

ЛАБОРАТОРНА РОБОТА № 2

1. Паспорт лабораторної роботи

Параметр	Зміст
Тема	Додавання і віднімання 8-бітових чисел, прапорці CF/OF, зсуви, формат Q4.4, таблиці істинності, ДДНФ/ДКНФ і мінімізація картою Карно.
Мета	Навчитися послідовно виконувати двійкові арифметичні операції та аналізувати й мінімізувати булеву функцію трьох змінних.
Результат	Звіт із проміжними обчисленнями, перевірки та відповідями на питання викладача.

2. Короткі теоретичні відомості

2.1. Додавання та прапорці

Двійкове додавання виконують справа наліво — від LSB (Least Significant Bit — молодший значущий біт) до MSB (Most Significant Bit — старший значущий біт). У кожному розряді додають два біти та перенесення на вході. Сума 2 або 3 утворює перенесення 1 до наступного розряду.

x	y	C _{in}	S	C _{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

CF (Carry Flag — прапорець перенесення) дорівнює кінцевому перенесенню за межі 8 бітів і використовується для контролю беззнакового результату. OF (Overflow Flag — прапорець знакового переповнення) дорівнює 1, коли два числа однакового знака дали результат протилежного знака. CF і OF описують різні інтерпретації тієї самої бітової операції.

2.2. Віднімання через додатковий код

Різницю $P-Q$ обчислюють як $P+(-Q)$. Щоб утворити 8-бітовий додатковий код (two's complement — доповнення до двох) числа $-Q$, біти коду Q інвертують і додають 1. Після додавання кінцеве перенесення поза 8-бітовою сіткою відкидають. Результат обов'язково перевіряють десятковим відніманням.

2.3. Зсуви

Скорочення	Повна назва	Дія
LSL	Logical Shift Left — логічний зсув ліворуч	Біти рухаються ліворуч, справа входять нулі; старші біти втрачаються.
LSR	Logical Shift Right — логічний зсув праворуч	Біти рухаються праворуч, зліва входять нулі; молодші біти втрачаються.
ASR	Arithmetic Shift Right — арифметичний зсув праворуч	Біти рухаються праворуч, а зліва копіюється знаковий біт.

2.4. Формат Q4.4 та IEEE 754

Q4.4 — формат із фіксованою крапкою (fixed-point format): 4 біти цілої і 4 біти дробової частини. Ціна молодшого біта дорівнює $2^{-4}=1/16$. Для кодування беззнакового числа R обчислюють $I=R \cdot 16$ і записують I у 8 бітах; для декодування обчислюють $R=I/16$.

IEEE 754 binary32 — 32-бітовий формат із рухомою крапкою (floating-point format): S (Sign — знак), 1 біт; E (Exponent — порядок), 8 бітів; F (Fraction — дробова частина мантиї), 23 біти. У практичних завданнях цієї роботи IEEE 754 не кодується, але структура формату входить до питань захисту.

2.5. Булеві операції та таблиця істинності

Булева змінна має два значення: 0 або 1. У роботі використовуються AND (І, кон'юнкція, $\wedge$), OR (АБО, диз'юнкція, $\vee$), XOR (виключне АБО, $\oplus$) та NOT (НЕ, заперечення). Заперечення позначається надрисковою рисою: $\bar{A}$. Порядок виконання: дужки, NOT, AND, XOR, OR.

A	B	$A \wedge B$	$A \vee B$	$A \oplus B$	$\bar{A}$
0	0	0	0	0	1
0	1	0	1	1	1

A	B	$A \wedge B$	$A \vee B$	$A \oplus B$	$\bar{A}$
1	0	0	1	1	0
1	1	1	1	0	0

Для трьох змінних A, B, C таблиця істинності має $2^3=8$ рядків. Номер набору визначається двійковим кодом ABC: 000_2 має індекс 0, 001_2 — 1, ..., 111_2 — 7. Проміжні стовпці зменшують імовірність помилки.

2.6. ДДНФ, ДКНФ і карта Карно

Мінтерм містить усі змінні в кон'юнкції та дорівнює 1 лише на одному наборі: для біта 1 змінну записують прямо, для біта 0 — із рисою. ДДНФ (досконала диз'юнктивна нормальна форма; PDNF — Perfect Disjunctive Normal Form) є диз'юнкцією мінтермів рядків $F=1$.

Макстерм містить усі змінні в диз'юнкції та дорівнює 0 лише на одному наборі: для біта 0 змінну записують прямо, для біта 1 — із рисою. ДКНФ (досконала кон'юнктивна нормальна форма; PCNF — Perfect Conjunctive Normal Form) є кон'юнкцією макстермів рядків $F=0$.

У карті Карно для трьох змінних рядки відповідають $A=0$ і $A=1$, а стовпці BC розміщено за кодом Ґрея: 00, 01, 11, 10. Одиниці об'єднують у прямокутні групи 1, 2, 4 або 8 клітинок. Обирають найбільші групи; протилежні краї є сусідніми, перекриття дозволене, діагональне сусідство — ні.

A / BC	00	01	11	10
0	m_0	m_1	m_3	m_2
1	m_4	m_5	m_7	m_6

3. Повний приклад виконання

3.1. Додавання 44+23

$44_{10}=00101100_2$, $23_{10}=00010111_2$. Додаємо справа наліво, фіксуючи перенесення.

i	x_i	y_i	C_{in}	S_i	C_{out}
0	0	1	0	1	0
1	0	1	0	1	0
2	1	1	0	0	1
3	1	0	1	0	1
4	0	1	1	0	1
5	1	0	1	0	1
6	0	0	1	1	0
7	0	0	0	0	0

Отримано $01000011_2=67_{10}$. Кінцеве перенесення 0, тому $CF=0$. Обидва операнди додатні й результат додатний, тому $OF=0$.

3.2. Віднімання 81–29

- $81_{10}=01010001_2$; $29_{10}=00011101_2$.
- Інверсія 29: 11100010_2 .
- Додаємо 1: 11100011_2 — код -29.
- $01010001_2+11100011_2=1\ 00110100_2$.
- Відкидаємо кінцеве перенесення: $00110100_2=52_{10}$. Перевірка: $81-29=52$.

3.3. Зсув і Q4.4

ASR (арифметичний зсув праворуч) коду 11110000_2 на 2 позиції: $11110000_2 \rightarrow 11111000_2 \rightarrow 11111100_2$. Знаковий біт 1 копіюється зліва. Початкове число -16, результат -4.

Кодування 3,25 у Q4.4: $I=3,25 \cdot 16=52$; $52_{10}=00110100_2$. Розділення полів: $0011.0100_2=3+4/16=3,25$.

3.4. Булева функція $F = \bar{A}VC$

i	A	B	C	$\bar{A}$	F
0	0	0	0	1	1
1	0	0	1	1	1
2	0	1	0	1	1
3	0	1	1	1	1
4	1	0	0	0	0
5	1	0	1	0	1
6	1	1	0	0	0
7	1	1	1	0	1

Одиничні набори: $\Sigma m(0,1,2,3,5,7)$. Нульові набори: ПМ(4,6).

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

ДКНФ: $(\bar{A}\bar{B}VC) \wedge (\bar{A}\bar{B}\bar{V}C)$.

На карті Карно група m_0, m_1, m_2, m_3 дає $\bar{A}$, а група m_1, m_3, m_5, m_7 дає C . Мінімальна ДНФ: $F = \bar{A}VC$. Перевірка множиною мінтермів дає ті самі індекси 0,1,2,3,5,7.

4. Завдання та порядок виконання

Завдання 1. 8-бітова арифметика

- Запишіть X і Y у 8-бітовій сітці. Виконайте $X+Y$ справа наліво та покажіть C_{in} і C_{out} у кожному розряді.
- Запишіть 8-бітову суму та CF . Інтерпретуйте X , Y і суму як знакові числа в додатковому коді; визначте OF і поясніть його одним реченням.
- Для $P-Q$ інвертуйте 8 бітів Q , додайте 1, а потім додайте отриманий код до P . Відкиньте кінцеве перенесення та виконайте десяткову перевірку.

Завдання 2. Зсув і Q4.4

- Виконайте заданий зсув коду Z на k позицій у фіксованій 8-бітовій сітці. Покажіть кожний крок, заповнювальні та втрачені біти.
- Закодуйте R у беззнаковому форматі Q4.4: обчисліть $I=R \cdot 16$, переведіть I у 8-бітовий код і виконайте зворотну перевірку $I/16$.

Завдання 3. Канонічні форми

- Побудуйте таблицю істинності $F(A,B,C)$ для наборів 000...111. Додайте потрібні проміжні стовпці.
- Випишіть індекси рядків $F=1$ і $F=0$. Побудуйте повні ДДНФ і ДКНФ, не пропускаючи жодної змінної в термах.

Завдання 4. Мінімізація

- Перенесіть значення F на карту Карно з порядком стовпців BC : 00, 01, 11, 10.
- Позначте всі вибрані групи та для кожної випишіть незмінні літерали.
- Запишіть мінімальну ДНФ і перевірте її на всіх 8 наборах або порівнянням множин мінтермів.

5. Варіанти завдань

У першій таблиці всі операнди подано 8-бітовими кодами. У другій таблиці R наведено в десятковій системі.

№	X_2	Y_2	P_2	Q_2
1	00101101	00011011	01001001	00011100
2	11001000	01100100	01011011	00100101
3	01100100	00111100	01101000	00101101
4	10010110	10000010	01111110	00111010
5	11111111	00000001	01010101	00010011
6	01111111	00000001	01100011	00101011
7	10000000	11111111	01110101	00110100
8	01000000	00100000	01000100	00010101
9	11110000	00100000	01101101	01000000
10	01011010	01000110	01110011	01001101
11	10110100	01011010	01111010	00110111
12	11010010	01010000	01100000	00100110
13	01000110	10111110	01101111	01000101
14	01101110	00011110	01011000	00100111
15	10001100	10001100	01101001	00101111
16	00100001	01100011	01110111	00111101
17	10101010	00111100	01100101	00110110

№	Z_2	Операція	R_{10}	$F(A,B,C)$
1	00101101	LSL на 1	1,5	$(A \wedge B) \vee C$
2	10110100	LSR на 2	2,25	$A \vee (B \wedge C)$
3	11110000	ASR на 2	3,75	$(A \oplus B) \wedge C$
4	01011001	LSL на 1	4,5	$(A \oplus B) \vee C$
5	10011000	ASR на 1	5,125	$(\bar{A} \wedge B) \vee (A \wedge C)$
6	01100110	LSR на 1	6,375	$(A \wedge \bar{B}) \vee C$
7	11101000	ASR на 2	7,625	$\bar{A} \vee (B \wedge C)$
8	00110111	LSL на 2	8,75	$(A \wedge C) \vee (B \wedge C)$
9	11010000	LSR на 2	9,5	$(A \wedge B) \vee (B \wedge C)$
10	10101100	ASR на 1	10,25	$(\bar{A} \wedge C) \vee (B \wedge \bar{C})$
11	01010101	LSR на 2	11,75	$(A \wedge \bar{B}) \vee (A \wedge C)$
12	11100100	ASR на 2	12,125	$(\bar{A} \wedge \bar{B}) \vee C$
13	00111100	LSL на 1	13,375	$(A \wedge B) \vee (\bar{A} \wedge C)$
14	10000000	ASR на 1	14,625	$(A \oplus B) \vee (A \wedge C)$
15	01111001	LSR на 1	15,5	$(A \wedge \bar{B} \wedge C) \vee (A \wedge B \wedge C)$
16	11001100	ASR на 2	2,875	$(\bar{A} \wedge B) \vee (\bar{A} \wedge C)$
17	00101011	LSL на 2	6,5625	$(A \wedge B) \vee (A \wedge \bar{B}) \vee C$

6. Вимоги до звіту

- Номер, тема, мета та номер варіанта.
- Повні проміжні обчислення для кожного завдання.
- Таблиця істинності, ДДНФ, ДКНФ і заповнена карта Карно.
- Незалежні перевірки арифметичних операцій, Q4.4 та мінімізації.
- Короткий висновок: що було обчислено, які прапорці отримано і до якого виразу мінімізовано функцію.

7. Питання для захисту

1. Що таке перенесення на вході та перенесення на виході розряду?
2. За якими правилами додають два біти без перенесення на вході?
3. Що показує CF?
4. Що показує OF?
5. Як визначити OF під час додавання?
6. Чому CF і OF не можна вважати одним і тим самим?
7. Як виконати віднімання $A-B$ за допомогою суматора?
8. Як утворити додатковий код від'ємного числа?
9. Чим LSL відрізняється від LSR?
10. Чим ASR відрізняється від LSR?
11. Які біти називають втраченими під час зсуву?
12. Що означає формат Q4.4?
13. Як закодувати число у беззнаковому Q4.4?
14. З яких полів складається IEEE 754 binary32?
15. Що таке булева змінна та булева функція?
16. Як працюють AND, OR, XOR і NOT?
17. Який пріоритет логічних операцій використано в роботі?
18. Скільки рядків має таблиця істинності для n змінних?
19. Що таке літерал, мінтерм і макстерм?
20. Як утворити мінтерм за рядком таблиці істинності?
21. Як утворити макстерм за рядком таблиці істинності?
22. Що таке ДДНФ і ДКНФ?
23. Чому стовпці карти Карно розташовують у коді Грея?
24. Які групи дозволено утворювати на карті Карно?
25. Чи є протилежні краї карти Карно сусідніми та чи можна перекривати групи?
26. Як перевірити результат мінімізації?

8. Контрольний список перед здаванням

- Усі коди мають рівно 8 бітів.
- Для додавання показано перенесення, а CF і OF не переплутано.
- Для віднімання показано інверсію Q і додавання 1.
- Для зсуву вказано заповнювальні та втрачені біти.
- Для Q4.4 виконано пряме й зворотне перетворення.
- ДДНФ побудовано за одиничними, ДКНФ — за нульовими наборами.
- Стовпці карти Карно йдуть у порядку 00, 01, 11, 10.
- Мінімізований вираз перевірено на всіх наборах.