CMSC 130 – Logic Design and Digital Computer Circuits
Handout # 2: COMPUTER ARITHMETIC

Recall Binary Addition:

0 + 0 = 0	1 + 0 = 1
0 + 1 = 1	1 + 1 = 10

Subtraction Using Complements

Given:
Minuend
– Subtrahend
Difference

1. 10’s complement

a. $(72532 - 3250)_{10}$

M	=		72532
10’s complement of S	=	+	96750
Sum	=		169282
Discard end carry 10^5	=	–	100000
Answer	=		69282

b. $(3250 - 72532)_{10}$

M	=		03250
10’s complement of S	=	+	27468
Sum	=		30718

There is no end carry.

Answer (10's comp.)	=	–	69282

2. 2’s complement

Let: X = 101 0100
Y = 100 0011

a. X – Y

X	=		1010100
2’s complement of Y	=	+	0111101
Sum	=		10010001
Discard end carry 2^7	=	–	10000000
Answer	=		10001

b. Y – X

Y	=		1000011
2’s complement of X	=	+	0101100
Sum	=		1101111

There is no end carry.

Answer (2's comp.)	=	–	10001

3. 1’s complement

Let: X = 101 0100
Y = 100 0011

a. X – Y

X	=		1010100
1’s complement of Y	=	+	0111100
Sum	=		10010000
End-around carry	=	+	1
Answer	=		10001

b. Y – X

Y	=		1000011
1’s complement of X	=	+	0101011
Sum	=		1101110

There is no end carry.

Answer (1's comp.)	=	–	10001

Overflow/underflow Detection

Overflow

- occurs when an arithmetic operation yields a result that is greater than the range's positive limit
- when the last carry-in is not equal to the carry-out.

Underflow

- occurs when an arithmetic operation yields a result that is lesser than the range's negative limit
- when the last carry-in is not equal to the carry-out.

Using 5 bits:

1. positive + positive

1.a)

```
          0                 last carry-in
          01001     (9)
    +     00101     (5)
        0 01110     (14)    since, last carry-in = carry-out
                            no overflow/underflow
```

1.b)

```
          1                 last carry-in
          01101     (13)
    +     00111     (7)
        0 10100             since, last carry-in ≠ carry-out
                            overflow (should be 20)
```

2. negative + negative

1's complement
2.a)

```
          0                 last carry-in
          11000     (-7)
    +     10000     (-15)
        1 01000
    +         1             end-round carry
          01001     (9)     underflow (should be -22)
```

2's complement
2.b)

```
              0                      last carry-in
              11001        (-7)
       +      10001        (-15)
            1 01010        (10)      underflow (should be -22)
```

BCD Addition

BCD is from 0000 to 1001. If the sum is less than or equal to 1001, the answer is still valid.

Example:

```
1.    4     0100
     +5     0101
      9     1001  Valid!

But if we have:
      4     0100
     +8     1000
     12     1100 (not in BCD)

To correct the answer, add 6 (0110) to it, so
            0100
          + 1000
            1100
          +  0110
       0001 0010                      = 12

2.    8     1000
     +9     1001
     17     0001
          + 0110
       0001 0111                      = 17

            1
3. 156     0001   0101   0110
  +427     0100   0010   0111
   583     0101   1000   1101
                         0110
                         0011

           0101   1000   0011         = 583
```