Name: ____ Lab Section: _______

Introduction: For the decimal to binary/octal/hex conversions, use either the subtraction method or the division/multiplication methods described in class and in Chapter 2 of your ECOA2e text. For many problems, I give you the answer; that means you must ***show all your work*** on securely attached (stapled) separate sheets. Your answers must be in order and each answer must be numbered consecutively. Your answer must ***demonstrate clearly*** that you have a method for getting from the question to the correct answer (and not the other way around).

See the Class Notes here: **`http://teaching.idallen.com/cst8214/08w/`**

Part I – decimal to binary (unsigned), including fractions

References: ECOA2e Section 2.3, 2.4 and ECOA2e Chapter 2 slides 6-10, 18-24

1. Write down all the powers of two from zero (“1”) to 16 (“65,536”).
2. Write down all the negative powers of two from -1 (“0.5”) to -4 (“0.0625”).
3. Convert 25_{10} decimal to 11001_2 binary (see ECOA2e Chapter 2 slide 8).
4. Convert 147_{10} decimal to 10010011_2 binary (see ECOA2e page 43).
5. Convert 0.75_{10} to 0.11_2 binary (see ECOA2e Chapter 2 slide 19).
6. Convert 0.8125_{10} to 0.1101_2 binary (see ECOA2e Chapter 2 slide 21).
7. Convert 0.34375_{10} to 0.0101_2 binary, stopping at four fractional bits (ECOA2e Ex. 2.7 p.45)

Part II – binary, octal (base 8), and hexadecimal (base 16)

References: ECOA2e Sections 2.2, 2.3.3 and ECOA2e Chapter 2 slides 25-27.

When converting from binary to octal or hex digits, always start grouping binary bits from the ***right*** of the binary number. If you run out of bits for the last group of bits on the left, assume the leftmost missing bits are zero, e.g. treat a left-over 11_2 as 011_2 (when converting to octal) or as 0011_2 (when converting to hexadecimal).

8. Write down the sixteen binary four-bit patterns for the hex digits **0,…,9,A,…,F** along with their decimal equivalents.

9. Convert the following binary value to octal (base 8) by using groups of three bits, starting from the ***right***. *Hint: Your answer will have eight octal digits and the rightmost octal digit will be “0”:* $110011101010100001111000_2$

   Answer: ____

10. Convert the following binary to hexadecimal (base 16) by using groups of four bits, starting from the ***right***. *Hint: Your answer will have eight hex digits and the rightmost hex digit will be “A”*: $11101101001110011100000000011010_2$

    Answer: ____

11. Convert the following binary to hexadecimal. *Hint: Your answer will have eight hex digits, all different:* $1011000010010111111001001110110_2$

    Answer: ____

Part III – miscellany

Reread the introduction to this lab. You must ***show all your work clearly***.

12. Convert 6235_8 octal to $C9D_{16}$ hexadecimal (ECOA2e Example 2.9 p.46).
13. What are the largest and smallest integers an 8-bit word can hold using an unsigned representation?
14. What are the largest and smallest integers an 8-bit word can hold using a sign-magnitude representation? (p.47)
15. What are the largest and smallest integers an 8-bit word can hold using a one's complement representation?
16. What are the largest and smallest integers an 8-bit word can hold using a two's complement representation?
17. What are the largest and smallest integers a 16-bit word can hold using an unsigned representation?
18. What are the largest and smallest integers a 16-bit word can hold using a sign-magnitude representation?
19. What are the largest and smallest integers a 16-bit word can hold using a one's complement representation?
20. What are the largest and smallest integers a 16-bit word can hold using a two's complement representation?
21. Convert 23_{10} to 8-bit 00010111_2 binary one's complement (ECOA2e Example 2.16 p.53).
22. Write 23_{10} in octal and hexadecimal.
23. Convert -9_{10} to 8-bit 11110110_2 binary one's complement (ECOA2e Example 2.16 p.53).
24. Write 11110110_2 in octal and hexadecimal.
25. Convert -23_{10} to 8-bit 11101000_2 binary one's complement.
26. How do you know that a two's-complement addition has overflowed?
27. Convert 23_{10} to 8-bit 00010111_2 binary two's complement (ECOA2e Example 2.19 p.54).
28. Write 00010111_2 in octal and hexadecimal.
29. Convert -9_{10} to 8-bit 11110111_2 binary two's complement (ECOA2e Example 2.19 p.54).
30. Write 11110111_2 in octal and hexadecimal.
31. Convert -23_{10} to 8-bit 11101001_2 binary two's complement (ECOA2e Example 2.19 p.54).
32. Write 11101001_2 as octal and hexadecimal.
33. Write 10010011_2 as octal and hexadecimal.
34. Convert 8-bit 10010011_2 binary *unsigned* to 147_{10} decimal.
35. Convert 8-bit 10010011_2 binary *sign-magnitude* to -19_{10} decimal (note the negative).
36. Convert 8-bit 10010011_2 binary *one's complement* to -108_{10} decimal (note the negative).
37. Convert 8-bit 10010011_2 binary *two's complement* to -109_{10} decimal (note the negative).
38. Copy the left column of ECOA2e Table 2.2 p.63 and perform the given two's complement additions. Without looking, fill in the remaining four columns based on the results. Add another column that states whether the result is correct if treated as ***unsigned*** math instead of two's complement. (Note: this table 2.2 has one printing error in it.)
39. Convert 16-bit two's complement $1A8C_{16}$ to $6,796_{10}$ decimal.
40. Convert 16-bit two's complement $7FFF_{16}$ to $32,767_{10}$ decimal.
41. Convert 16-bit two's complement 8000_{16} to decimal $-32,768_{10}$ (note the negative).
42. Convert 16-bit two's complement $A123_{16}$ to decimal $-24,285_{10}$ (note the negative).
43. Convert 16-bit two's complement $FFFF_{16}$ to decimal -1_{10} (note the negative).
44. Circle the negative numbers (16-bit two's complement):
    Answer here: $6FFF_{16}$ $7FFF_{16}$ 8000_{16} 8001_{16} $9FC5_{16}$ $A123_{16}$ $BFFF_{16}$
45. Add 16-bit two's complement $ABCD_{16}$ to $7FFF_{16}$ and give the Result, Carry, and Overflow. Is the result correct?
46. Add 16-bit two's complement $8A9C_{16}$ to $ABCD_{16}$ and give the Result, Carry, and Overflow. Is the result correct?
47. Add 16-bit two's complement 9999_{16} to 4321_{16} and give the Result, Carry, and Overflow. Is the result correct?