

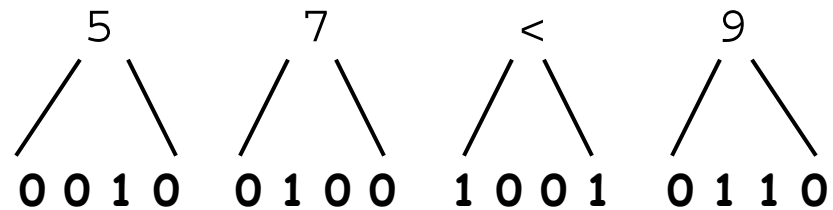
Binary Codes

- Digital systems and circuits work with signals that have only one of two states corresponding to digital 1 and 0.
- Any discrete element of information among a group of quantities (elements) can be represented by a binary code.
- One bit can represent up to two elements (1 or 0).
- A group of 2^n distinct elements requires a minimum of n bits. That means to code a group of m elements, we need to use n bits such that: $2^n \geq m$.
- Examples:
 - A group of four elements can be represented by two bit code [00, 01, 10, and 11].
 - A binary code to represent the decimal digits [0-9], must contain at least four bits because $(2^4=16) \geq 10 \geq (2^3=8)$.

BCD Code

Decimal Digit	8-4-2-1 Code (BCD)
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1

Convert $57<9_{43}$ to BCD Code



Note this is very different from converting to binary which yields:

100111000000_2

Weighted Codes

- BCD is a weighted code

Each binary digit has a position or weight (w) which can be added algebraically to compute the decimal digit

$$N = w_3a_3 + w_2a_2 + w_1a_1 + w_0a_0$$

- Other weighted codes are possible

8 4 -2 -1 code

2 4 2 1 code

etc.

Weighted Codes (cont.)

Decimal Digit	BCD 8421		Excess-3	84-2-1	2421	
0	0000		0011	0000	0000	MSB = 0
1	0001		0100	0111	0001	
2	0010		0101	0110	0010	
3	0011		0110	0101	0011	
4	0100		0111	0100	0100	MSB = 1
5	0101		1000	1011	1011	
6	0110		1001	1010	1100	
7	0111		1010	1001	1101	
8	1000		1011	1000	1110	
9	1001		1100	1111	1111	
			Self complement codes			

BCD Addition

- The addition of two BCD digits with a possible carry from the previous less significant pair of digits results in a sum in the range 0 to a maximum of $(9+9+1=19)$. There is a difference in the representation of the sum in binary and in BCD code.
 - If $0 \leq \text{Sum} \leq 9$ then, sum in BCD = sum in binary,
 - If $10 \leq \text{Sum} \leq 19$ then, sum in BCD consists of 8 bits which is not equal to the sum in binary. The correction needed in the sum to represent it in BCD is the addition of 6 to the binary sum.

- Examples:

$$\begin{array}{r} 4 \quad 0100 \\ +5 \quad 0101 \\ \hline 9 \quad 1001 \end{array}$$

$$\begin{array}{r} 4 \quad 0100 \\ +8 \quad 1000 \\ \hline 12 \quad 1100 \\ +0110 \\ \hline 10010 \end{array}$$

$$\begin{array}{r} 8 \quad 1000 \\ +9 \quad 1001 \\ \hline +17 \quad 10001 \\ +0110 \\ \hline 10111 \end{array}$$

Error Detection Code

A common method to achieve error detection in binary information transmission from one location to another is by means of a parity bit. A parity bit is defined as an extra bit included with a message to make the total number of 1's transmitted either odd or even.

	Odd Parity		Even Parity
Message	P	Message	P
0000	1	0000	0
0001	0	0001	1
0010	0	0010	1
0011	1	0011	0
0100	0	0100	1
0101	1	0101	0
0110	1	0110	0
0111	0	0111	1
1000	0	1000	1
1001	1	1001	0
1010	1	1010	0
1011	0	1011	1
1100	1	1100	0
1101	0	1101	1
1110	0	1110	1
1111	1	1111	0

2 out of 5 Code

Decimal Digit	2-out-of-5 Code
0	0 0 0 1 1
1	0 0 1 0 1
2	0 0 1 1 0
3	0 1 0 0 1
4	0 1 0 1 0
5	0 1 1 0 0
6	1 0 0 0 1
7	1 0 0 1 0
8	1 0 1 0 0
9	1 1 0 0 0

- Every code has exactly 2 ones
- Not a weighted code
- Useful for certain types of error detection
- Very difficult to add two numbers

Gray Code

Number	Gray Code
0	0 0 0 0
1	0 0 0 1
2	0 0 1 1
3	0 0 1 0
4	0 1 1 0
5	0 1 1 1
6	0 1 0 1
7	0 1 0 0
8	1 1 0 0
9	1 1 0 1
10	1 1 1 1
11	1 1 1 0
12	1 0 1 0
13	1 0 1 1
14	1 0 0 1
15	1 0 0 0

- Only one bit changes with each number increment
- Not a weighted code
- Useful for translating from an analog quantity (shaft encoding)

ASCII Code [American Standard Code for Information Interchange] (partial)

Character	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀
c	1	1	0	0	0	1	1
d	1	1	0	0	1	0	0
e	1	1	0	0	1	0	1
f	1	1	0	0	1	1	0
g	1	1	0	0	1	1	1
h	1	1	0	1	0	0	0
i	1	1	0	1	0	0	1
j	1	1	0	1	0	1	0
k	1	1	0	1	0	1	1
l	1	1	0	1	1	0	0
m	1	1	0	1	1	0	1
n	1	1	0	1	1	1	0
o	1	1	0	1	1	1	1
p	1	1	1	1	0	0	0
q	1	1	1	0	0	0	1

Used for handling numbers, letters, characters...etc.

- This type of code is called alpha-numeric code.

- To handle 10 decimal digits, 26 small letters, 26 capital letters, and at least 20 other characters, the code must cater for at least 82 elements.

$(128=2^7) \geq 82 \geq (64=2^6)$.

- Therefore we need seven bits for the code.

Convert help to ASCII

h e l p

1101000 1100101 1101100 1111000

Codes - Summary

- Bits are bits...
 - Modern digital devices represent everything as collections of bits
 - A computer is one such digital device
- You can encode anything with sufficient 1's and 0's
 - Text (ASCII, EBCDIC)
 - Computer programs (C code, assembly code, machine code)
 - Sound (.wav, .mp3, ...)
 - Pictures (.jpg, .gif, .tiff)