

WHAT IS ASCII AND HOW IS IT USED TO CONVERT NUMBERS AND LETTERS INTO BINARY FORM?

All electronic communications involving computers are expressed in binary language which involves only two symbols. These are 0 for off and 1 for on. Since normal communications between humans over large distances is both in letters and numbers it earlier became necessary to come up with a coding method capable of handling both cases. The result was a coding system developed in this country in the early 1960s now known as ASCII(American Standard Code for Information Interchange). We want here to show how this code works.

Our starting point is to write down the first few integers and underneath them give their binary form. Also we add a third row giving the 26 letters of the alphabet written in sequential order. We summarize this information in the following three tables -

1	2	3	4	5	6	7	8	9	10
0	10	11	100	101	110	111	1000	1001	1010
A	B	C	D	E	F	G	H	I	J

11	12	13	14	15	16	17	18	19	20
1011	1100	1101	1110	1111	10000	10001	10010	10011	10100
K	L	M	N	O	P	Q	R	S	T

21	22	23	24	25	26
10101	10111	11000	110001	110010	110111
U	V	W	X	Y	Z

Now the major idea behind ASCII is that we can assign a number for each letter of the alphabet and then convert it to binary form. The difficulty, as seen from the above tables, is that one can not yet distinguish between a binary expression being a number or letter. Certainly, as things stand, 110001 could refer to either the number 24 or the letter X. Likewise 1000 could be 8 or H. To remedy this difficulty ASCII lengthens the number of bits for each element to eight and then replaces the first three digits on the left by 010 to indicate a letter. Thus the letter M in binary reads 01001101 and S in binary is 01010011. A number has the first four symbols on the left replaced by 0011. That is, the number 2 is written as 00110010. Using eight elements to represent a letter or number allows extra space for punctuation marks, small letters, and mathematical operation signs. The basic idea behind the ASCII coding procedure is that each symbol has a specific binary number attached to it. The original ASCII coding of 1963 involved 128 characters . Later extensions added several hundred more. If you convert the binary 01010011 into decimal, one gets the ASCII number of 83. Thus S in ASCII is 83 which converts to 01010011 in binary. There are available on the internet numerous conversion tables. One of the best is found at-

<http://www.rapidtables.com/convert/number/ascii-hex-bin-dec-converter.htm>

This link allows rapid conversion from ASCII to binary to decimal . One recovers the ASCII symbol by converting the binary. Thus the binary representation 00110111 yields 55 in ASCII and corresponds to the decimal 7. We could of course have anticipated this result by noting from the above tables that the ending 111 in binary corresponds to the decimal number 7. Had one changed the fourth symbol on the left in the binary expansion to 0 , the expansion would correspond to the letter G. Whole phrases can be readily converted into binary form by using the conversion table. So, for instance, the distress signal SOS would read-

01010011 01001111 01010011

in binary. Although this chain is quite a bit longer than the Morse Code signal-

000 111 000

it is far superior in the sense that eight bit representations allow for the inclusion of numerous additional symbols . A few of these additional symbols, expressed in binary, are-

00101011=+ 00101101=- 00101111=/ 00101100=, 00101110=.
00111011=; 00111110=> 00111100=< 00100011=# 01111110=~
00101000=(00101001=) 00100001=! 00100110=& 00101010=*

Also a space corresponds to 00100000 .

Consider another letter phrase:

TO BE OR NOT TO BE THAT IS THE QUESTION

In binary using ASCII it reads-

01010100 01001111 00100000 01000010 01000101 00100000 01001111
01010010 00100000 01001110 01001111 01010100 00100000 01010100
01001111 00100000 01000010 01000101 00100000 01010100 01001000
01000001 01010100 00100000 01001001 01010011 00100000 01010100
01001000 01000101 00100000 01010001 01010101 01000101 01010011
01010100 01001001 01001111 01001110

You will recognize at once that the grouping of the first eight bits indicates

a letter (010) and from the above tables its extension of 10100 is found in the 20-T column. Thus the letter is T.

In cryptography one would disguise this batch of elements by multiplying things by a large random number before sending the encrypted message out to a receiver(and anyone else who may be listening). Only persons familiar with the random number being used will be able to decipher the message by dividing the encrypted message by the random number being used. As we will show in an upcoming article, even certain semi-random numbers may be used for an effective encryption. For most cryptography applications it is sufficient to deal only with the thirty six ASCII symbols representing the numbers 0 through 9 and the capital letters A through Z. A table giving all 36 groups follows-

ASCII CONVERSION TABLE , DECIMAL TO BINARY

00110000	0	01000001	A	01001110	N
00110001	1	01000010	B	01001111	O
00110010	2	01000011	C	01010000	P
00110011	3	01000100	D	01010001	Q
00110100	4	01000101	E	01010010	R
00110101	5	01000110	F	01010011	S
00110110	6	01000111	G	01010100	T
00110111	7	01001000	H	01010101	U
00111000	8	01001001	I	01010110	V
00111001	9	01001010	J	01010111	W
		01001011	K	01011000	X
		01001100	L	01011001	Y
		01001101	M	01011010	Z

ASCII Number is recoverable by converting the binary group to decimal. So 01001010 becomes 74 and represents J

U.H.Kurzweg
October 8, 2017

