

BINARY CODED SYSTEM:

In general, coding is the process of assigning a group of binary digits to represent multivalued items of information. Binary coded decimal (BCD) is a system of writing numerals that assigns a four-digit binary code to each digit 0 through 9 in a decimal (base- 10) numeral. The four-bit BCD code for any particular single base-10 digit is its representation in binary notation, as follows:

0	1	2	3	4	5	6	7	8	9
0000	0001	0010	0011	0100	0101	0110	0111	1000	1001
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓

Numbers larger than 9, having two or more digits in the decimal system, are expressed digit by digit, For example, the BCD interpretation of the base-10 number 1895 is

1 8 9 5 ie., 1895 = 0001 1000 1001 0101
0001 1000 1001 0101

The binary equivalents of 1, 8, 9, and 5, always in a four-digit format, go from left to right.

Binary codes are broadly classified into Numeric codes, Alphanumeric codes and Error detecting codes. Numeric codes are further classified into weighted codes and non-weighted codes. The most obvious way of encoding digits is "natural BCD" (NBCD), where each decimal digit is represented by its corresponding four-bit binary value. This is also called "8421" encoding. Standard binary coded decimal code is commonly known as a weighted 8421 BCD code, with 8, 4, 2 and 1 representing the weights of the different bits starting from the most significant bit (MSB) and proceeding towards the least significant bit (LSB). The weights of the individual positions of the bits of a BCD code are: $2^3 = 8$, $2^2 = 4$, $2^1 = 2$, $2^0 = 1$.

The main advantage of the Binary Coded Decimal system is that it is a fast and efficient system to convert the decimal numbers into binary numbers as compared to the pure binary system. However, the disadvantage is that BCD code is inefficient as the states between 1010 (decimal 10), and 1111 (decimal 15) are not used.

In non-weighted code, there is no positional weight i.e. each position within the binary number is not assigned a prefixed value. No specific weights are assigned to bit position in non-weighted code. The non-weighted codes are classified to

- a) The Excess-3 code
- b) The Gray code

EXCESS-3 CODE:

Excess-3 code is an important BCD code, is a 4 bit code and used with BCD numbers as weights are not assigned, it is a kind of non- weighted codes. Excess-3 code was used on some older computers, cash registers and hand held portable electronic calculators. The Excess-3 code for a given decimal number is determined by adding '3' to each decimal digit in the given number and then replacing each digit of the newly found decimal number by its four bit binary equivalent. The table gives the Excess-3 code.

Decimal	8421	Excess-3
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

Decimal to Excess-3 code:

Excess-3 code of 24 is obtained as

2 4

+3 +3

5 7

0101 0111

Thus, Excess-3 code of 24 is 0101 0111.

Similarly, Excess-3 code for $(597)_{10}$ and $(14.57)_{10}$ is

$(597)_{10} = (100011001010)$

$(14.57)_{10} = (01000111.10001010)$

1.21.1 Excess-3 to Decimal:

From given Excess-3 code, the equivalent decimal number can be determined by first splitting the number into four-bit groups, starting from radix point and then subtracting 0011 from each four-bit group. This gives us 8421 BCD equivalent of the given Excess-3 code, which can then be converted into the equivalent decimal number.

Example:

Determine the decimal equivalent for the Excess-3 code 1000110.

0100 0110

Then Subtracting 0011 from each group,

$$\begin{array}{r} 0100 \\ - 0011 \\ \hline 0001 \end{array} \quad \begin{array}{r} 0110 \\ - 0011 \\ \hline 0011 \end{array}$$

The new number as 00010011. Its decimal equivalent is 13.

GRAY CODE

The disadvantage of the previous codes is that several bits change state between adjacent counts. The Gray code is unique in that successive counts result in only one bit change. For example, 7 (0111) to 8 (1000) in binary, or BCD, all four bits change state. In Gray code, however, 7 (0100) to 8 (1100) require a single bit change.

The switching noise generated by the associated circuits may be intolerable in some environments. The same change with Gray code undergoes only a single bit change consequently, less noise is generated. Shaft encoders used for receiver tuning dials often use Gray code.

The Gray code is widely used for encoding the position of the rotary shaft and for data transmission using PSK.

The Gray code was designed by Frank Gray at Bell Labs and patented in 1953. It is an unweighted binary code in which two successive values differ only by 1 bit. Owing to this feature, the maximum error that can creep into a system using the binary Gray code to encode data is much less than the worst-case error encountered in the case of straight binary encoding. Table 2.3 lists the binary and Gray code equivalents of decimal numbers 0–15. An examination of the four-bit Gray code numbers, as listed in Table 2.3, shows that the last entry rolls over to the first entry. That is, the last and the first entry also differ by only 1 bit. This is known as the *cyclic property* of the Gray code. Although there can be more than one Gray code for a given word length, the term was first applied to a specific binary code for non-negative integers and called the *binary-reflected Gray code* or simply the Gray code.

There are various ways by which Gray codes with a given number of bits can be remembered. One such way is to remember that the least significant bit follows a repetitive pattern of ‘2’ (11, 00, 11,), the next higher adjacent bit follows a pattern of ‘4’ (1111, 0000, 1111,) and so on. We can also generate the n-bit Gray code recursively by prefixing a ‘0’ to the Gray code for n–1 bits to obtain the first 2^{n-1} numbers, and then prefixing ‘1’ to the reflected Gray code for n–1 bits to obtain the remaining 2^{n-1} numbers. The reflected Gray code is nothing but the code written in reverse order. The process of generation of higher-bit Gray codes using the reflect and- prefix method is illustrated in Table 2.4. The columns of bits between those representing the

Gray codes give the intermediate step of writing the code followed by the same written in reverse order.

Table 2.3 Gray code.

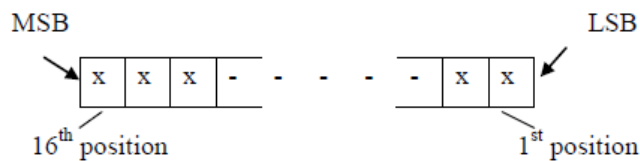
Decimal	Binary	Gray	Decimal	Binary	Gray
0	0000	0000	8	1000	1100
1	0001	0001	9	1001	1101
2	0010	0011	10	1010	1111
3	0011	0010	11	1011	1110
4	0100	0110	12	1100	1010
5	0101	0111	13	1101	1011
6	0110	0101	14	1110	1001
7	0111	0100	15	1111	1000

Table 2.4 Generation of higher-bit Gray code numbers.

One-bit Gray code	Two-bit Gray code		Three-bit Gray code		Four-bit Gray code	
0	0	00	00	000	000	0000
1	1	01	01	001	001	0001
	1	11	11	011	011	0011
	0	10	10	010	010	0010
			10	110	110	0110
			11	111	111	0111
			01	101	101	0101
			00	100	100	0100
					100	1100
					101	1101
					111	1111
					110	1110
					010	1010
					011	1011
					001	1001
					000	1000

2.3.1 Binary–Gray Code Conversion

- The given binary code is shifted to the right by one bit.
- Discard the last bit (the LSB) from the obtained bits.
- Exclusive-ORing the given and obtained bits result in the equivalent Gray code.



Ex

Compute the Gray code for the binary number 11010

Solution

Binary code	1 1 0 1 0
	\oplus 1 1 0 1
Gray code	1 0 1 1 1

Ex

Compute the Gray code for the binary number 10001101

Solution

Binary code	1 0 0 0 1 1 0 1
	\oplus 1 0 0 0 1 1 0
Gray code	1 1 0 0 1 0 1 1

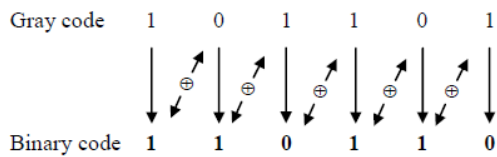
Gray-to-Binary Conversion

- The first bit, the leftmost of the given Gray code, becomes the MSB of the Binary code.
- Exclusive-ORing the second Gray code bit with the MSB of the binary code yields the second binary bit.
- Exclusive-ORing the third Gray code bit with the second binary code yields the third binary bit.
- Exclusive-ORing the fourth Gray code bit with the third binary code yields the fourth binary bit. And so on.

Ex

Compute the binary code for the Gray code 101101

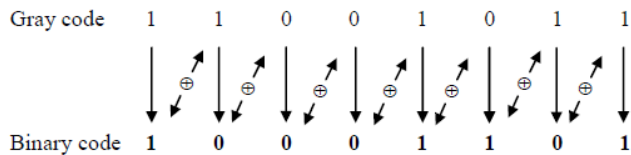
Solution



Ex

Compute the binary code for the Gray code 11001011

Solution



Applications

1. The Gray code is used in the transmission of digital signals as it minimizes the occurrence of errors.
2. The Gray code is preferred over the straight binary code in angle-measuring devices. Use of the Gray code almost eliminates the possibility of an angle misread, which is likely if the angle is represented in straight binary. The cyclic property of the Gray code is a plus in this application.
3. The Gray code is used for labelling the axes of Karnaugh maps, a graphical technique used for minimization of Boolean expressions.
4. The use of Gray codes to address program memory in computers minimizes power consumption. This is due to fewer address lines changing state with advances in the program counter.
5. Gray codes are also very useful in genetic algorithms since mutations in the code allow for mostly incremental changes. However, occasionally a one-bit change can result in a big leap, thus leading to new properties.

Exercise

Find (a) the Gray code equivalent of decimal 13 and (b) the binary equivalent of Gray code number 1111

ALPHANUMERIC CODE:

Alphanumeric codes are also called character codes due to their certain properties. These codes are basically binary. These codes are used to write alphanumeric data, including data, letters of the alphabet, numbers, mathematical symbols and punctuation marks which can be easily understandable and can be processed by the computers. Input output devices such as keyboards, monitors, mouse can be interfaced using these codes. A complete alphanumeric code would include the 26 lowercase letters, 26 uppercase letters, 10 numeric digits, 7 punctuation marks and anywhere from 20 to 40 other characters such as +, /, *, # and so on. That is it represents all of the various characters and functions that are found on a standard typewriter or computer keyboard. The most common alphanumeric codes used are ASCII code, EBCDIC code and Unicode.

1. ASCII CODE

The full form of ASCII code is American Standard Code for Information Interchange. It is a seven-bit code based on the English alphabet. In 1967 this code was first published and since then it is being modified and updated. ASCII code has 128 characters some of which are enlisted below to get familiar with the code.

Dec	Octal	Hex	Binary	Symbol	Description
1	001	01	00000001	SOH	Start of Heading
2	002	02	00000010	STX	Start of text
3	003	03	00000011	ETX	End of text
4	004	04	00000100	EOT	End of transmission
5	005	05	00000101	ENQ	Enquiry
6	006	06	00000110	ACK	Acknowledgement
7	007	07	00000111	BEL	Bell
8	010	08	00001000	BS	Back Space
9	011	09	00001001	HT	Horizontal Tab
10	012	0A	00001010	LF	Line Feed
11	013	0B	00001011	VT	Vertical Tab
12	014	0C	00001100	FF	Form Feed
13	015	0D	00001101	CR	Carriage Return
14	016	0E	00001110	SO	Shift Out/X-On
15	017	0F	00001111	SI	Shift In/X-O

Example:

The following is a message encoded in ASCII code. What is the message?

01001000 1000101 1001100 1010000

Solution:

Convert each 7 bit code to its Hexadecimal equivalent

0100	1000	0100	0101	0100	1100	0101	0000
4	8	4	5	4	C	5	0

The result are 48 45 4C 50

Locate these hexadecimal values in table ASCII and determine the character represented by each. The result are: H E L P

2. EBCDIC CODE:

The EBCDIC stands for Extended Binary Coded Decimal Interchange Code. IBM invented this code to extend the Binary Coded Decimal which existed at that time. All the IBM computers and peripherals use this code. It is an 8 bit code and therefore can accommodate 256 characters. Below is given some characters of EBCDIC code to get familiar with it.

Char	EBCDIC	HEX	Char	EBCDIC	HEX	Char	EBCDIC	HEX
A	1100 0001	C1	P	1101 0111	D7	4	1111 0100	F4
B	1100 0010	C2	Q	1101 1000	D8	5	1111 0101	F5
C	1100 0011	C3	R	1101 1001	D9	6	1111 0110	F6
D	1100 0100	C4	S	1110 0010	E2	7	1111 0111	F7
E	1100 0101	C5	T	1110 0011	E3	8	1111 1000	F8
F	1100 0110	C6	U	1110 0100	E4	9	1111 1001	F9
G	1100 0111	C7	V	1110 0101	E5	blank
H	1100 1000	C8	W	1110 0110	E6
I	1100 1001	C9	X	1110 0111	E7	(...	...
J	1101 0001	D1	Y	1110 1000	E8	+
K	1101 0010	D2	Z	1110 1001	E9	\$
L	1101 0011	D3	0	1111 0000	F0	*
M	1101 0100	D4	1	1111 0001	F1)
N	1101 0101	D5	2	1111 0010	F2	-
O	1101 0110	D6	3	1111 0011	F3	/		

3. UNICODE

Encodings such as ASCII, EBCDIC and their variants do not have a sufficient number of characters to be able to encode alphanumeric data of all forms, scripts and languages. As a result, these encodings do not permit multilingual computer processing. In addition, these encodings suffer from incompatibility. Two different encodings may use the same number for two different characters or different numbers for the same characters. For example, code 4E (in hex) represents the upper-case letter 'N' in ASCII code and the plus sign '+' in the EBCDIC code. Unicode, developed jointly by the Unicode Consortium and the International Organization for Standardization (ISO), is the most complete character encoding scheme that allows text of all forms and languages to be encoded for use by computers. It not only enables the users to handle practically any language and script but also supports a comprehensive set of mathematical and technical symbols, greatly simplifying any scientific information exchange. The Unicode standard has been adopted by such industry leaders as HP, IBM, Microsoft, Apple, Oracle, Unisys, Sun, Sybase, SAP and many more.