Cambridge IGCSE

Computer Science

Section 1

Methods of Error Detection

Unit 2:
Data Transmission

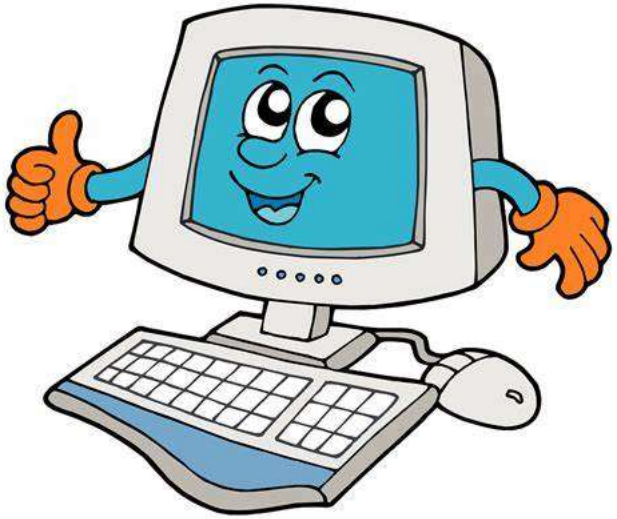
Objectives

- Understand the need to check for errors after data transmission and how these errors occur
- Describe the processes involved in odd and even parity check, check sum and echo check
- Describe how a check digit is used to detect errors in data entry and examples of its use e.g. ISBN and barcodes
- Describe how an automatic repeat request (ARQ) can be used to establish that data is received without error

Transmission errors

- When data is transmitted, it doesn't always arrive in the same format that it was sent. This could be due to:
  - **Electrical interference**
  - **Power surges**
  - **Synchronisation issues due to skewing of data**
  - **Wear and tear on the cable or connectors**
  - **Problems with packet switching**
- Data could be lost, data could be gained or data could be changed - bits flip from 1s to 0s and 0s to 1s

Transmission errors



1101

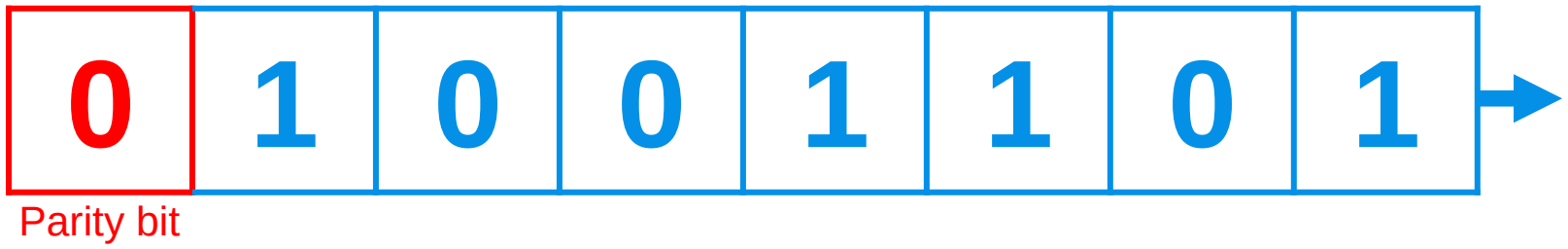
1100

Error checking

- Data corruption can be a serious problem in computing - think about all the critical systems that rely on computers
- It is important therefore, to have methods to check for data transmission errors
- Methods include:
  - **Parity checks**
  - **Check digits**
  - **Checksums**
  - **Echo checks**
  - **Automatic Repeat reQuests (ARQ)**

Parity checks

- Used to check if data has been changed in transmission
- Counts the **number of bits with a value of 1**, in a byte of data
- Computers use either **odd** or **even** parity
- When the parity is set as EVEN, there should be an even number of 1s in the data received



Parity checks

- When the parity is set as ODD, there should be an odd number of 1s in the data received
- When sending a byte of data, one bit is used as a **parity bit**
- This bit is set to a 1 or 0 to make the total number of 1s or 0s in the byte (including the parity bit) odd or even depending on the parity check used

Parity bits and blocks

Parity Checking

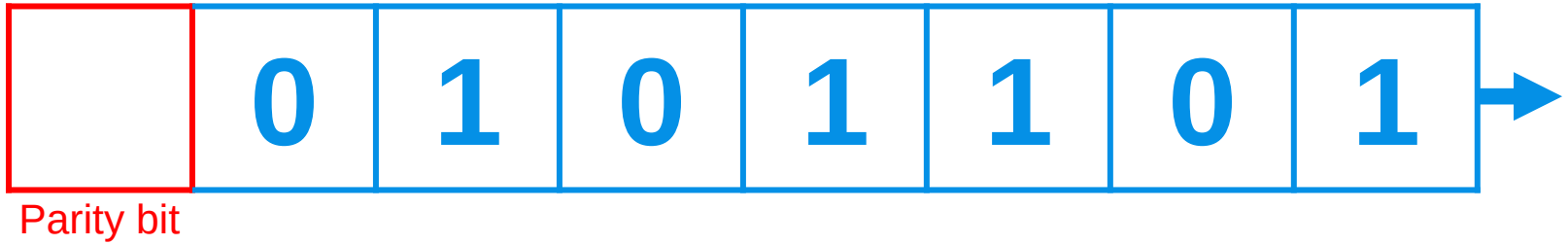
Part of our toolkit for **error detection**

A way of trying to establish if binary data has changed during **transmission**

Odd and even parity

- What would the value of the parity bit be for the following, using **even** parity?

- Give the parity bit value here using **odd** parity?



Using parity for error detection

- Whether to use odd or even parity is first agreed between the sending and receiving computers
- When data is transmitted, the parity bit is set at the transmitting end and parity is checked at the receiving end
- If the wrong number of bits are ‘on’, an error has occurred
- The receiving computer notifies the transmitting end and the data is resent

Using parity for error detection

How could a parity check allow an erroneous byte past detection?

Using parity for error detection

How could a parity check allow an erroneous byte past detection?

Example, even parity is being used ...

Data sent

Data received

Using parity for error detection

How could a parity check allow an erroneous byte past detection?

Example, even parity is being used ...

Data sent

Data received

Using parity for error detection

Looking at just one byte of data, it is difficult to find which bit might have the error. **Parity blocks** can be used to locate an erroneous bit.

	Parity bit	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
Byte 1	1	1	1	1	0	1	1	0
Byte 2	1	0	0	1	0	1	0	1
Byte 3	0	1	1	1	1	1	1	0
Byte 4	1	0	0	0	0	0	1	0
Byte 5	0	1	1	0	1	0	0	1
Byte 6	1	0	0	0	1	0	0	0
Byte 7	1	0	1	0	1	1	1	1
Byte 8	0	0	0	1	1	0	1	0
Byte 9	0	0	0	1	0	0	1	0
Parity byte	1	1	0	1	0	0	0	1

Activity

Find the parity bits for each of the following bytes:

	1	1	0	1	1	0	1	even parity being used
	0	0	0	1	1	1	1	even parity being used
	0	1	1	1	0	0	0	even parity being used
	1	1	1	0	1	0	0	odd parity being used
	1	0	1	1	0	1	1	odd parity being used
	1	1	1	1	1	1	0	even parity being used
	1	1	1	1	1	1	0	odd parity being used
	1	1	0	1	0	0	0	odd parity being used
	0	0	0	0	1	1	1	even parity being used
	1	1	1	1	1	1	1	odd parity being used

Activity

The following block of data was received after transmission from a remote computer. Even parity was being used by both sender and receiver. One of the bytes has been changed during the transmission stage. Locate where this error is and suggest a corrected byte value.

	Parity bit	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
Byte 1	1	1	0	0	0	0	0	0
Byte 2	0	0	1	1	1	1	0	0
Byte 3	0	1	0	0	0	1	1	1
Byte 4	1	0	1	0	1	1	1	1
Byte 5	0	0	0	1	0	0	0	1
Byte 6	0	0	1	1	1	1	1	1
Byte 7	1	0	1	1	0	1	0	0
Byte 8	0	1	0	1	0	0	0	1
Byte 9	1	1	1	0	0	1	0	0
Parity byte	0	0	1	1	1	0	0	1

Check digits

- A check digit is an additional digit at the end of a string of other numbers designed to check for mistakes in input or transmission
- Printed books and other products have a unique barcode with an **ISBN** (International Standard Book Number) or **EAN** (European Article Number)
- The first 12 digits of the **barcode** are the unique item number, the 13th is the check digit

Calculating the check digit

Example using the **Modulo 10** system

ISBN	5	0	1	4	0	1	6	1	5	0	8	2	1
Weight													
Multiplication													
Addition	Add all the numbers												
Remainder	Find the remainder when divided by 10												
Subtraction	Subtract the result from 10												

Calculating the check digit

Example using the **Modulo 10** system

ISBN	5	0	1	4	0	1	6	1	5	0	8	2	1
Weight	1	3	1	3	1	3	1	3	1	3	1	3	
Multiplication													
Addition	Add all the numbers												
Remainder	Find the remainder when divided by 10												
Subtraction	Subtract the result from 10												

Calculating the check digit

Example using the **Modulo 10** system

ISBN	5	0	1	4	0	1	6	1	5	0	8	2	1
Weight	1	3	1	3	1	3	1	3	1	3	1	3	
Multiplication	5	0	1	12	0	3	6	3	5	0	8	6	
Addition	Add all the numbers												
Remainder	Find the remainder when divided by 10												
Subtraction	Subtract the result from 10												

Calculating the check digit

Example using the **Modulo 10** system

ISBN	**5**	**0**	**1**	**4**	**0**	**1**	**6**	**1**	**5**	**0**	**8**	**2**	**1**
Weight	1	3	1	3	1	3	1	3	1	3	1	3	
Multiplication	5	0	1	12	0	3	6	3	5	0	8	6	
Addition	Add all the numbers												49
Remainder	Find the remainder when divided by 10												
Subtraction	Subtract the result from 10												

Calculating the check digit

Example using the **Modulo 10** system

ISBN	5	0	1	4	0	1	6	1	5	0	8	2	1
Weight	1	3	1	3	1	3	1	3	1	3	1	3	
Multiplication	5	0	1	12	0	3	6	3	5	0	8	6	
Addition	Add all the numbers												49
Remainder	Find the remainder when divided by 10												9
Subtraction	Subtract the result from 10												

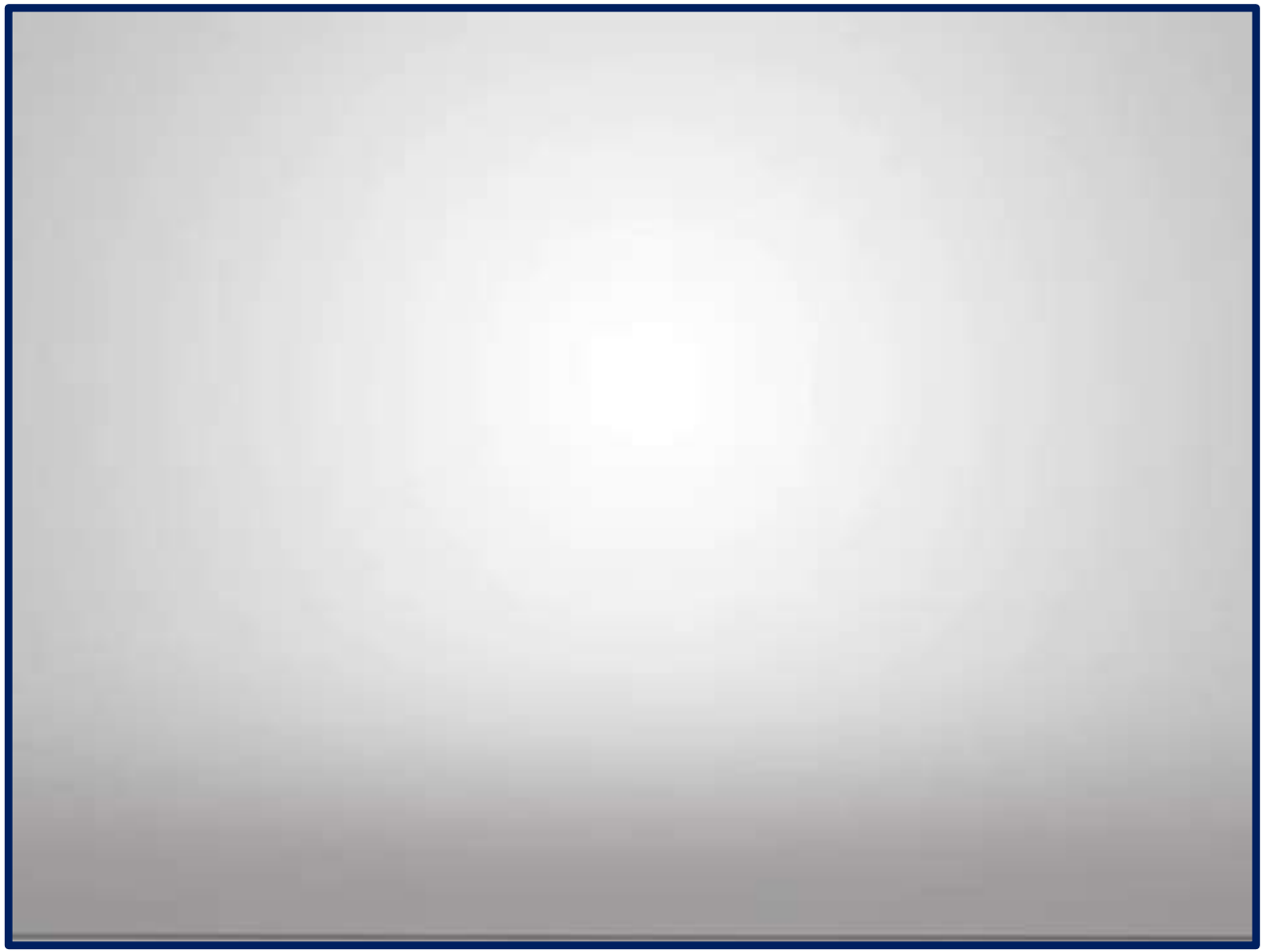
Calculating the check digit

Example using the **Modulo 10** system

The numbers match
-> data is correct

ISBN	5	0	1	4	0	1	6	1	5	0	8	2	1
Weight	1	3	1	3	1	3	1	3	1	3	1	3	
Multiplication	5	0	1	12	0	3	6	3	5	0	8	6	
Addition	Add all the numbers												49
Remainder	Find the remainder when divided by 10												9
Subtraction	Subtract the result from 10												1

Calculating the check digit



Worksheet - Error checking

- Complete **Task 1** to calculate the check digit on the back of a book, or use the one below:

Checksums

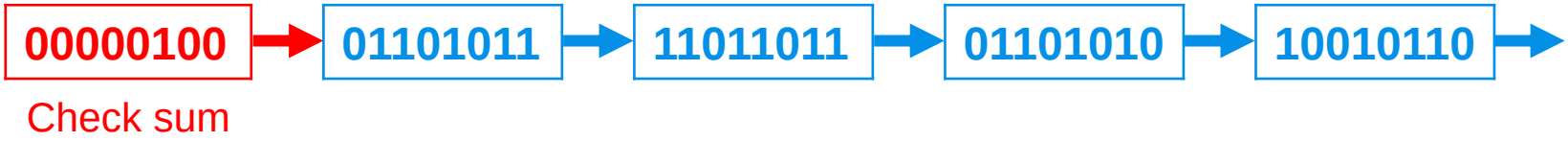
- A checksum works in a similar way to a check digit
  - The checksum of all bytes in a data transmission is calculated using an **algorithm** that is agreed and used by both sender and receiver

algorithm - a set of rules to be followed in calculations and problem-solving operations

  - The checksum is transmitted with the data
  - The receiving computer recalculates the checksum based on the data it received and compares it with the checksum sent with the data
  - If it does not match, the data may have been altered or corrupted during transmission and a request is made to resend the data
- Checksums are used with credit card numbers

Simple checksum example

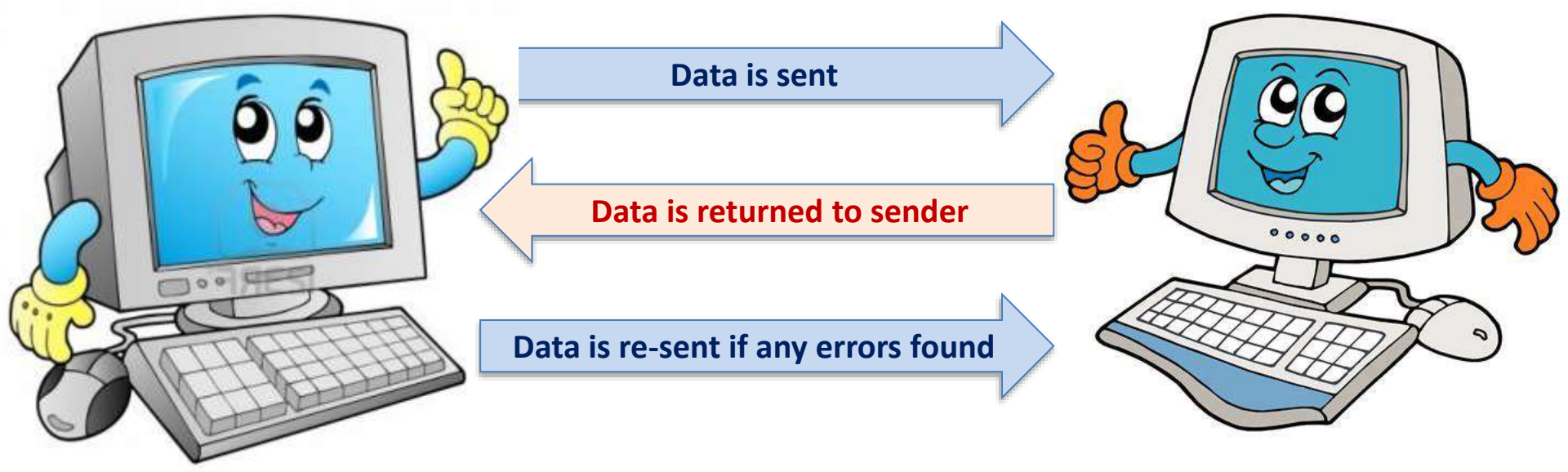
- The number of bytes in the transmission is counted as 4
- The checksum is given the value 4 in binary and sent with the real data



- What weaknesses are there in the method used in this example?

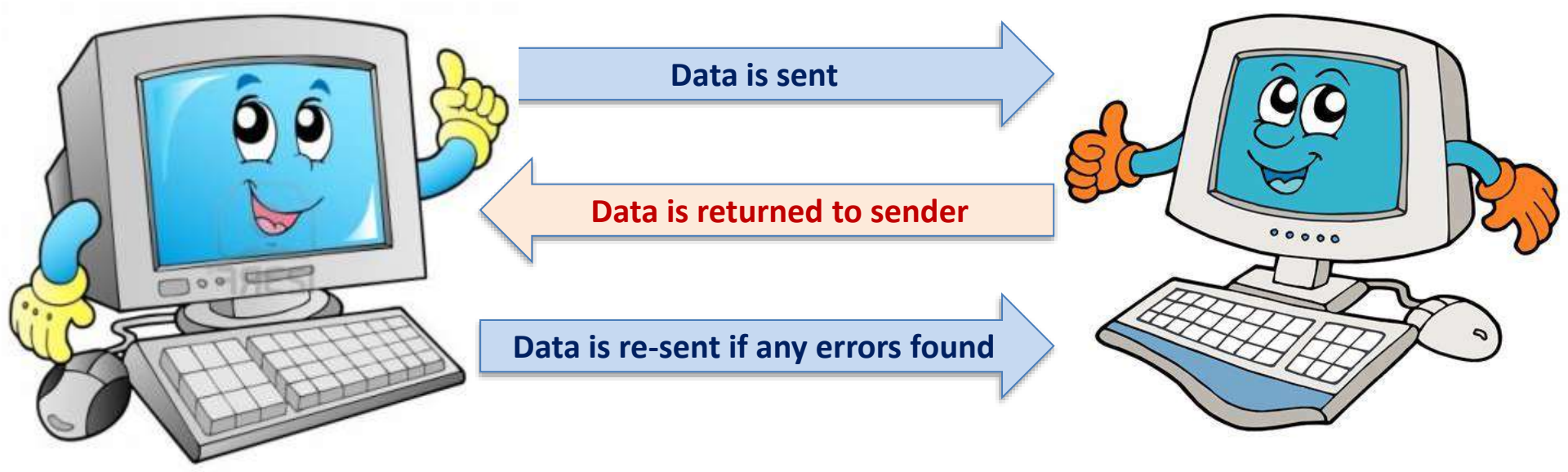
Echo check

- When data is sent to a device, a copy of that same data is sent back to the sender, by the receiver
- The sending computer compares what it receives with what it originally sent, to check for any differences



Echo check

- Differences between the data suggest errors in transmission
- An error-free echo check indicates that no data was changed or corrupted
- Why might Echo Check not be very reliable?



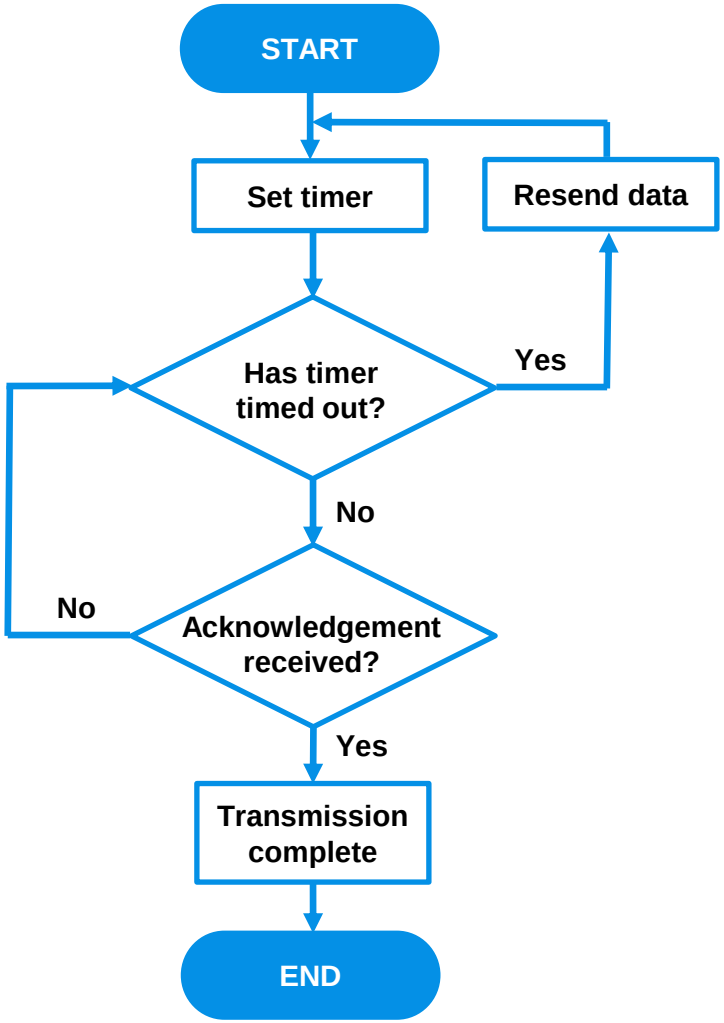
Automatic Repeat reQuests (ARQs)

- An **Automatic Repeat Request** (**ARQ**) is another way to check that data has been received correctly after transmission
- The receiving computer will automatically return an **acknowledgement** to the sender that the data was received correctly - a positive acknowledgement
- If the data is not received correctly, a negative acknowledgement is sent back to the sending computer, and in this case, the data would be retransmitted

Automatic Repeat reQuests (ARQs)

- The simplest **Stop-and-wait ARQ** will resend the data if an acknowledgement is **not** received within a specific time period, this is called a **timeout**; it will assume that there was an error in the transmission
- The data will keep being resent until a positive acknowledgement is received or until a limit of resends has been reached
    - It is sometimes used with mobile phone network data

ARQ flowchart



Worksheet - Error checking

- Complete **Task 2**

Vocabulary

- **transmission**
- **error**
- **parity bit**
- **parity check**
- **parity block**
- **check digit**
- **checksum**
- **algorithm**
- **echo**
- **echo check**
- **automatic repeat request**
- **acknowledgement**
- **timeout**
- **positive**
- **negative**

Summary

- Errors occur during data transmission and so methods are required to detect and correct these
- Processes involved include:
  - Odd and even parity checks
  - Checksum
  - Echo check
- Check digits also detect errors and are used in bar codes and international standard book numbers (ISBN)
- ARQs use positive and negative acknowledgements and timeouts to establish whether data has been received without error

Past paper question examples ...

9 The contents of three binary registers have been transmitted from one computer to another. **Even parity** has been used as an error detection method.

The outcome after transmission is:

Register A and **Register C** have been transmitted **correctly**.

Register B has been transmitted **incorrectly**.

Complete the **Parity bit** for each register to show the given outcome.

	Parity bit							
Register A		0	1	0	0	1	0	1
Register B		1	0	0	0	0	0	1
Register C		1	0	0	0	0	1	1

[3]

Past paper question examples ...

(b) The system uses parity bits to check for errors during data transmission.

The outcome of four bytes after transmission is:

Byte 1	Byte 2	Byte 3	Byte 4
00110011	01010100	10110100	01110111

One of the bytes has been transmitted incorrectly.

Identify the byte that was transmitted incorrectly.

Byte .....

Explain how you identified the byte that was transmitted incorrectly.

.....

.....

.....

.....

[4]

Past paper question examples ...

(iii) The library's archive system uses an error detection and correction system that combines a parity check with an automatic repeat request (ARQ).

Describe how this system uses the parity check and ARQ.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [6]

Past paper question examples ...

10 Data is valuable to a company.

(a) Companies use error detection methods to make sure that data is accurate.

One error detection method is the use of a check digit.

Explain what is meant by a check digit and how it is used to detect errors.

.....

.....

.....

.....

.....

.....

.....

..... [4]

Past paper question examples ...

4 Two error detection methods that Allison's computer uses are check digit and checksum.

(a) Give **two** similarities between the check digit and checksum methods.

1 .....

.....

2 .....

.....

[2]

Past paper question examples ...

(b) Identify **one other** error detection method that Allison's computer could use.

Describe how the method checks for errors.

Method .....

Description .....

.....

.....

.....

.....

.....

[4]

Extras ...

Printout slides … 8 ->