THE BINARY NUMBER SYSTEM

Dr. Robert P. Webber, Longwood University

Our civilization uses the **base 10** or **decimal** place value system. Each digit in a number represents a power of 10. For example, 365.42 means

$$3*10^2 + 6*10^1 + 5*10^0 + 4*10^{-1} + 2*10^{-2}$$

In general, think of the place values as powers of 10.

$$\cdots \underset{10^3}{\_\_} \quad \underset{10^2}{\_\_} \quad \underset{10^1}{\_\_} \quad \underset{10^0}{\_\_} \,.\, \underset{10^{-1}}{\_\_} \quad \underset{10^{-2}}{\_\_} \cdots$$

There is nothing sacred about base 10. We could just as well use **base 2**, also called **binary**.

$$\cdots \underset{2^3}{\_\_} \quad \underset{2^2}{\_\_} \quad \underset{2^1}{\_\_} \quad \underset{2^0}{\_\_} \,.\, \underset{2^{-1}}{\_\_} \quad \underset{2^{-2}}{\_\_} \cdots$$

For example, 1101.1 is a binary number. It means

$$1*2^3 + 1*2^2 + 0*2^2 + 1*2^1 + 1*2^{-1} = 8 + 4 + 1 + ½ = 13 ½ \text{ (base 10)}$$

That is, $1101.1_2 = 13.5_{10}$, where the subscript indicates the base.

In general, to convert binary to decimal, just expand out using place values. Try converting 1011010_2 to base 10. Did you get 90?

To convert a base 10 integer to binary, think of filling in blanks in the binary place value chart, starting as far left as possible. That is, pull out the largest power of 2 possible from the integer. Subtract it from the original number, and repeat with the remainder. Continue until the remainder is 0 or 1.

Example: Convert 85_{10} to binary.

Solution: 64 (= 2^6) is the largest power of 2 that is less than or equal to 85. Put 1 in the 64's place and subtract it off.

$$\underset{2^6}{\underline{1}} \quad \underset{2^5}{\_\_} \quad \underset{2^4}{\_\_} \quad \underset{2^3}{\_\_} \quad \underset{2^2}{\_\_} \quad \underset{2^1}{\_\_} \quad \underset{2^0}{\_\_} \qquad\qquad 85 - 64 = 21$$

Repeat, this time using 21. The largest power of 2 less than or equal to 21 is 16 ($=2^3$). Put 1 in the 16's position and subtract it off. Note that this also requires us to put 0 in the 32's position!

1	0	1				
2^6	2^5	2^4	2^3	2^2	2^1	2^0

$21 - 16 = 5$

Continuing, $5 - 4 = 1$. Put 1 in the 4's position (and 0 in the 8's position).

1	0	1	0	1		
2^6	2^5	2^4	2^3	2^2	2^1	2^0

$5 - 4 = 1$

Since the remainder is 1, put 1 in the 1's position (and 0 in the 2's position).

1	0	1	0	1	0	1
2^6	2^5	2^4	2^3	2^2	2^1	2^0

That is, $85_{10} = 1010101_2$.

Try counting in binary, starting with 0. Do you get 0, 1, 10, 11, 100, 101, 110, 111, 1000, …?

To convert a base 10 fraction to binary, assume that the denominator is a power of 2. Other fractions are beyond the scope of these notes.

With that assumption, write the integer numerator in binary, write the denominator as a power of 2, and shift the binary point left the number of places indicated by the exponent.

Example: Convert $\frac{5}{8}$ to binary.

Solution: First, $5_{10} = 4 + 1 = 101_2$. Since $8 = 2^3$, shift the binary point three places to the left. $\frac{5}{8} = \frac{5}{2^3} = .101_2$

Convert a mixed base 10 number to base 2 by writing the mixed number as an improper fraction and proceeding as above.

Example: Convert 17 ¼ to base 2.

Solution: 17 ¼ = 69/4 . Now $69 = 64 + 4 + 1$, so $69_{10} = 1000101_2$. Next, since $4 = 2^2$, move the binary point two places to the left. Thus 17 ¼ = 10001.01_2 .

BASE 16 (HEXADECIMAL) NUMBERS

Just as we use base 10 and base 2 place values, we could use other bases. In particular, **base 16**, also called **hexadecimal**, is used frequently in computer science.

$$\cdots \underset{16^3}{\_\_} \quad \underset{16^2}{\_\_} \quad \underset{16^1}{\_\_} \quad \underset{16^0}{\_\_} \;.\; \underset{16^{-1}}{\_\_} \quad \underset{16^{-2}}{\_\_} \cdots$$

Since $16^2 = 256$, $16^3 = 4096$, and so on, hexadecimal place values increase rapidly!

Convert between base 10 and base 16 as we did earlier with binary.

Example: Convert 158_{16} to decimal.

Solution: $158_{16} = 1*16^2 + 5*16 + 8$

$= 256 + 80 + 8$

$= 344_{10}$

Example: Convert 643_{10} to base 16.

Solution: $643_{10} = 2*256 + 131$

$= 2*256 + 8*16 + 3$

$= 2*16^2 + 8*16^1 + 3*16^0 = 283_{16}$

A problem arises when we try to count in hexadecimal. What comes after 9?

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ?

What should be the tenth positive integer? It cannot be written as 10, because

$10_{16} = 1*16 + 0 = 16_{10}$

We must have another symbol for the tenth digit. A little thought should convince you that we need symbols for the eleventh through the fifteenth digit, too. This will give 16 digits in all, including 0. As a general principle, base *n* requires *n* digits, including 0. Each digit must be a single symbol. The common agreement in mathematics is to use the letters *A, B, C, D, E,* and *F* as the new symbols.

Base 2 digits: 0, 1 (A **bit** is a binary digit)

Base 10 digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Base 8 digits: 0, 1, 2, 3, 4, 5, 6, 7

Base 16 digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, *A*, *B*, *C*, *D*, *E*, *F*

Example: Convert $10A3_{16}$ to base 10.

Solution: $10A3_{16} = 1*16^3 + 0*16^2 + A*16^1 + 3*16^0$

$= 1*4096 + 10*16 + 3*1$

$= 4096 + 160 + 3 = 4259_{10}$

Use the extended set of digits as needed to convert decimal to base 16.

Example: Convert 445_{10} to hexadecimal.

Solution: $445_{10} = 1*256 + 189$

$= 1*256 + 11*16 + 13$

$= 1*16^2 + B*16^1 + D*16^0 = 1BD_{16}$

Converting between base 16 and base 2 is particularly useful. You can do this by going through base 10, of course, but it is much quicker to convert directly. Since $16 = 2^4$, one hexadecimal digit is the equivalent of four binary digits. To convert from base 2 to base 16, just group the base 2 bits by fours, starting on the right. Convert each group of four bits to its equivalent digit in base 16.

Example: Convert 1011011001_2 to hexadecimal.

Solution: Group the bits by fours, starting on the right. For emphasis, write two leading zeros to round out the first group of bits.

1011011001 = 0010 1101 1001

Now $0010_2 = 2_{16}$, $1101_2 = 13_{10} = D_{16}$, and $1001_2 = 9_{16}$, so $1011011001_2 = 2D9_{16}$.

Example: Convert $B605_{16}$ to base 2.

Solution: Expand each hexadecimal digit to its four-bit binary form.

$B_{16} = 11_{10} = 1011_2$, $6_{16} = 0110_2$, $0_{16} = 0000_2$, $5_{16} = 0101_2$.

Therefore, $B605_{16} = 1011011000000101_2$

ASCII

ASCII (**A**merican **S**tandard **C**ode for **I**nformation **I**nterchange) is a code used to represent characters inside a computer. A **character** is any single keystroke on the keyboard. Characters are usually indicated by including them in single quotes. For example, 'A' and 'a' are characters. So are '!", '.', and '6'.

A **byte** is the unit of storage for a character. In computers, 1 byte = 8 bits, so ASCII is a 1 byte, or 8 bit, code. There is a separate eight bit combination of 0's and 1's for every individual character on a computer keyboard. (As an historical note, ASCII was originally defined as a seven bit code. The left-most bit is not used in true ASCII, and we will always write it as 0.) Many computer books have tables of ASCII values, and you can find them in lots of locations on the Web. Here is the address of a helpful one. It gives the codes in their base 10 (dec), base 16 (hx), and base 8 (oct) forms.

http://www.lookuptables.com/

For example, the string *hi* in ASCII would be $68\ 69_{16}$, or 0110100 01101001. Note that *Hi* would be $48\ 69_{16}$, or 01001000 01101001. Upper case letters have different codes from lower case letters.

As you can see, the binary ASCII code representation of a string of characters can get very long and difficult to read. Consequently, the codes are often shown in their hexadecimal or decimal forms.

Example: Decode the message 43 4*D* 53 43 20 31 32 31, which is written in hexadecimal ASCII.

Solution: Look up the codes in the ASCII table to find the message *CMSC* 121 . Notice that the space character has its own ASCII code.

Computer documentation of such things as memory addresses and the contents of memory locations are often given in hexadecimal ASCII. Here is a way to actually see some ASCII code. On a PC, type and save a short file using a text editor such as Notepad. Give it a short name such as `temp`, and do not use any extension. It does not matter what you put in the file. Next, open a command window and open the file you saved using the command `DEBUG` followed by the file name. Then give a `-d` command (for display). The contents of your file will be shown in ASCII using hexadecimal notation. To exit DEBUG and return to command level, enter the command `–q` (for quit).

EXERCISES

1. Convert the binary numbers to decimal.

   a. 10110
   b. 11100111
   c. 101.011

2. Convert the decimal numbers to binary.

   a. 86
   b. 131
   c. $14\frac{5}{8}$

3. Convert the base 10 numbers to hexadecimal.

   a. 68
   b. 543
   c. 127

4. Convert the hexadecimal numbers to decimal.

   a. 10D
   b. 345
   c. BABE

5. Convert the base 16 numbers to binary.

   a. 53
   b. 94B0
   c. 3ED

6. Convert the binary numbers to hexadecimal.

   a. 1010 0101
   b. 0011 0000 1101 1111
   c. 1001 0111 0110 1000

7. Convert the base 10 number 36 to

   a. binary
   b. hexadecimal

8. Convert the binary number 1000 1010 to

a. hexadecimal
b. decimal

9. Convert the base 16 number 3C5 to

a. binary
b. base 10

10. Convert the base 10 number 130.4375 to binary.

11. Convert the binary number 1001101.1001 to decimal.

12. Convert the hexadecimal number A2.4 to base 10.

13. Convert the English expression ***Memorial day 2007*** to ASCII. Show the code in hexadecimal form.

14. Convert the ASCII code (shown in hexadecimal) to English.

43 4D 53 43 20 32 30 34-2E 42 30 33 20 53 75 6D 6D 65 72 20 31