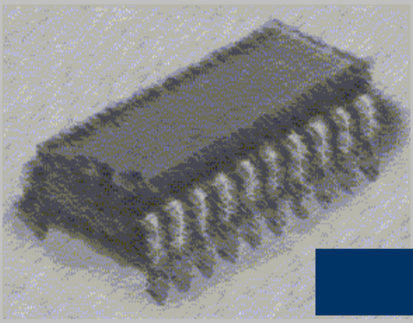
Exercises

Lesson 1: The Datapath



Computer Structure and Organization

Graduate in Computer Sciences
Graduate in Computer Engineering

1. Add following numbers by using a 16 bits look ahead adder.

   A = 0001 1010 0011 0011 and B = 1110 0101 1110 1011

   Please, obtain gi y pi values.

2. Add following numbers by using a 16 bits look ahead adder.

   A = 0111 1110 0000 1011 and B = 1111 1101 1100 1011

   Please, obtain gi y pi values.

3. Add following numbers by using a 16 bits block look ahead adder. Blocks are 4 bits length.

   A = 0001 1010 0011 0011 and B = 1110 0101 1110 1011

   Please, obtain gi, pi, Pi y Gi and output carry C4. Carry in is 1

4. Add following numbers by using a 16 bits block look ahead adder. Blocks are 4 bits length.

   A = 0111 1110 0000 1011 and B = 1111 1101 1100 1011

   Please, obtain gi, pi, Pi y Gi and output carry C4. Carry in is 1

5. Use a floating point subtract-adder to add numbers represented as following:

   - Exponent: 8 bits, excess 2^{8-1}.
   - Mantissa: 8 bits, complement 2, normalized, fractional and without implicit bit

   a) Perform add operation of numbers: A and B, A: 1000 0011 **|** 0110 0011 y B: 1000 0110 **|** 1001 1100 (exponent **|** mantissa)

b) ¿Is there any different with the real result? ¿Why?

6. Do the same exercise as above but supposing the subtract-adder is provided with two guard digits and a sticky bit. The subtract-adder uses nearest rounding:

   a) Perform add operation over A: 1000 0110 | 0110 0010 and B: 1000 0010 | 1001 0000 (exponent | mantis)

   b) ¿Is there any different with the real result? ¿Why?

7. Lets a floating-point numbers subtract-adder. Numbers are represented as following: mantissa normalized, fractional without implicit bit, C1 expressed and exponent excess 2^{4-1}represented, The adder is also provided with two guard digits and a sticky bit. Nearest rounding is used.

   a) Obtain A+B A: 1101 | 10110 y B: 1101 | 01011 (exponent | mantissa

8. Lets A: 110011 y B: 010001, both numbers are binary represented, get A x B using add-shift algorithm

9. Get A x B, A: 0100110 y B: 0110010, both numbers are C2 expressed.

10. Get A x B, A: 111110 y B: 1111111 both numbers are C2 expressed.

11. Get quotient of the division of D: 0100100 y d: 1001, both numbers are binary expressed.

    a) Use with restoring division algorithm.

    b) Use without restoring division algorithm.

12. Get quotient of the division of D: 0100110 y d: 1110, both numbers are binary expressed.

    a) Use with restoring division algorithm.

    b) Use without restoring division algorithm.

13. Get quotient of the division of D: 0100101 y d: 0111, both numbers are binary expressed.

    a) Use with restoring division algorithm.

    b) Use without restoring division algorithm.

14. Lets a floating-point adder used in numbers represented with:

    - Exponent excess 2^{n-1} expressed (8 bits).
    - Mantissa complement 1, normalized, fractional, without implicit bit represented (8 bit)

    Adder has two guard digits, a sticky bit and adder uses nearest rounding method.

    Get A+B (exponent || mantissa):

    - A = 1000 0111 **||** 1010 0000
    - B = 1000 0010 **||** 0110 0000