

Part 1: Representations

4-Bit & 5-Bit Sign-Magnitude, 1's Complement, and 2's Complement Conversions

Exercise 1: 4-Bit & 5-Bit Representations

4-Bit System Problems

Express the following decimal integers using **4-bit binary** representation:

Numbers: +6, -6, +2, -2

Range: [-8 to +7]

1 sign bit + 3 magnitude bits

5-Bit System Problems

Express the following decimal integers using **5-bit binary** representation:

Numbers: +13, -13, +9, -9

Range: [-16 to +15]

1 sign bit + 4 magnitude bits

Exercise 1 Solution Matrix (4-Bit vs 5-Bit)

Decimal	Width	Sign-Magnitude	1's Complement	2's Complement
+6 / -6	4-Bit	0110 / 1110	0110 / 1001	0110 / 1010
+2 / -2	4-Bit	0010 / 1010	0010 / 1101	0010 / 1110
+13 / -13	5-Bit	01101 / 11101	01101 / 10010	01101 / 10011
+9 / -9	5-Bit	01001 / 11001	01001 / 10110	01001 / 10111

(to 5-bit) sign-extends positive numbers with leading 0s and 2's complement negative numbers with leading 1s.

Part 2: Binary Arithmetic

2's Complement Addition in 4-Bit and 5-Bit Systems

Exercise 2: Binary Arithmetic Problems

4-Bit Addition Exercises

Compute using 4-bit 2's complement:

1) $5 + (-3)$ | $(+5 = 0101, -3 = 1101)$

2) $3 + (-6)$ | $(+3 = 0011, -6 = 1010)$

Observe when carry-out occurs and how result signs reflect operand magnitudes.

5-Bit Addition Exercises

Compute using 5-bit 2's complement:

1) $11 + (-5)$ | $(+11 = 01011, -5 = 11011)$

2) $7 + (-12)$ | $(+7 = 00111, -12 = 10100)$

Verify that discarding the MSB carry yields the correct signed sum.

Exercise 2 Solutions (4-Bit & 5-Bit)

System	Operation	Binary Sum	Discard Carry	Decimal
4-Bit	5 + (-3)	0101 + 1101	(1) 0010 → 0010	+2
4-Bit	3 + (-6)	0011 + 1010	(0) 1101 → 1101	-3
5-Bit	11 + (-5)	01011 + 11011	(1) 00110 → 00110	+6
5-Bit	7 + (-12)	00111 + 10100	(0) 11011 → 11011	-5

Note on Carries: Generating a carry out of the MSB does NOT mean overflow occurred! In 2's complement, carry-out is ignored when adding numbers of opposite signs.

Exercise 3: Overflow in 4-Bit vs 5-Bit

4-Bit Overflow

5 + 4 in 4-bit:

0101 (+5) + 0100 (+4)

= 1001 (-7)

Range: [-8, +7]

Correct sum (+9) exceeds max +7. Pos + Pos yielded Neg. **Overflow!**

5-Bit Overflow

11 + 9 in 5-bit:

01011 (+11) + 01001 (+9)

= 10100 (-12)

Range: [-16, +15]

Correct sum (+20) exceeds max +15. Pos + Pos yielded Neg. **Overflow!**

Valid Additions

4-Bit: 6 + (-4) = 0010 (+2)

5-Bit: 13 + (-8) = 00101 (+5)

Opposite Signs

Adding a positive and negative number can never overflow in either system. **No Overflow!**