Base Conversion

When converting between bases we're primarily concerned with four bases used for numbers:
2 (binary), 8 (octal), 10 (decimal), and 16 (hex).

As we'll show in a minute, converting between bases 2, 8, and 16 is fairly simple and painless. The only real work involves converting to or from decimal. Here's how that's done:

Suppose you have a decimal number, say, 245_{10}. To convert to any other base from decimal (*<u>and only from decimal!</u>*) we use the **<u>Div/Mod</u>** technique. In this technique, you are determining what the coefficients are for the powers of the base – these coefficients are the digits of the number for that base.

Let's convert from <u>decimal</u> to <u>octal</u>:

Note the desired base is used here

245_{10}:

245 DIV 8 = 30
245 MOD 8 = **5**
30 DIV 8 = 3
30 MOD 8 = **6**
3 DIV 8 = 0
3 MOD 8 = **3**

Answer is: **3 6 5**

Div = Division result as a whole number
Mod = Remainder after division

Start at bottom of column and write the MOD results from left to right

Answer: 365_8

Similarly, convert from <u>decimal</u> to <u>hex</u> (***note the conversion to hex digits***):

245_{10}:

245 DIV 16 = 15
245 MOD 16 = 5
15 DIV 16 = 0
15 MOD 16 = 15

Answer is: F 5 (Remember: 15_{10} is represented as F_{16} in hex)

Answer: $F5_{16}$

Now, to convert from any base to Decimal, use the **<u>Powers of the Base</u>** technique:

$F5_{16} = 15 \cdot 16^1 + 5 \cdot 16^0 = 240 + 5 = 245_{10}$ [Note that we are using powers of 16]

Another example:

$AB14_{16} = 10 \cdot 16^3 + 11 \cdot 16^2 + 1 \cdot 16^1 + 4 \cdot 16^0 = 10 \cdot 4096 + 11 \cdot 256 + 16 + 4$
$= 40960 + 2816 + 20 = 43796_{10}$

$A_{16} = 10_{10}$
$B_{16} = 11_{10}$

Once you have a number in hex, octal, or binary, converting between the three is a simple matter of arranging bits. Since each bit in a binary number represents a power of 2, we can simply add the values of the bits to get the value we want:

8's column 4's column 2's column 1's column

1 1 1 1_2

e.g. 1 1 1 1 $= 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 8 + 4 + 2 + 1 = 15_{10}$

0 1 1 0 $= 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 0 + 4 + 2 + 0 = 6_{10}$

When converting to binary, take the decimal value of each digit to be converted, and use this technique to find the bit values that will add up to the decimal value of the digit. If you have a hex number, convert it to binary by changing each digit into **<u>four bits</u>**:

F 5_{16} = 1111 0101 (because $F_{16} = 15_{10} = 1111_2$ and $5_{16} = 5_{10} = 0101_2$)

So the binary equivalent is 11110101_2.

More examples:

<u>$A6C_{16}$ to binary</u>:

A_{16} is equivalent to 10_{10} – to get the binary digits to add up to 10, we use 8 and 2 (because 8+2=10). No 4's and no 1's:

$A_{16} = 10_{10} = 1010_2$

$6_{16} = 6_{10} = 4 + 2 \rightarrow 0110_2$

$C_{16} = 12_{10} = 8 + 4 \rightarrow 1100_2$

$A6C_{16} = 1010\ 0110\ 1100_2$ Final answer.

If you have an octal number, convert it to binary by changing each digit into **<u>3 bits</u>** (where the columns are the 4's column, 2's column, and 1's column):

e.g. $756_8 = 111\ 101\ 110_2$ (because $7_8 = 111_2$ and $5_8 = 101_2$ and $6_8 = 110_2$)

<u>Starting with a binary number</u>:

If you have a binary number, just regroup (**<u>starting from the RIGHT</u>**) into sets of 3 to get octal, or sets of 4 to get hex:

$11001111000101100_2 =$

11 001 111 000 101 100 = 317054_8

or

1 1001 1110 0010 1100 = $19E2C_{16}$

Complete example:

Convert 3290_{10} to bases 2, 8, and 16:

Recommended method: Decimal → Octal → Binary → Hex

Decimal → Octal: **Div Mod**

3290 DIV 8 = 411
3290 MOD 8 = 2
411 DIV 8 = 51
411 MOD 8 = 3
51 DIV 8 = 6
51 MOD 8 = 3
6 DIV 8 = 0
6 MOD 8 = 6

Answer: 6 3 3 2_8

Octal → Binary: **3 bits**
6 3 3 2_8 → 110 011 011 010_2 Binary answer

Binary → Hex: **4 bits**
110011011010_2 → (Group in four bits, starting on right) 1100 1101 1010_2 → 12_{10} 13_{10} 10_{10} → CDA_{16}