

## **ЛАБОРАТОРНА РОБОТА № 3**

---

### **Синтез і дослідження комбінаційних цифрових пристроїв**

Кількість індивідуальних варіантів: 17

**Методичні вказівки та індивідуальні завдання**

## 1. Паспорт лабораторної роботи

| Параметр | Зміст                                                                                                                                              |
|----------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| Тема     | Синтез комбінаційної схеми та дослідження дешифратора, пріоритетного шифратора, мультиплексора, демультимплексора і компаратора.                   |
| Мета     | Навчитися переходити від таблиці істинності до структури на логічних елементах і стандартних комбінаційних вузлах, а також перевіряти їхні виходи. |

## 2. Короткі теоретичні відомості

### 2.1. Комбінаційна схема та послідовність синтезу

Комбінаційний цифровий пристрій (combinational digital circuit) формує виходи лише за поточними значеннями входів:  $Y=F(X)$ . Внутрішній стан і пам'ять попередніх наборів відсутні. Однаковий стабілізований набір входів завжди дає однаковий набір виходів.

1. визначити входи, виходи, їхній зміст, розрядність і активні рівні;
2. побудувати таблицю істинності або функціональну таблицю;
3. записати булеві функції виходів і за потреби мінімізувати їх;
4. вибрати реалізацію на логічних елементах або стандартних вузлах;
5. перевірити всі допустимі набори та оцінити логічну глибину і затримку.

### 2.2. Логічні елементи і логічна глибина

У роботі використовуються NOT (НЕ, інвертор), AND (І, кон'юнктор) та OR (АБО, диз'юнктор). Заперечення позначається надрисковою рисою:  $\bar{A}$ . Для реалізації мінімальної диз'юнктивної нормальної форми інвертори утворюють заперечені літерали, елементи AND формують добутки, а OR об'єднує їх у вихідну функцію.

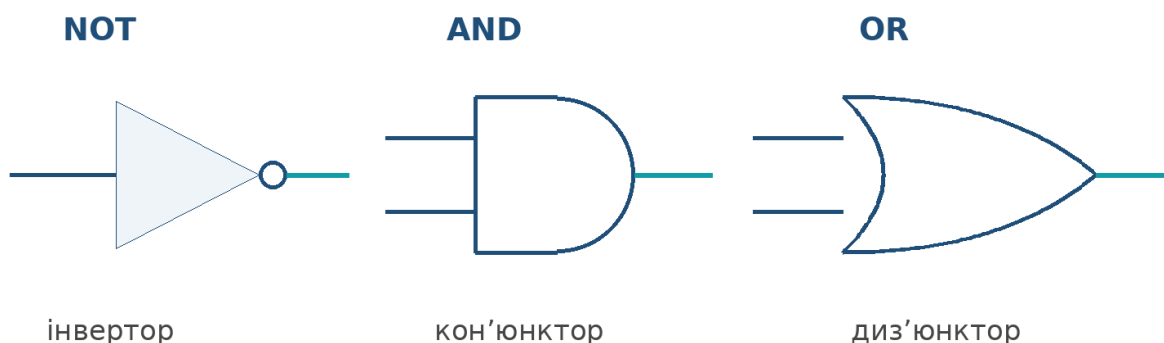


Рисунок 1. Стандарти умовні позначення елементів NOT, AND та OR.

Логічна глибина (logic depth) — максимальна кількість послідовно пройдених елементів від входу до виходу. Затримка поширення (propagation delay) кожного елемента додається вздовж шляху. Різні затримки можуть короткочасно утворити глітч (glitch) — хибний імпульс під час перемикавання.

### 2.3. Дешифратор і шифратор

Дешифратор (decoder) перетворює  $n$ -бітовий код на один із максимум  $2^n$  виходів. Для active-high моделі та  $EN=1$  активним є рівно один вихід  $Y_k$ , де  $k$  дорівнює десятковому значенню вхідного коду.

За  $EN=0$  усі виходи неактивні. Виходи дешифратора 3 у 8 відповідають мінтермам  $m_0 \dots m_7$ , тому функцію  $F = \sum m(k)$  можна реалізувати об'єднанням потрібних  $Y_k$  через OR.

Шифратор (encoder) виконує зворотнє перетворення one-hot  $\rightarrow$  code. Простий шифратор припускає тільки один активний вхід. Пріоритетний шифратор (priority encoder) допускає кілька одиниць і кодує вхід із найвищим пріоритетом. У роботі  $D_3$  має найвищий пріоритет. Вихід V (Valid — ознака дійсності) дорівнює 1, якщо активний хоча б один вхід.

## 2.4. Мультимплексор і реалізація функції

Мультимплексор (multiplexer, MUX) передає на вихід один із інформаційних входів. MUX 4 у 1 має входи  $I_0 \dots I_3$  та адресні входи  $S_1 S_0$ . У цій роботі  $S_1=A$ ,  $S_0=B$ , а залежність від C переноситься на інформаційні входи.

| Ф при C=0 | Ф при C=1 | Підключення $I_k$ |
|-----------|-----------|-------------------|
| 0         | 0         | 0                 |
| 0         | 1         | C                 |
| 1         | 0         | $\bar{C}$         |
| 1         | 1         | 1                 |

Для кожної фіксованої пари AB потрібно взяти два рядки таблиці — з  $C=0$  і  $C=1$  — та за таблицею вище визначити  $I_0$ ,  $I_1$ ,  $I_2$  або  $I_3$ . Після підключення обов'язково перевіряють усі вісім наборів ABC.

## 2.5. Демультимплексор і компаратор

Демультимплексор (demultiplexer, DEMUX) спрямовує один інформаційний вхід D на один із виходів, обраний кодом  $S_1 S_0$ . Для DEMUX 1 у 4 при  $D=1$  активується  $Y_s$ , а при  $D=0$  всі виходи дорівнюють 0. Дешифратор вибирає лінію за кодом, а DEMUX додатково передає на вибрану лінію значення D.

Цифровий компаратор (digital comparator) формує взаємовиключні ознаки G:  $P>Q$ , E:  $P=Q$  та L:  $P<Q$ . Для будь-якої пари коректних чисел рівно один із цих виходів дорівнює 1.

## 3. Повний приклад виконання

Нехай  $F(A,B,C) = \sum m(1,2,6,7)$ .

| i | A | B | C | F |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 2 | 0 | 1 | 0 | 1 |
| 3 | 0 | 1 | 1 | 0 |
| 4 | 1 | 0 | 0 | 0 |
| 5 | 1 | 0 | 1 | 0 |
| 6 | 1 | 1 | 0 | 1 |

| i | A | B | C | F |
|---|---|---|---|---|
| 7 | 1 | 1 | 1 | 1 |

ДДНФ:  $(\bar{A}\bar{B}AC) \vee (\bar{A}B\bar{A}\bar{C}) \vee (A\bar{B}A\bar{C}) \vee (A\bar{B}AC)$ .

Після мінімізації:  $F = B\bar{A}\bar{C} \vee \bar{A}\bar{B}AC \vee A\bar{B}$ .

Реалізація AND–OR–NOT потребує: інверторів — 3, елементів AND — 3, елементів OR — 1; разом 7. Максимальна логічна глибина за окремого підрахунку NOT дорівнює 3.

Через дешифратор 3 у 8:  $F = Y_1 \vee Y_2 \vee Y_6 \vee Y_7$ .

| AB | Номер входу | F(C=0) | F(C=1) | Підключення |
|----|-------------|--------|--------|-------------|
| 00 | I0          | 0      | 1      | C           |
| 01 | I1          | 1      | 0      | $\bar{C}$   |
| 10 | I2          | 0      | 0      | 0           |
| 11 | I3          | 1      | 1      | 1           |

Отже,  $I_0 = C$ ,  $I_1 = \bar{C}$ ,  $I_2 = 0$ ,  $I_3 = 1$ ,  $S_1 = A$ ,  $S_0 = B$ .

## 4. Завдання та порядок виконання

### Завдання 1. Таблиця істинності та мінімізація

- За множиною мінтермів свого варіанта заповніть таблицю істинності для  $ABC = 000 \dots 111$ .
- Запишіть повну ДДНФ.
- Перенесіть одиниці на карту Карно й отримайте мінімальну ДНФ.
- Перевірте, що мінімізований вираз формує одиниці саме на заданих індексах.

### Завдання 2. Реалізація на AND–OR–NOT

- Намалюйте схему мінімального виразу стандартними умовними позначеннями.
- Підпишіть усі входи, проміжні сигнали та вихід F.
- Порахуйте кількість NOT, AND і OR та визначте максимальну логічну глибину.

### Завдання 3. Реалізація через дешифратор і MUX

- Для дешифратора 3 у 8 запишіть, які виходи  $Y_0 \dots Y_7$  потрібно об'єднати елементом OR.
- Для MUX 4 у 1 прийміть  $S_1 = A$  і  $S_0 = B$ . За парами  $F(C=0)$ ,  $F(C=1)$  визначте  $I_0 \dots I_3$  як 0, 1, C або  $\bar{C}$ .
- Перевірте MUX на всіх восьми наборах.

### Завдання 4. Аналіз стандартних вузлів

- Визначте  $Y_0 \dots Y_3$  дешифратора 2 у 4 за кодом  $A_1A_0$  і входом EN.
- Для пріоритетного шифратора  $D_3 \dots D_0$  з пріоритетом  $D_3 > D_2 > D_1 > D_0$  визначте  $Q_1Q_0$  та V.
- Для DEMUX 1 у 4 визначте  $Y_0 \dots Y_3$  за D і  $S_1S_0$ .
- Порівняйте 3-бітові P і Q та визначте G, E, L.

**Рекомендований розподіл часу**

| Етап                              | Час      |
|-----------------------------------|----------|
| Ознайомлення і таблиця істинності | 8–10 хв  |
| Мінімізація та схема AND–OR–NOT   | 12–15 хв |
| Дешифратор і MUX                  | 12–15 хв |
| Стандартні вузли                  | 8–10 хв  |
| Перевірка та висновок             | 5–10 хв  |

## 5. Варіанти завдань

| №  | Функція                      |
|----|------------------------------|
| 1  | $F(A,B,C)=\Sigma m(1,2,6,7)$ |
| 2  | $F(A,B,C)=\Sigma m(0,3,4,7)$ |
| 3  | $F(A,B,C)=\Sigma m(1,3,4,6)$ |
| 4  | $F(A,B,C)=\Sigma m(0,2,5,7)$ |
| 5  | $F(A,B,C)=\Sigma m(2,3,5,6)$ |
| 6  | $F(A,B,C)=\Sigma m(0,1,6,7)$ |
| 7  | $F(A,B,C)=\Sigma m(1,4,5,6)$ |
| 8  | $F(A,B,C)=\Sigma m(0,3,5,6)$ |
| 9  | $F(A,B,C)=\Sigma m(2,4,6,7)$ |
| 10 | $F(A,B,C)=\Sigma m(1,2,3,5)$ |
| 11 | $F(A,B,C)=\Sigma m(0,2,4,7)$ |
| 12 | $F(A,B,C)=\Sigma m(1,3,6,7)$ |
| 13 | $F(A,B,C)=\Sigma m(0,1,3,4)$ |
| 14 | $F(A,B,C)=\Sigma m(2,3,4,5)$ |
| 15 | $F(A,B,C)=\Sigma m(0,5,6,7)$ |
| 16 | $F(A,B,C)=\Sigma m(1,2,4,6)$ |
| 17 | $F(A,B,C)=\Sigma m(0,2,3,7)$ |

| № | Decoder $A_1A_0$ / EN | Priority encoder $D_3D_2D_1D_0$ | DEMUX D / $S_1S_0$ | Comparator P / Q |
|---|-----------------------|---------------------------------|--------------------|------------------|
| 1 | 10 / 1                | 0111                            | 1 / 10             | 101 / 011        |
| 2 | 01 / 1                | 1010                            | 1 / 01             | 010 / 110        |
| 3 | 11 / 0                | 0011                            | 1 / 11             | 111 / 100        |
| 4 | 00 / 1                | 1101                            | 0 / 10             | 001 / 001        |
| 5 | 11 / 1                | 0100                            | 1 / 00             | 100 / 101        |
| 6 | 10 / 0                | 1001                            | 1 / 01             | 110 / 010        |
| 7 | 01 / 1                | 0001                            | 1 / 11             | 011 / 111        |

| №  | Decoder $A_1A_0$ / EN | Priority encoder $D_3D_2D_1D_0$ | DEMUX D / $S_1S_0$ | Comparator P / Q |
|----|-----------------------|---------------------------------|--------------------|------------------|
| 8  | 10 / 1                | 1110                            | 0 / 00             | 101 / 101        |
| 9  | 00 / 0                | 0110                            | 1 / 10             | 000 / 001        |
| 10 | 11 / 1                | 1000                            | 1 / 01             | 111 / 011        |
| 11 | 01 / 0                | 0101                            | 1 / 00             | 010 / 010        |
| 12 | 10 / 1                | 0010                            | 0 / 11             | 100 / 001        |
| 13 | 00 / 1                | 1011                            | 1 / 10             | 011 / 100        |
| 14 | 11 / 0                | 0000                            | 1 / 01             | 110 / 110        |
| 15 | 01 / 1                | 1100                            | 1 / 11             | 001 / 101        |
| 16 | 10 / 1                | 0110                            | 0 / 00             | 101 / 110        |
| 17 | 00 / 0                | 1001                            | 1 / 10             | 111 / 000        |

## 6. Вимоги до звіту

- номер, тема, мета і номер варіанта;
- таблиця істинності, ДДНФ, карта Карно та мінімальна ДНФ;
- схема AND–OR–NOT зі стандартними позначеннями і підрахунком елементів;
- реалізація функції через дешифратор 3 у 8 та MUX 4 у 1;
- розраховані виходи дешифратора, пріоритетного шифратора, DEMUX і компаратора;
- короткий висновок із результатами перевірки.

## 7. Критерії оцінювання

| Складова                               | Бали |
|----------------------------------------|------|
| Таблиця істинності, ДДНФ і мінімізація | 2    |
| Схема AND–OR–NOT, складність і глибина | 2    |
| Реалізація через дешифратор            | 1.5  |
| Реалізація через MUX і перевірка       | 2    |
| Аналіз стандартних вузлів              | 1.5  |
| Оформлення, висновок і захист          | 1    |

## 8. Питання для захисту

1. Що називають комбінаційним цифровим пристроєм?
2. Чим комбінаційна схема відрізняється від послідовної?
3. Які основні етапи синтезу комбінаційної схеми?

4. Що таке активний рівень сигналу?
5. Які функції виконують NOT, AND та OR?
6. Чому NAND і NOR називають універсальними елементами?
7. Що таке логічна глибина?
8. Що таке затримка поширення та глітч?
9. Що виконує дешифратор?
10. Що означає кодування one-hot?
11. Для чого потрібен вхід EN?
12. Як реалізувати  $F = \sum m(k)$  через дешифратор?
13. Що виконує шифратор?
14. Чому звичайний шифратор не можна коректно використовувати за кількох активних входів?
15. Як працює пріоритетний шифратор і навіщо потрібен V?
16. Що виконує мультиплексор?
17. Скільки адресних входів потрібно MUX із  $2^n$  інформаційними входами?
18. Запишіть рівняння MUX 2 у 1.
19. Як реалізувати функцію трьох змінних через MUX 4 у 1?
20. Коли інформаційний вхід MUX потрібно підключити до  $\bar{C}$ ?
21. Що виконує демультимплексор?
22. Чим DEMUX відрізняється від дешифратора?
23. Які виходи має цифровий компаратор?
24. Чому виходи G, E та L є взаємовиключними?
25. Як перевірити реалізацію комбінаційної функції?
26. Що таке критичний шлях комбінаційної схеми?