Homework

-

Module

3

Reference

:

Chapter 2, Sections

2.5

-

2.7

1

.

Add

these pairs of binary numbers.

1. 1011 + 0001 = 12 or 1100
2. 0000 + 1010 = 10 or 1010
3. 1100 + 0011 = 15 or 1111
4. 0101 + 0110 = 11 or 1011
5. 1111 + 0001 = 16 or 10000

Note: When no specific notation is given, binary numbers are treated as unsigned numbers by default.

2. Add the following 2’s complement binary numbers. Also express each answer in decimal.

1. 01 + 1011 = -4
2. 11 + 01010101 = 88
3. 0101 + 110 = -5
4. 01 + 10 = -1

3. Add the following unsigned binary numbers. Also express each answer in decimal.

1. 01 + 1011 = 12
2. 11 + 01010101 = 88
3. 0101 + 110 = 11
4. 01 + 10 = 3

4. The following binary numbers are 4-bit 2’s complement integers. Which of the following operations generate overflow?

a. 1100 + 0011 = 1111 -4 + 3 = -1

1. 1100 + 0100 = 0000 -4 + 4 = 0
2. 0111 + 0001 = 1000 7 + 1 = -8 Overflow
3. 1000 - 0001 = same as 1000 + 1111 = 0111 = -8 + (-1)

= 7 Overflow

1. 0111 + 1001 = 0000 7 + (-7) = 0

1. How can you tell that overflow has resulted from the addition of two 2’s complement binary numbers?

If both operands are positive and the result is negative or vice versa.

1. How can you tell that overflow has resulted from the addition of two unsigned binary numbers?

If there are more bits than are allocated for it. For example, if there are 4 bits allocated and the result is 5 bits then it would be an overflow.

1. It’s not possible to generate overflow when you add a positive 2’s complement binary number to a negative 2’s complement binary number. Why?

Because if both operands are one sign, then the result is the opposite that’s when an overflow happens but since we use a negative and a positive sign either a positive or negative result is okay.

1. Compute the following.
2. 01010111 AND 11010111 = 01010111
3. 101 AND 110 = 100
4. 11100000 AND 10110100 = 10100000
5. 00011111 AND 10110100 = 00010100
6. (0011 AND 0110) AND 1101 = 0000
7. 0011 AND (0110 AND 1101) = 0000

9. Compute the following.

1. 01010111 OR 11010111 = 11010111
2. 101 OR 110 = 111
3. 11100000 OR 10110100 = 11110100
4. 00011111 OR 10110100 = 10111111
5. (0101 OR 1100) OR 1101 = 1101
6. 0101 OR (1100 OR 1101) = 1101

10. Compute the following.

1. NOT(1011) OR NOT(1100) = 0111
2. NOT(1000 AND (1100 OR 0101)) = 0111
3. NOT(NOT(1101)) = 1101
4. (0110 OR 0000) AND 1111 = 0110

11. Convert the following unsigned binary numbers to hexadecimal.

1. 1101 0001 1010 1111 = D1AF
2. 001 1111 = 1F
3. 1 = 1
4. 1110 1101 1011 0010 = EDB2

12. Convert the following hexadecimal numbers to binary

1. 0x10 = 0001 0000
2. 0x801 = 1000 0000 0001
3. 0xF731 = 1111 0111 0011 0001
4. 0x0F1E2D = 0000 1111 0001 1110 0010 1101
5. 0xBCAD = 1011 1100 1010 1101

13. Convert the following hexadecimal representations of 2’s complement binary numbers to decimal numbers.

a. 0xF0 = -16

1. 0x7FF = 2047
2. 0x16 = 22
3. 0x8000 = -32768

14. Convert the following decimal numbers to hexadecimal representations of 2’s complement numbers.

1. 256 = 100
2. 111 = 6F
3. 123,456,789 = 75BCD15
4. -44 = 2C

15. Perform the following additions. The hexadecimal numbers below are in 2’s complement notation. Provide your answers in hexadecimal. a. 0x025B + 0x26DE

1. 0x7D96 + 0xF0A0
2. 0xA397 + 0xA35D
3. 0x7D96 + 0x7412

16. Perform the following logical operations. Express your answers in hexadecimal notation.

1. 0x5478 AND 0xFDEA
2. 0xABCD OR 0x1234
3. NOT((NOT(0xDEFA)) AND (NOT(0xFFFF)))
4. 0x00FF XOR 0x325C

17. Fill in the truth table for the equations given. The first line is done as an example.

Q1 = NOT(NOT(X) OR (X AND Y AND Z))

Q2 = NOT((Y OR Z) AND (X AND Y AND Z))

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | Y | Z | Q1 | Q2 |
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 |  |  |
| 0 | 1 | 0 |  |  |
| 0 | 1 | 1 |  |  |
| 1 | 0 | 0 |  |  |
| 1 0 1  1 1 0  1 1 1 | | | |  | |
| 18. Fill in the truth table for the equations given. The first line is done as an example. | | | | | |

Q1 = NOT(A AND B)

Q2 = NOT(NOT(A) AND NOT(B))

|  |  |  |  |
| --- | --- | --- | --- |
| A  0 | B | Q1 | Q2 0 |
| 0 | 1 |
| 0 | 1 |  |  |
| 1 | 0 |  |  |
| 1 | 1 |  |  |
|  |  |  |  |

19. The following hexadecimal numbers represent strings of ASCII values. Translate these hexadecimal numbers into strings of characters by interpreting each pair of hex digits (8 bits) into the character represented by its ASCII value.

1. 0x48656C6C6F21
2. 0x68454C4C4F21
3. 0x436f6d70757465727321
4. 0x4C432D32

1. What arithmetic operation can be used to convert the binary representation for the number 3 (0000 0011) into the ASCII representation for the character ‘3’ (0011 0011)? What about converting the binary value 4 into the ASCII character ‘4’? What about any digit?

1. Convert the following 2’s complement binary numbers into 8-bit 2’s complement binary numbers without changing their values. a. 1010
2. 011001
3. 1111111000
4. 01