

ЛЕКЦІЯ 2

Системи числення та представлення інформації

1. Система числення та її основа

Система числення — сукупність правил запису й читання чисел за допомогою визначеного набору цифр. Основа системи b показує кількість допустимих цифр. У системі з основою b використовуються цифри від 0 до $b-1$.

Система	Основа	Допустимі цифри	Приклад
Двійкова (binary)	2	0, 1	101101_2
Вісімкова (octal)	8	0...7	316_8
Десяткова (decimal)	10	0...9	206_{10}
Шістнадцяткова (hexadecimal)	16	0...9, A...F	$7B_{16}$ або $0x7B$

Нижній індекс біля числа вказує основу: 1011_2 , 73_8 , 59_{10} , $2F_{16}$. У програмному коді часто застосовують префікси: $0b$ для двійкового, $0o$ для вісімкового та $0x$ для шістнадцяткового запису. Якщо основу не зазначено, число зазвичай вважають десятковим, але в технічному тексті основу краще вказувати явно.

2. Позиційний принцип

У позиційній системі значення цифри залежить від її позиції. Позиції цілої частини нумерують справа наліво, починаючи з нуля. Вага позиції i дорівнює b^i , де b — основа. Значення числа обчислюють як суму добутків цифр на їхні ваги: $N = \sum d_i \cdot b^i$. Символ Σ (сигма) означає сумування за всіма позиціями.

Докладний приклад: 316_8 у десятковій системі

- Основа дорівнює 8, тому ваги позицій справа наліво: 8^0 , 8^1 , 8^2 .
- Цифра 6 стоїть у позиції 0: $6 \cdot 8^0 = 6 \cdot 1 = 6$.
- Цифра 1 стоїть у позиції 1: $1 \cdot 8^1 = 1 \cdot 8 = 8$.
- Цифра 3 стоїть у позиції 2: $3 \cdot 8^2 = 3 \cdot 64 = 192$.
- Сума: $192 + 8 + 6 = 206$. Отже, $316_8 = 206_{10}$.

Докладний приклад: $2F_{16}$ у десятковій системі

У шістнадцятковій системі $F=15$. Тому $2F_{16} = 2 \cdot 16^1 + 15 \cdot 16^0 = 32 + 15 = 47_{10}$. Перевірка: $47 \div 16$ дає частку 2 та остачу 15, яку позначають F .

3. Перетворення десяткового цілого числа у двійкове

Для перетворення додатного десяткового цілого числа послідовно ділять число й отримані частки на 2. Остача кожного ділення є черговим бітом. Перша остача відповідає LSB (Least Significant Bit — молодший значущий біт), тому остаточний код читають від останньої остачі до першої.

Докладний приклад: 45_{10} у двійковій системі

Ділення	Частка	Остача
$45 \div 2$	22	1
$22 \div 2$	11	0

Ділення	Частка	Остача
$11 \div 2$	5	1
$5 \div 2$	2	1
$2 \div 2$	1	0
$1 \div 2$	0	1

Остачі знизу вгору: 101101_2 . Перевірка за вагами: $1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 32 + 8 + 4 + 1 = 45$. Отже, $45_{10} = 101101_2$.

Докладний приклад: 26_{10} у двійковій системі

$26 \div 2 = 13$, остача 0; $13 \div 2 = 6$, остача 1; $6 \div 2 = 3$, остача 0; $3 \div 2 = 1$, остача 1; $1 \div 2 = 0$, остача 1. Читаємо остачі у зворотному порядку: 11010_2 . Перевірка: $16 + 8 + 2 = 26$.

4. Перетворення двійкового числа у десяткове

Кожний біт множать на вагу його позиції 2^i і додають добутки. MSB (Most Significant Bit — старший значущий біт) розташований зліва та має найбільшу вагу. LSB розташований справа і має вагу $2^0 = 1$.

Докладний приклад: 101101_2 у десятковій системі

Позиція i	5	4	3	2	1	0
Біт d_i	1	0	1	1	0	1
Вага 2^i	32	16	8	4	2	1
Добуток	32	0	8	4	0	1

Сума ненульових добутків: $32 + 8 + 4 + 1 = 45$. Отже, $101101_2 = 45_{10}$.

5. Взаємне перетворення двійкової, вісімкової та шістнадцяткової систем

Оскільки $8 = 2^3$, одна вісімкова цифра точно відповідає трьом бітам. Оскільки $16 = 2^4$, одна шістнадцяткова цифра відповідає чотирьом бітам. Для цілих чисел біти групують справа наліво; неповну старшу групу доповнюють нулями зліва. Дописування нулів зліва не змінює значення цілого числа.

Десяткове	Двійкова трійка	Вісімкова	Двійкова четвірка	Шістнадцяткова
0	000	0	0000	0
1	001	1	0001	1
2	010	2	0010	2
3	011	3	0011	3

Десяткове	Двійкова трійка	Вісімкова	Двійкова четвірка	Шістнадцяткова
4	100	4	0100	4
5	101	5	0101	5
6	110	6	0110	6
7	111	7	0111	7
8	—	—	1000	8
9	—	—	1001	9
10	—	—	1010	A
11	—	—	1011	B
12	—	—	1100	C
13	—	—	1101	D
14	—	—	1110	E
15	—	—	1111	F

Докладний приклад: двійкове число у вісімкове

Для 11010110_2 утворюємо трійки справа: 11 | 010 | 110. Доповнюємо старшу групу: 011 | 010 | 110. За таблицею $011_2=3_8$, $010_2=2_8$, $110_2=6_8$. Отже, $11010110_2=326_8$. Перевірка: $3 \cdot 64 + 2 \cdot 8 + 6 = 214$; двійкове число також дорівнює $128 + 64 + 16 + 4 + 2 = 214$.

Докладний приклад: 0x3A у двійкову та десяткову системи

Префікс 0x вказує на шістнадцятковий запис. Цифрі 3 відповідає 0011_2 , цифрі A=10 — 1010_2 . Отже, $0x3A=00111010_2$. Десяткове значення: $3 \cdot 16 + 10 = 58$. Незалежна перевірка двійкового коду: $32 + 16 + 8 + 2 = 58$.

6. Розрядність і беззнакові числа

Розрядність (bit width — бітова ширина) — кількість бітів, відведених для коду. Для n-бітового беззнакового формату (unsigned format) найменший код містить n нулів і представляє 0. Найбільший код містить n одиниць і представляє $2^n - 1$. Діапазон: $0 \dots 2^n - 1$.

Докладний приклад: 8-бітовий беззнаковий формат

Кількість комбінацій: $2^8=256$. Найменше значення: $00000000_2=0$. Найбільше: $11111111_2=128+64+32+16+8+4+2+1=255$. Значення 256 потребує вже 9 бітів: 100000000_2 .

Розрядність n	Кількість кодів	Unsigned-діапазон
4	16	0...15

Розрядність n	Кількість кодів	Unsigned-діапазон
6	64	0...63
8	256	0...255
16	65536	0...65535

7. Знакові способи кодування

Для подання додатних і від'ємних цілих чисел потрібно визначити правило інтерпретації старшого біта. У навчальній практиці розглядають прямий код, обернений код і додатковий код. Бітова послідовність не містить назви формату, тому її значення завжди слід визначати разом із розрядністю та правилом кодування.

7.1. Прямий код

У прямому коді (sign-magnitude — знак і модуль) MSB є бітом знака: 0 означає плюс, 1 — мінус; решта бітів містять модуль. Для 8 бітів $+13=00001101_2$, а $-13=10001101_2$. Формат має два нулі: 00000000_2 і 10000000_2 .

7.2. Обернений код

Обернений код (ones' complement — доповнення до одиниці) від'ємного числа утворюють інверсією всіх бітів додатного коду: 0 замінюють на 1, а 1 — на 0. Для $+13=00001101_2$ отримуємо $-13=11110010_2$. Тут також існують два нулі: 00000000_2 та 11111111_2 .

7.3. Додатковий код

Додатковий код (two's complement — доповнення до двох) від'ємного числа утворюють інверсією всіх бітів модуля та додаванням 1. Для n бітів той самий результат можна отримати як $2^n - |x|$. Додатковий код має один нуль і дозволяє використовувати один суматор для додавання та віднімання, тому є основним знаковим форматом сучасних процесорів.

Формат 8 бітів	+13	-13	Нуль	Діапазон
Прямий	00001101	10001101	Два коди	-127...127
Обернений	00001101	11110010	Два коди	-127...127
Додатковий	00001101	11110011	Один код	-128...127

8. Кодування від'ємного числа у додатковому коді

Докладний приклад: -13 у 8 бітах

1. Модуль: $|-13|=13$.
2. Додатний 8-бітовий код: $13_{10}=00001101_2$.
3. Інверсія всіх бітів: 11110010_2 .
4. Додавання 1: $11110010_2 + 1_2 = 11110011_2$.
5. Перевірка другим способом: $2^8 - 13 = 256 - 13 = 243$; $243_{10} = 11110011_2$.

Докладний приклад: –5 у трьох форматах

Модуль 5 має 8-бітовий код 00000101_2 . Прямий код: установити біт знака — 10000101_2 . Обернений код: інвертувати всі біти — 11111010_2 . Додатковий код: до оберненого додати 1 — 11111011_2 .

9. Декодування додаткового коду

Якщо MSB дорівнює 0, код представляє невід'ємне число й читається як звичайне беззнакове. Якщо MSB дорівнює 1, число від'ємне. Його можна декодувати двома рівноправними способами: інвертувати код і додати 1 для отримання модуля або від беззнакового значення U відняти 2^n .

Докладний приклад: декодування 11110011_2

1. MSB=1, тому число від'ємне.
2. Перший спосіб: $11110011_2 \rightarrow$ інверсія $00001100_2 \rightarrow +1 = 00001101_2 = 13$. Отже, значення –13.
3. Другий спосіб: беззнакове $U=243$. Для 8 бітів $243-256=-13$.
4. Обидва способи дали однаковий результат.

10. Однакові біти — різні значення

Код 11111011_2 як 8-бітове unsigned-число дорівнює 251. Як 8-бітове число в two's complement він дорівнює $251-256=-5$. Як набір прапорців кожна позиція могла б мати окреме логічне значення. Отже, без формату не можна однозначно відповісти, що означає послідовність бітів.

Код	Інтерпретація	Значення
11111011	8-bit unsigned	251
11111011	8-bit two's complement	–5
11111011	Набір незалежних прапорців	Сім прапорців 1, один прапорець 0

11. Терміни та позначення

Термін	Пояснення
MSB	Most Significant Bit — старший значущий біт.
LSB	Least Significant Bit — молодший значущий біт.
Unsigned	Беззнаковий формат, усі біти представляють невід'ємну величину.
Signed	Знаковий формат, що передбачає подання додатних і від'ємних значень.
Інверсія	Заміна кожного біта на протилежний: $0 \leftrightarrow 1$.
Розрядна сітка	Фіксована кількість позицій, у межах яких зберігається код.