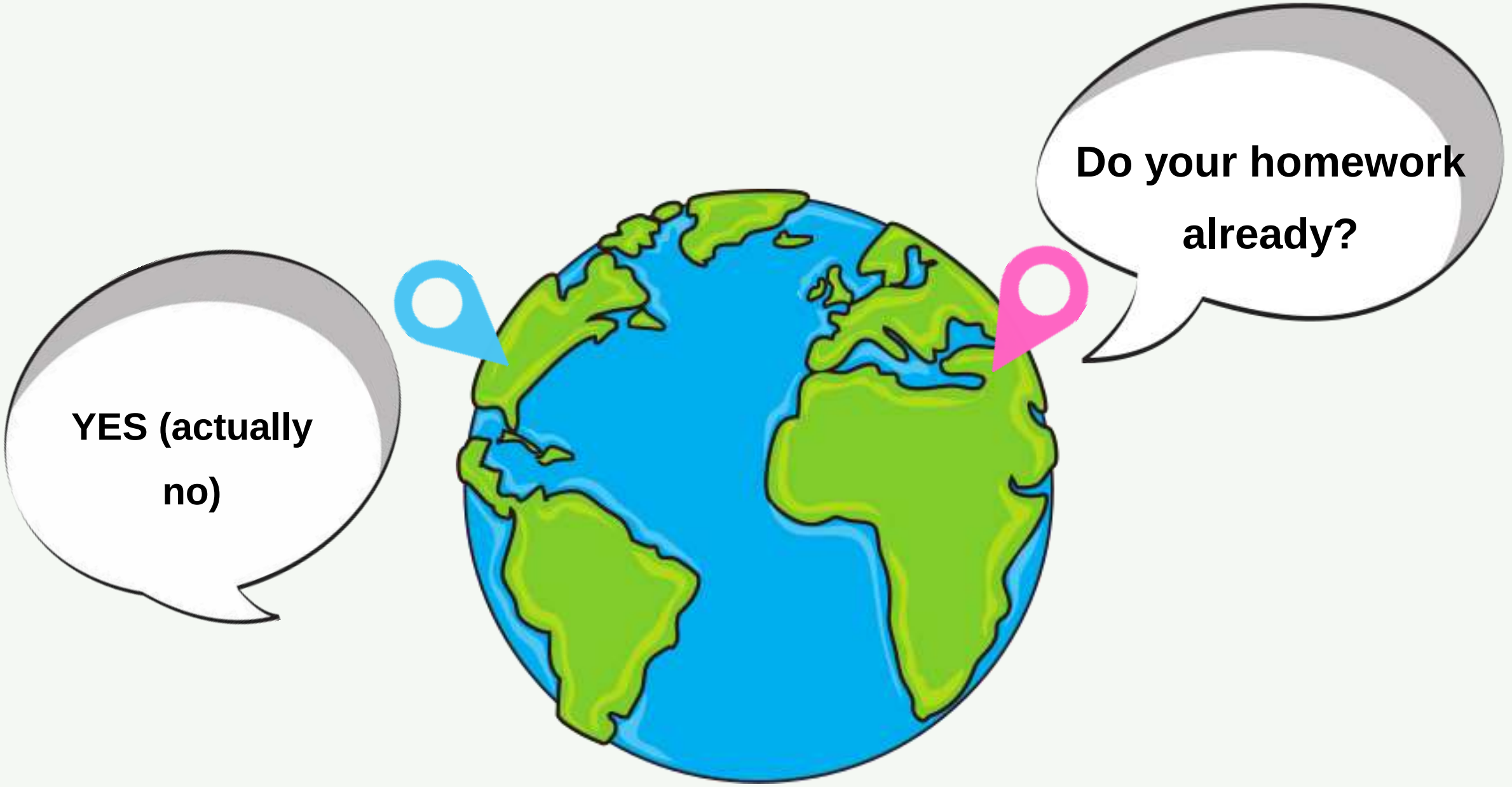
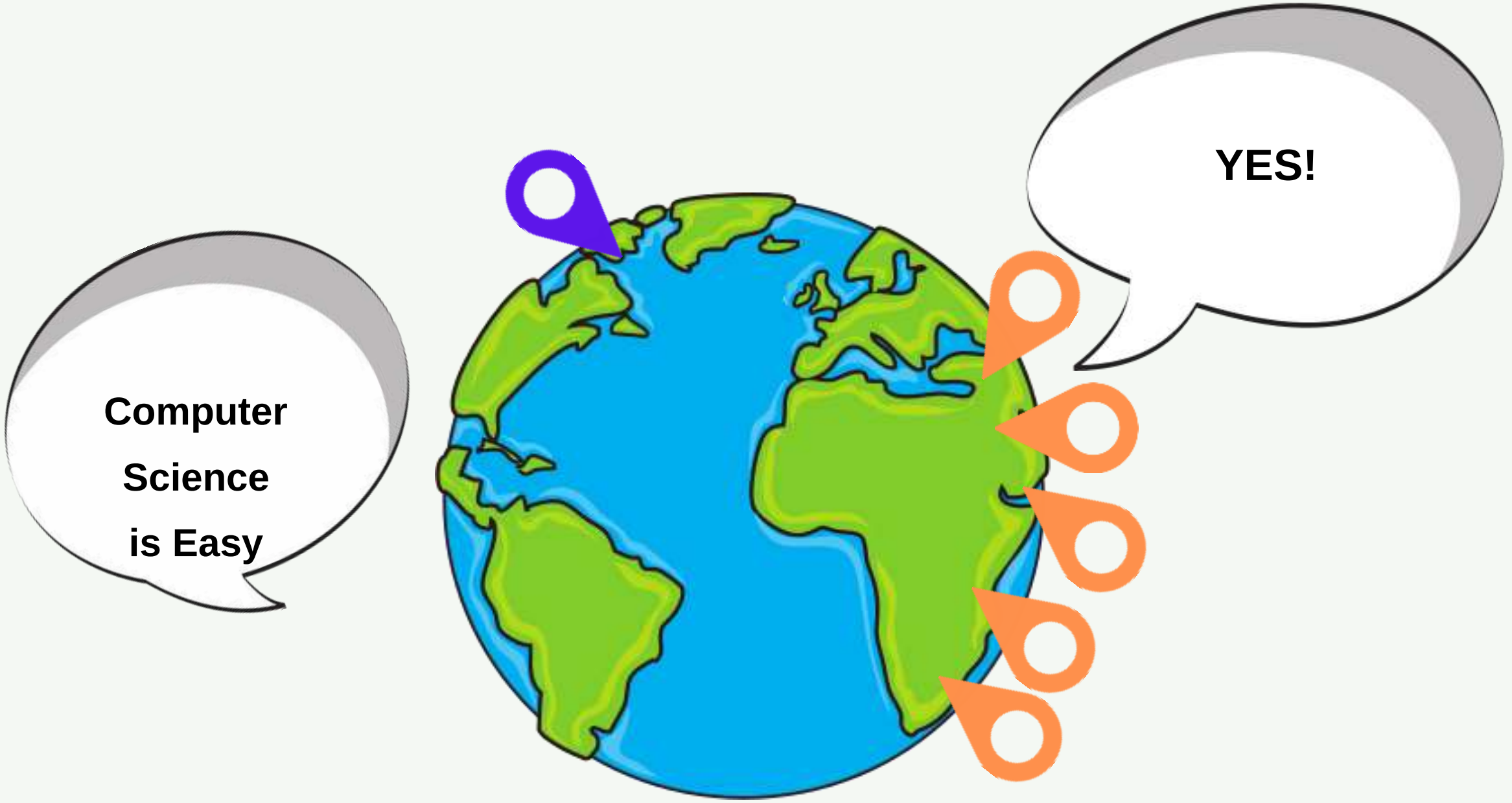
CHAPTER 2.1

Types and modes of Data Transmission

IGCSE COMPUTER SCIENCE


Do your homework already?
YES (actually no)

HOW IS DATA (TEXT, IMAGE, AND SOUND) TRANSMITTED?



LESSON OBJECTIVE

Packet Structure

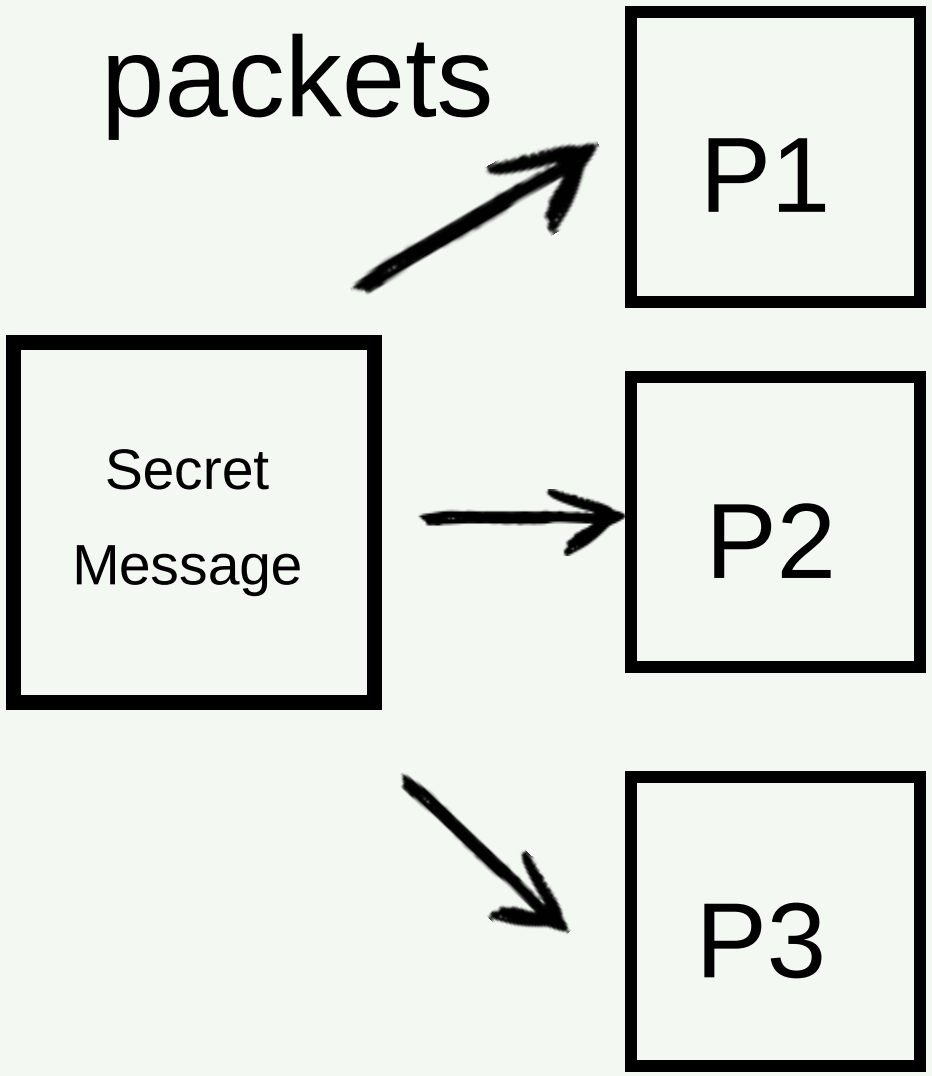
Packet Switching

Types of Data Transmission

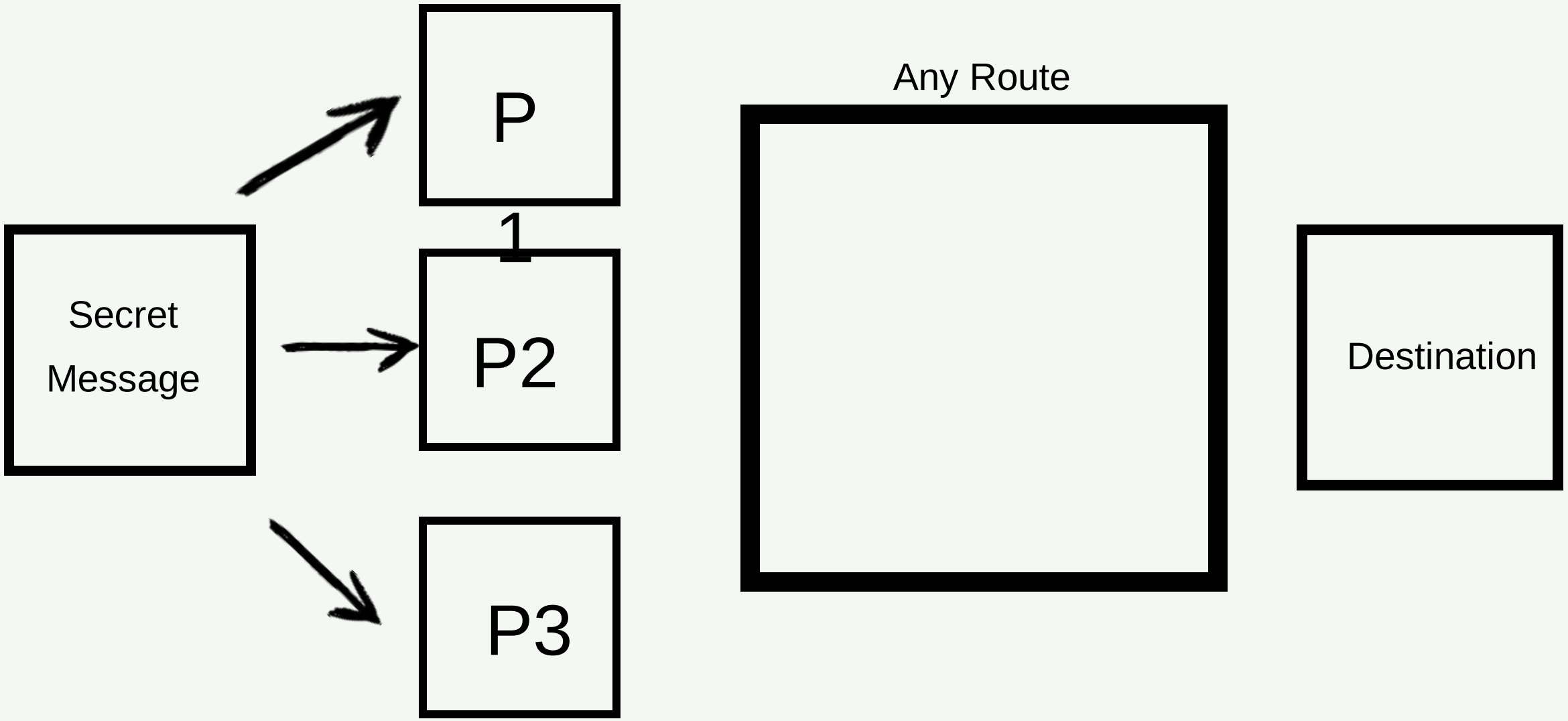
Data Packets

- Data sent over long distances is usually broken up into data packets (datagrams).
- Packet size is roughly 64KiB.
- This makes it easier to control than a long continuous stream of data.
- Each Packet can be sent along a different route to its destination.

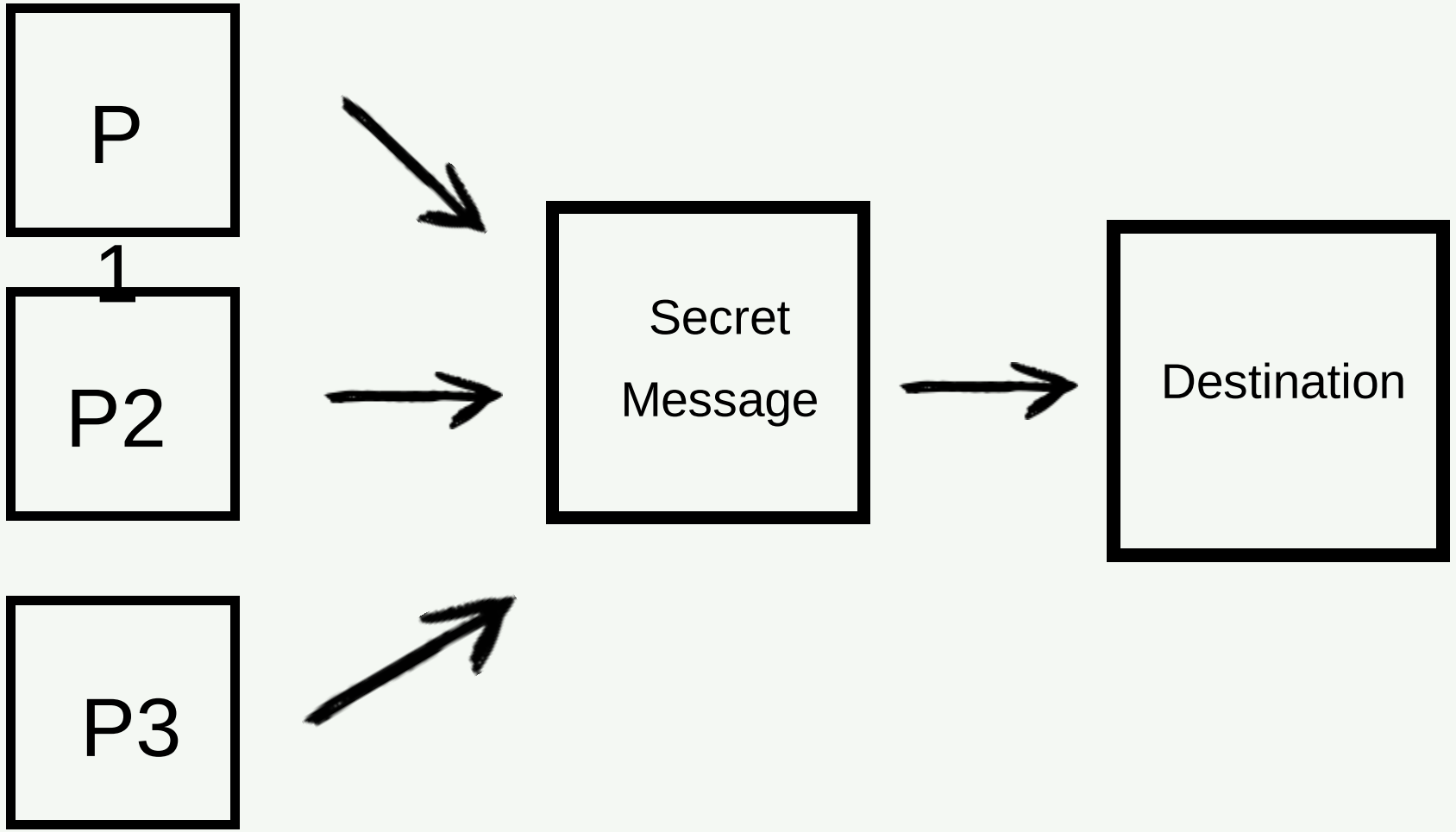
Data is broken down into packets



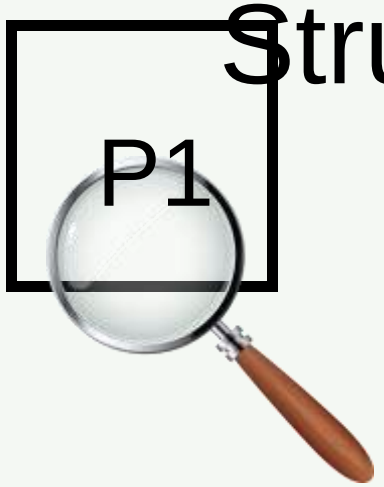
Each packet can be sent along a different route



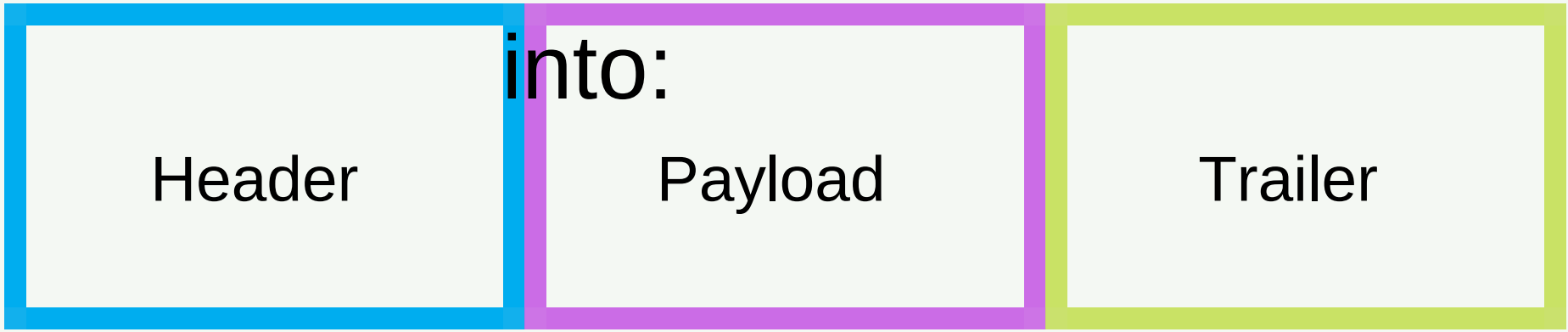
One disadvantage : Data needs to be **reassembled** when it reaches the destination.



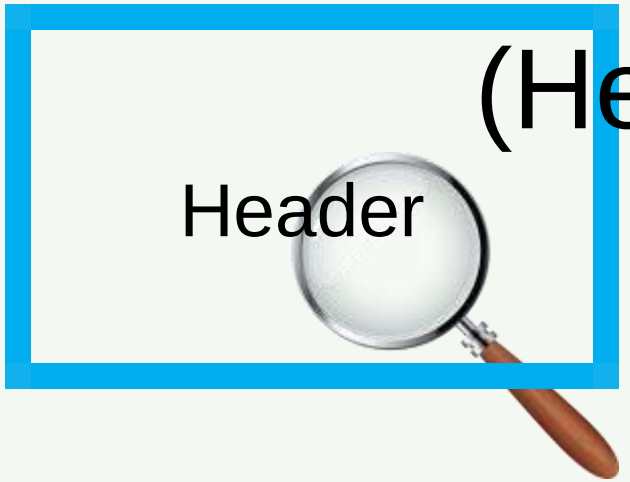
Packet Structure



A typical packet is split up into:



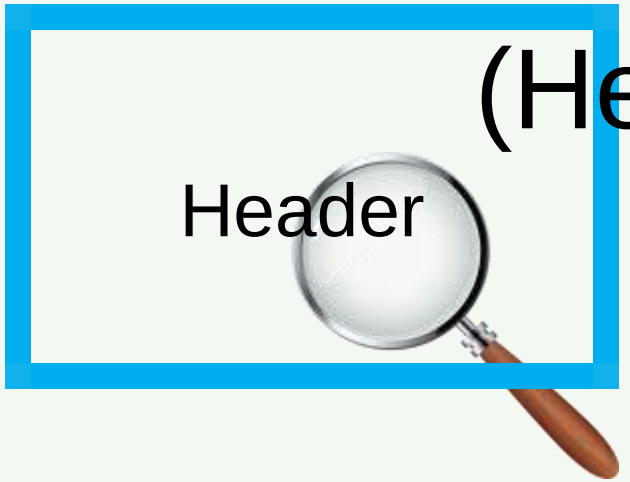
Packet Structure (Header)



- Sender IP Address
- Receiver IP Address
- Sequence number of the packet.
  - This is to ensure that all the packets can be reassembled correctly once they reach the destination.
- Size of the packet
  - This is to ensure that the receiving station can check if all of the packets have arrived.

Packet Structure

(Header)

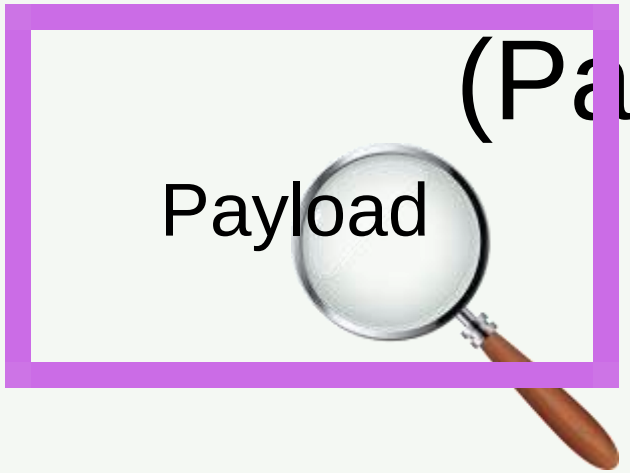


- Sender IP Address
- Receiver IP Address
- Sequence number of the packet.
  - This is to ensure that all the packets can be reassembled correctly once they reach the destination.
- Size of the packet
  - This is to ensure that the receiving station can check if all of the packets have arrived.

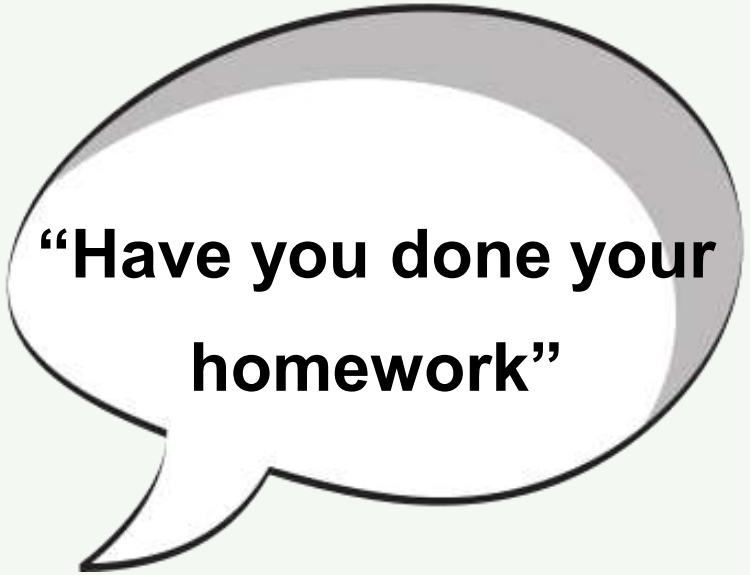
Remember SSSR

Packet Structure

(Payload)



The actual data in the packet.



Packet Structure (Trailer)

- Some ways to identify the end of the packet. This is essential to allow each packet to be separated from each other as they travel from the sending to receiving station.

Packet Structure (Trailer)

2. Some form of error checking to ensure packet arrives error free.

Cyclic Redundancy Checks

- The sending computer will add up all the 1-bits in the payload and store it as a hex value in the trailer before it is sent.

Eg. Payload	Number of 1 bit	Hexadecimal	Trailer
1110100110100 1111010010101	→ 1 5	→ F	→ F

Cyclic Redundancy Checks

Trailer

2. Once the packet arrives, the receiving computer recalculate the number of 1-bits in the payload.

Eg.

Payload

111010011010

0

111101001010

1

→ Number of 1 bit → 1 5 → Hexadecimal → F

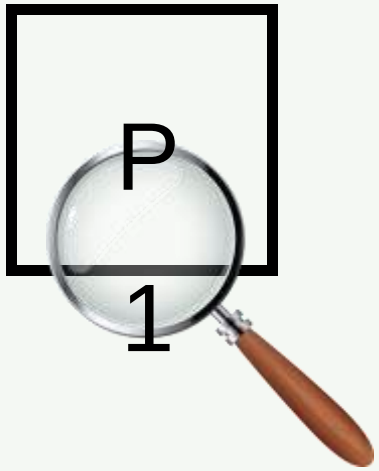
Compare

Trailer

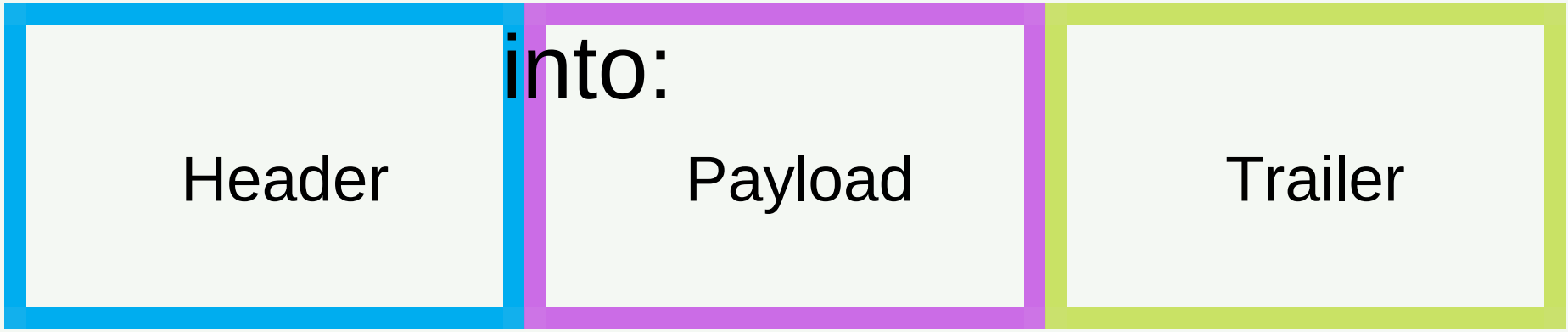
F

Does error occur?

Packet Structure



A typical packet is split up into:

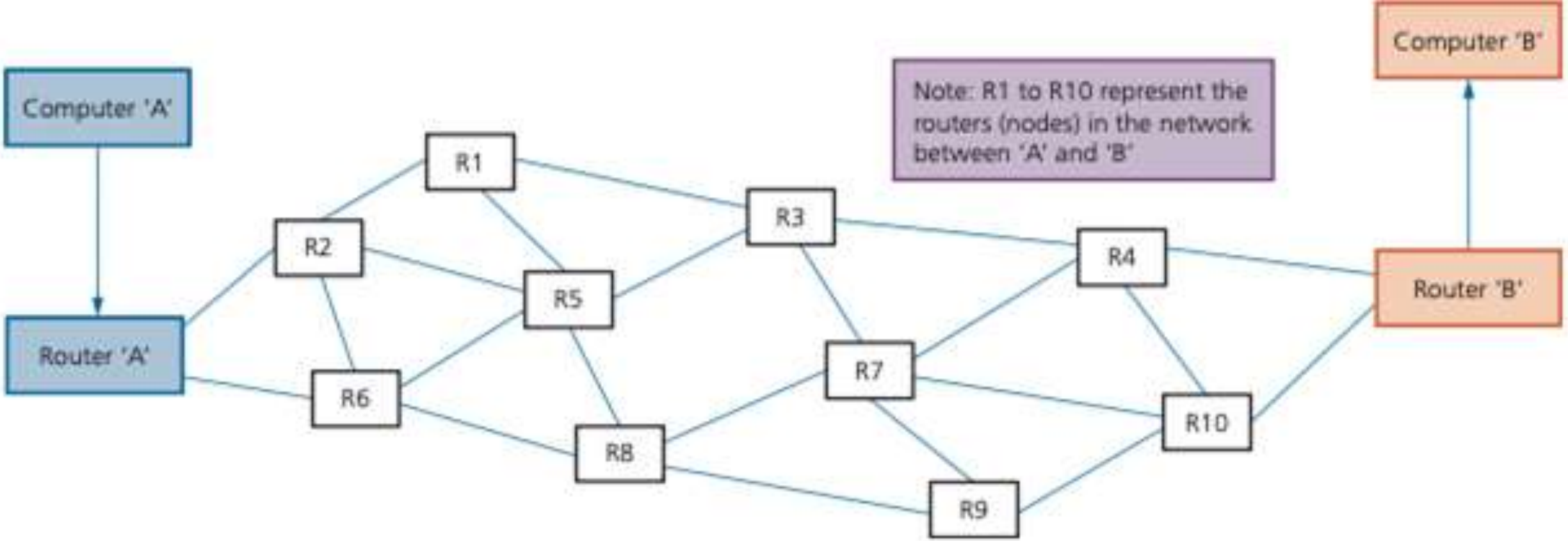


Packet Switching

- Packet switching is a method of data transmission in which a message is broken up into a number of packets. (Payload)
- Each packet can be sent independently from start point to end point.
- At the destination, the packets will need to be reassembled into their correct order.
- At each stage in the transmission there are nodes that contain a router.
- Each router will determine which route the packet needs to take, in order to reach its destination (The destination IP address is used in this part of the process).

Packet Switching

- Packet switching is a method of data transmission in which a message is broken up into a number of packets. (Payload)
- Each packet can be sent independently from start point to end point.
- At the destination, the packets will need to be reassembled into their correct order.
- At each stage in the transmission there are nodes that contain a router.
- Each router will determine which route the packet needs to take, in order to reach its destination (The destination IP address is used in this part of the process).



1. The router will determine the route of each packet.
2. Routing Selection depends on the number of packets waiting to be processed at each node.
3. The shortest possible path available is always selected.
4. Packets can arrive in a different order compared to the way they were sent.

Benefits of packet switching

- There is no need to tie up a single communication line.
- A high data transmission rate is possible

Drawbacks of packet switching

- Data can be lost and need to be re-sent
- Delay at the destination whilst the packets are being re-ordered)

Data Transmission

- There are several different methods of transmitting data depending on the types of hardware and connections being used.

Types of data transmission

- Serial
- Parallel

Transmission Mode

- Simplex
- Half-duplex
- Full-duplex

Data Transmission

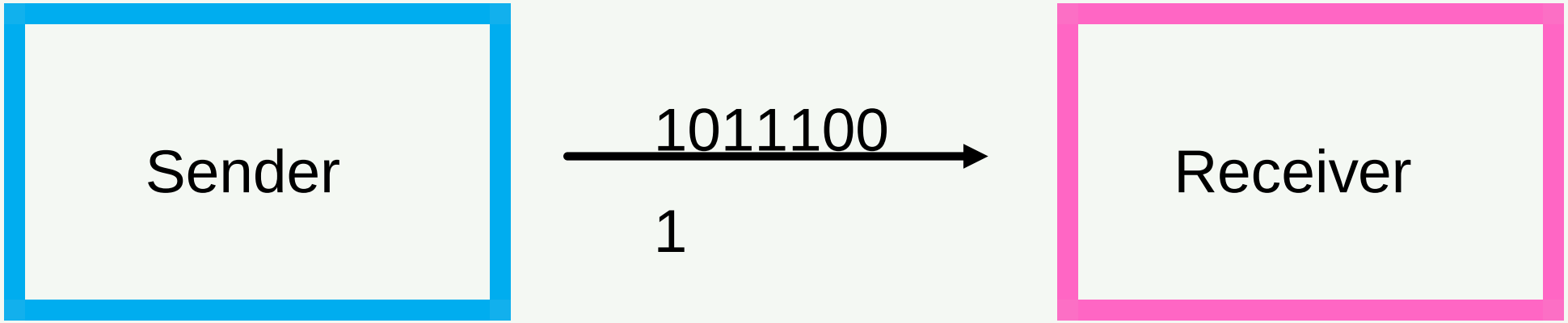
- There are several different methods of transmitting data depending on the types of hardware and connections being used.

Types of data transmission

- Serial
- Parallel

Types of data transmission - Serial

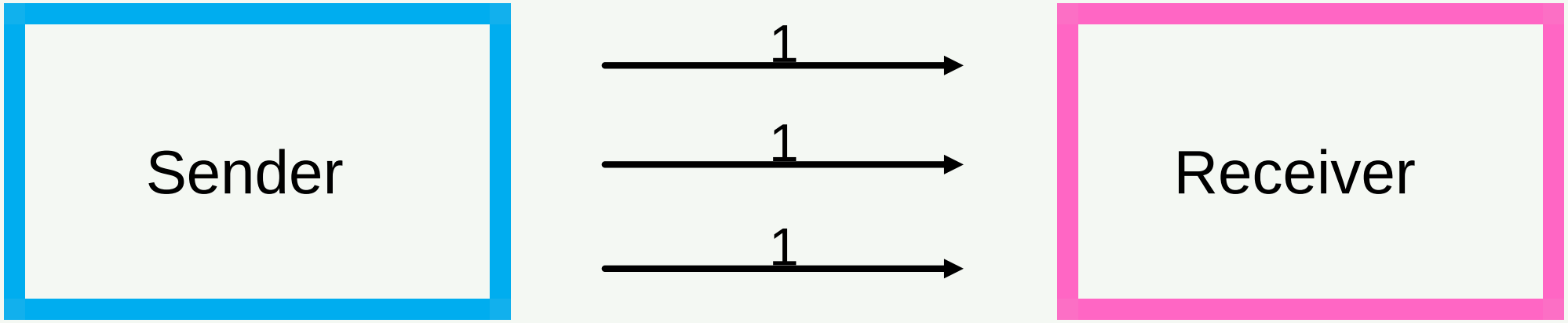
- Serial data transmission occurs when data is sent ONE BIT AT A TIME over a single wire/channel.
- Bits are sent one after the other as a single stream.



Example : Computer is connected directly to the printer via a USB connection.

Types of data transmission - Parallel

- Parallel data transmission occurs when SEVERAL BITS OF DATA are sent down SEVERAL CHANNELS/WIRES all at the same time.
- Each channel/wire transmits one bit



Example : Internal circuits in a computer.

	Advantage	Disadvantage
Serial	• Reliable in transmitting data over long distance • Fewer Errors	• Data transmission can be slow as only one channel is used
Parallel	• Faster data transmission as multiple channels are used • Parallel data transmission works well over short distances.	• Expensive • Synchronisation errors can happen due to the separation of data across different channels

Data Transmission

- There are several different methods of transmitting data depending on the types of hardware and connections being used.

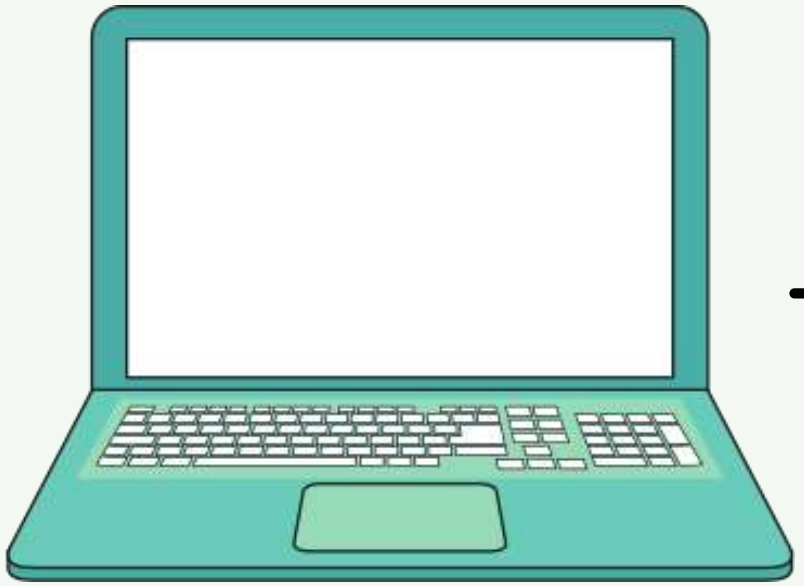
Transmission Mode

- Simplex
- Half-duplex
- Full-duplex

Transmission mode - Simplex

- **Simplex** mode occurs when data can be sent in **ONE DIRECTION ONLY** (for example, from sender to receiver).

LAPTOP TO PRINTER

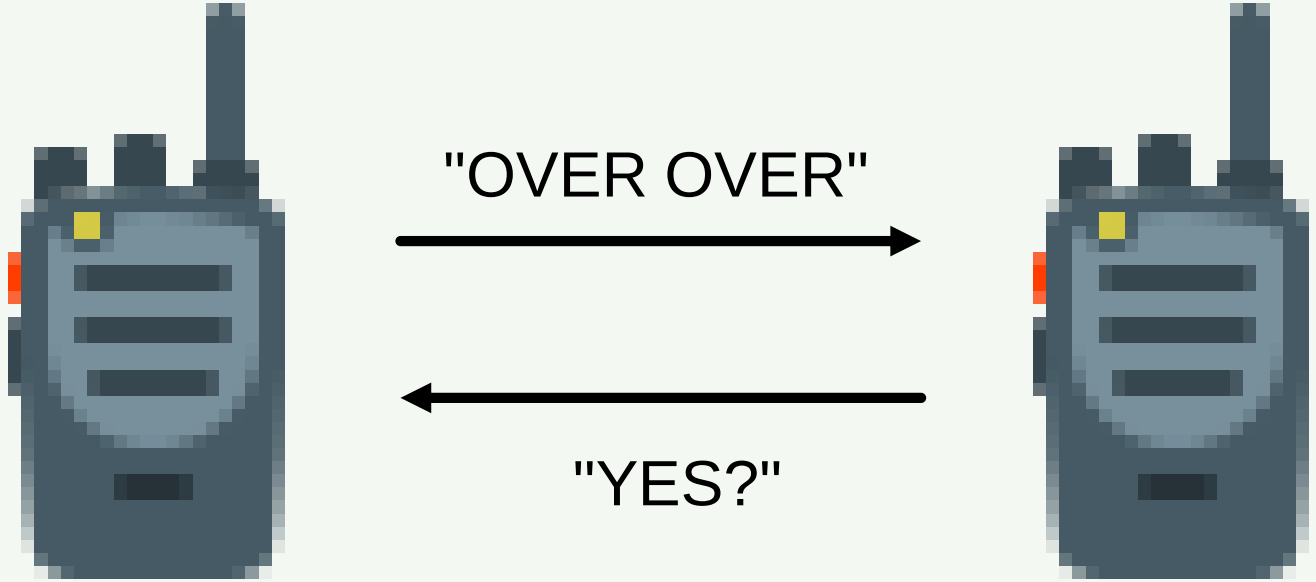


Print this page for me

Transmission mode - Half-duplex

- **Half-duplex** mode occurs when data is sent in **BOTH DIRECTIONS** but **NOT AT THE SAME TIME** (for example, data can be sent from ‘A’ to ‘B’ and from ‘B’ to ‘A’ along the same transmission line, but they can’t both be done at the same time).

WALKIE TALKIE



Transmission mode - Full-duplex

- **Full-duplex** mode occurs when data can be sent in **BOTH DIRECTIONS AT THE SAME TIME** (for example, data can be sent from ‘A’ to ‘B’ and from ‘B’ to ‘A’ along the same transmission line simultaneously).

BROADBAND

INTERNET

CONNECTION

	Advantage	Disadvantage
Simplex	• The process can use full bandwidth for the channel	• Two way communication is not possible
Half-Duplex	• Enable two-way transmission using full bandwidth	• Delay in the communication
Full-Duplex	• Fastest duplex transmission method due to each communication using its own simplex channel. No delay	• Some networks are not able to utilise the technology.

DIY

Which types of data transmission are being described:

data is sent one bit at a time in one direction only

Types of data transmission

Serial

Parallel

Transmission Mode

Simplex

Half-duplex

Full-duplex

Students, draw anywhere on this slide!

DIY

Which types of data transmission are being described:

data is being sent 8 bits at a time in one direction only

Types of data transmission

Serial

Parallel

Transmission Mode

Simplex

Half-duplex

Full-duplex

Students, draw anywhere on this slide!

DIY

Which types of data transmission are being described:

data is being sent 16 bits at a time in both directions simultaneously

Types of data transmission

Serial

Parallel

Transmission Mode

Simplex

Half-duplex

Full-duplex

Students, draw anywhere on this slide!

DIY

Which types of data transmission are being described:

data is sent one bit at a time in both directions simultaneously

Types of data transmission

- [ ] Serial
- [ ] Parallel

Transmission Mode

- [ ] Simplex
- [ ] Half-duplex
- [ ] Full-duplex

Students, draw anywhere on this slide!

PAST YEAR QUESTIONS

(a) **Five** statements are given about duplex data transmission.

Tick (✓) to show if the statement is **True** or **False**.

Statement	True (✓)	False (✓)
Duplex data transmission can be either serial or parallel		
Duplex data transmission is when data is transmitted both ways, but only one way at a time		
Duplex data transmission is always used to connect a device to a computer		
Duplex data transmission is when data is transmitted both ways at the same time		
Duplex data transmission automatically detects any errors in data		

[5]

PAST YEAR QUESTIONS

Statement	True (✓)	False (✓)
Duplex data transmission can be either serial or parallel	✓	
Duplex data transmission is when data is transmitted both ways, but only one way at a time		✓
Duplex data transmission is always used to connect a device to a computer		✓
Duplex data transmission is when data is transmitted both ways at the same time	✓	
Duplex data transmission automatically detects any errors in data		✓

PAST YEAR QUESTIONS

Sonia shares files with her friends. The method of data transmission she uses is half-duplex serial transmission.

(a) Describe how data is transmitted using half-duplex serial data transmission.

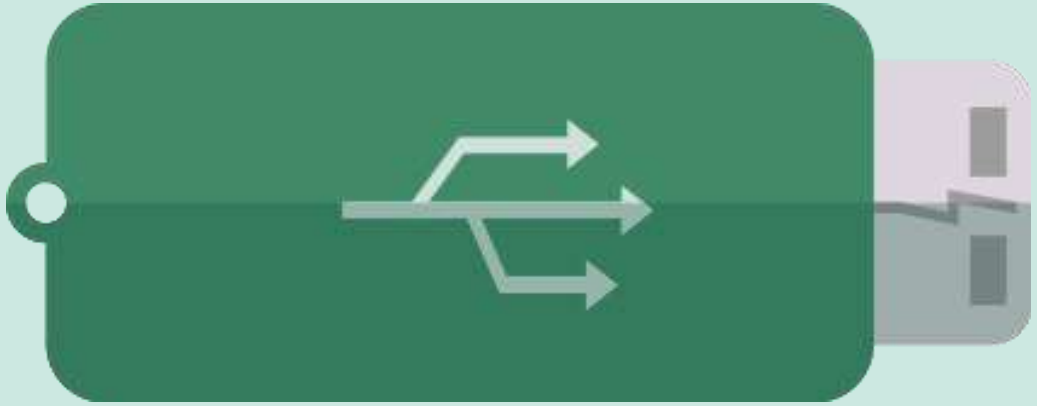
.....

.....

.....

PAST YEAR QUESTIONS

Question	
5(a)	∝ Data is sent down a single wire … ∞ … one bit at a time ∞ Data is sent in both directions … ∞ … but only one direction at a time



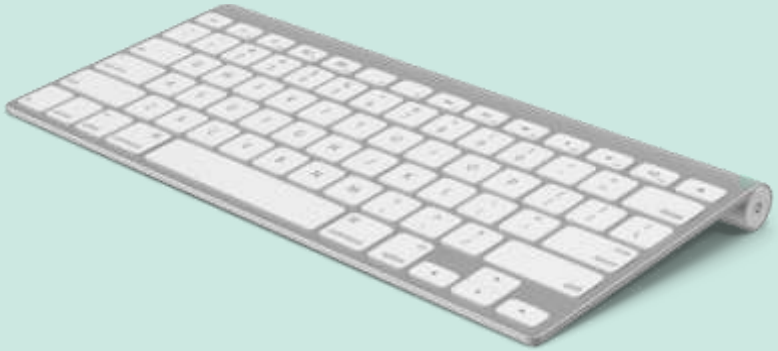
Chapter 2.2

Universal Serial Bus (USB)

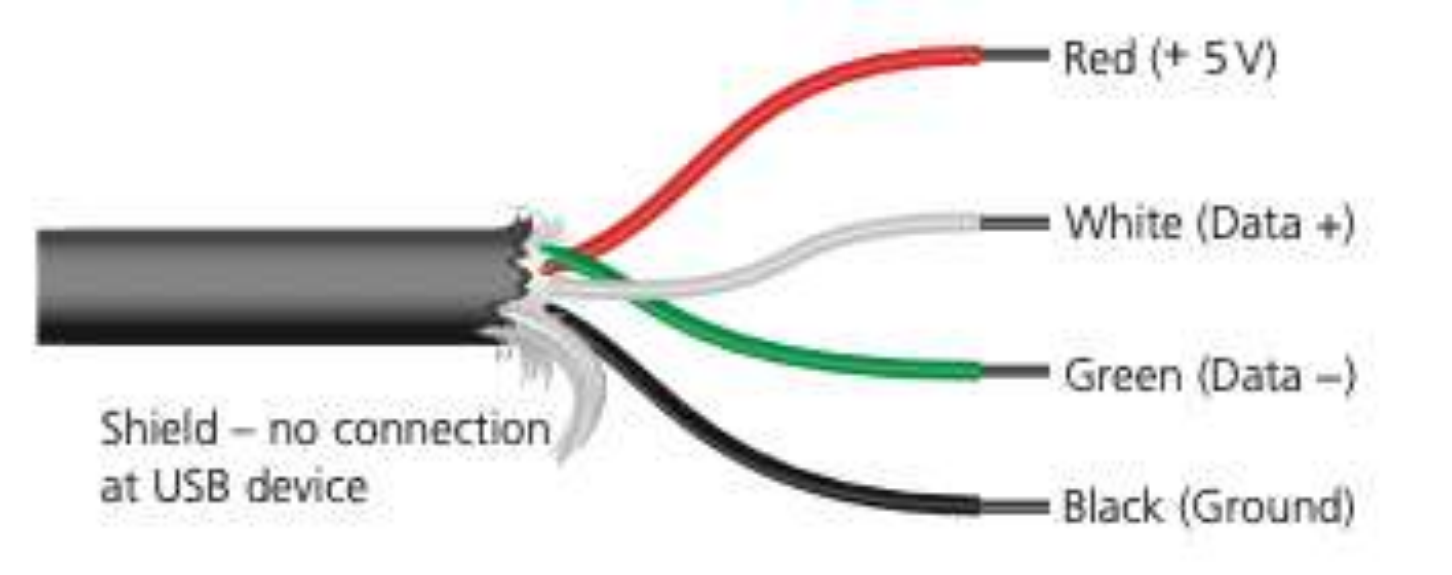
YEAR 10

Universal Serial Bus (USB)

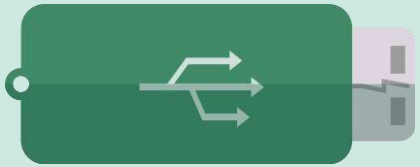
- Universal serial bus is a form of SERIAL data transmission.
- USB is a type of communication port that has been built into computers in order to do away with other older forms of port.
- USB is now the most common type of input/output port found on computers and has led to a standardisation method for the transfer of data between devices and a computer.
- USB allows both half-duplex and full-duplex data transmission.



USB Cable



- The USB cable consists of a four-wired shielded cable, with
  - two wires for power (red and black)
  - two wires (white and green) are for data transmission.

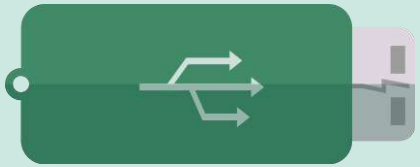


When a device is plugged into a computer using one of the USB ports:

- Computer automatically detects that a device is present (this is due to a small change in the voltage on the data signal wires in the USB cable).
- The device is automatically recognised, and the appropriate device driver software is loaded up so that the computer and device can communicate effectively
- If this is not available, the user is prompted to download the appropriate driver software

USB-C

- A new type of USB connector, referred to as USB-C, is now becoming more common in laptops and tablets/phone .
- This is a 24-pin symmetrical connector which means it will fit into a USB-C port either way round.
- It is much smaller and thinner than older USB connectors, offers 100 watt (20 volt) power connectivity, which means full-sized devices can now be charged and it can carry data at 10 gigabits per second (10 Gbps); this means it can now support 4K video delivery.



Advantages and disadvantages of USB

- Ubiquitous. It is everywhere.
- Able to supply power and data (external power source is not needed)
- Backward compatible as well as suitable adaptor is used

- Data transmission speed is still relatively slow.
- Transmission quality will deteriorate when the cable becomes too long (not more than 5m)

PAST YEAR QUESTIONS

(c) Julia uses a USB connection to transfer data onto her USB flash memory drive.

(i) One benefit of using a USB connection is that it is a universal connection.

State **two** other benefits of using a USB connection.

Benefit 1 .....

.....

Benefit 2 .....

.....

[2]

(ii) Identify the type of data transmission used in a USB connection.

..... [1]

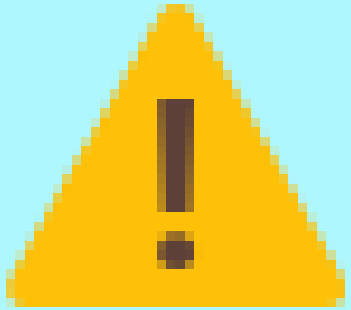
PAST YEAR QUESTIONS

2(c)(i)	Any **two** from: – It cannot be inserted incorrectly – Supports different transmission speeds – High speed **transmission** – **Automatically** detected (not connected) // automatically downloads drivers – Powers the device (for data transfer) – Backward compatible	**2**
2(c)(ii)	– Serial	**1**

Chapter 2.3

Methods of error detection

IGCSE Computer Science



HELP!

Can you raed tihs?

"I cnduo't bvleiee taht I culod aulaclty uesdtannrd waht I was rdnaieg. Unisg the icndeblire pweor of the hmuan mnid, aocdcrnig to rseecrah at Cmabridge Uinervtisy, it dseno't mttaer in waht oderr the lterets in a wrod are, the olny irpoamtnt tihng is taht the frsit and lsat ltteer be in the rhgit pclae. The rset can be a taotl mses and you can sitll raed it whoutit a pboerlm.

Tihs is bucseae the huamn mnid deos not raed ervey ltteer by istlef, but the wrod as a wlohe.

Aaznmig, huh? Yeah and I awlyas tghhuot slelinpg was ipmorantt! See if yuor fdreins can raed tihs too"

(from an unknown source at Cambridge University)

The need to check for errors.

- When data is transmitted, there is always a risk that it may be corrupted, lost or even gained.
- Errors can occur during data transmission due to:

1. **Interference (all types of cable can suffer from electrical interference, which can cause data to be corrupted or even lost)**
2. **Problems during packet switching (this can lead to data loss – or it is even possible to gain data!)**
3. **Skewing of data (this occurs during parallel data transmission and can cause data corruption if the bits arrive out of synchronisation)**

Five error detection methods

Parity Checks

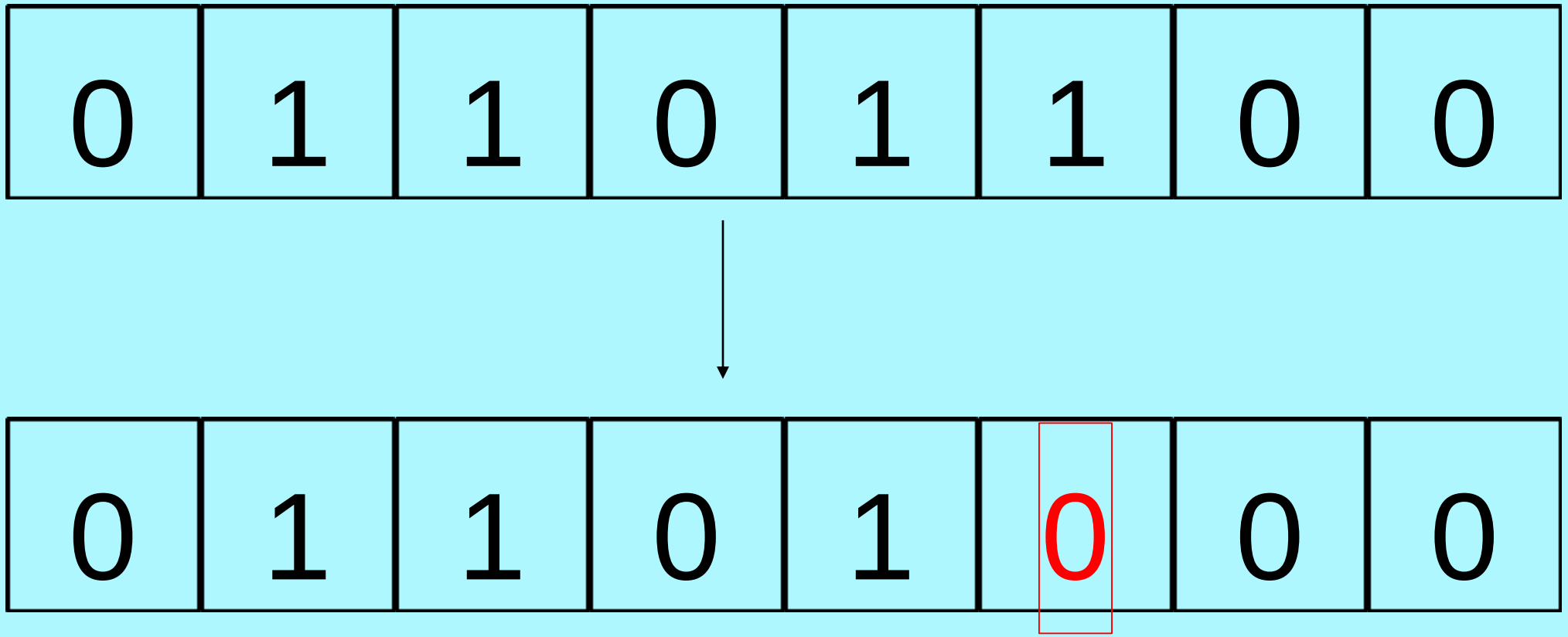
Echo Check

Check Digit

Checksum

Automatic Repeat Request

How do we know that bits are being transmitted correctly?



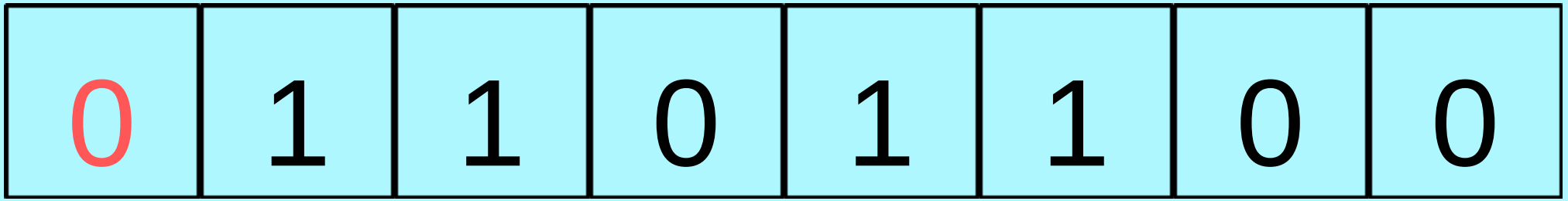
Parity Checks

- Parity checking is one method used to check whether data has been changed or corrupted following data transmission. This method is based on the number of 1-bits in a byte of data.
- The parity check can be either called **EVEN** and **ODD**
- One of the bits in the byte (usually the most significant bit or left-most bit) is reserved for a **parity bit**.

SETTING UP THE PARITY BIT - EVEN PARITY

The parity bit is set according to whether the parity being used is even or odd. For example, consider the byte:

What should I put here if I want an EVEN number of 1s?



0, awesome!

SETTING UP THE PARITY BIT - ODD PARITY

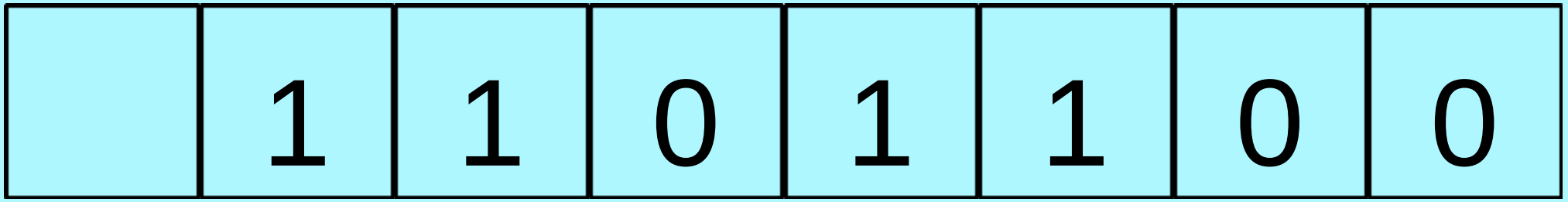
The parity bit is set according to whether the parity being used is even or odd. For example, consider the byte:

What should I put here if I want an ODD number of 1s?

1	1	1	0	1	1	0	0

1, awesome!

Steps to set up a parity bit:



- Count the number of 1s
- Ask a question:
  - Even parity: What digit should I put in the empty box if I want an EVEN number of 1s?
  - Odd parity: What digit should I put in the empty box if I want an ODD number of 1s?
  - Decide yourself based on the question!

How does parity check helps detect errors?

Message

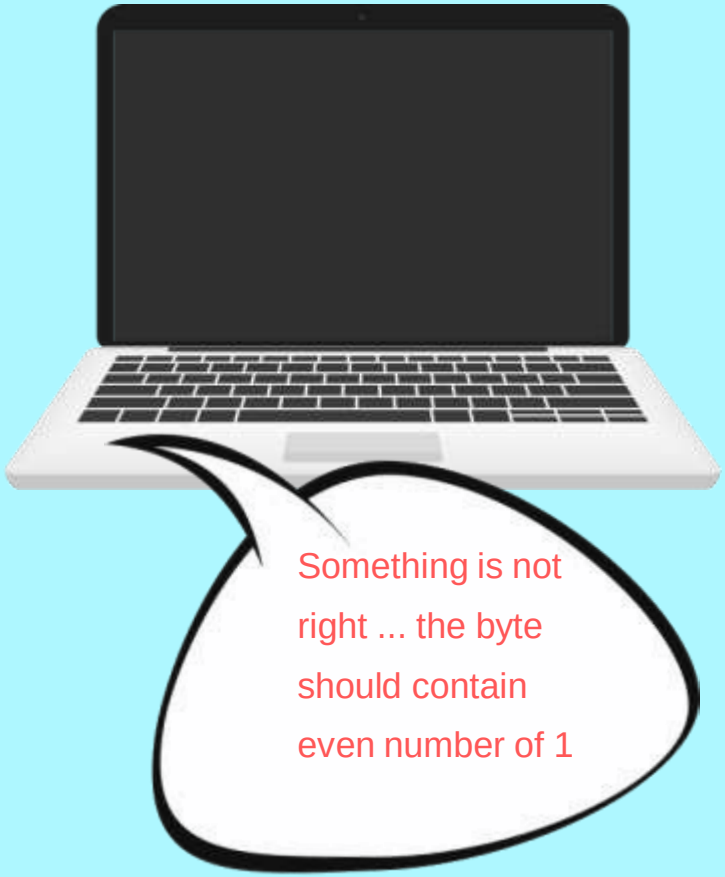
0	1	1	0	1	1	0	0

Let’s say that even parity is used

Let's say the message is corrupted...

0	1	1	0	0	1	0	0

Let’s say that even parity is used



The computer will detect that an error has occurred during the transmission of the byte.

The error is detected by the recipient’s computer re-calculating the parity of the byte sent.

Do it yourself - does any error occur?

1)

Message (even parity is used)

1	1	1	0	1	1	0	1

2)

Message (odd parity is used)

1	1	1	1	1	1	1	1

Opps, there is a problem with parity check ...

Message

0	1	0	1	1	1	0	0

Even parity

is used

Parity check will not work if two bits are corrupted!

Message

0	1	1	1	1	1	0	1

Even parity

is used

In all these cases, the byte has clearly been corrupted, but number of ‘1’ bits remains even. Therefore, no error would be flagged in spite of the obvious errors in transmission.

Remember: When two bits (an even number of bits) are corrupted, parity check fails.

Checksum

1. When a block of data is about to be transmitted, the checksum is calculated from the block of data. The calculation is done using an agreed algorithm (this algorithm has been agreed by sender and receiver) .

1	2	3	4	5	6	7	8
25	37	15	45	32	13	6	

Algorithm:

1. Sum all the numbers.
2. Divide the sum by 7. Let the remainder of the division to be the checksum.

Checksum

1. When a block of data is about to be transmitted, the checksum is calculated from the block of data. The calculation is done using an agreed algorithm (this algorithm has been agreed by sender and receiver) .
2. The checksum is then transmitted with the block of data

1	2	3	4	5	6	7	checksum
25	37	15	45	32	13	6	5

Checksum

1. When a block of data is about to be transmitted, the checksum is calculated from the block of data. The calculation is done using an agreed algorithm (this algorithm has been agreed by sender and receiver) .
2. The checksum is then transmitted with the block of data
3. At the receiving end, the checksum is recalculated by the computer using the block of data (the agreed algorithm is used to find the checksum) .The re-calculated checksum is then compared to the checksum sent with the data block.

1	2	3	4	5	6	7	checksum
25	37	15	45	32	13	6	5

Checksum

1. When a block of data is about to be transmitted, the checksum is calculated from the block of data. The calculation is done using an agreed algorithm (this algorithm has been agreed by sender and receiver) .
2. The checksum is then transmitted with the block of data
3. At the receiving end, the checksum is recalculated by the computer using the block of data (the agreed algorithm is used to find the checksum) .The re-calculated checksum is then compared to the checksum sent with the data block
4. If the two checksums are the same, then no transmission errors have occurred.

1	2	3	4	5	6	7	checksum
25	37	15	45	32	13	6	5

Checksum

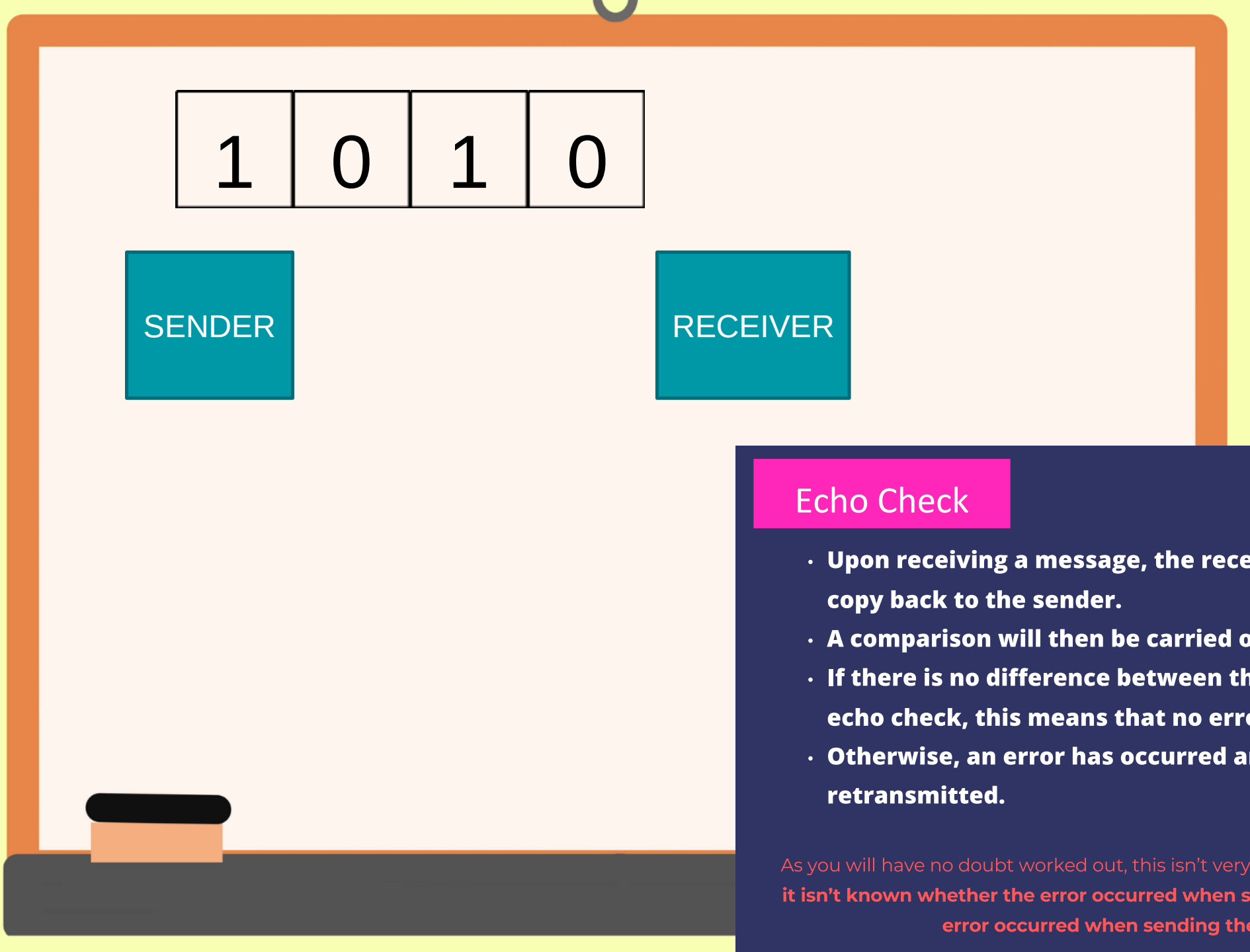
1. When a block of data is about to be transmitted, the checksum is calculated from the block of data. The calculation is done using an agreed algorithm (this algorithm has been agreed by sender and receiver) .
2. The checksum is then transmitted with the block of data
3. At the receiving end, the checksum is recalculated by the computer using the block of data (the agreed algorithm is used to find the checksum) .The re-calculated checksum is then compared to the checksum sent with the data block
4. If the two checksums are the same, then no transmission errors have occurred. Otherwise an error has occured and a request is made to re-send the block of data.

1	2	3	4	5	6	7	checksum
25	37	15	25	32	13	6	5

Echo Check

- **Upon receiving a message, the receiver will immediately send a copy back to the sender.**
- **A comparison will then be carried out.**
- **If there is no difference between the two sets of data during the echo check, this means that no error has occurred.**
- **Otherwise, an error has occurred and the data will be retransmitted.**

As you will have no doubt worked out, this isn't very reliable. If the two sets of data are different**, it isn't known whether the error occurred when sending the data in the first place, or if the error occurred when sending the data back for checking.**



EXAMPLES - ECHO CHECK IN ACTION

Echo Check

- **Upon receiving a message, the receiver will immediately send a copy back to the sender.**
- **A comparison will then be carried out.**
- **If there is no difference between the two sets of data during the echo check, this means that no error has occurred.**
- **Otherwise, an error has occurred and the data will be retransmitted.**

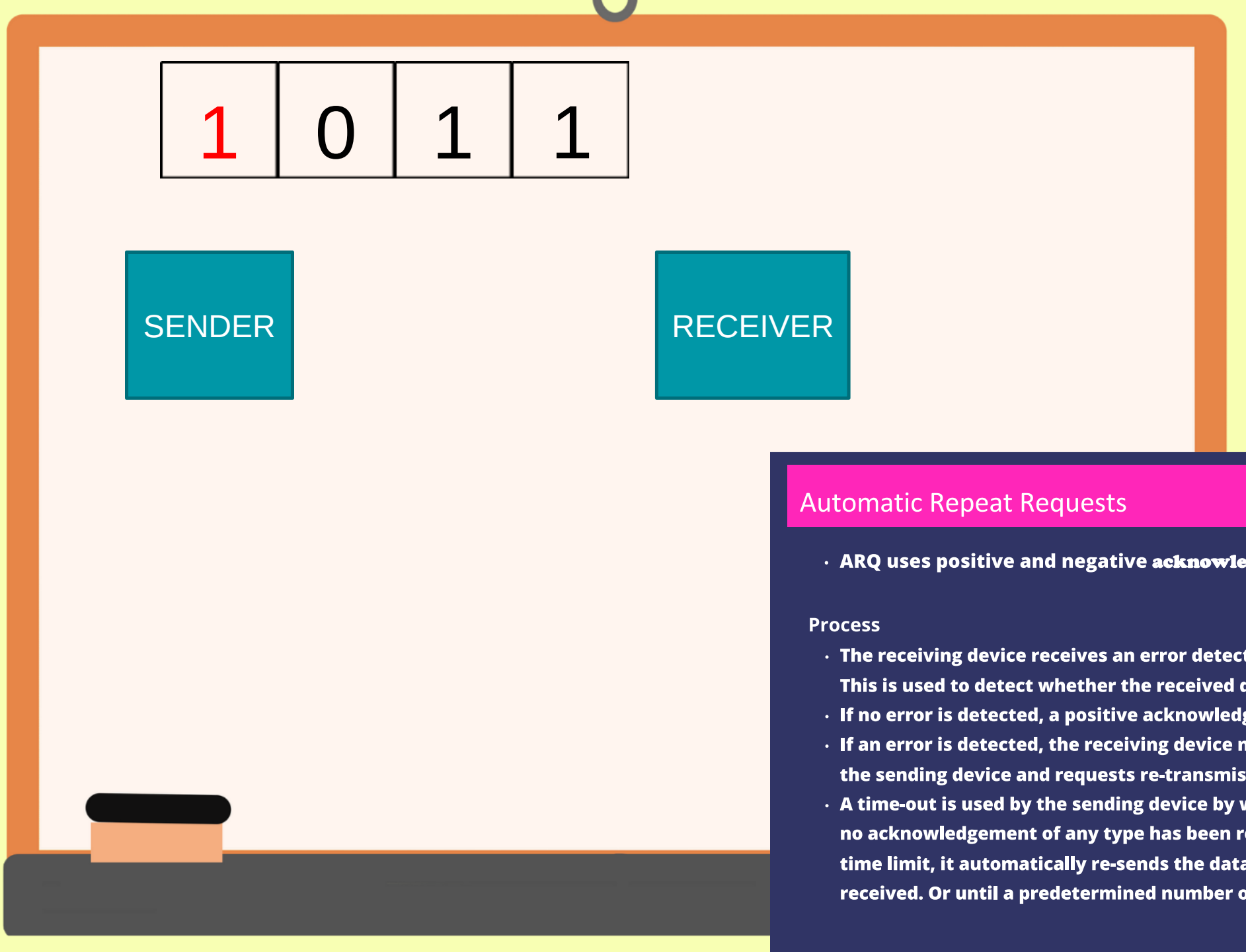
As you will have no doubt worked out, this isn't very reliable. If the two sets of data are different, **it isn't known whether the error occurred when sending the data in the first place, or if the error occurred when sending the data back for checking.**

Automatic Repeat Requests

- ARQ uses positive and negative **acknowledgements and timeout.**

Process

- The receiving device receives an error detection code as part of the data transmission. This is used to detect whether the received data contains any transmission error.
- If no error is detected, a positive acknowledgement is sent back to the sending device.
- If an error is detected, the receiving device now sends a negative acknowledgement to the sending device and requests re-transmission of the data.
- A time-out is used by the sending device by waiting a predetermined amount of time. If no acknowledgement of any type has been received by the sending device within this time limit, it automatically re-sends the data until a positive acknowledgement is received. Or until a predetermined number of re-transmissions has taken place.



EXAMPLES - ARQ IN ACTION

Automatic Repeat Requests

- **ARQ uses positive and negative acknowledgements and timeout.**

Process

- **The receiving device receives an error detection code as part of the data transmission. This is used to detect whether the received data contains any transmission error.**
- **If no error is detected, a positive acknowledgement is sent back to the sending device.**
- **If an error is detected, the receiving device now sends a negative acknowledgement to the sending device and requests re-transmission of the data.**
- **A time-out is used by the sending device by waiting a predetermined amount of time. If no acknowledgement of any type has been received by the sending device within this time limit, it automatically re-sends the data until a positive acknowledgement is received. Or until a predetermined number of re-transmissions has taken place.**

Check Digits

- **The error detection systems described above help to spot errors during the transmission of data between two different devices. Sometimes, data discrepancies can occur due to human input errors.**

1. **An incorrect digit entered, for example 5327 entered instead of 5307**
2. **Transposition errors where two numbers have changed order, for example 5037 instead of 5307**
3. **Omitted or extra digits, for example 537 instead of 5307 or 53107 instead of 5307**
4. **Phonetic errors, for example 13 (thirteen), instead of 30**

Check Digits - ISBN 13 methods

1. **The check digit in ISBN 13 is the thirteenth digit in the number.**

2 . The thirteenth digit is generated using the other 12 digits in a number.

Different methods are used for different system when calculating the thirteenth digit.

3. The thirteenth digit will help to identify whether an error has occurred or not.

Check Digits - ISBN 13 methods

- **Add all the odd numbered digits together (Not the 13rd number)**
- **A**dd all the even numbered digits together and multiply the result by 3
- **A**dd the results from 1 and 2 together and divide by 10
- **T**ake the remainder, if it is zero then use this value, otherwise subtract the remainder from 10 to find the check digit.

EXAMPLES - CHECK DIGIT IN ACTION

1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11st	12nd	13rd
9	7	9	3	1	6	1	4	8	4	1	0	9

International Standard Book Numbers

Check Digits - ISBN 13 methods

- **Add all the odd numbered digits together (Not the 13rd number)**
- **A**dd all the even numbered digits together and multiply the result by 3
- **A**dd the results from 1 and 2 together and divide by 10
- **T**ake the remainder, if it is zero then use this value, otherwise subtract the remainder from 10 to find the check digit.

PAST YEAR QUESTIONS

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 YEAR QUESTIONS

Question	Answer
9	1 mark per each correct parity bit:

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

PAST YEAR QUESTIONS

(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 YEAR QUESTIONS

Question	Answer
10(a)	**Four** from: ∞ Validation method ∞ Used to check data entry ∞ Digit is calculated from data // by example ∞ Digit is appended / added to data ∞ Digit is recalculated when data has been input ∞ Digits are compared ∞ If digits are different, error is detected // If digits match, no error is detected

PAST YEAR QUESTIONS

(d) Errors can occur when data is transmitted, stored or entered into a system.

Darius could use an error detection method to find whether errors have occurred.

One error detection method he could use is a checksum.

(I) Describe how a checksum detects errors.

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [5]

PAST YEAR QUESTIONS

4(d)(i)	– Sending device creates value from calculation on **data** // By example – Value is transmitted with the data – Receiving device performs same calculation – Values are compared **after transmission** // If values do not match … – … an error is detected

PAST YEAR QUESTIONS

(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.

.....

.....

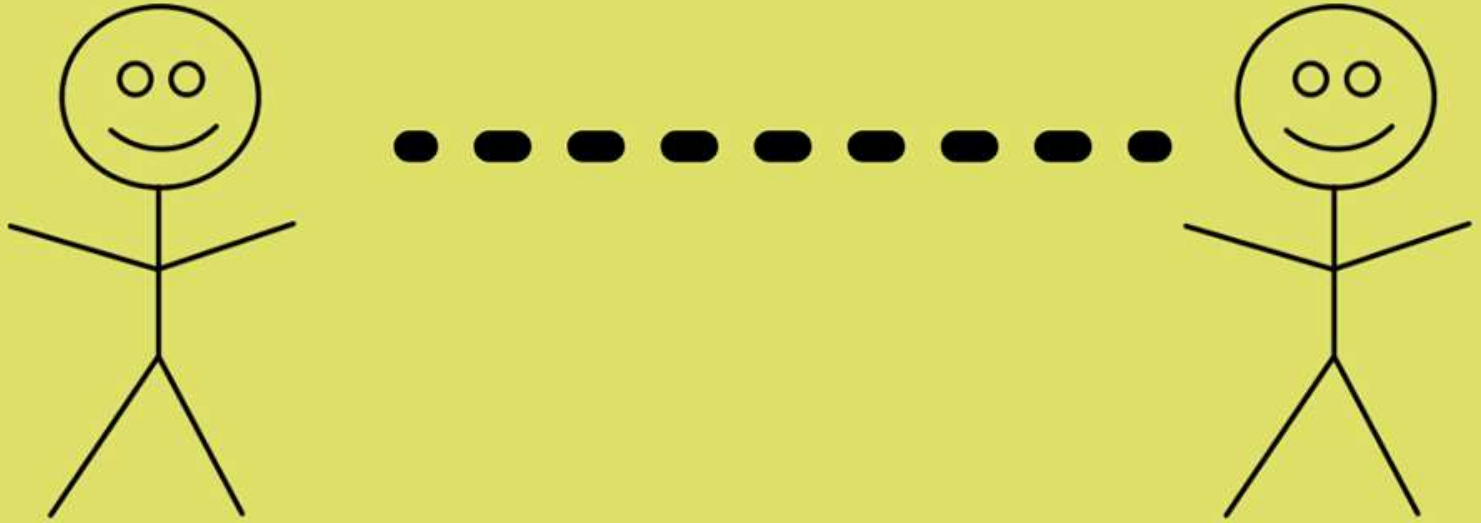
PAST YEAR QUESTIONS

5(b)	**One** mark for correct byte (Byte) 2 // 01010100 **Three** from: ∞ Added up / counted the 1s / 0s ∞ Even parity used // 3 bytes are even ∞ Byte 2 uses odd parity // 1 byte is odd

Chapter 2.4

Symmetric and Asymmetric Encryption

The purpose of encryption

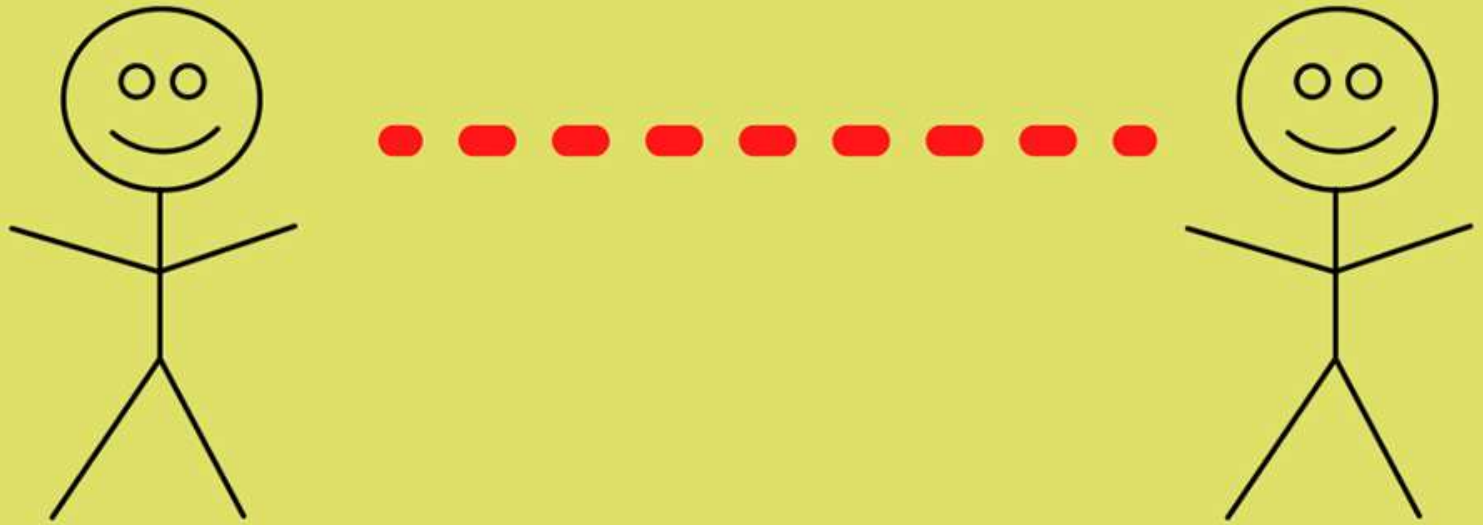


MAIN REASON:

Message could be intercepted by a hacker

Encryption cannot prevent a message from being intercepted, but it stops it from making sense to the hacker.

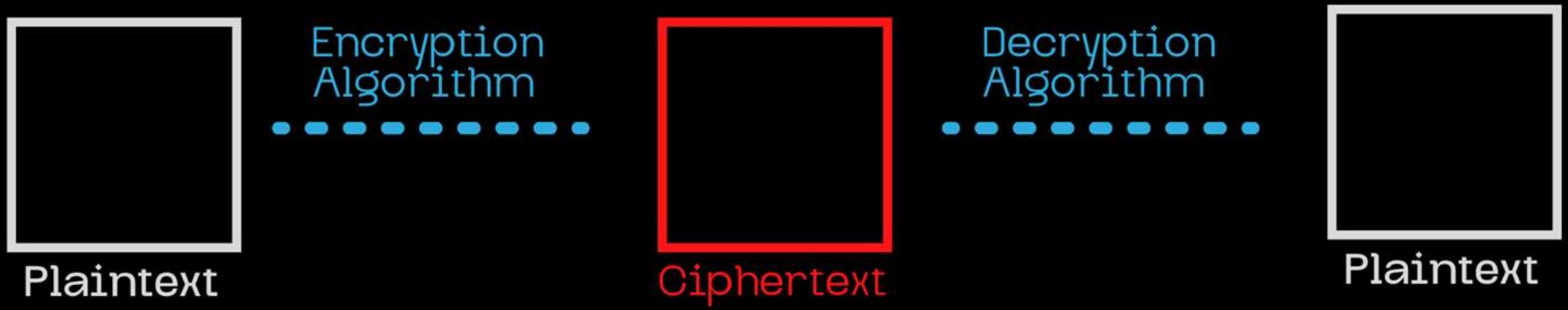
Encryption cannot prevent a message from being intercepted, but it stops it from making sense to the hacker.



Wait, I don't get it....

Plaintext and Ciphertext

Plaintext and Ciphertext



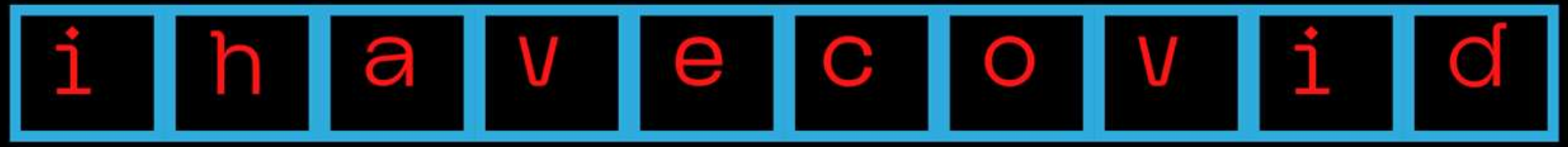
Symmetric Encryption

1. The symmetric encryption uses an encryption key
2. The same key is used to encrypt and decrypt the encoded message.
3. One method is called the Caesar Cipher.

Symmetric Encryption

Original message: I have covid

Key: 4578



Symmetric Encryption - Encrypt

Original message: I have covid

Key: 4578

i	h	a	v	e	c	o	v	i	d
4	5	7	8	4	5	7	8	4	5

Symmetric Encryption - Encrypt

	i	h	a	v	e	c	o	v	i	d
INCREASE	4	5	7	8	4	5	7	8	4	5

Symmetric Encryption - Encrypt

i	h	a	v	e	c	o	v	i	d
4	5	7	8	4	5	7	8	4	5
m	m	h	d	i	h	v	d	m	i

mmhdihvdmi

Symmetric Encryption - Decrypt

m	m	h	d	i	h	v	d	m	i
4	5	7	8	4	5	7	8	4	5

Symmetric Encryption - Decrypt

	m	m	h	d	i	h	v	d	m	i
DECREASE	4	5	7	8	4	5	7	8	4	5

Symmetric Encryption - Decrypt

DECREASE

m	m	h	d	i	h	v	d	m	i
4	5	7	8	4	5	7	8	4	5
i	h	a	v	e	c	o	v	i	d

Why symmetric?

Encrypt

+4

i	h	a	v	e	c	o	v	i	d
4	5	7	8	4	5	7	8	4	5
m	m	h	d	i	h	v	d	m	i

Decrypt

-4

m	m	h	d	i	h	v	d	m	i
4	5	7	8	4	5	7	8	4	5
i	h	a	v	e	c	o	v	i	d

Drawbacks of symmetric encryption

1. The real difficulty is keeping the encryption key a secret.
2. The issue of security is always the main drawback of symmetrical encryption, since a single encryption key is required for both sender and recipient

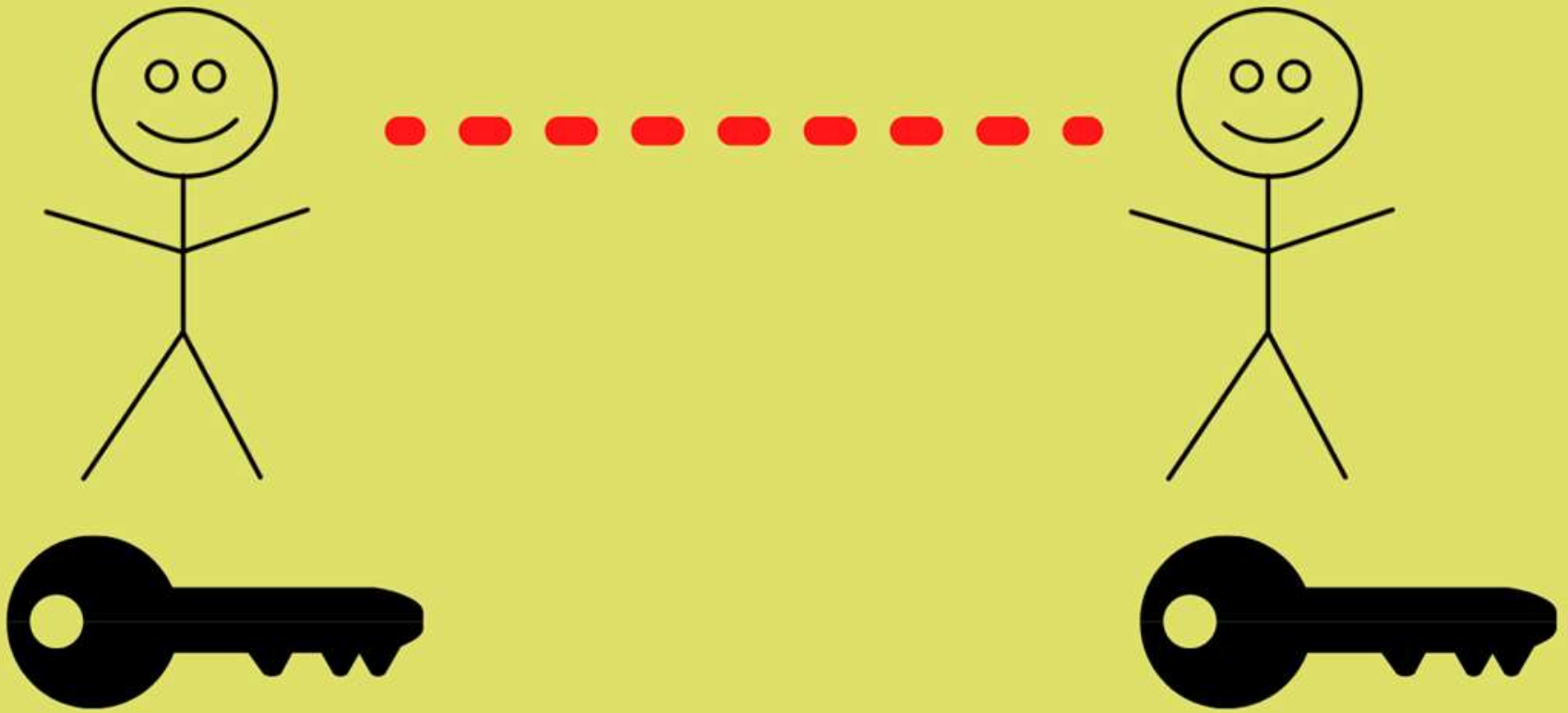
Symmetric Encryption - Encrypt

Original message: I have covid

Key: 4578

i	h	a	v	e	c	o	v	i	d
4	5	7	8	4	5	7	8	4	5

Encryption key is required for both sender and recipient.



Asymmetric Encryption

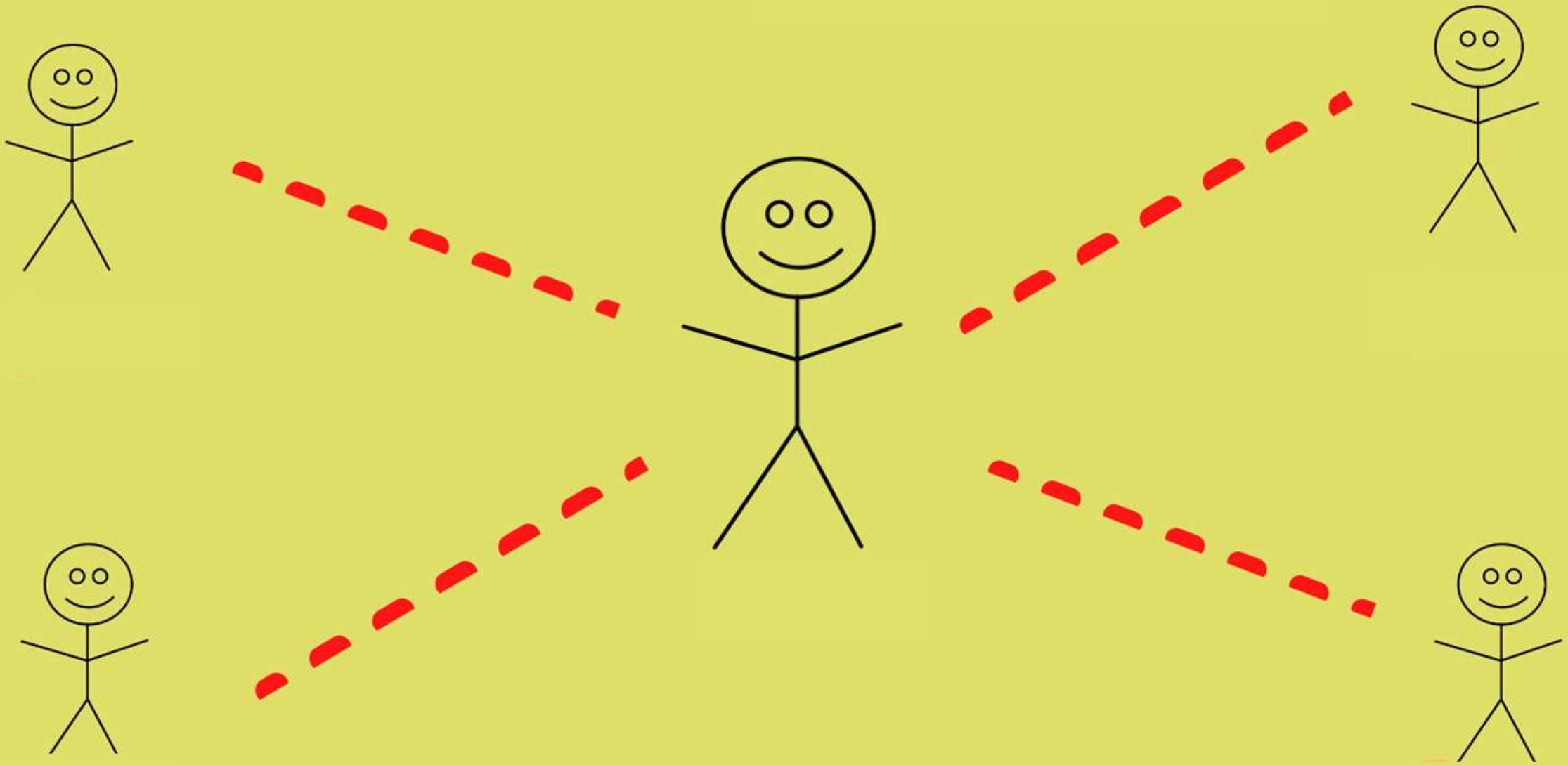
1. Aysmmetric encryption was developed to overcome the security problems associated with symmetric encryption.
2. It makes use of two keys called the public key and the private key:

    a. public key (made available to everybody)
    b. private key (only known to the computer user).

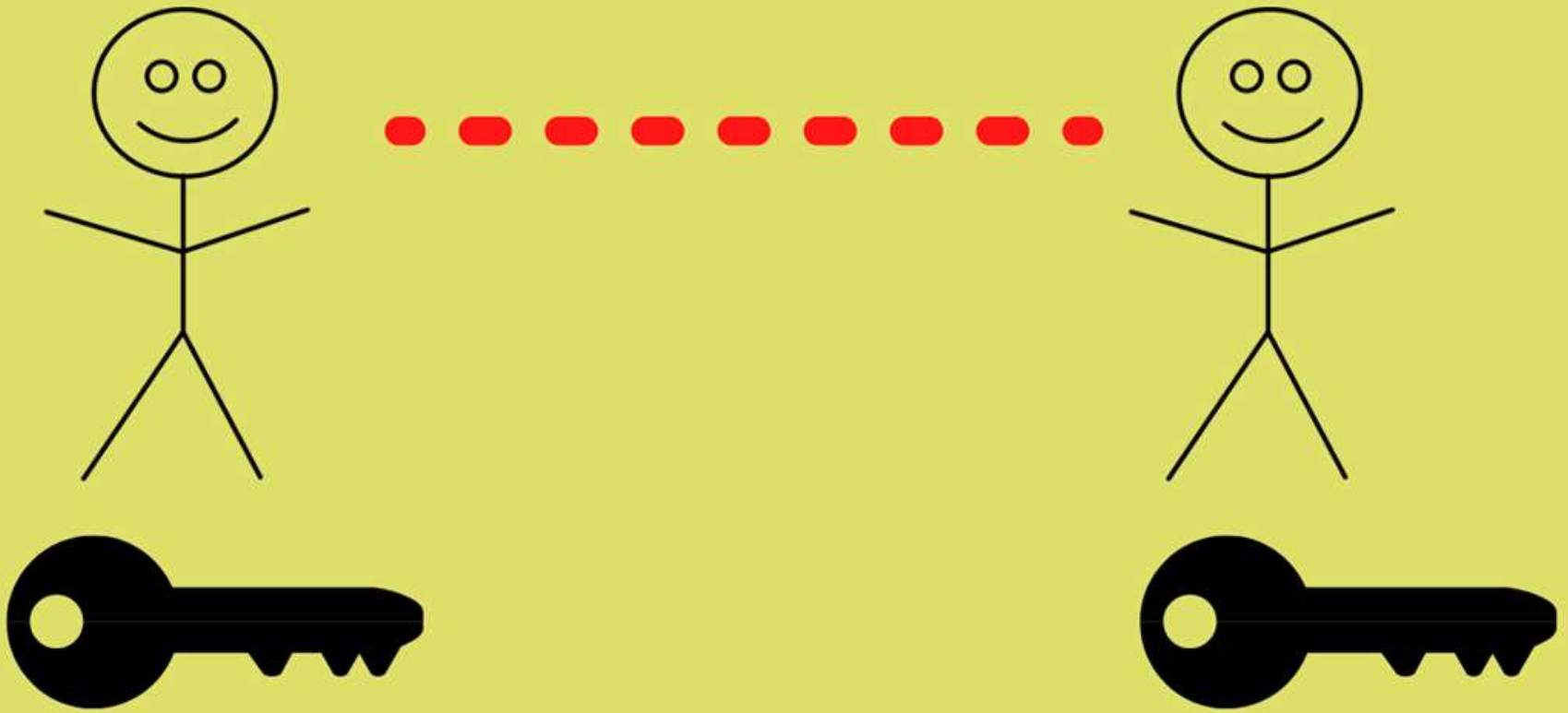
Asymmetric Encryption

1. When a message is ciphered ready for transmission by a sender, the receiver will generate a pair of keys, a public key and a private key.

Asymmetric Encryption



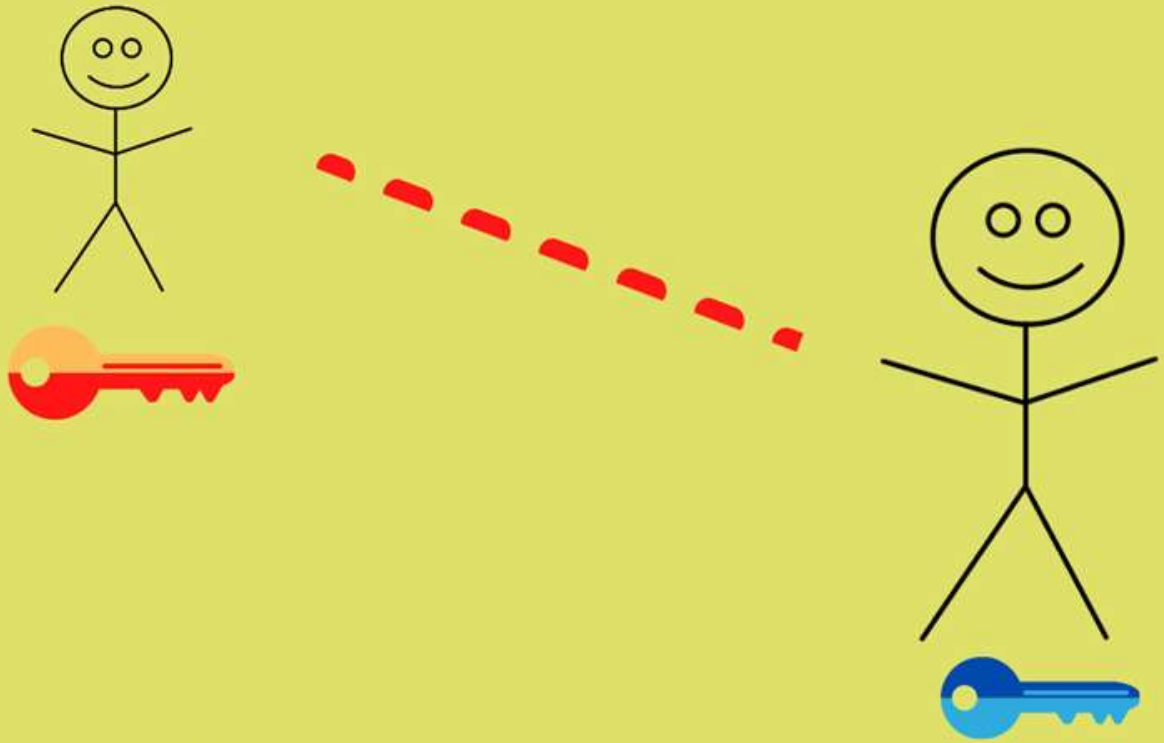
Symmetric Encryption



Important notes

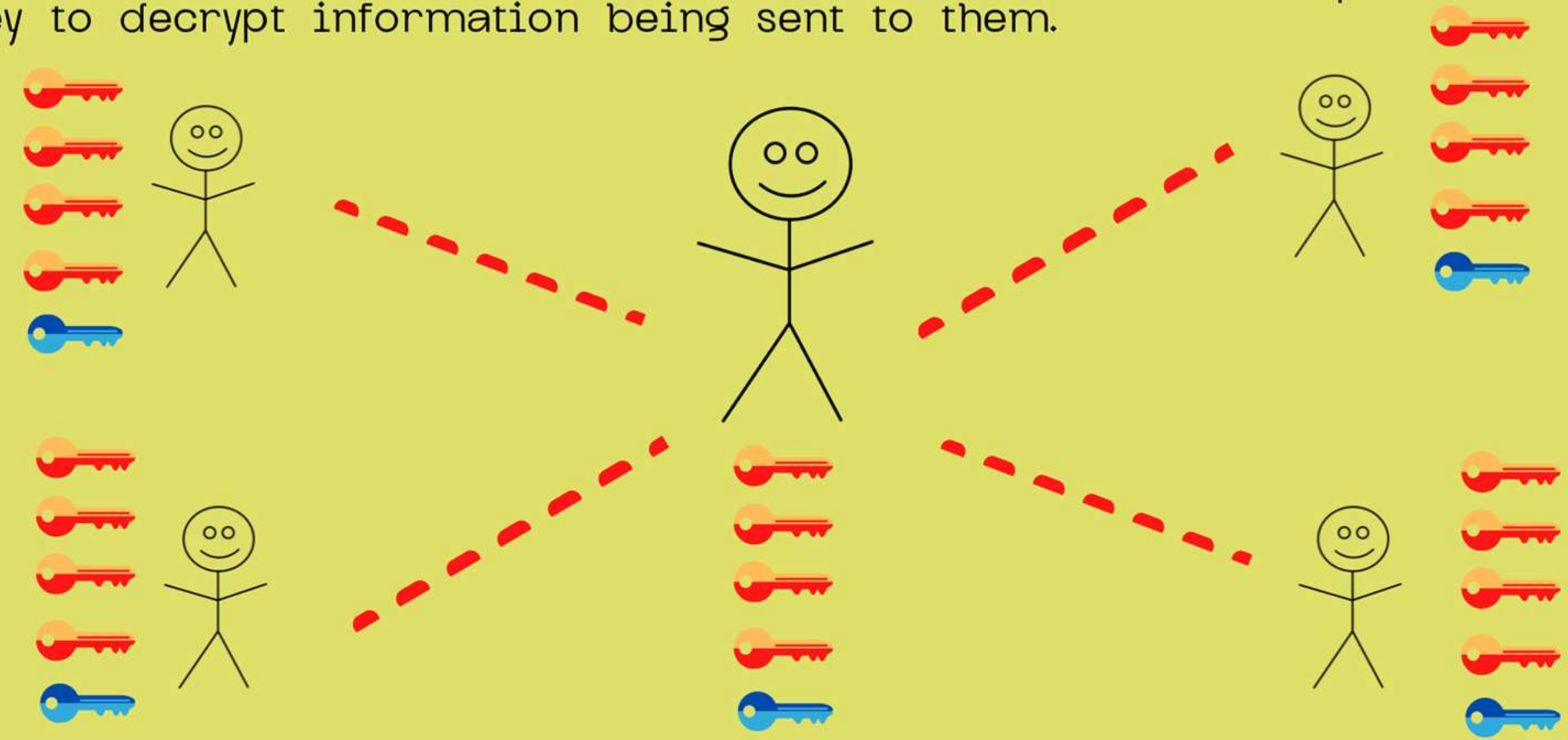
Public key cannot be used to decrypt a message.

Private key is never shared



As a result, asymmetric encryption is more secure than symmetric encryptioon

However, if a two-way communication is required between all five workers, then they all need to generate their own matching public and private keys. Once this is done, all users then need to swap public keys so that they can send encrypted documents/files/messages between each other. Each worker will then use their own private key to decrypt information being sent to them.



PAST YEAR QUESTIONS

(c) Data is encrypted using 128-bit symmetric encryption before it is transmitted.

(i) Explain what is meant by encryption.

.....

.....

.....

..... [2]

(ii) State how the strength of the encryption can be improved.

.....

..... [1]

PAST YEAR QUESTIONS

2(c)(i)	Any **two** from: • Scrambles data • … making it meaningless/unintelligible • Uses an algorithm / key • Data / plain text is changed to cipher text
2(c)(ii)	Any **one** from: • Increase the length of the key // use more than 128 bits • Uses a more complex encryption algorithm