- Errors of transmission or errors in memory storage can be detected by checking parity bits.
- ASCII coding is used for transmission and storage of characters.
- The Hamming distance is a measure of the difference between two code segments.
- Hamming codes contain more error check bits which allow two or more errors to be detected and corrected.

Computers store binary numbers
not decimal digits or letters.

Assign a number in range 0 to 255_D or 0 to FF_H to each character.

ASCII code is most common assignment.

"The cat is on the mat."
is stored (in HEX) as:
54 68 65 20 63 61 74 20 69 73 20 6F 6E 20
74 68 65 20 6D 61 74 2E
within the computer.

Transmission and memory errors sometimes occur.
Add a parity bit and agree on a convention.
This gives extra information which allows the detection of errors.

Suppose we use an EVEN parity convention.

If data word is 9B = 10011011
There are 5 1's so we add a 1 in the parity bit position to get 100110111 which has nine bit and an even number of 1's.

If the received word does not have EVEN parity then an error has occurred.

Simple parity checks will detect single bit errors but not double errors.

Not a problem if errors occur randomly.

BUT errors can be due to bursts of interference.

M out of N code.

Each word has n bits and m of these are always 1's.

How many 2 out of 5 codes are there?

All single errors will be detected and some double errors.

Berger code.

Count up the number of 1's in the data part. Express this in binary and put at end to form the full word.

0110111 5 one bits or 0101
01101110101 Full Berger code.

Ideal situation allows errors to be detected and repaired without need for repeated transmission.

Hamming distance is measure of difference between data words.

2^n distinct words with n bits.

Choose subset which differ in 2 bit positions.

Add a check bit and it is then possible to repair data.

The **minimum Hamming distance** of a set of code words is the minimum number of bits of a code word that must be changed in order to convert one valid code word of the set into another valid code word of the set.

x		0100
y		0011
z		1010
w		1001
then x and y	differ in	3 bit positions
x and z	differ in	3 bit positions
x and w	differ in	3 bit positions
y and z	differ in	2 bit positions
y and w	differ in	2 bit positions
z and w	differ in	2 bit positions

This set of code words has a minimum Hamming distance of 2, that is $d = 2$.

It can be shown by a theoretical analysis that if C is the number of bit errors that can be corrected and D is the number of bit errors that can be detected then:

$$d = C + D + 1 \text{ and } D \geq C$$

In the 4 bit code words shown above, $d = 2$ so that $C + D = 2 - 1 = 1$ and since $D \geq C$ the only valid solution is $C = 0$ and $D = 1$ so that we can detect a single error bit but we cannot correct the error.

Extra check bits are appended
If there are m bits in the data word and there are p check bits appended then the required number of bits for correction of errors in the $m + p$ bits of the code word is given by the value of p which satisfies the **Hamming relationship**:

$$2^p \geq m + p + 1$$

where m = number of data bits and p = number of check bits and the resulting code is called the **Hamming code**.

m	4	5	6	7	8	9	10	16	32	200
p	3	3	4	4	4	4	4	5	6	8

In constructing the Hamming code, the convention is that the kth check bit is put in the $2^{(k-1)}$th position in the Hamming code word.

Bit fun	p_1	p_2	d_1	p_3	d_2	d_3	d_4
Pos no	1	2	3	4	5	6	7
Pos no	001	010	011	100	101	110	111
P of	1357	2367		4567			

To form a parity bit; take the parity bit position number (row 3) and locate the single 1 bit. The parity is formed using the bits in the code for which the position number (in binary) has a 1 in the same place as the parity bit. The fourth row shows the numbers of the bits which are used to form the parity bit at that position.

7 bit Hamming code for the data word 0111 using even parity.

p_1	p_2	0	p_3	1	1	1
p_1 	 p_2 	0 0 	 p_3	1 1	 1 1	1 1 1
0	0	0	1	1	1	1

Recover the correct data word from the received 7 bit code 0111001, given that odd parity is in use.

p_1	p_2	d_1	p_3	d_2	d_3	d_4	function
0	1	1	1	0	0	1	received
0		1		0		1	error
	1	1			0	1	no error
			1	0	0	1	error
0	1	1	1	1	0	1	corrected