Unsigned Conversions from Decimal or to Decimal and other Number Systems

In all digital design, analysis, troubleshooting, and repair you will be working with binary numbers (or base 2). It is sometimes necessary to convert from the binary number system to the decimal number system, because we are most familiar with and understand the decimal number system. After all, would you rather have someone give you $11110100_2$ dollars or $500$ dollars.

There are several methods for converting between number systems, and you will have to decide which method to use. For those who are having a little trouble with this, I would learn how to convert from binary to decimal and from decimal to binary very well. Converting between binary, octal, and hexadecimal are not very difficult because octal and hexadecimal are really just short-hand versions of binary. So, let's start with converting between binary and decimal using unsigned numbers:

**Decimal Numbers:**

Although we are very familiar with the decimal system, there are some facts that are worth reviewing - as a starting point:

The decimal (base 10) system has a total of 10 symbols representing the magnitude of a number. We know these symbols as the ten digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9. We need to express numbers that are much larger than 9, and so different "placements" of digits represent different values. We know the number 539 means that we have 5 hundreds, 3 tens, and 9 ones. We don't normally think about it, but in essence we are saying that the magnitude of this number is:

\[ 5 \times 10^2 + 3 \times 10^1 + 9 \times 10^0 = 500 + 30 + 9 = 539 \]

What this means is that the magnitude (or value) of this number is: \[ \text{symbol} \times \text{base}^n + \text{symbol(x-1)} \times \text{base}^{n-1} + \text{symbol(x-2)} \times \text{base}^{n-2} + \text{symbol(x-3)} \times \text{base}^{n-3} + \ldots + \text{symbol(0)} \times \text{base}^0 \]

where: \( \text{symbol} \) is the digit we use to express the magnitude of a number in the current system
\( \text{base} \) is the base of the current number system
\( n \) is the exponent of the base representing that place-holder
and \( x \) is the place-holder (we always start with the 1s place which is the base to the power of 0: \( X^0=1 \), or any number raised to the power of 0 is equal to 1)

We can use this method to convert from any other number system to decimal.

**Convert Binary to Decimal**

Using the method described above, you can convert from any number system to decimal. We need to know the base of the number system and the symbols used to represent digits. In converting binary to decimal, we know that the binary system uses a base of 2 and uses the symbols 0 and 1 (you cannot have more symbols than the base, and we always start with 0). In many texts you will see the word radix. Radix means the base of a number system (or logarithm). To convert from any other number system to decimal, use radix expansion (or base expansion) as shown below:

**Example: Convert Binary 110010111101_2 to Decimal**

**Solution:**
Convert by means of radix expansion. That is, the relationship between the decimal value and the binary can be determined by the following relationship:

\[
\begin{align*}
1 \times 2^{11} + 1 \times 2^{10} + 0 \times 2^9 + 0 \times 2^8 + 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\
= 2048 + 1024 + 0 + 0 + 128 + 0 + 32 + 16 + 8 + 4 + 0 + 1 \\
= 3261
\end{align*}
\]

**Convert Octal to Decimal**

Using the method described above, you can convert from octal (base 8) to decimal. The symbols used for the octal number system are: 0, 1, 2, 3, 4, 5, 6, and 7. The base of the octal number system is 8. To convert from octal to decimal, we will again use the radix expansion method:

**Example: Convert Octal 7241_8 to Decimal**

**Solution:** Convert by means of radix expansion.

\[
decimal = 7 \times 8^3 + 2 \times 8^2 + 4 \times 8^1 + 1 \times 8^0 \quad \text{replacing the values for the base to the exponents we have,}
\]

\[
\begin{align*}
&= 7 	imes 512 + 2 \times 64 + 4 \times 8 + 1 \\
&= 3584 + 128 + 32 + 1 \\
&= 3745
\end{align*}
\]

\[
\text{decimal} = 3745
\]
Convert Hexadecimal to Decimal

One method to convert from hexadecimal (base 16) to decimal is radix expansion. The symbols used for the hexadecimal number system are: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F. We are not used to seeing symbols we generally use for words to be used for numbers, but it is better than creating completely new symbols. The base of the hexadecimal number system is 16. To convert from hexadecimal (or hex for short) to decimal, we will again use the radix expansion method:

Example: Convert Hexadecimal A62E₁₆ to Decimal (sometimes you will see an "X" or an "h" used for a subscript to denote a hex number)

Solution: Convert by means of radix expansion.

\[
\text{decimal} = A \times 16^3 + 6 \times 16^2 + 2 \times 16^1 + E \times 16^0 \]

Replacing the values for the base to the exponents we have,

\[
\text{decimal} = 10 \times 4096 + 6 \times 256 + 2 \times 16 + 14 \times 1 \]

Now we need to replace those hex symbols for numbers larger than ten to their decimal equivalent

\[
\text{decimal} = 40960 + 1536 + 32 + 14 = 42542_{10}
\]

Converting from Decimal to Another Number System

Converting from decimal to another number system requires a series of divisions.

The algorithm to convert from decimal to another number system is:

1. Divide the decimal number by the radix (or base). The first pass through this algorithm, this will be the beginning or original number. For all other passes, this will be the whole number result of the division (the example below should make this statement clearer).

2. The remainder of the division becomes the next least significant digit (LSD) in the result.

3. Check to see if the whole number result of the division is 0. If it is, the algorithm is complete. If not, go back to step #1.

Example: Convert Decimal 312 to binary.

Solution:

Convert by means of successive division as illustrated below:
As the decimal number is divided by the radix, 2, the remainder is kept. Each remainder represents one digit of the final answer, starting at the least significant position. This same methodology can be used to convert from decimal to any other radix, for example, octal and hexadecimal.

Example: Convert Decimal 623 to Octal.

Solution:

Convert by means of successive division as illustrated below:

\[
\begin{align*}
\frac{623}{8} &= 77 \text{ remainder } 7 \\
\frac{77}{8} &= 9 \text{ remainder } 5 \\
\frac{9}{8} &= 1 \text{ remainder } 1 \\
\frac{1}{8} &= 0 \text{ remainder } 1
\end{align*}
\]

So, the result is \(1157_8\)

Example: Convert Decimal 44,094 to hexadecimal.

Solution:
Convert by means of successive division as illustrated below:

\[
\begin{align*}
\frac{44094}{16} &= 2755 \quad \text{remainder} = 14 = E \\
\frac{2755}{16} &= 172 \quad \text{remainder} = 3 \\
\frac{172}{16} &= 10 \quad \text{remainder} = 12 = C \\
\frac{10}{16} &= 0 \quad \text{remainder} = 10 = A
\end{align*}
\]

So, the result is AC3E

---

**Other Unsigned Conversions**

Because the octal and hexadecimal (and quartal - base 4, although that is not seen much these days) are just short-hand forms for binary, the conversions are not as complicated as they are for conversions between decimal and any other number system. To convert from binary to another system, group the binary digits (bits) into groups of \( \log_2(\text{radix}) \), and then replace the group of bits with its equivalent radix symbol. For example, to convert from binary to octal, group the bits into groups of 3 (\( \log_2(8) = 3 \)), and then replace the bits with the octal digits.

To convert from quartal, octal, or hexadecimal to binary, all you need to do is replace the digits with their equivalent group of binary digits.

**Convert Binary 110101111101\_2 to Octal**

**Solution:**

Combine the binary bits into groups of three from the binary point (far right for integers), then combine into octal characters. The equivalents table for octal to binary or binary to octal conversions are:

<table>
<thead>
<tr>
<th>Binary</th>
<th>Octal</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
</tr>
<tr>
<td>010</td>
<td>2</td>
</tr>
<tr>
<td>011</td>
<td>3</td>
</tr>
<tr>
<td>100</td>
<td>4</td>
</tr>
<tr>
<td>101</td>
<td>5</td>
</tr>
<tr>
<td>110</td>
<td>6</td>
</tr>
<tr>
<td>111</td>
<td>7</td>
</tr>
</tbody>
</table>

\[110\ 101\ 111\ 101\_2 = 6\ 5\ 7\ 5\_8\]

**Convert Binary 110101111101\_2 to Hexadecimal**

**Solution:**

Combine the binary bits into groups of four from the binary point (far right for integers), then combine into hexadecimal characters. The equivalents table for hex to binary or binary to hex conversions are:

<table>
<thead>
<tr>
<th>Binary</th>
<th>Hex</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>0</td>
</tr>
<tr>
<td>0001</td>
<td>1</td>
</tr>
<tr>
<td>0010</td>
<td>2</td>
</tr>
<tr>
<td>0011</td>
<td>3</td>
</tr>
<tr>
<td>0100</td>
<td>4</td>
</tr>
<tr>
<td>0101</td>
<td>5</td>
</tr>
<tr>
<td>0110</td>
<td>6</td>
</tr>
<tr>
<td>0111</td>
<td>7</td>
</tr>
<tr>
<td>1000</td>
<td>8</td>
</tr>
<tr>
<td>1001</td>
<td>9</td>
</tr>
<tr>
<td>1010</td>
<td>A</td>
</tr>
<tr>
<td>1011</td>
<td>B</td>
</tr>
<tr>
<td>1100</td>
<td>C</td>
</tr>
<tr>
<td>1101</td>
<td>D</td>
</tr>
<tr>
<td>1110</td>
<td>E</td>
</tr>
<tr>
<td>1111</td>
<td>F</td>
</tr>
</tbody>
</table>

\[1101\ 0111\ 1101\_2 = D\ 7\ D\_{16}\]

**Convert Hexadecimal CBD\_16 to Binary**

---
Solution:

It is probably simplest to memorize the binary-hex equivalents and then just replace one for the other. The equivalents table for hex to binary or binary to hex conversions are:

<table>
<thead>
<tr>
<th>Hex</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0000</td>
<td>0001</td>
<td>0010</td>
<td>0011</td>
<td>0100</td>
<td>0101</td>
<td>0110</td>
<td>0111</td>
<td>1000</td>
<td>1001</td>
<td>1010</td>
<td>1011</td>
</tr>
</tbody>
</table>

Hex = C B D

binary = 1100 1111 1111

Convert Octal 62578 to Binary

Solution:

Using the same method above, replace the octal digits by their binary equivalents. The equivalents table for octal to binary or binary to octal conversions are:

<table>
<thead>
<tr>
<th>Octal</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>000</td>
<td>001</td>
<td>010</td>
<td>011</td>
<td>100</td>
<td>101</td>
<td>110</td>
<td>111</td>
</tr>
</tbody>
</table>

Octal = 6 2 5 7

binary = 110 010 101 111

binary = 110010111112