Binary Arithmetic

Addition/Subtraction:

```
15 + 7                  00001111        =   15   upper
                        00000111        =   7    lower
                        00010110        =   22   sum



----------------------------------------------------------
15 - 7                  00001111        =   15   upper
                        00000111        =   7    lower

convert subtrahend      00000111    original is changed to
                        11111000    ones complement
                               1    add 1
                        11111001    twos complement


                        00001111        =   15   minuend
                        11111001        =   -7   subtrahend
                ==> 1   00001000        =   8    difference
```

Note the high-order carry is dropped by the hardware.

```
----------------------------------------------------------
7 - 15                  00000111        =   7    upper
                        00001111        =   15   lower

convert subtrahend      00001111    original is changed to
                        11110000    ones complement
                               1    add 1
                        11110001    twos complement


                        00000111        =   7    minuend
                        11110001        =   -15  subtrahend
                        11111000        =   -8   difference
```

```
-15 + -15                 11110001          =    -15  upper
                          11110001          =    -15  lower

Note that each operand is already in twos complement form.


                          11110001          =    -15  minuend
                          11110001          =    -15  subtrahend
               ==> 1      11100010          =    -30  difference

Note the high-order carry is dropped by the hardware.




------------------------------------------------------------
127 - 127                 01111111          =    127 upper
                          01111111          =    128 lower

convert subtrahend        01111111     original is changed to
                          10000000     ones complement
                                 1     add 1
                          10000001     twos complement


                          01111111          =    127  minuend
                          10000001          =    -127 subtrahend
               ==> 1      00000000          =    0    difference

Note the high-order carry is dropped by the hardware.
```

(Though not shown, number all above number representations are extended as needed with high-order bits.)

Multiplication:

```
7 x 5                00000111             = 7 multiplicand
                     00000101             = 5 multiplier
                          111
                         0000
                        11100
                       000000
                      0000000
                     00000000
                    000000000
                   0000000000
    00...        ..0000100011             = 35 product

-----------------------------------------------------------

7 x (-5)             00000111             =   7
                     11111011             =   -5
                          111
                         1110
                        00000
                       111000
                      1110000
                     11100000
                    111000000
                   1110000000
    11...        ..1111011101             =   -35

-----------------------------------------------------------

(-3) x (-2)          11111101             =   -3
                     11111110             =   -2
                          000
                         1010
                        10100
                       101000
                      1010000
                     10100000
                    101000000
                   1010000000
    00...        ..0000000110             =   +6
```

Division:

```
                          000111                 =    7 quotient
53 / 7              0111 |110101 (dividend)
                          1001   (converted twos complement)
                         101100
                          11001
                         1001011 (high-order carry is dropped)
                          111001
                         1000100                 =    4 remainder
```

(Please note that the twos complement is extended as needed with high-order 1-bits)

```
                          00011000               =    24
122 / 5             0101 |01111010
                          1011
                         100101
                          11011
                         100000010               =    2 remainder
```