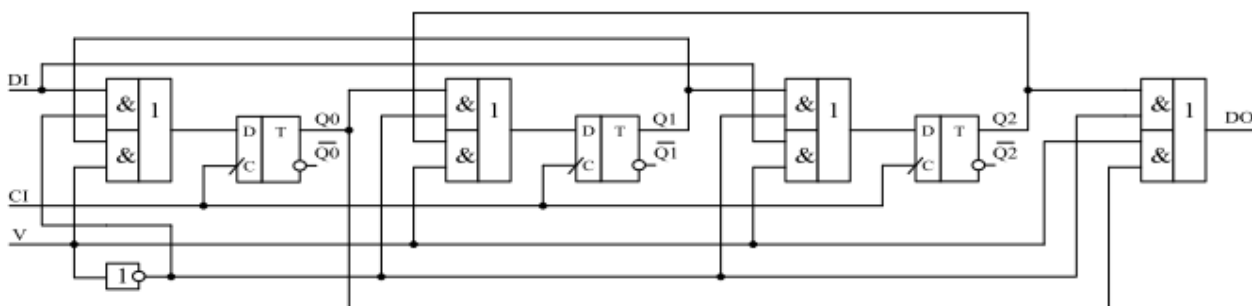


Рис. 7.5. Принципова схема реверсивного зсувного регістра

## Універсальний послідовно-паралельний регістр

Універсальний послідовно-паралельний регістр повинен дозволяти:

- приймати інформацію в послідовному або паралельному коді в залежності від сигналу керування;
- формувати на виході паралельний код;
- формувати на виході послідовність даних у вигляді послідовного коду, зсув якого можливий як в одному так і в іншому напрямі в залежності від другого сигналу керування.

Для того, щоб побудувати універсальний регістр доцільно використовувати RSD-тригери (рис. 7.6). Входи R і S використовують для запису паралельного коду. Вхід D – для побудови зсувного регістру.

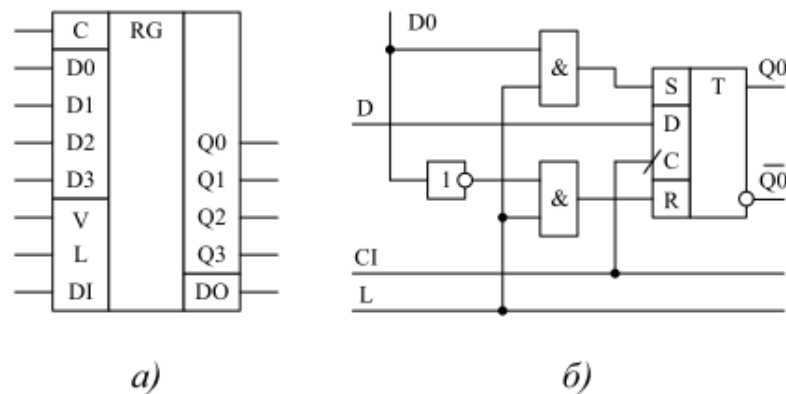


Рис. 7.6. Схематичне зображення універсального регістра (а) та принципова схема одного розряду (б)

На схематичному зображенні універсального регістра (рис. 7.6,а) введені наступні позначення: C – вхід синхронізації; D0, D1, D2, D3 – входи паралельного коду; V – напрям зсуву; L – дозвіл запису паралельного коду; DI – вхід послідовного коду; DO – вихід послідовного коду; Q0, Q1, Q2, Q3 – виходи паралельного коду.

## Перетворювач паралельного коду в послідовний

З метою економії портів вводу-виводу використовують схеми перетворювачів паралельного коду в послідовний. На рис. 7.7 показано приклад такої схеми.

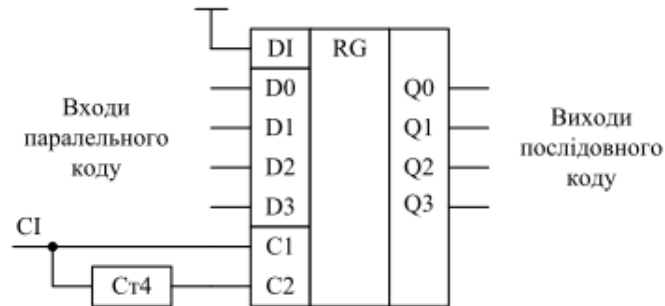


Рис. 7.7. Функціональна схема перетворювача паралельного коду в послідовний

У даній схемі: DI – вхід послідовних даних; D0, D1, D2, D3 – вхід паралельного коду; C1– синхроімпульс для зсуву інформації в регістрі; C2 – завантаження паралельного коду; Ст4 – схема ділення частоти ( $f_{C1}/f_{C2}=4$ ); Q0, Q1, Q2, Q3 – виходи послідовного коду. На рис. 7.8 показані часові діаграми роботи схеми перетворення паралельного коду в послідовний (рис. 7.7).

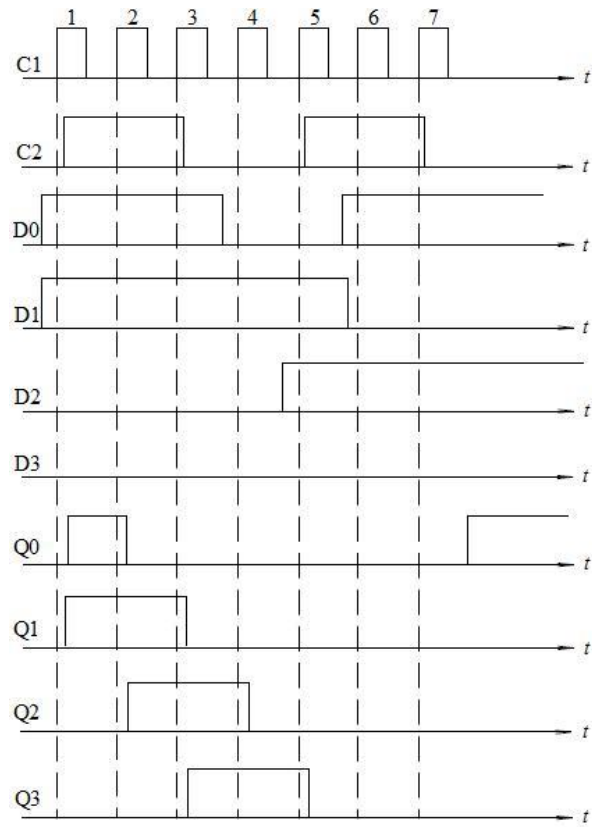


Рис. 7.8. Часові діаграми роботи схеми перетворювача паралельного коду в послідовний

## Перетворювач послідовного коду в паралельний

Перетворення послідовного коду в паралельний часто використовується з метою підвищення швидкості бітового потоку. Приклад схеми такого перетворювача показаний на рис. 7.9.

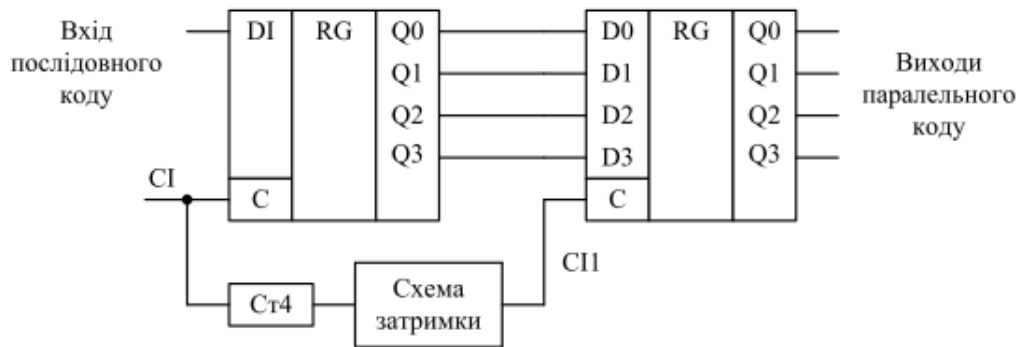


Рис. 7.9. Функціональна схема перетворювача послідовного коду в паралельний

В даній схемі СІІ формується через кожні чотири імпульси СІ. За чотири такти вся інформаційна послідовність коду вводиться в перший регістр. Як тільки вона буде введена здійснюється запис коду в другий регістр, на виході якого і буде сформовано паралельний код.

### Моделювання роботи регістра за допомогою програми NI Multisim

За допомогою програми NI Multisim змодельуємо роботу послідовного регістра на чотирьох RSD-тригерах. Для цього дослідимо принципову схему даного регістра.

Для того, щоб змодельувати потрібну схему, необхідно розмістити елементи схеми на робочому аркуші. Для розміщення елемента досить клацнути лівою кнопкою “миші” на піктограмі потрібної бібліотеки або через меню Place/Component вибрати необхідний елемент. За замовчуванням панель компонентів відображається у вікні програми.

Дотримуючись вище описаних дій, дослідимо принципову схему для вивчення роботи послідовного регістра на чотирьох RSD-тригерах, яка міститься в файлі **registr\_priklad.ms10**.

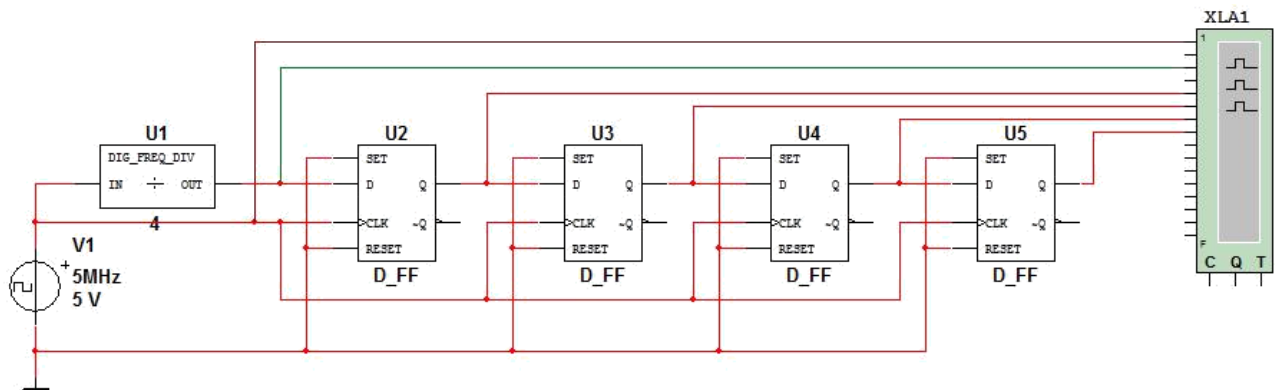


Рис. 7.10. Принципова схема послідовного регістра на чотирьох RSD-тригерах

Для того, щоб запустити моделювання, необхідно натиснути на значок піктограми із зображенням зеленого трикутника або на вимикач, розташований у правій верхній частині робочого вікна програми, або через меню Simulate викликати команду Run.

Після запуску моделювання необхідно два рази клацнути на зображення логічного аналізатора лівою кнопкою “миші” для того, щоб подивитися результати моделювання. Щоб часові діаграми відображалися на білому фоні необхідно у вікні логічного аналізатора натиснути на кнопку Reverse.

На рис. 7.11 наведені часові діаграми роботи послідовного регістра на чотирьох RSD-тригерах (рис. 7.10).



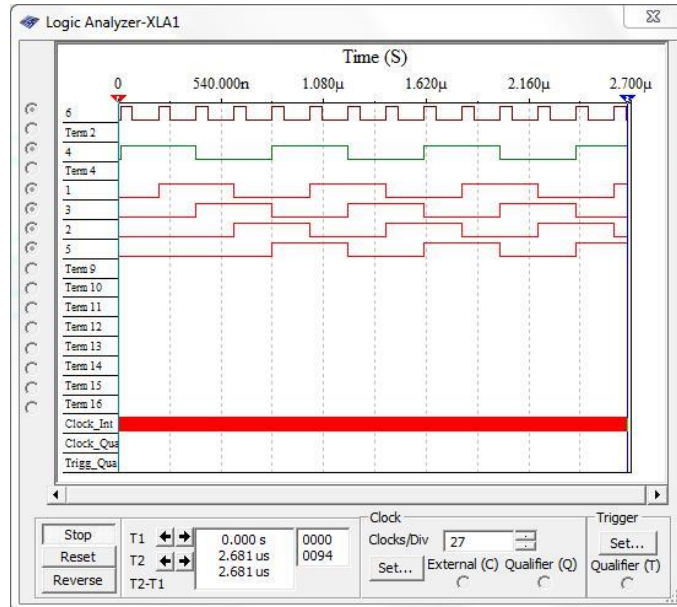


Рис. 7.11. Часові діаграми роботи послідовного регістра на чотирьох RSD-тригерах

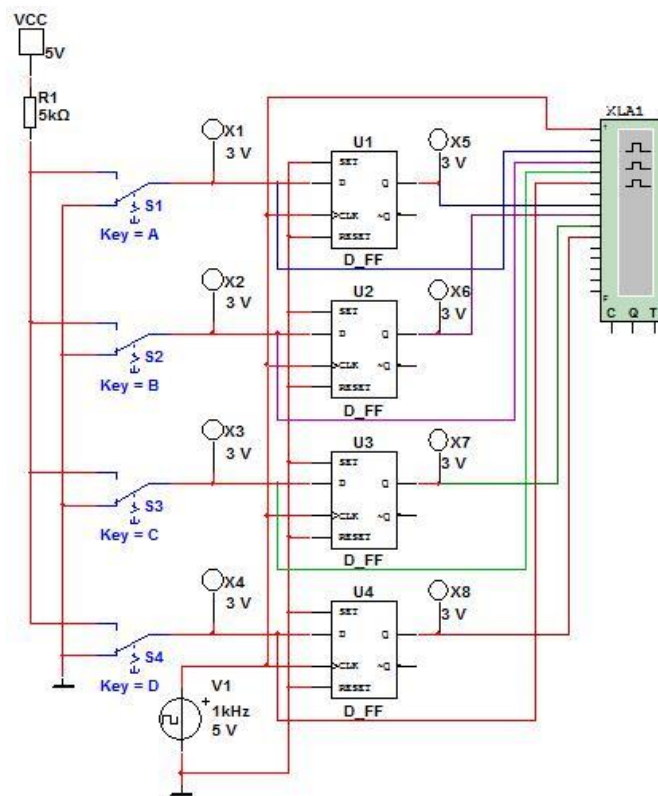


Рис. 7.12. Принципова схема моделі для дослідження паралельного регістра (05\_parallel\_registr.ms10)

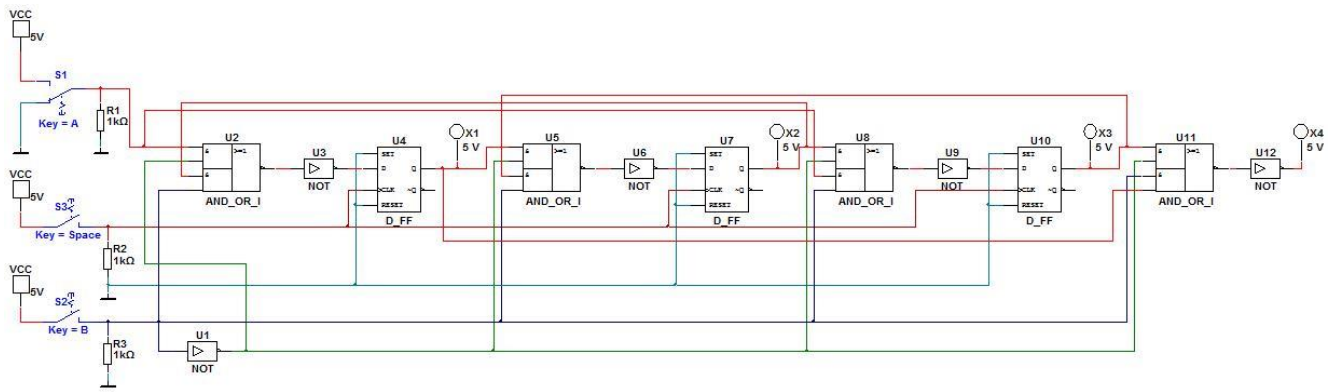


Рис. 7.13. Принципова схема моделі для дослідження реверсивного зсувного регістра (05\_revers\_zsuvniy\_registr.ms10)

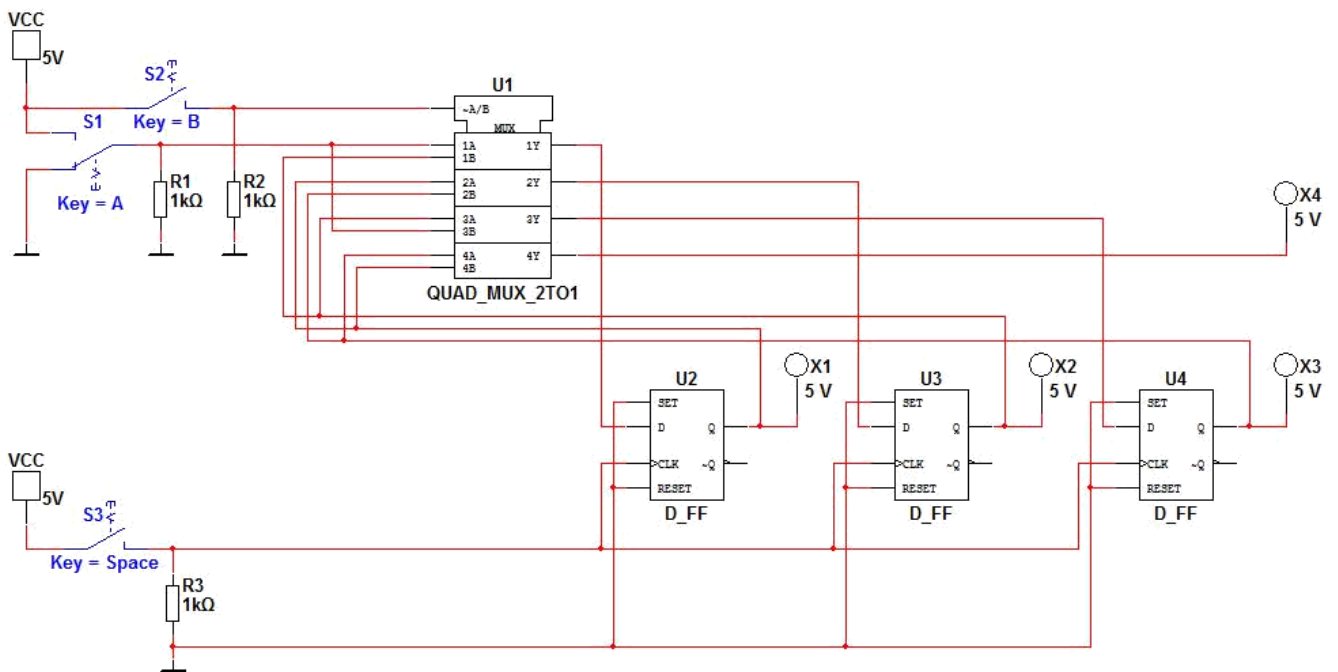


Рис. 7.14. Принципова схема моделі для дослідження реверсивного зсувного регістра з використанням мультиплектора (05\_revers\_zsuvniy\_registr\_mux.ms10)

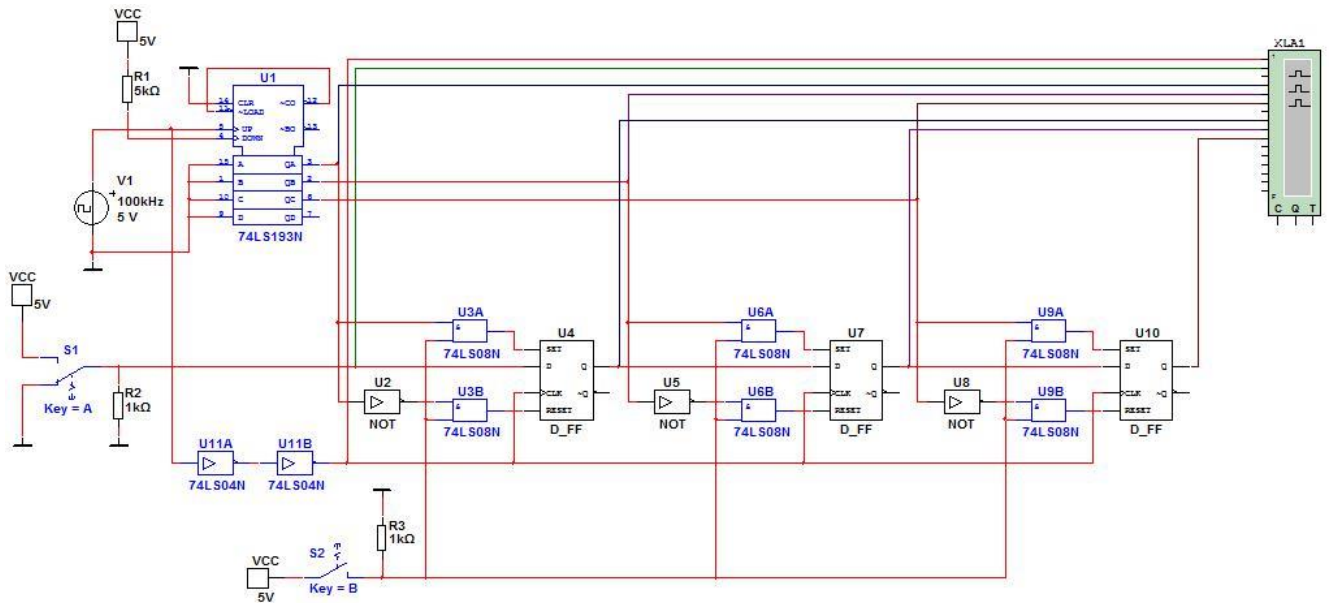


Рис. 7.15. Принципова схема моделі для дослідження універсального послідовно-паралельного регістра (05\_univers\_posl\_paral\_registr.ms10)

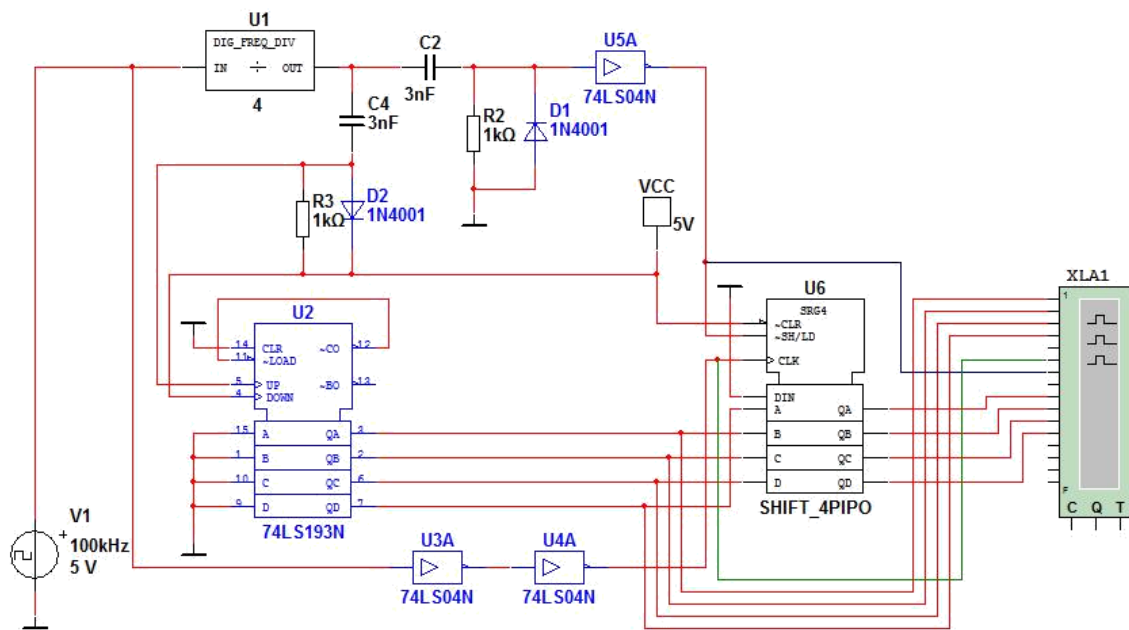


Рис. 7.16. Принципова схема моделі для дослідження перетворення паралельного коду у послідовний (05\_paralelniy\_v\_poslidovniy.ms10)

## Порядок виконання роботи

1. Для вивчення принципу роботи паралельного регістра дослідити схему, зображену на рис. 7.12, яка міститься в файлі 05\_paralel\_registr.ms10, в програмі NI Multisim. За допомогою перемикачів встановити вхідний код паралельного регістра. За допомогою логічного аналізатора та індикаторів отримати результати моделювання даної схеми. Пояснити отримані результати.

2. Для вивчення принципу роботи реверсивного зсувного регістра дослідити схеми, зображену на рис. 7.13 та рис. 7.14, які містяться в файлах 05\_revers\_zsuvniy\_registr.ms10 та 05\_revers\_zsuvniy\_registr\_mux.ms10 відповідно, в програмі NI Multisim. За вибором викладача, дослідити одну з двох схем. Комбінуючи положення перемикача S1 задати послідовний код регістра. Комбінуючи положення перемикача S3 задати синхроімпульси даної схеми. За допомогою індикаторів спостерігати результати моделювання даної схеми. За допомогою перемикача S2 змінювати напрямок зсуву інформації. Дослідити роботу реверсивного зсувного регістра з різним напрямком лічення. Пояснити отримані результати.

3. Для вивчення принципу роботи універсального послідовно-паралельного регістра дослідити схему, зображену на рис. 7.15, яка міститься в файлі 05\_univers\_posl\_paral\_registr.ms10, програмі NI Multisim. Комбінуючи положення перемикача S2 дослідити роботу універсального послідовно-паралельного регістра в двох режимах. При роботі в режимі послідовного регістра, за допомогою перемикача S1 задати послідовний код. За допомогою логічного аналізатора отримати результати моделювання даної схеми. Пояснити отримані результати.

4. Для вивчення принципу роботи схеми перетворення паралельного коду в послідовний дослідити схему, зображену на рис. 7.16, яка міститься в файлі 05\_paralelniy\_v\_poslidovniy.ms10, в програмі NI Multisim. За допомогою логічного аналізатора отримати результати моделювання даної схеми. Пояснити отримані результати.

## **Контрольні запитання**

1. Що таке регістр? Які приклади його застосування?
2. Класифікація регістрів.
3. Пояснити принцип роботи паралельного регістра. Навести приклад часових діаграм, котрі характеризують роботу паралельного регістра.
4. Пояснити принцип роботи послідовного регістра. Навести приклад часових діаграм, котрі характеризують роботу послідовного регістра.
5. Пояснити принцип роботи універсального послідовно-паралельного регістра. Навести приклад часових діаграм, котрі характеризують роботу універсального послідовно-паралельного регістра.
6. Пояснити принцип перетворення паралельного коду в послідовний. Навести приклад принципової схеми такого перетворення та часові діаграми її роботи.
7. Пояснити принцип перетворення послідовного коду в паралельний. Навести приклад принципової схеми такого перетворення та часові діаграми її роботи.

## **Звіт про виконання лабораторної роботи**

У звіті треба навести часові діаграми сигналів для кожного з досліджуваних регістрів. Схему регістра або навести у звіті, або дати посилання на рисунок, де вона наведена..

Навести для кожної схеми час затримки перемикання всього регістра.