Introduction to COMPUTER CODES

The foundation for programming computers—binary, octal, hexadecimal, and BCD codes.

BY R. T. ROBBINS

COMPUTERS and other digital equipment communicate in "languages" based on 1's and 0's that must be mastered if any meaningful communication between human and machine is to be accomplished. There are four number systems in common use in electronics today. For human-to-human communication, we use the 0 through 9 "decimal," or base-10 (n_{10}), number system. The other three are the digital 1/0-based systems, which include: binary (base-2, or n_2), octal (base-8, or n_8), and hexadecimal (base-16, or n_{16}).

Converting from one to any other numbering system is a relatively simple matter. Let us review how these systems compare with each other and interact.

The Binary System. The binary code is the basic digital machine numbering system. In essence, it is a simple 1/0 representation of the decimal system. Its great disadvantage, and the reason it is rarely used in computers, is that binary equivalents of decimal numbers become increasingly ponderous with each succeeding number

To illustrate how to use the binary numbering system, let us convert 69_{10}, 230_{10}, 1976_{10}, and 52801_{10} to binary. (Note: It is important to use subscripts to identify the numbering system when more than one system appears in your notes. The numbering system subscripts are detailed above.) The four numbers we have selected at random will illustrate the drawback of the binary system.

The binary system is a simple 1/0 format. Going from right to left, as each position is filled by a 1 and another 1 is added to it, the result is 0 and a 1 is carried to the next place to the left. A table of binary equivalents for the decimal numbers 0 through 15 would appear as follows:

Decimal	Binary
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

Note the logical progression of the numbers in binary and the increasing number of digits used as you count up. Note also that the points at which you add digits to the left occur at the power-of-two (2^n) points—2,4,8,16,32, etc. And one final note: 0 is always considered a number possibility in all digital machine languages.

Now we can make some step-by-step conversions. The first step is to list across the page the decimal equivalent of each power of two that will be required for making the conversion, as shown in Fig. 1. Each location will

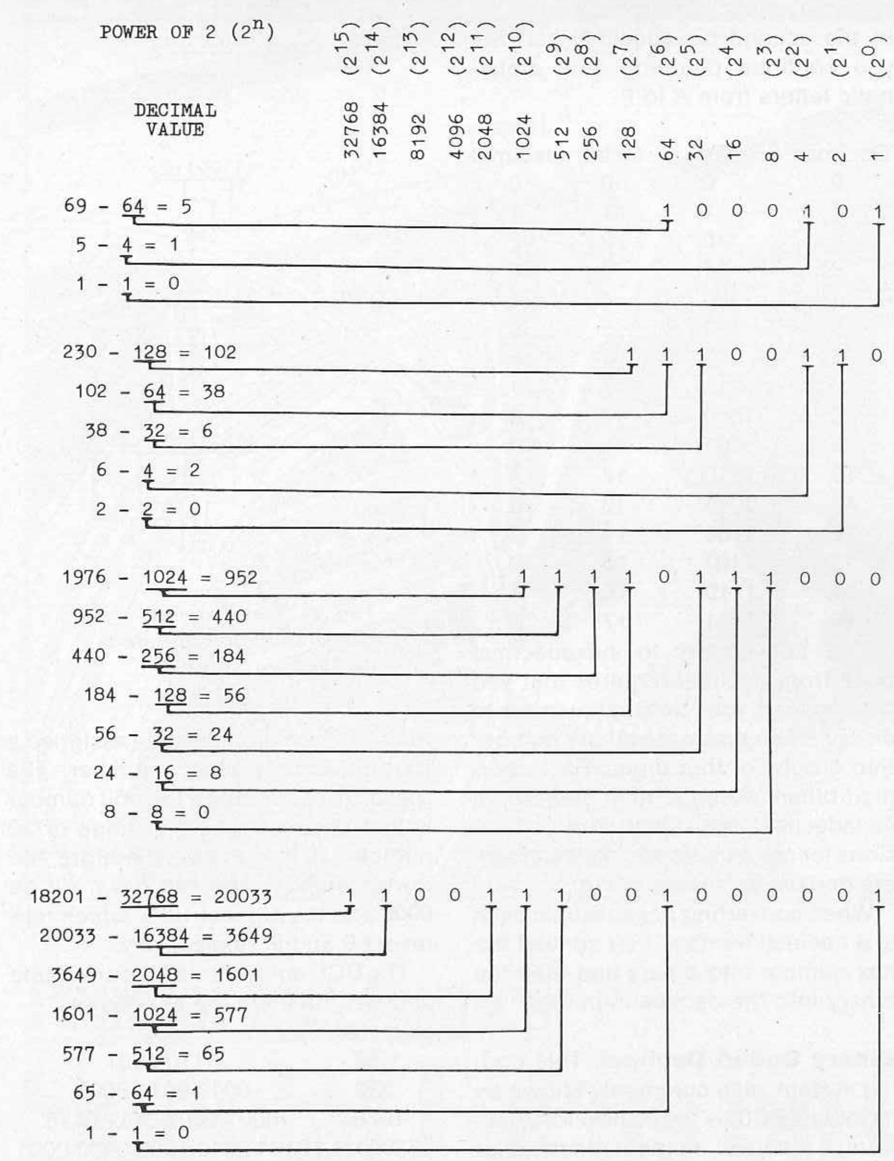


Fig. 1. Converting decimal numbers to equivalents in binary code. The final binary conversions are at the right.

$69_{10} = 1000101_2$

$230_{10} = 11100110_2$

$1976_{10} = 11110111000_2$

$52801_{10} = 1100111001000001_2$

be filled in with a 1 or a 0 (binary "bit"), depending on whether or not its value is used during computations. If the value is used, you enter a 1; otherwise, you enter a 0. (Leading 0's need not be entered.)

Start your conversion by placing a 1 under the largest decimal number that doesn't exceed the number you're converting. Next, subtract the number under which you placed a 1 from the number you're converting to determine which is the next number under which you must place a 1. Always keep in mind that you must never have a negative result. Only positive results or 0 after subtraction are acceptable. If you must skip numbers to enter a 1, fill in the blank spaces with 0's. Continue subtracting numbers until you've filled each column and your final result equals exactly zero. If, after obtaining a 0 result, you still have empty spaces to the right, fill in 0's. To double-check your conversion, simply add all the power-of-2 numbers under which you've entered 1's and check that the result is the decimal number with which you started. (Note: This is the procedure to use when converting *any* binary number back to its decimal equivalent.)

The final conversions to binary for our sample decimal numbers are shown at the bottom of Fig. 1. Note how many more digits must be filled in as the numbers increase in magnitude.

Once you understand the binary code, the octal and hexadecimal codes are much easier to understand and learn.

The Octal Code. Much of the ponderousness of the binary code can be eliminated by switching to the more efficient octal code. An octal number is only one-third as long as its binary equivalent. Once a number has been converted from decimal to binary, separate it into groups of three digits, working from right to left.

In any group of three digits, the minimum will be 000 and the maximum will be 111. Now, to get the equivalent octal number, simply translate the three binary numbers back to decimal, which means the smallest number you will have will be 0 (000) and the largest will be 4+2+1=7 (111). Bearing in mind that 0 is always a number possibility, you now have a base-8 number system:

Decimal	Binary	Octal
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	10
9	1001	11
10	1010	12
11	1011	13
12	1100	14
13	1101	15
14	1110	16
15	1111	17

When converting from binary to octal, you may end up with only one or two digits in the left-end group. To handle this situation, simply assume that the missing digits are 0's.

The octal equivalents of the sample decimal numbers chosen above are shown in Fig. 2.

The Hexadecimal Code. Greatest efficiency is obtained when the hexadecimal code is used to format decimal numbers. Here, the binary number is separated into four-digit slices (right to left). Now, the maximum number capable of representation is 1111_2, or $8+4+2+1=15_{10}$. You still count up from 0 in the usual decimal manner, but only to 9. This still leaves six numbers to fill in to reach the 16 maximum. However, you cannot enter 10 as the next number because that is the first numeral in the second hexadecimal set. So, you continue counting with alphabetic letters from A to F:

Decimal	Binary	Octal	Hexa-decimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

The conversion to hexadecimal code from decimal requires that you first convert your decimal number to binary. Then break the binary number into groups of four digits. Each four-digit binary group is then assigned a hexadecimal equivalent. The conversions for our sample decimal numbers are derived as shown in Fig. 3.

When converting hexadecimal back to a decimal number, first convert the hex number into binary and then the binary into the decimal number.

Binary Coded Decimal. This coding system, also commonly known by its initials BCD, is the easiest to obtain from a decimal number. Instead of considering the decimal number as a whole, treat each digit in the number as a separate entity when converting

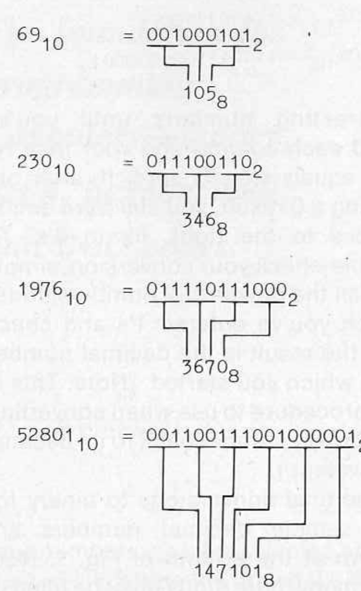


Fig. 2. Octal equivalents of four decimal numbers.

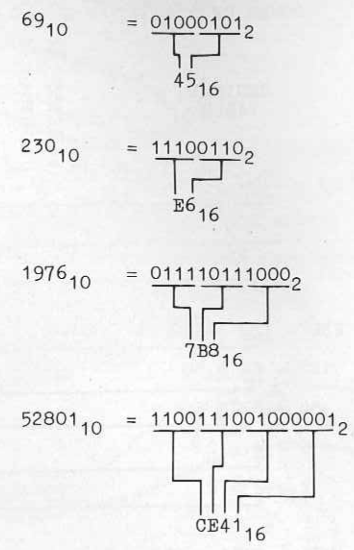


Fig. 3. Hexadecimal conversions for sample decimal numbers.

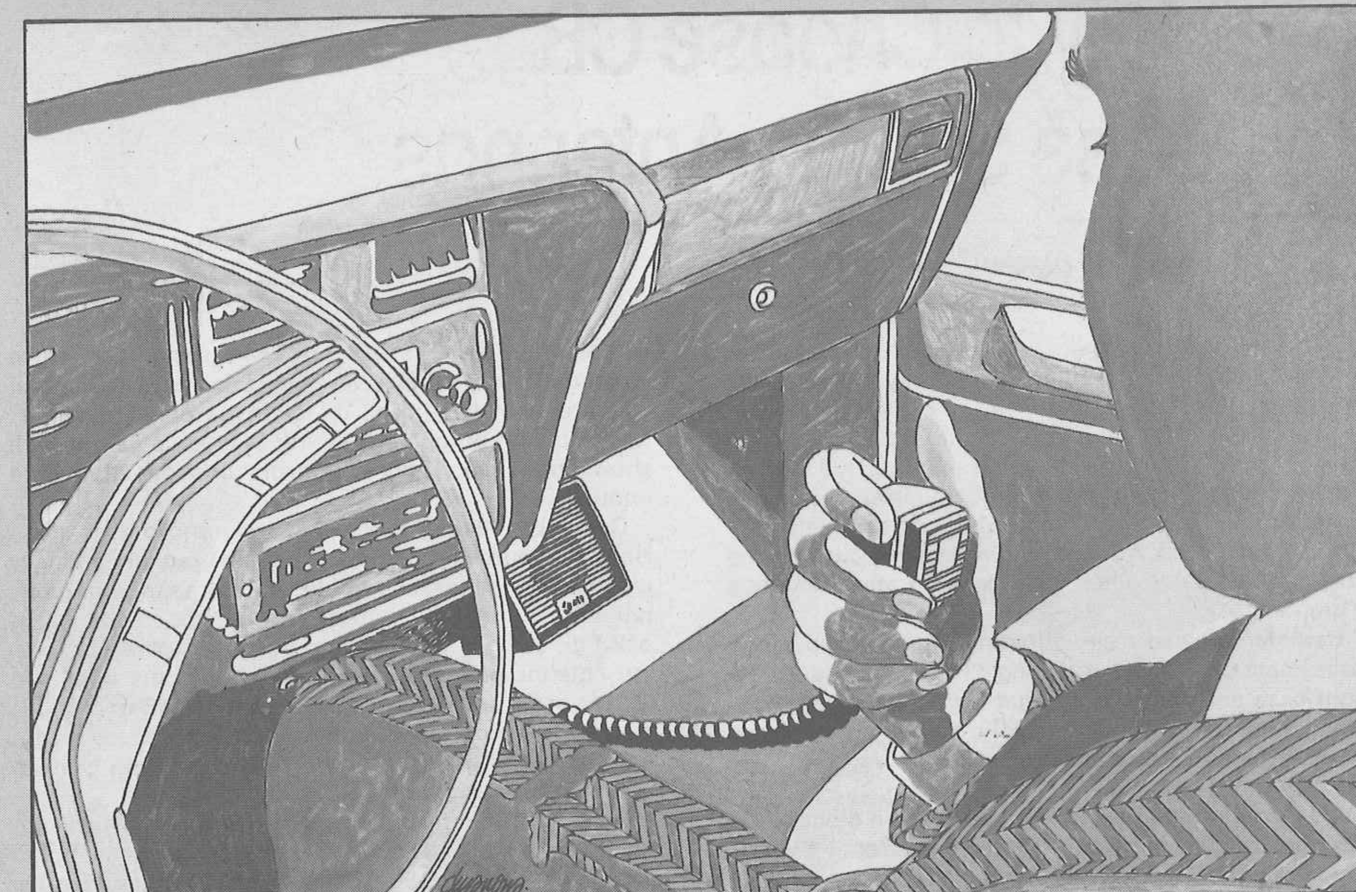
to BCD. Each digit must be assigned a four-bit, strictly binary number. The reason for assigning a four-bit number is that decimal covers a range of 10 numbers (0 through 9). Therefore, the lowest number you can have will be 0000 and the highest 1001, which represent 0 and 9, respectively.

The BCD equivalents for our sample decimal numbers are as follows:

Decimal	BCD
69	0110 1001
230	0010 0011 0000
1976	0001 1001 0111 0110
52801	0101 0010 1000 0000 0001

Note that when a BCD number is written, an obvious space must be left between each four-bit binary slice for identification purposes. BCD is the only numbering system that should have such spaces. Therefore, It doesn't need a subscript, except in the special case where only a single decimal digit is to be converted, in which case you can use a BCD subscript.

Summing Up. We have discussed four basic numbering systems used in digital equipment. Of these, hexadecimal is the most efficient but has the disadvantage of requiring both numeral and alphabet symbols. Slightly less efficient, the octal system has the advantage of employing only easy-to-understand decimal numbers. The least efficient numbering systems are binary and BCD. BCD is by far the least efficient numbering system, requiring five times the number of spaces (including the required blank space) of its equivalent decimal number. ◈



Popular Electronics®
Special Focus on CB RADIO

MORE and more people are turning to Citizens Band Radio to satisfy their two-way radio communication needs. They might be motivated by a desire to have an emergency line in an automobile, check on traffic conditions by listening to trucker's conversations (now mostly on Channel 19 rather than Channel 10), participate in public service activities, develop new friends, enjoy general conversations when alone in the home, contact the business office while driving, and so on.

Coupled with the no-test requirement and relatively low equipment cost, it's not surprising that the growth of CB Radio (Class D) has been astonishingly great. As an example, the last report received from the FCC indicated that almost 300,000 new CB license applications were received in the month of November 1975! And the FCC projects that there will be about 15-million CB'ers in the U.S.A. by 1979.

This certainly has been a good year for CB. Operating restrictions were relaxed (including removal of the "hobby" prohibition), license fees were reduced from $20 to $4 for five years, and interstation channels were increased, among other benefits. By relaxing CB rules, however, the FCC expects full compliance with present restrictions, which include use of one's Call Letters, transmission-time limits, no obscene language, etc. It's to every CB'er's advantage to press for compliance with the rules. Remember, what the FCC gives it can take away!

This special 16-page insert examines a variety of CB areas: (1) Base-station CB antennas, which can significantly increase communication range, are given authoritative treatment. (2) A pictorial presentation shows how easy it really is to install a mobile CB system. (3) A CB highway assistance program in Washington, D.C. underlines the important public service performed by many CB'ers. (4) An easy-to-build communication timer simplifies complying with the new FCC ruling on communication time. (5) A glossary of CB equipment terms will enable buyers to understand specification and feature claims in product literature. (6) Finally, a list of CB radio equipment and antenna manufacturers and addresses for reference purposes.