APPENDIX A

Binary Number System—The Language of Computers

This appendix is for those readers who are interested in understanding how 1s and 0s can translate into something easier to understand.

Binary System (Converting Binary to Decimal)

The numbering system that computers use is called the *binary system* (also referred to as *base 2*). The system may seem awkward at first glance, but it uses the exact same logic as the base 10 (or decimal system) you use every day. You probably have not thought about this in a while, but think back to the days when you first learned to count and do basic math. Typically, you were taught to start with the "ones" column and count until the highest unit is reached, and then move over to the "tens" column. This continues on an on with successive powers (1, 10, 100, 100, etc.).

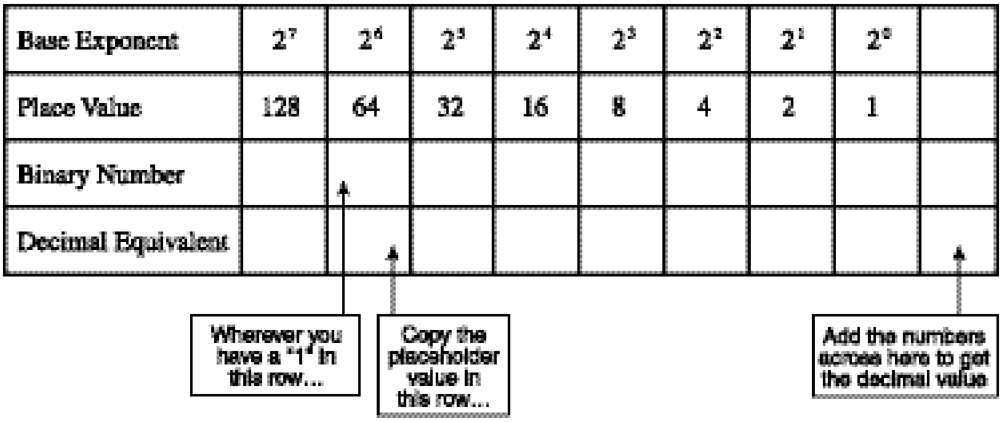
The binary system uses the same mechanics, it just has fewer digits to work with. With the binary system, the columns or placeholders are 1, 2, 4, 8, etc.). Table A-1 shows how the numbers 0–4 are written in binary and decimal form.

Table A-1 Binary to Decimal Equivalent

Binary	Decimal	Notes
0000	0	0 is the same in both systems.
0001	1	1 is the same in both systems.
0010	2	What we do with two in binary is equivalent to what happens with a ten in decimal. We have run out of places in the first column so we put a one in the next highest column—i.e., the twos column. 0010 = 1 * 2 + 0 * 1 = 2
0011	3	Now we have a 1 in the "twos" column and a 1 in the "ones" column, which gives us a 3 (2 + 1). 0011 = 1 * 2 + 1 * 1 = 3
0100	4	Again, when we add 1 to 3, that's adding binary 0001 to 0011. We run out of places in both the "ones" and "twos" columns, so we have to carry the 1 to the "fours" column. 0100 = 1 * 4 + 0 * 2 + 0 * 1 = 4

OK, this looks nasty at first, but it will start to make sense as you read on. If you ever want to do the conversion yourself, just make a table like the one shown in Figure A-1.

Figure A-1 Binary Conversion Table



Base Exponent	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Place Value	128	64	32	16	8	4	2	1	
Binary Number									
Decimal Equivalent									

For example, if you want to convert the binary number 10100111, simply drop it in the table. If the number is larger than 8 digits, just add exponents to the left. For this example, the table looks like Table A-2.

Table A-2 Binary Conversion Table

Base Exponent	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Place Value	128	64	32	16	8	4	2	1	
Binary Number	1	0	1	0	0	1	1	1	
Decimal Equivalent	128	0	32	0	0	4	2	1	128+32+4+2+1 = 167

NOTE The same table can also be use to perform the reverse operation (going from decimal to binary). In this case, find the largest place value that is less than the decimal number (in the previous example 128 is the largest place value number that is less than 167). Place a 1 in the binary number slot below the placeholder (and 0s in the spaces to the left), subtract the placeholder from the original number (167–128 = 39) and repeat the process until your last subtraction equals 0.

Hexadecimal System

Another common numbering system used in computers is *hexadecimal* (also called *base 16*). Hexa-decimal has lots of applications in computers, including MAC addresses and security keys (like WEP). Hexadecimal is interesting from the standpoint that we only have 10 single digit numbers (0–9) but we need 16 numbers for hexadecimal. The solution is to use letters A through F to represent the values 11 to 15. Table A-3 shows binary, decimal, and hexadecimal numbers.

Table A-3 Decimal and Hexadecimal Equivalents

Binary	Hex	Decimal
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

Using hexadecimal, a very large binary string of 1s and 0s can be represented with just a few hexadecimal numbers by breaking the binary number into groups of four and then using the hexadecimal equivalent; for example, 1101100101001111 can be written as 1101 1001 0100 1111 which, from Table A-3, equals D94F in hexadecimal (or 55,631 in decimal).

How to Tell Whether a Number Is Binary, Decimal, or Hexadecimal

Because these three number systems share digits—for example, all have 0 and 1—how can you tell which system is being used? For example, take the number 101. This could be any of the following:

Decimal 101 = 101

Binary 101 = 5

Hexadecimal 101 = 257

Obviously, this could lead to some confusion. In general, there are a few hints to look for:

- If a number contains only 1s and 0s, it is likely binary, but could still be decimal or hexadecimal.
- If a number contains decimal digits 0 through 9, it is most certainly not binary, probably decimal, but still could be hexadecimal.
- If a number contains decimal digits 0 through 9, and also letters A through F, it is not binary, not decimal, and very likely hexadecimal.
- More precise writers will make sure to include a qualifier that identifies which number system is in use, for example: 0x101.

Here, the 0x preceding the number tells us that hexadecimal is being represented.