We will use **fixed-point** numbers when we wish to express values in our computer that have noninteger values. A fixed-point number contains two parts. The first part is a variable integer, called *I*. The variable integer will be stored on the computer. The second part of a fixed-point number is a fixed constant, called the **resolution** Δ.

The fixed constant will NOT be stored on the computer. The fixed constant is something we keep track of while designing the software operations. The value of the number is the product of the variable integer times the fixed constant. The integer may be signed or unsigned. An unsigned fixed-point number is one that has an unsigned variable integer. A signed fixed-point number is one that has a signed variable integer.

Help

The **precision** of a number system is the total number of distinguishable values that can be represented. The precision of a fixed-point number is determined by the number of bits used to store the variable integer. On most microcontrollers, we can use 8, 16, or 32 bits for the integer. With **binary fixed point** the fixed constant is a power of 2.

Note: The use of fixed point numbers will NOT be required for this course.

CHECKPOINT 2.21

Convert the unsigned decimal number 1234 to 16-bit hexadecimal.

Hide Answer

1234 = 4*256+13*16+2 = 0x04D2.

CHECKPOINT 2.22

Convert the unsigned decimal number 10000 to 16-bit binary.

Hide Answer

10000 = 8192+1024+512+256+16 = 0010011100010000_2.

CHECKPOINT 2.23

Convert the 16-bit hex number 0x1234 to signed decimal.

Hide Answer

1*4096+2*256+3*16+4 = 4660.

CHECKPOINT 2.24

Convert the 16-bit hex number 0xABCD to signed decimal.

Hide Answer

-32768 + 2*4096+11*256+12*16+13 = -21555.

CHECKPOINT 2.25

Convert the signed decimal number 1234 to 16-bit hexadecimal.

Hide Answer

1234 = 4*256+13*16+2 = 0x04D2.

CHECKPOINT 2.26

Convert the signed decimal number –10000 to 16-bit binary.

Hide Answer

-10000 = -32768 +16384+4096+2048+128+64+32+16 = 1101100011110000_2.

Help

Help